

Automatique

Identification,

Représentation d'état des systèmes linéaires

7 janvier 2019

Avant-propos

Ce cours est destiné aux étudiants en troisième année de l'école d'ingénieurs ISTIA, option *Systemes Automatisés et Génie Informatique*. Il se situe dans la continuité du module "Modélisation et simulation" (3-ième année), et en préambule des modules d'Automatique avancée (en 4-ième et 5-ième année).

Il est supposé que les étudiants ont suivi lors de leurs années d'études précédentes un module d'Automatique générale. Sont notamment considérées comme assimilées par l'étudiant les notions suivantes sur les systèmes linéaires en temps continu :

- représentations entrée-sortie : réponse impulsionnelle et fonction de transfert ;
- précision dynamique et statique ;
- conditions et critères de stabilité.

Table des matières

1	Introduction	1
1.1	Introduction	1
2	Identification des systèmes continus	3
2.1	Introduction	3
2.1.1	Position du problème	3
2.1.2	Rappels sur les modèles de systèmes linéaires en temps continu et en temps discret	4
2.1.3	Démarche pour l'identification	5
2.2	Caractérisation	5
2.2.1	Caractérisation à l'aide d'une réponse indicielle	5
2.2.2	Caractérisation à l'aide d'une réponse à un signal aléatoire : méthode du rapport des déterminants instrumentaux	7
2.3	Estimation à l'aide d'une réponse indicielle	8
2.3.1	Gain statique	8
2.3.2	Système d'ordre 1	8
2.3.3	Système d'ordre 2	9
2.3.4	Systèmes d'ordre supérieur à 2, méthode de <i>Strejc</i>	11
2.4	Estimation à l'aide d'une réponse à un signal aléatoire	12
2.4.1	Méthode (hors-ligne) des moindres carrés ordinaires	13
2.4.2	Méthode (en-ligne) des moindres carrés récursifs	15
3	Représentation d'état des systèmes linéaires	17
3.1	Introduction et rappels	17
3.1.1	Stabilité	20
3.1.2	Précision	20
3.2	Introduction au formalisme d'état	20
3.2.1	Représentation d'état	20
3.2.2	Variables d'état	21
3.2.3	Représentations d'état semblables	22
3.3	Représentation d'état des systèmes linéaires en temps continu	23
3.3.1	D'une représentation d'état vers une représentation entrée-sortie	23
3.3.2	D'une représentation entrée-sortie vers une représentation d'état	24
3.3.3	Simulation à partir de la représentation d'état	26
3.3.4	Solution des équations d'état	26
3.3.5	Stabilité	27
3.3.6	Commandabilité	28
3.3.7	Observabilité	28
3.4	Représentation d'état des systèmes linéaires en temps discret	29
3.4.1	Solution des équations d'état	29
3.4.2	Stabilité	29
3.4.3	Commandabilité	30
3.4.4	Observabilité	30
3.4.5	D'une représentation entrée-sortie vers une représentation d'état	30
3.4.6	Etude de systèmes continus <i>via</i> un calculateur : systèmes échantillonnés	31
3.5	Commande par retour d'état	34

3.5.1	Commande par placement des pôles	35
3.5.2	Régime statique du système bouclé	38
3.5.3	Ajout d'un effet intégral	40
3.6	Reconstruction de l'état	40
3.6.1	Principe du reconstruteur d'état	41
3.6.2	Synthèse de la matrice L_{obs}	42
3.6.3	Principe de séparation	43

Chapitre 1

Introduction

1.1 Introduction

L'automatique est une science qui traite de la modélisation, de l'analyse, de l'identification et de la commande des systèmes dynamiques. Un but courant est de réaliser la commande de systèmes : par exemple, de faire varier les entrées pour que les sorties aient des valeurs fixées à l'avance (consigne), tout en satisfaisant certains critères (sensibilité ou insensibilités à certaines entrées, précision...). Pour cela, on utilise un *modèle dynamique* des systèmes, c'est-à-dire une description mathématique (par exemple, des équations différentielles) du comportement dynamique du système.

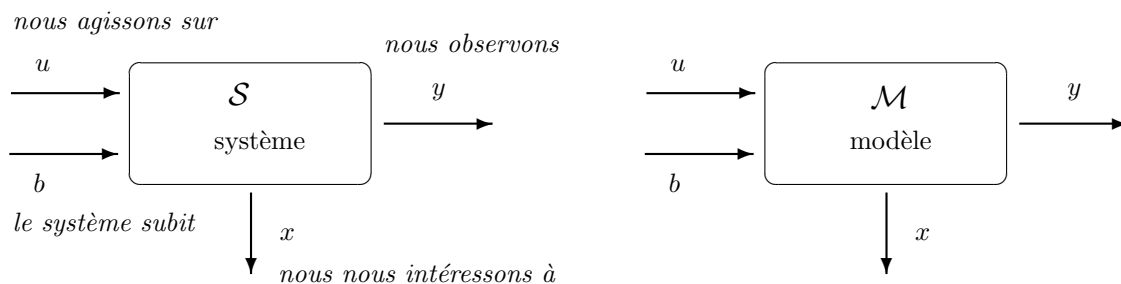


FIGURE 1.1 – Schématisation d'un système et de son modèle

Il y a deux démarches possibles pour élaborer un modèle :

- à partir de la connaissance des phénomènes intervenant (lois de la Physique, de la Chimie, de la Biologie,...) on construit le modèle en écrivant des équations de conservation (masse, moment, énergie,...), des équations de bilan... Cette activité est généralement appelée *modélisation*, et on obtient alors des *modèles dits de connaissance* ou phénoménologiques.
- à partir de données expérimentales (mesures de certaines grandeurs au sein du système généralement soumis à des excitations choisies), on extrapole un modèle mathématique. Cette activité est plutôt appelée *identification*, et on obtient des *modèles de représentation* ou *comportementaux*.

La première partie de ce cours sera dédiée aux techniques d'identification (un cours a été consacré à la modélisation lors du premier semestre).

L'histoire de l'automatique proprement dite commence un peu avant 1930 avec l'élaboration de la théorie de l'Automatique dite fréquentielle. Les modèles manipulés sont de type entrée-sortie (dans le schéma ci-dessus, seuls les signaux u , b et y sont considérés) et prennent souvent la forme de fonctions de transfert. À partir de 1960, émerge la notion d'état (en définissant dans le schéma ci-dessus, le signal x) et de nombreuses techniques ont été développées en utilisant la représentation d'état comme modèle du système.

La deuxième partie de ce cours sera dédiée à la représentation d'état des systèmes. On y abordera notamment la commande par retour d'état ainsi que la reconstruction d'état.

- Un système (et son modèle) est dit *déterministe* si pour chaque entrée $u(t)$, il n'existe qu'une seule sortie possible $y(t)$. Au contraire, dans un modèle non déterministe ou *stochastique*, il y a plusieurs sorties possibles, chacune d'elles étant affectée d'une certaine probabilité.
- Un système (et son modèle) est dit *linéaire*, si le principe de superposition s'applique :

$$\mathcal{S}(k_1 \cdot u_1(t) + k_2 \cdot u_2(t)) = k_1 \cdot \mathcal{S}(u_1(t)) + k_2 \cdot \mathcal{S}(u_2(t)) .$$

- Un système (et son modèle) est dit *stationnaire* si les relations entre l'entrée et la sortie sont indépendantes du temps (les caractéristiques du système sont invariantes dans le temps).
- Un système (et son modèle) est dit *causal* si la valeur de la sortie à un instant t_0 , $y(t_0)$, ne dépend pas des valeurs des valeurs de $u(t)$ pour $t > t_0$ (tous les systèmes physiques sont causaux).

Dans ce cours, on restreint notre attention aux systèmes déterministes, linéaires, stationnaires et causaux.

Chapitre 2

Identification des systèmes continus

2.1 Introduction

L'*identification* consiste à déterminer une description mathématique (un *modèle*) d'un procédé à partir de données expérimentales.

2.1.1 Position du problème

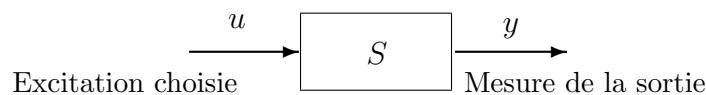
On s'intéresse ici uniquement à des modèles linéaires : cette hypothèse est "classique" et assez peu restrictive (un système non linéaire considéré autour d'un point de fonctionnement s'approche notamment bien comme un système linéaire).

On distingue deux classes de modèles linéaires :

- les *modèles non paramétriques* : par exemple, la réponse impulsionnelle qui établit une suite de points (théoriquement infinie) correspondant à la réponse du système à une impulsion (physiquement impossible à générer parfaitement),
- les *modèles paramétriques* : équations différentielles (en temps continu), équations aux différences (en temps discret), fonctions de transfert avec un nombre fini de paramètres.

On se restreint ici aux modèles paramétriques¹. De plus, on se limite à des systèmes mono-variables (une seule entrée et une seule sortie).

Le problème posé se résume donc par



S linéaire et mono-entrée mono-sortie

Choix d'un modèle paramétrique de S ?

1. Le lecteur peut se reporter par exemple à [Borne et al.] pour un exposé sur les méthodes d'identification non-paramétriques telles que les méthodes de déconvolution et de corrélation.

2.1.2 Rappels sur les modèles de systèmes linéaires en temps continu et en temps discret

	TEMPS DISCRET	TEMPS CONTINU
Convolution	$y(k) = (h * u)(k)$ $= \sum_{i=0}^k h(i)u(k-i)$	$y(t) = (h * u)(t)$ $= \int_0^t h(\tau)u(t-\tau)d\tau$
FdT	$H(z) = \frac{b_n z^n + b_{n-1} z^{n-1} + \dots + b_1 z + b_0}{z^n + a_{n-1} z^{n-1} + \dots + a_1 z + a_0}$ $\mathcal{Z}[h(k)] = H(z)$ $Y(z) = H(z)U(z)$	$H(p) = \frac{b_n p^n + b_{n-1} p^{n-1} + \dots + b_1 p + b_0}{p^n + a_{n-1} p^{n-1} + \dots + a_1 p + a_0}$ $\mathcal{L}[h(t)] = H(p)$ $Y(p) = H(p)U(p)$
Equ. diff.	$y(k+n) + a_{n-1}y(k+n-1) + \dots + a_0y(k)$ $= b_n u(k+n) + b_{n-1}u(k+n-1) + \dots + b_0u(k)$ C.I. $y(0), y(1), \dots, y(n-1)$	$y^{(n)} + a_{n-1}y^{(n-1)} + \dots + a_1\dot{y} + a_0y$ $= b_n u^{(n)} + b_{n-1}u^{(n-1)} + \dots + b_1\dot{u} + b_0u$ C.I. $y(0^+), \dot{y}(0^+), \dots, y^{(n-1)}(0^+)$

Remarques 1

- Comme précisé à la sous-section précédente, on ne considère que des modèles paramétriques, et la réponse impulsionnelle (rappelée ci-dessus dans la représentation par convolution) n'est plus évoquée dans la suite du document.
- Dans la littérature, il est courant de distinguer dans la fonction de transfert le degré du polynôme au numérateur (de coefficients notés b_i) de celui du polynôme au dénominateur (de coefficients notés a_i). Ils sont souvent appelés respectivement le degré et l'ordre du système. L'ordre est supérieur (bien souvent strictement avec $b_n=0$ ci-dessus) au degré pour traduire la causalité du système. Ici, on considère le cas le plus général où le degré est possiblement égal à l'ordre n .
- On rappelle qu'il est possible de passer d'une des représentations à une autre à l'aide d'une transformation adéquate. En particulier, si on dispose d'un modèle sous la forme d'une fonction de transfert en temps continu, il est possible d'en déduire une équation différentielle (à l'aide d'une transformée de Laplace inverse). Réciproquement, on peut passer d'une équation différentielle à une fonction de transfert.
- Dans beaucoup de cas, un système en temps discret correspond à un système continu bloqué échantillonné étudié à partir d'un ordinateur (comme schématisé sur la figure 2.1). Cette chaîne de traitement sera détaillée dans ce support quand nous étudierons la représentation d'état de systèmes échantillonnés.

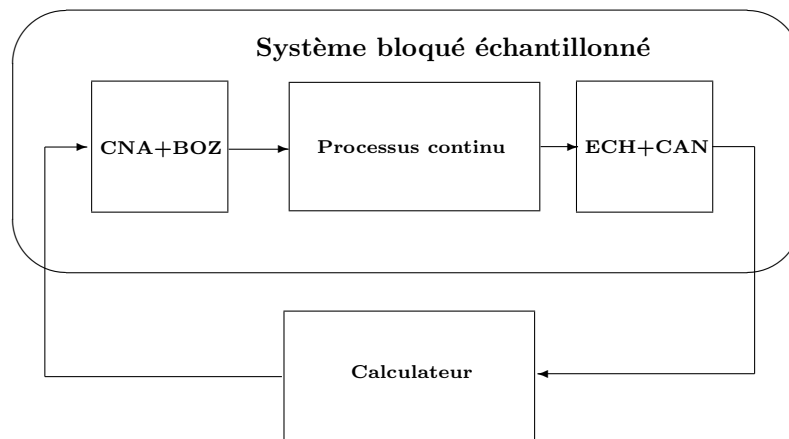


FIGURE 2.1 – Système continu perçu comme un processus bloqué échantillonné par le calculateur

2.1.3 Démarche pour l'identification

Il y a deux étapes principales dans le travail d'identification :

1. la première consiste à fixer la forme des équations : c'est la *caractérisation* abordée dans la section 2.2,
2. la seconde consiste à trouver les valeurs numériques des coefficients qui interviennent dans ces équations : c'est l'*estimation des paramètres* abordée dans les sections 2.3 et 2.4. Ces valeurs numériques sont déterminées pour que le comportement du modèle soit le plus proche possible de celui du système : cette "proximité" se mesure à l'aide d'un *critère*.

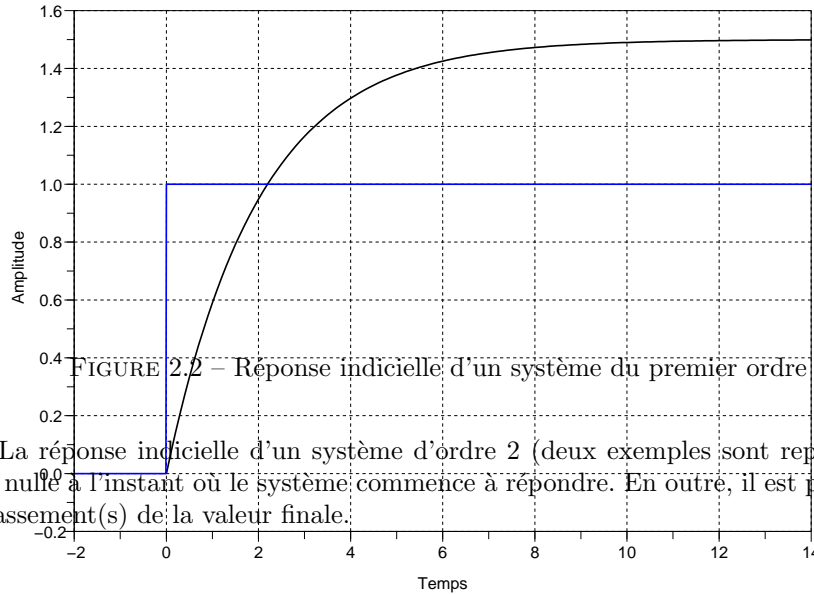
2.2 Caractérisation

L'objet de la caractérisation est de fixer la structure du modèle. On s'intéresse ici seulement à des modèles paramétriques de systèmes linéaires mono-variables et on a rappelé dans la sous-section 2.1.2 les modèles possibles. A la vue de ceux-ci, il apparaît que le problème de caractérisation se résume alors à choisir une valeur pour n mise en jeu dans la fonction de transfert et les équations différentielles.

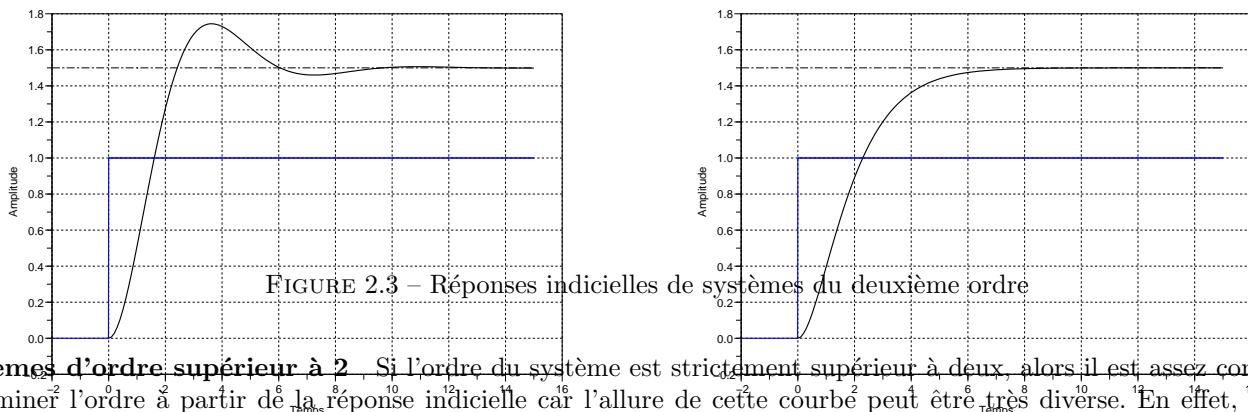
2.2.1 Caractérisation à l'aide d'une réponse indicielle

On rappelle ici des éléments, souvent bien connus, qui permettent de caractériser un système à partir d'observations sur sa réponse à un échelon.

Système d'ordre 1 La réponse indicielle d'un système d'ordre 1 (exemple représenté sur la figure 2.2) est caractérisée par une dérivée non nulle à l'instant où le système commence à répondre. On a également une absence de dépassement de la valeur finale, mais cela ne permet pas de reconnaître un système d'ordre 1 puisque des systèmes d'ordre supérieur à 1 peuvent également présenter cette caractéristique (voir paragraphes suivants).



Système d'ordre 2 La réponse indicielle d'un système d'ordre 2 (deux exemples sont représentés sur la figure 2.3) est caractérisée par dérivée nulle à l'instant où le système commence à répondre. En outre, il est possible (mais pas nécessaire!) qu'il y ait un (des) dépassement(s) de la valeur finale.



Systèmes d'ordre supérieur à 2 Si l'ordre du système est strictement supérieur à deux, alors il est assez compliqué de déterminer l'ordre à partir de la réponse indicielle car l'allure de cette courbe peut être très diverse. En effet, la réponse d'un tel système peut être vue comme la superposition des réponses de sous-systèmes d'ordre un et deux. La réponse du système est généralement marquée par certains de ses pôles qui sont dits *dominants* et qui correspondent à des constantes de temps élevées ou à des amortissements faibles (réponse temporelle lente ou très oscillatoire). Deux approches peuvent être envisagées :

- approcher le système comme un système d'ordre 1 ou 2 en l'assimilant au(x) pôle(s) dominant(s),

- utiliser une méthode de caractérisation plus "élaborée" pour déterminer précisément l'ordre du système (dans ce manuscrit, on présente une méthode au paragraphe 2.2.2 qui peut être utilisée à cet effet).

Retards purs Il existe des systèmes (généralement à "dynamiques lentes", tels que certains systèmes thermiques ou hydrauliques) pour lesquels il y a un retard entre l'instant d'excitation en entrée et l'instant de réaction en sortie du système. On a tracé sur la figure 2.4, la réponse indicielle d'un système du premier ordre retardé.

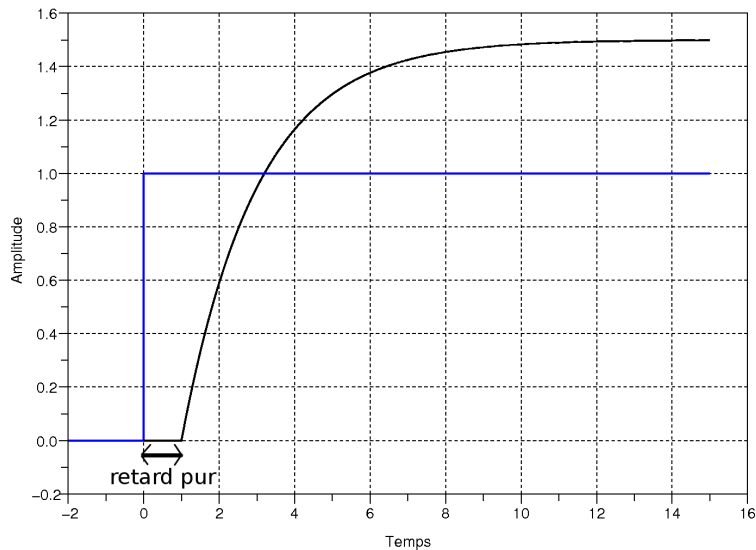


FIGURE 2.4 – Réponse indicielle d'un système du premier ordre retardé

Si on note τ la valeur du retard pur observé, et r est le nombre de pas discret traduisant le retard pur τ , on obtient les fonctions de transfert suivantes pour un système d'ordre n :

TEMPS DISCRET	TEMPS CONTINU
$H(z) = \frac{b_n z^n + b_{n-1} z^{n-1} + \dots + b_0}{z^n + a_{n-1} z^{n-1} + \dots + a_0} \mathbf{z}^{-r}$	$H(p) = \frac{b_n p^n + b_{n-1} p^{n-1} + \dots + b_0}{p^n + a_{n-1} p^{n-1} + \dots + a_0} \mathbf{e}^{-p\tau}$

Pour un système bloqué-échantillonné, on prend $r = \lfloor \tau/\Delta \rfloor$ où Δ désigne la période d'échantillonnage et $\lfloor \cdot \rfloor$ est la partie entière par défaut.

En considérant la fonction de transfert en temps discret, on peut remarquer que :

$$\begin{aligned} H(z) &= \frac{b_n z^n + b_{n-1} z^{n-1} + \dots + b_0}{z^n + a_{n-1} z^{n-1} + \dots + a_0} \mathbf{z}^{-r} \\ &= \frac{b_n z^n + b_{n-1} z^{n-1} + \dots + b_0}{z^{n+r} + a_{n-1} z^{n+r-1} + \dots + a_0 z^r} \\ &= \frac{\beta_{n+r} z^{n+r} + \beta_{n+r-1} z^{n+r-1} + \dots + \beta_0}{z^{n+r} + \alpha_{n+r-1} z^{n-1+r} + \dots + \alpha_1 z + \alpha_0}, \end{aligned}$$

avec

- $\beta_{n+r} = \dots = \beta_{n+1} = 0$ et $\beta_n = b_n, \dots, \beta_0 = b_0$,
- $\alpha_{n+r-1} = a_{n-1}, \dots, \alpha_r = a_0$ et $\alpha_{r-1} = \dots = \alpha_0 = 0$.

On retiendra donc que, d'un point de vue formel, un système d'ordre n avec r retards purs peut être étudié comme un système d'ordre $n+r$ (avec certains coefficients nuls).

2.2.2 Caractérisation à l'aide d'une réponse à un signal aléatoire : méthode du rapport des déterminants instrumentaux

Dans les paragraphes précédents, on a rappelé comment reconnaître des systèmes d'ordre 1 et 2 (voire d'ordre supérieur) à partir de leur réponse indicielle. Cette façon de procéder laisse une place à "l'appréciation de l'ingénieur" puisqu'il doit interpréter des courbes. Au contraire, on présente un test qui de "façon mécanique" va permettre de déterminer la valeur de n , c'est-à-dire l'ordre d'un système (ou encore la somme de l'ordre et du nombre de retards purs éventuels). En outre, ce test ne nécessite pas l'utilisation d'un échelon comme entrée, mais plutôt un signal aléatoire.

Le test du rapport des déterminants instrumentaux est basé essentiellement sur les conditions de rang d'une matrice, dite *matrice d'information*, contenant les couples de mesures entrée-sortie. Pour cela, on considère avoir réalisé N mesures de l'entrée appliquée $u(i)$ et de la réponse en sortie $y(i)$, $i = 1, \dots, N$, la matrice d'information Q_i à l'étape i s'écrit :

$$Q_i = \frac{1}{N} \sum_{k=i}^{N-i} \begin{pmatrix} u(k) \\ u(k+1) \\ u(k-1) \\ u(k+2) \\ \vdots \\ u(k-i+1) \\ u(k+i) \end{pmatrix} \begin{pmatrix} y(k+1) & u(k+1) & \dots & y(k+i) & u(k+i) \end{pmatrix}. \quad (2.1)$$

Notez que Q_i est une matrice carrée de dimension $2i \times 2i$. La matrice d'information Q_{i+1} à l'étape $i+1$ est construite à partir de la matrice Q_i :

$$Q_{i+1} = \frac{1}{N} \sum_{k=i+1}^{N-(i+1)} \begin{pmatrix} \mathbf{u}(k) \\ \mathbf{u}(k+1) \\ \mathbf{u}(k-1) \\ \mathbf{u}(k+2) \\ \vdots \\ \mathbf{u}(k-i+1) \\ \mathbf{u}(k+i) \\ u(k-i) \\ u(k+i+1) \end{pmatrix} \begin{pmatrix} \mathbf{y}(k+1) & \mathbf{u}(k+1) & \dots & \mathbf{y}(k+i) & \mathbf{u}(k+i) & y(k+i+1) & u(k+i+1) \end{pmatrix},$$

avec Q_{i+1} une matrice carrée de dimension $(2i+2) \times (2i+2)$.

La structure de la matrice d'information est "emboîtée", car, par exemple, la matrice Q_2 contient tous les éléments de la matrice Q_1 :

$$Q_2 = \frac{1}{N} \sum_{k=2}^{N-2} \begin{pmatrix} \mathbf{u}(k)\mathbf{y}(k+1) & \mathbf{u}(k)\mathbf{u}(k+1) & u(k)y(k+2) & u(k)u(k+2) \\ \mathbf{u}(k+1)\mathbf{y}(k+1) & \mathbf{u}(k+1)\mathbf{u}(k+1) & u(k+1)y(k+2) & u(k+1)u(k+2) \\ u(k-1)y(k+1) & u(k-1)u(k+1) & u(k-1)y(k+2) & u(k-1)u(k+2) \\ u(k+2)y(k+1) & u(k+2)u(k+1) & u(k+2)y(k+2) & u(k+2)u(k+2) \end{pmatrix}$$

Le rapport des déterminants instrumentaux (RDI) est donné est par

$$RDI(i) = \frac{|Q_i|}{|Q_{i+1}|} \quad (2.2)$$

Pour chaque valeur de i , la procédure de détermination de l'ordre

- construit les matrices Q_i et Q_{i+1} définies par (2.1),
- et évalue le rapport des déterminants instrumentaux défini par (2.2),
- enfin, **on retient comme valeur de n (ordre du système) la valeur de i pour laquelle la valeur absolue du rapport $RDI(i)$ augmente rapidement pour la première fois.**

2.3 Estimation à l'aide d'une réponse indicielle

2.3.1 Gain statique

Le gain statique correspond au quotient entre l'amplitude de la réponse et l'amplitude de l'échelon appliqué en entrée (voir par exemple la figure 2.5). L'amplitude de la réponse s'évalue en faisant la différence entre la valeur initiale de la sortie notée y_{init} (avant que le système commence à réagir à l'échelon) et la valeur finale de la réponse notée y_{fin} .

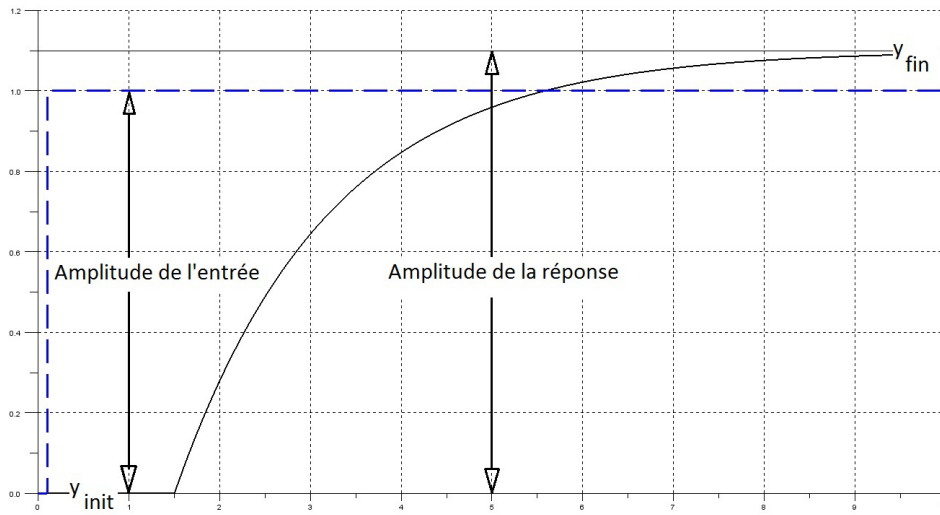


FIGURE 2.5 – Relevé des amplitudes pour l'estimation du gain statique.

2.3.2 Système d'ordre 1

Rappelons que la fonction de transfert d'un premier ordre s'écrit typiquement comme suit.

— En temps continu :

$$H(p) = K \frac{1}{1 + Tp}$$

— En temps discret (processus bloqué échantillonné) :

$$\begin{aligned} H(z) &= \frac{z-1}{z} \mathcal{Z} \left[\frac{H(p)}{p} \right] \\ &= K \frac{1-z_0}{z-z_0} \end{aligned}$$

avec $z_0 = e^{-\frac{\Delta}{T}}$ (la période d'échantillonnage Δ devant être choisie en respectant : $0,25T < \Delta < 1,25T$).

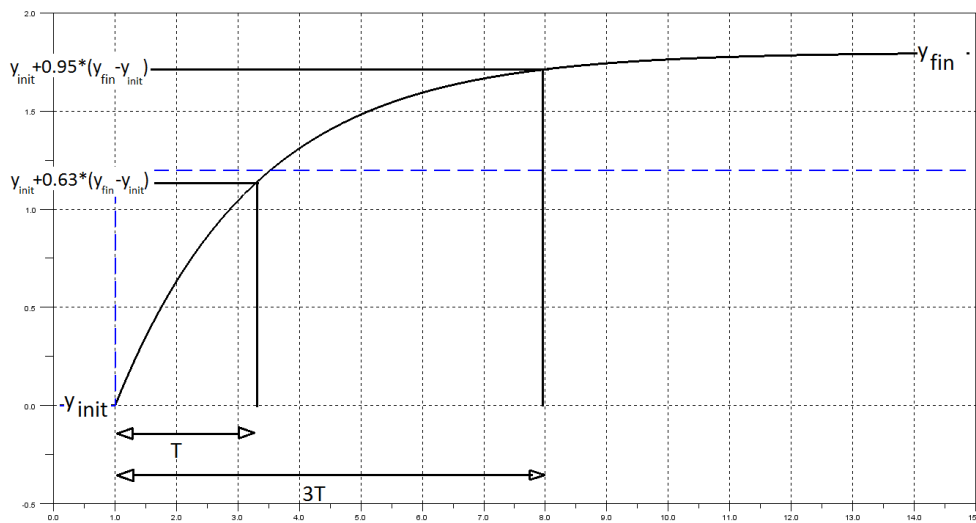


FIGURE 2.6 – Estimation à partir de la réponse indicielle

Dans le paragraphe précédent, on a précisé comment estimer le gain statique K . On peut estimer l'autre paramètre T , appelé la constante de temps, à partir de relevés sur la réponse indicielle comme, par exemple, indiqué sur la figure 2.6.

2.3.3 Système d'ordre 2

Rappelons que la fonction de transfert d'un système du deuxième ordre en temps continu s'écrit typiquement

$$H(p) = K \frac{1}{1 + 2\xi \frac{p}{\omega_n} + \left(\frac{p}{\omega_n}\right)^2}$$

avec

ξ	coefficient d'amortissement
ω_n	pulsation propre non amortie

Pour l'estimation de ces paramètres, on distingue deux cas :

- si le système est résonnant amorti ($0 < \xi < 1$), on obtient une réponse indicielle du type de celle représentée sur la figure 2.7, et en relevant le dépassement d et le temps de montée t_m (ou le temps de pic t_{pic}) sur la courbe, on peut faire une estimation des paramètres ξ et ω_n à l'aide des relations suivantes :

Dépassement (en pourcentage de la valeur finale)	$D\% = \frac{d \times 100}{\text{Amplitude de la réponse}} = 100e^{-\pi\xi/\sqrt{1-\xi^2}}$
Temps de montée	$t_m = \frac{1}{\omega_n \sqrt{1-\xi^2}} (\pi - \arccos \xi)$
Temps de pic	$t_{pic} = \frac{\pi}{\omega_n \sqrt{1-\xi^2}}$

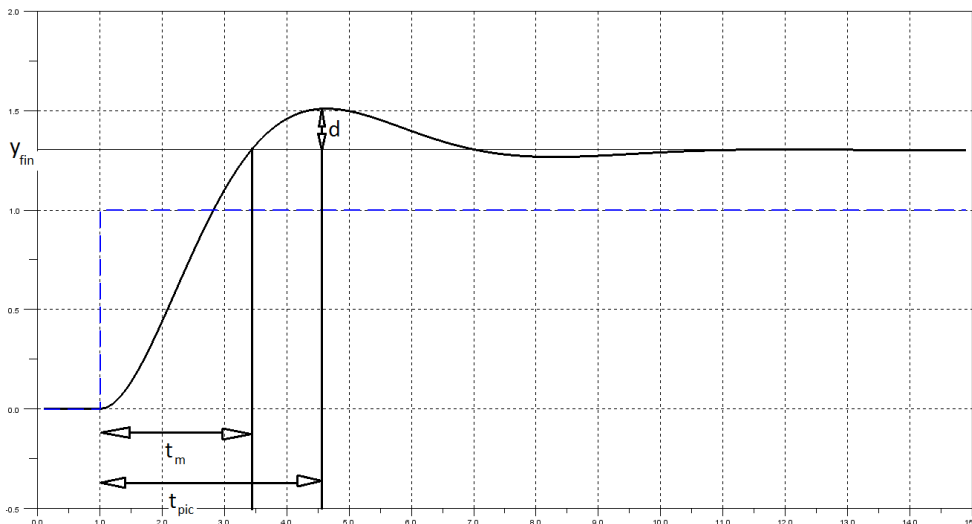
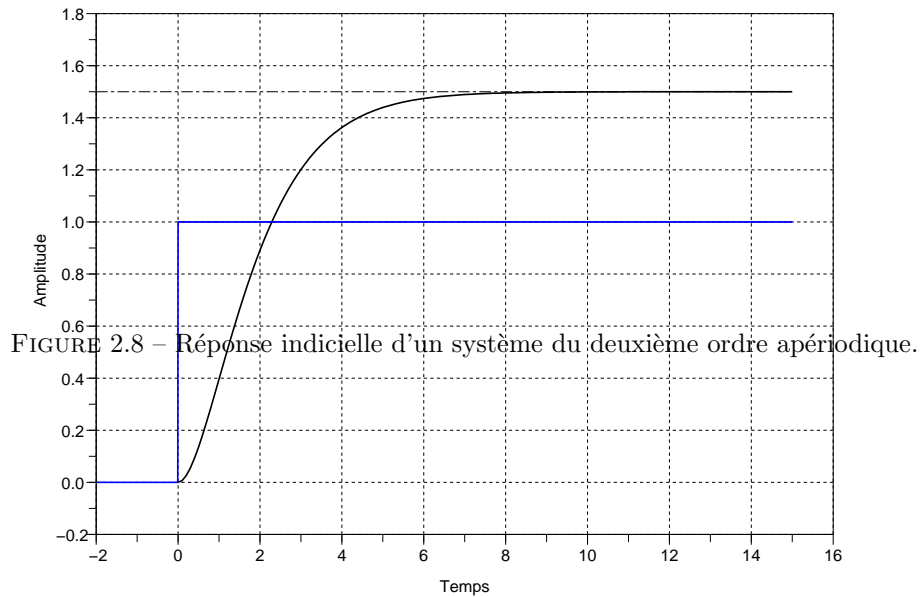


FIGURE 2.7 – Réponse indicielle d'un système du deuxième ordre résonnant amorti.

- si le système est apériodique ($\xi \geq 1$), on obtient une réponse indicielle du type de celle représentée sur la figure 2.8.



La fonction de transfert peut se mettre sous la forme :

$$H(p) = \frac{K}{(1 + T_1 p)(1 + T_2 p)},$$

avec

$$\xi = \frac{1}{2} \frac{T_1 + T_2}{\sqrt{T_1 T_2}} \text{ et } \omega_n = \frac{1}{\sqrt{T_1 T_2}}$$

On peut tracer la tangente au point d'inflexion pour obtenir une valeur approchée de T_1 et T_2 (voir la figure 2.9), et ainsi, on peut donc estimer ξ et ω_n .

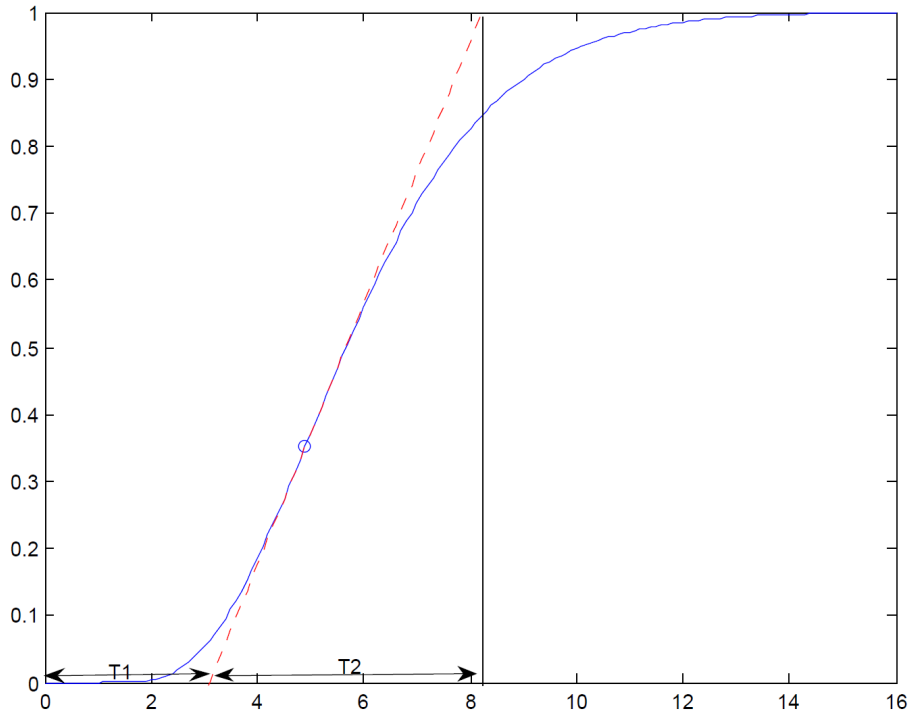


FIGURE 2.9 – Méthode pour obtenir les valeurs de T_1 et T_2 pour un deuxième ordre apériodique.

Rappelons qu'en temps discret, la fonction de transfert d'un deuxième ordre s'écrit typiquement

$$H(z) = \frac{b_1 z + b_0}{z^2 + a_1 z + a_0}$$

avec la période d'échantillonnage choisie telle que $0,25 < \omega_n \Delta < 1,25$ pour un processus bloqué échantillonné. On a les relations suivantes :

<p>si $\xi < 1$ en notant $\alpha = e^{-\xi \omega_n \Delta}$, $\omega_p = \omega_n \sqrt{1 - \xi^2}$</p>	<p>si $\xi \geq 1$ en notant $z_1 = e^{-\Delta/T_1}$, $z_2 = e^{-\Delta/T_2}$,</p>
<p>$a_0 = \alpha^2$</p> <p>$a_1 = -2\alpha \cos(\omega_p \Delta)$</p> <p>$b_0 = \alpha^2 + \alpha \left[\xi \frac{\omega_n}{\omega_p} \sin(\omega_p \Delta) - \cos(\omega_p \Delta) \right]$</p> <p>$b_1 = 1 - \alpha \left[\xi \frac{\omega_n}{\omega_p} \sin(\omega_p \Delta) + \cos(\omega_p \Delta) \right]$</p>	<p>$a_0 = z_1 z_2$</p> <p>$a_1 = -(z_1 + z_2)$</p> <p>$b_0 = K \left(z_1 z_2 - \frac{T_1 z_2 - T_2 z_1}{T_1 - T_2} \right)$</p> <p>$b_1 = K \left(\frac{1 - T_1 z_1 - T_2 z_2}{T_1 - T_2} \right)$</p>

2.3.4 Systèmes d'ordre supérieur à 2, méthode de *Strejc*

Comme pour la caractérisation, il peut s'avérer compliqué de réaliser l'estimation d'un système d'ordre supérieur à 2 en utilisant sa seule réponse indicielle (car celle-ci peut prendre des formes très diverses). En particulier, si cette réponse indicielle ne présente pas de dépassement, alors il existe des méthodes telles que celle dite de *Strejc* et celle dite de *Broïda* qui permettent de réaliser l'estimation. Elles peuvent apparaître comme des généralisations de la méthode proposée ci-dessus pour un deuxième ordre apériodique. Nous expliquons ici rapidement la méthode de *Strejc*.

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

$$H(p) = \frac{K}{(1 + Tp)^n}$$

Les paramètres à estimer sont donc :

- le gain statique K (en se référant au paragraphe ci-dessus),
- la constante de temps T ,
- et l'ordre n .

La méthode peut se décomposer selon les étapes suivantes :

1. On trace la tangente au point d'inflexion pour déterminer deux valeurs T_1 et T_2 (comme sur la figure 2.9)
2. On déduit l'ordre n de la valeur de $\frac{T_1}{T_2}$ en utilisant le tableau ci-dessous. Entre deux lignes du tableau, on choisit la valeur de n la plus petite.
3. On détermine la constante de temps T à partir de la valeur de $\frac{T_1}{T}$ ou de $\frac{T_2}{T}$ dans le tableau ci-dessous.

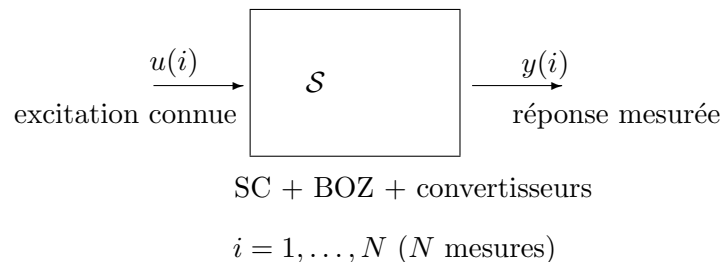
n	$\frac{T_1}{T}$	$\frac{T_2}{T}$	$\frac{T_1}{T_2}$
3	0,8	3,7	0,22
4	1,42	4,46	0,32
5	2,10	5,12	0,41
6	2,81	5,70	0,49

2.4 Estimation à l'aide d'une réponse à un signal aléatoire

Pour réaliser l'estimation des paramètres d'un modèle à l'aide d'une réponse à un signal aléatoire, on distingue deux types de procédures, dites "hors-ligne" et "en-ligne", qui peuvent être décrites comme suit.

1) Procédure hors-ligne

On considère l'expérimentation suivante :



u est choisie de manière à exciter le processus dans toute sa bande passante.

Typiquement, on applique une Séquence Binaire Pseudo-Aléatoire (SBPA) dont le spectre approche celui d'un bruit blanc (énergie équirépartie sur toutes les fréquences).

On souhaite construire un modèle \mathcal{M} à partir de ces données :

- (a) On a choisi une structure pour \mathcal{M} lors de la caractérisation. Il s'agira ici d'une équation aux différences (ou de façon équivalente la fonction de transfert en \mathcal{Z}) :

$$y_M(i+n) = -a_{n-1}y_M(i+n-1) - \dots - a_0y_M(i) + b_nu(i+n) + \dots + b_0u(i) \quad (2.3)$$

- (b) On souhaite maintenant établir le vecteur des paramètres

$$\theta = \begin{pmatrix} a_0 \\ \vdots \\ a_{n-1} \\ b_0 \\ \vdots \\ b_n \end{pmatrix} \quad (2.4)$$

tel que la sortie du modèle $y_M(i)$ soit proche de $y(i)$ pour $i = 1, \dots, N$, si on applique en entrée du modèle $u(i)$. Pour pouvoir qualifier la qualité des paramètres choisis pour le modèle, on va utiliser un critère J à minimiser. Il est défini à partir de l'erreur de sortie ε :

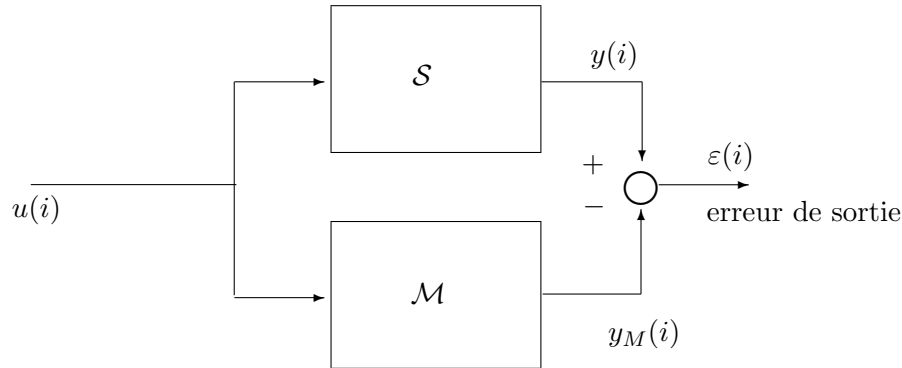


FIGURE 2.10 – Erreur de sortie

Et ce critère J est donc fonction des paramètres :

$$J(\theta) = f(\varepsilon).$$

Au paragraphe 2.4.1, nous présenterons une telle procédure d'estimation hors-ligne appelée *méthode des moindres carrés ordinaires* et bien connue pour son efficacité. Dans la littérature, on peut trouver d'autres méthodes hors-ligne d'estimation paramétrique, citons notamment *la méthode de maximum de vraisemblance* ou encore *la méthode du principe de pointage*.

I) Procédure en ligne

On peut aussi mettre en œuvre une procédure "en ligne" d'estimation des paramètres. Plus précisément, on estime de façon itérative, à chaque instant i les paramètres θ à partir des entrées appliquées et des mesures de la sortie jusqu'à cet instant. L'estimation se fait en quelque sorte au "fil de l'eau" et on a alors le schéma suivant :

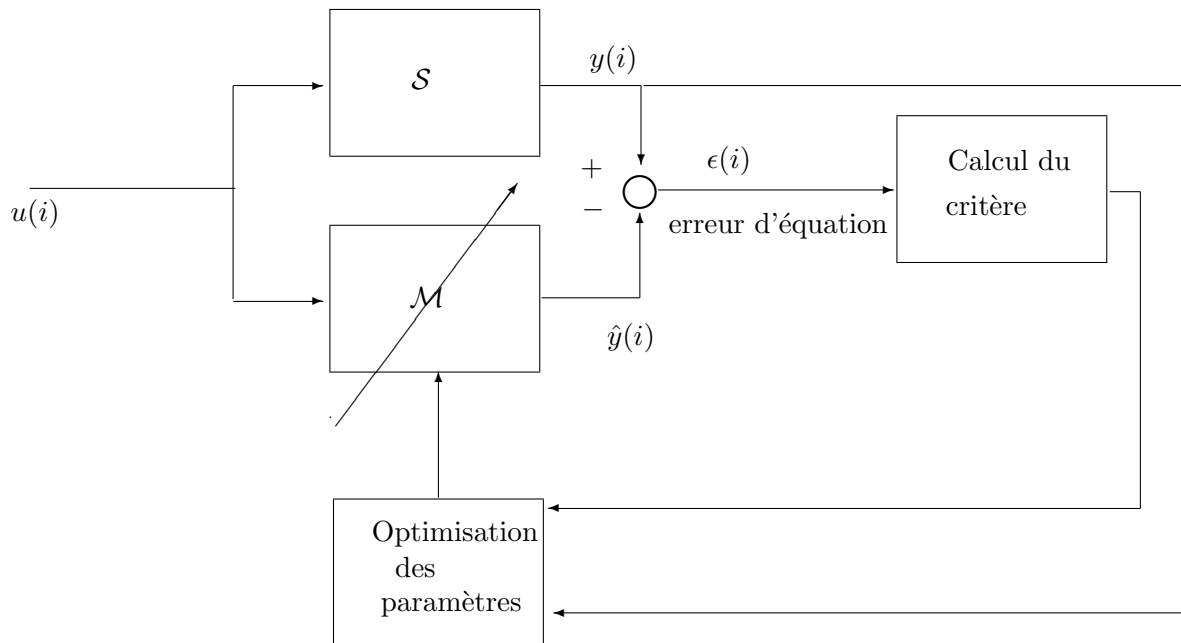


FIGURE 2.11 – Principe pour l'estimation "en ligne" des paramètres

Cette approche est la seule valable si l'identification est utilisée en contrôle adaptatif ou sur des processus variants avec le temps (*non stationnaires*) : on peut ainsi actualiser l'estimation des paramètres au fil de l'évolution du comportement

du processus.

Au paragraphe 2.4.2, nous présenterons la *méthode des moindres carrés récursifs* pour illustrer cette approche.

2.4.1 Méthode (hors-ligne) des moindres carrés ordinaires

Supposons que l'on ait N couples de mesures (entrées $u(i)$, sorties $y(i)$) du procédé. L'idéal serait que ces données soient telles que

$$y(i) = \hat{y}(i)$$

avec $\hat{y}(i)$ la sortie estimée par

$$\hat{y}(i+n) = -a_{n-1}y(i+n-1) - \dots - a_0y(i) + b_nu(i+n) + \dots + b_0u(i),$$

et $\theta^\top = (a_0 \dots a_{n-1} \ b_0 \dots b_n)$ les paramètres estimés.

De façon plus réaliste, on a

$$y(i) = \hat{y}(i) + \epsilon(i)$$

où $\epsilon(i) = y(i) - \hat{y}(i)$ est appelée l'*erreur d'équation*. On notera que pour l'erreur de sortie, $\epsilon(i)$, la sortie $y_M(i)$ est calculée à l'aide des sorties antérieures du modèle $y_M(i-1), \dots, y_M(i-n)$, alors que pour l'erreur d'équation, $\epsilon(i)$, la sortie estimée $\hat{y}(i)$ est calculée à l'aide des mesures antérieures $y(i-1), \dots, y(i-n)$. Pour l'ensemble des N mesures, on peut regrouper les équations sous la forme matricielle :

$$\underbrace{\begin{pmatrix} y(n+1) \\ y(n+2) \\ \vdots \\ y(N) \end{pmatrix}}_Y = \underbrace{\begin{pmatrix} -y(1) & \dots & -y(n) & u(1) & \dots & u(n+1) \\ -y(2) & \dots & -y(n+1) & u(2) & \dots & u(n+2) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ -y(N-n) & \dots & -y(N-1) & u(N-n) & \dots & u(N) \end{pmatrix}}_\Phi \underbrace{\begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_{n-1} \\ b_0 \\ \vdots \\ b_n \end{pmatrix}}_\theta + \underbrace{\begin{pmatrix} \epsilon(n+1) \\ \epsilon(n+2) \\ \vdots \\ \epsilon(N) \end{pmatrix}}_\epsilon$$

Pour estimer θ , nous choisissons un critère quadratique qui représente la somme des carrés des erreurs $\epsilon(i)$:

$$\begin{aligned} J(\theta) &= \sum_{i=n+1}^N \epsilon^2(i) \\ &= \epsilon^\top \cdot \epsilon \\ &= (Y - \Phi\theta)^\top \cdot (Y - \Phi\theta) \\ &= (Y^\top - \theta^\top \Phi^\top) \cdot (Y - \Phi\theta) \\ &= Y^\top Y - Y^\top \Phi\theta - \theta^\top \Phi^\top Y + \theta^\top \Phi^\top \Phi\theta \end{aligned}$$

On prend pour hypothèse que la matrice $\Phi^\top \Phi$ est inversible². L'expression de $J(\theta)$ peut alors se mettre sous la forme de deux termes dont l'un est indépendant de θ , à savoir :

$$J(\theta) = (\theta - (\Phi^\top \Phi)^{-1} \Phi^\top Y)^\top \Phi^\top \Phi (\theta - (\Phi^\top \Phi)^{-1} \Phi^\top Y) + Y^\top (\mathbf{I} - \Phi (\Phi^\top \Phi)^{-1} \Phi^\top) Y$$

A partir de cette expression, nous voyons que $J(\theta)$ est minimum si le nombre $(\theta - (\Phi^\top \Phi)^{-1} \Phi^\top Y)^\top \Phi^\top \Phi (\theta - (\Phi^\top \Phi)^{-1} \Phi^\top Y)^\top$ est nul, soit

$$\theta = \hat{\theta} = (\Phi^\top \Phi)^{-1} \Phi^\top Y \quad (2.5)$$

Méthodologie d'application de la méthode des moindres carrés ordinaires

On va suivre systématiquement la méthodologie suivante :

1. On commence par formuler le modèle choisi sous la forme d'une équation aux différences identique à l'équation (2.3), c'est-à-dire

$$y_M(i+n) = -a_{n-1}y_M(i+n-1) - \dots - a_0y_M(i) + b_nu(i+n) + \dots + b_0u(i)$$

Au prix d'un éventuel changement de variable, on formule donc l'équation aux différences de sorte à ce que l'itéré le plus ancien de u ait pour indice i .

2. Autrement dit $\Phi^\top \Phi$ est de rang plein et les vecteurs qui composent ses colonnes ne sont pas colinéaires. Notez que cette hypothèse n'est notamment pas vérifiée si les valeurs de l'entrée $u(i)$; $i = 1, 2, \dots, N$ sont constantes. Cela implique que la méthode ne peut en particulier pas être appliquée à partir d'un jeu de données expérimentales correspondant à une réponse indicielle du système.

2. Dans cette équation, on identifie n .
3. On construit les matrices Y et Φ .
4. On estime les paramètres à partir de la formule $\hat{\theta} = (\Phi^T \Phi)^{-1} \Phi^T Y$.
5. On évalue le critère $J(\hat{\theta})$ pour valider ou remettre en question l'estimation.

2.4.2 Méthode (en-ligne) des moindres carrés récurrents

La méthode d'identification par moindres carrés ordinaires nécessite d'avoir effectué toutes les mesures avant de pouvoir estimer le vecteur des paramètres $\hat{\theta}$.

On présente ici son adaptation "on-line" pour ré-actualiser la valeur des paramètres après chaque nouvelle acquisition. Autrement dit, on se pose le problème suivant :

Ayant calculé un vecteur des paramètres $\hat{\theta}_j$ à partir de j mesures

$$\hat{\theta}_j = (\Phi_j^T \Phi_j)^{-1} \Phi_j^T Y_j$$

Comment, après une nouvelle mesure $y(j+1)$,
obtenir le nouveau vecteur $\hat{\theta}_{j+1}$, connaissant $\hat{\theta}_j$?

Cette méthode itérative est qualifiée de *récurrente* car elle permet de calculer $\hat{\theta}_{j+1}$ sans reprendre l'ensemble du calcul $(\Phi_{j+1}^T \Phi_{j+1})^{-1} \Phi_{j+1}^T Y_{j+1}$ mais en l'exprimant à l'aide de $\hat{\theta}_j$ et d'une correction prenant en compte la mesure supplémentaire $y(j+1)$.

Notons

$$\Phi_j = \begin{pmatrix} \varphi(n+1) \\ \varphi(n+2) \\ \vdots \\ \varphi(j) \end{pmatrix} = \begin{pmatrix} -y(1) & \dots & -y(n) & u(1) & \dots & u(n+1) \\ -y(2) & \dots & -y(n+1) & u(2) & \dots & u(n+2) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ -y(j-n) & \dots & -y(j-1) & u(j-n) & \dots & u(j) \end{pmatrix}$$

$$\Phi_{j+1} = \begin{pmatrix} \Phi_j \\ \varphi(j+1) \end{pmatrix}$$

$$Y_j = \begin{pmatrix} y(n+1) \\ y(n+2) \\ \vdots \\ y(j) \end{pmatrix}$$

et

$$Y_{j+1} = \begin{pmatrix} Y_j \\ y(j+1) \end{pmatrix}.$$

Le vecteur $\hat{\theta}_{j+1}$, en prenant en compte la $(j+1)$ -ième mesure, s'écrit :

$$\begin{aligned} \hat{\theta}_{j+1} &= (\Phi_{j+1}^T \Phi_{j+1})^{-1} \Phi_{j+1}^T Y_{j+1} \\ &= [\Phi_j^T \Phi_j + \varphi^T(j+1)\varphi(j+1)]^{-1} (\Phi_j^T Y_j + \varphi^T(j+1)y(j+1)) \end{aligned}$$

L'idée est de faire apparaître dans cette expression une relation entre $\hat{\theta}_{j+1}$ et $\hat{\theta}_j$. Des manipulations (assez "lourdes") permettent d'obtenir le résultat suivant :

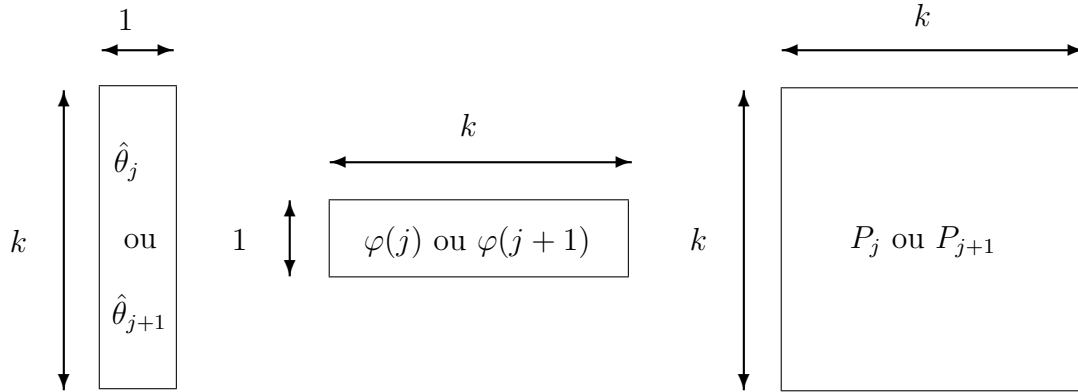
$$\hat{\theta}_{j+1} = \hat{\theta}_j + P_{j+1} \varphi^T(j+1) [y(j+1) - \varphi(j+1)\hat{\theta}_j] \quad (2.6)$$

avec

$$P_{j+1} = P_j - P_j \varphi^T(j+1) [\varphi(j+1)P_j \varphi^T(j+1) + 1]^{-1} \varphi(j+1)P_j \quad (2.7)$$

Remarques 2

- Notons $k = n + n + 1$, les dimensions des vecteurs et matrices mis en jeu sont :



On peut souligner que $\varphi(j+1)P_j\varphi^\top(j+1)$ est un scalaire, et que l'algorithme ne nécessite donc pas d'inversion matricielle (seulement des produits entre matrices, vecteurs ligne et vecteurs colonne).

- Dans l'équation (2.6), il apparaît bien que $\hat{\theta}_{j+1}$ est déduite de la valeur précédente $\hat{\theta}_j$ et d'un terme correctif qui tient compte de la nouvelle mesure.
- Il est possible d'introduire dans l'équation (2.7) un facteur d'oubli λ qui caractérise l'oubli progressif des mesures les plus anciennes :

$$P_{j+1} = \frac{P_j - P_j\varphi^\top(j+1)[\varphi(j+1)P_j\varphi^\top(j+1) + \lambda]^{-1}\varphi(j+1)P_j}{\lambda} \quad (2.8)$$

Ce facteur est utile lorsque l'on cherche à identifier un système dont les paramètres varient fortement au cours du temps (système dit non stationnaire). Typiquement, la valeur $\lambda = 0,95$ permet un oubli rapide et donc une poursuite d'une forte instationnarité. Classiquement, $0,95 \leq \lambda \leq 0,99$.

- Comme pour toute récurrence, il faut l'initialiser (valeurs de $\hat{\theta}_0$ et P_0). En l'absence d'information a priori sur les valeurs du vecteur des paramètres, on peut prendre :

$$\hat{\theta}_0 = \begin{pmatrix} 0 \\ \vdots \\ \vdots \\ 0 \end{pmatrix} \text{ et } P_0 = \alpha \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & 1 \end{pmatrix}$$

avec α très grand. Si on dispose d'informations sur $\hat{\theta}_0$, on prendra une valeur de α plus faible.

Chapitre 3

Représentation d'état des systèmes linéaires

3.1 Introduction et rappels

Dans la suite de ce support, on introduit des concepts et des résultats basés sur la représentation d'état des systèmes linéaires. Afin d'illustrer ce propos, un même exemple va être considéré tout au long du support.

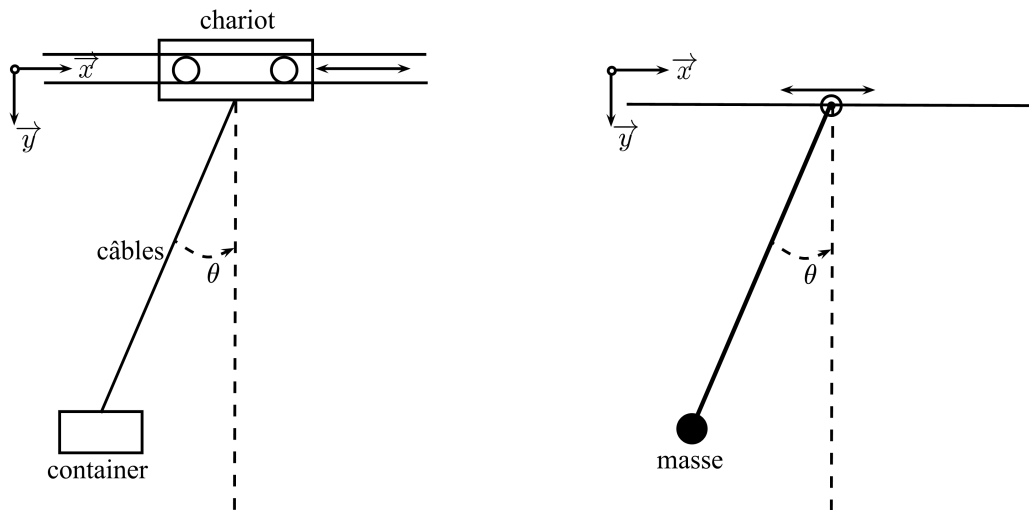
Exemple 1 *On s'intéresse au dispositif anti-ballant sur les grues portuaires telles que celles présentées sur la photo de la figure 3.1. Les conducteurs de grue doivent gérer le déplacement utile de la charge et maîtriser les balancements indésirables de celle-ci. Ce savoir-faire exige une grande expérience. Les industriels qui utilisent intensivement des grues cherchent à apporter une aide à la conduite de celles-ci, ou même à entièrement les automatiser. Décharger le grutier de cette tâche délicate qui consiste à compenser les balancements de la charge, c'est à coup sûr, augmenter la productivité et permettre au grutier de se concentrer sur l'aspect sécurité. C'est aussi normaliser les mouvements, éviter les fortes contraintes mécaniques et donc diminuer les coûts de maintenance. Divers fabricants ou équipementiers de grues ont déposé des brevets autour de cette problématique. Les grues portuaires sont principalement visées par ces techniques anti-ballant, grâce à la bonne connaissance du modèle mécanique (déplacement à longueur de câble fixe, suite de mouvements 1D, volume et forme des containers connus, ...).*



FIGURE 3.1 – Grue portuaire : portique à containers

On se focalise dans cet exemple sur le déplacement horizontal de l'ensemble chariot-câbles-container (le levage est considéré à l'arrêt). Plus précisément, on cherchera à asservir la vitesse du container tout en évitant son balancement (ballant). A des fins didactiques, on fait de plus les hypothèses simplificatrices suivantes (voir figure 3.2) :

- *L'ensemble chariot-câbles-container peut être modélisé comme un pendule dont le point d'accroche se déplace.*
- *On considère le repère (O, \vec{x}, \vec{y}) lié au bâti de la grue.*
- *Le chariot a une liaison glissière avec la flèche de la grue pour réaliser le mouvement de distribution.*
- *Compte tenu des accélérations mises en jeu, les câbles de masse négligeable, sont assimilés à un seul lien formant avec la charge un solide indéformable, en liaison pivot avec le chariot.*
- *Le container est assimilé à une masse ponctuelle placée au centre de gravité.*
- *Les frottements sont négligés.*



Schématisation de l'ensemble chariot-câble-container

Abstraction sous la forme d'un pendule

FIGURE 3.2 – Ensemble chariot-câbles-container étudié comme un pendule

Les grandeurs constantes dans le cadre de l'étude sont :

m	masse du container
L	longueur des câbles (considérés comme un solide indéformable) entre le point d'accroche et le centre de gravité du container
g	module de l'accélération de la pesanteur

Les variables (fonction du temps) associées à ce système seront :

p_m, v_m, a_m	respectivement position, vitesse et accélération de la masse (container) projetées selon l'axe \vec{x}
p_c, v_c, a_c	respectivement position, vitesse et accélération du chariot projetées selon l'axe \vec{x}
θ	angle entre les câbles et la verticale (orienté comme sur la figure 3.2 et compté positivement dans le sens trigonométrique)

On va prendre comme point de vue que :

- l'entrée du système est la vitesse horizontale du chariot v_c (à tout instant t , on peut fixer la valeur de $v_c(t)$);
- la sortie du système est la vitesse horizontale v_m de la masse (container).

On néglige les **perturbations (bruits)** qui peuvent intervenir sur le système : on pourrait notamment considérer le vent (grandeur extérieure subie qui a une influence sur le balancement du container). Comme déjà précisé précédemment, l'objectif sera d'asservir la vitesse de la masse tout en évitant le ballant.

Ce système est **déterministe** : une entrée v_c conduit à une seule sortie possible v_m .

Ce système est **causal** : la valeur de la sortie à un instant t_0 , $v_m(t_0)$, ne dépend pas du futur de l'entrée $v_c(t)$ pour $t > t_0$.

On peut utiliser les lois de la Physique pour effectuer une **modélisation** de ce système. On va notamment établir les équation différentielles qui régissent l'évolution des variables associées (pour disposer d'un modèle paramétrique) :

a) La définition du sinus dans un triangle rectangle nous donne que $p_c = p_m + L \sin \theta$. On peut en déduire une première relation liant la vitesse du chariot à celle de la masse :

$$v_c = v_m + L \frac{d \sin \theta}{dt}. \quad (3.1)$$

b) La masse se comporte comme un pendule. Le principe fondamental de la dynamique appliqué à la masse et projeté selon l'axe \vec{x} donne :

$$m a_m = T \sin \theta,$$

où \vec{T} est la traction de la masse sur les câbles. Puisque les câbles sont considérés comme indéformable, la force de traction \vec{T} et la composante du poids selon l'axe du câble s'annule, c'est-à-dire :

$$T = mg \cos \theta.$$

Ceci permet d'en déduire

$$ma_m = mg \sin \theta \cos \theta \quad (3.2)$$

Le système modélisé par les équations (3.1)-(3.2) est **stationnaire** puisque les valeurs des paramètres intervenant dans ces équations différentielles (à savoir m , g et L) sont supposées constantes au cours du temps. En revanche, il n'est **pas linéaire**. Pour disposer d'un modèle linéaire on va faire des approximations sur le comportement du système (le modèle sera moins précis mais plus aisé à manipuler car linéaire). On va notamment considérer que les variations de θ sont faibles et donc que $\cos \theta \approx 1$, $\sin \theta \approx \theta$. Ceci conduit à approcher les équations (3.1)-(3.2) sous la forme du modèle linéaire suivant :

$$\frac{d\theta}{dt} = -\frac{1}{L}v_m + \frac{1}{L}v_c \quad (3.3)$$

$$\frac{dv_m}{dt} (= a_m) = g\theta \quad (3.4)$$

Dans les deux paragraphes suivants, on rappelle les notions de stabilité et de précision qui nous seront utiles par la suite.

3.1.1 Stabilité

Un système linéaire est *stable* si pour toute entrée bornée, la sortie correspondante est bornée.

De manière équivalente, un système est *stable* si lorsque l'entrée s'annule, la sortie correspondante tend vers zéro lorsque t tend vers l'infini.

Théorème 1 (Condition de stabilité pour un système décrit par une FdT) Soit un système dynamique décrit par sa fonction de transfert $H(p)$. Il est stable si, et seulement si, tous les pôles de sa fonction de transfert $H(p)$ (c.-à-d., les racines du dénominateur de $H(p)$) sont à partie réelle strictement négative.

Exemple 2 On peut deviner que le système correspondant à l'ensemble chariot-câbles-container n'est pas stable. En effet, si on applique en entrée un échelon, puis que l'on annule l'entrée, alors les oscillations en sortie vont perdurer indéfiniment (les frottements sont négligés et l'amplitude des oscillations ne va donc pas diminuer) et la sortie ne va donc pas s'annuler asymptotiquement.

On vérifie que les racines du polynôme $1 + \frac{L}{g}p^2$ (dénominateur de la fonction de transfert) sont $\pm i\sqrt{\frac{g}{L}}$. Celles-ci ne sont pas à partie réelle strictement négative, ce qui confirme que le système est instable.

3.1.2 Précision

Un système asservi linéaire de sortie $y(t)$ est d'autant plus précis que la différence entre la sortie désirée $y_d(t)$ et la sortie réelle $y(t)$ est faible. La *précision* peut se mesurer par :

$$\varepsilon(t) \triangleq y_d(t) - y(t)$$

La *précision statique* ε_0 qualifie la valeur de cet écart $\varepsilon(t)$ quand t tend vers l'infini (régime permanent), c'est-à-dire :

$$\varepsilon_0 = \lim_{t \rightarrow \infty} \varepsilon(t).$$

On distingue parfois l'ordre de l'erreur stationnaire. Plus précisément, l'erreur stationnaire d'ordre n , notée ε_n , est relative à une entrée de la forme $U(p) = \frac{1}{p^n}$.

3.2 Introduction au formalisme d'état

3.2.1 Représentation d'état

Dans ce cours, on décrit les systèmes (processus physiques, biologiques, économiques,...) par des équations du type :

$$\begin{cases} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t) \end{cases} \quad (3.5)$$

sous l'hypothèse que le temps t est continu (c'est-à-dire qu'il prend ses valeurs dans \mathbb{R}).

- Le vecteur $u(t)$ est l'entrée (ou *commande*) du système. Sa valeur peut être choisie arbitrairement pour tout t .
- Le vecteur $y(t)$ est la *sortie* du système qui peut être mesurée.
- Le vecteur $x(t)$ est appelé *état* du système. Comme nous l'illustrerons par la suite, il représente la mémoire du système, c'est-à-dire l'ensemble des informations dont on a besoin pour prédire l'évolution du système connaissant l'entrée $u(t)$.
- Les matrices A , B , C et D ¹ sont appelées respectivement *matrices d'évolution*, *de commande*, *d'observation* et *directe*.
- La première des deux équations de (3.5) est appelée *équation d'évolution*. Il s'agit d'une équation différentielle qui permet de savoir vers où va se diriger l'état $x(t)$ sachant sa valeur à l'instant présent t et la commande $u(t)$ que nous sommes en train d'exercer.
- La deuxième équation s'appelle *équation d'observation*. Elle permet de calculer le vecteur de sortie $y(t)$ connaissant l'état et la commande à l'instant t . Contrairement à l'équation d'évolution, il ne s'agit pas d'une équation différentielle.
- Les équations (3.5) constituent la *représentation d'état* du système²
- Conventionnellement, on note n , m et p les dimensions respectives des vecteurs x , u et y . Les dimensions des matrices A , B , C et D sont donc respectivement $n \times n$, $n \times m$, $p \times n$ et $p \times m$.

Lorsque le système est étudié *via* un ordinateur, on ne s'intéresse pas continûment à son évolution, mais seulement en des instants *discrets* du temps (synchronisés sur l'horloge du processeur). Il est alors utile de considérer que le temps prend ses valeurs k dans \mathbb{Z} . On utilise alors la *représentation d'état en temps discret* sous la forme des équations de récurrence :

$$\begin{cases} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k) + Du(k) \end{cases} \quad (3.6)$$

À noter 1

Plusieurs remarques générales sur l'approche par représentation d'état :

- on étudie les systèmes dans le domaine temporel,
- les traitements font appel à l'algèbre matricielle,
- le formalisme s'applique naturellement aux systèmes multi-variables (plusieurs entrées et/ou plusieurs sorties).

3.2.2 Variables d'état

Rappelons que les systèmes linéaires en temps continu peuvent être représentés par leur réponse impulsionnelle³ h , et on a alors la relation entrée-sortie

$$\begin{aligned} y(t) &= (h * u)(t) \\ &= \int_0^t h(\tau)u(t - \tau)d\tau. \end{aligned} \quad (3.7)$$

En utilisant cette représentation, il apparaît que pour calculer la sortie à un instant t , on a besoin de connaître tout le passé (avant t) de l'entrée du système.

Si on considère cette fois un système linéaire du premier ordre représenté par son équation différentielle entrée-sortie :

$$\begin{cases} \dot{y}(t) - ay(t) &= bu(t) \\ y(0^+) &= y_0 \end{cases}$$

La solution de cette équation est

$$y(t) = y(0^+) \cdot e^{at} + \int_0^t b \cdot e^{a(t-x)} \cdot u(x)dx$$

Il apparaît également que pour calculer la sortie à un instant t , on a besoin de connaître tout le passé (avant t) de l'entrée du système.

1. Par la suite, il sera très souvent considéré que la matrice D est nulle.
 2. On se restreint aux systèmes linéaires, c'est-à-dire ceux pour lesquels la réponse à la somme pondérée de plusieurs excitations est égale à la somme pondérée des réponses à chacune des excitations prise séparément.
 3. La fonction de transfert est la transformée de Laplace de la réponse impulsionnelle, et on a alors la relation entrée-sortie

$$Y(p) = H(p) \cdot U(p)$$

Au contraire, à partir de sa représentation d'état, la sortie à l'instant t d'un même système linéaire peut être directement déduite de la connaissance de l'état à t . En effet, l'équation de sortie dans (3.5) donne $y(t)$ comme une fonction linéaire des composantes du vecteur d'état $x(t)$. En quelque sorte, la valeur du vecteur d'état résume tout le comportement passé du système.

Physiquement, la notion d'état peut être intuitée au travers des exemples suivants.

Exemple 3 *Considérons l'exemple d'un solide poussé sur une surface horizontale. Si on considère comme entrée du système la force extérieure qui pousse le solide. On peut se convaincre que pour deviner la position future de celui-ci, il n'est pas nécessaire de connaître depuis l'origine des temps la valeur de la force appliquée. La connaissance de la force à l'instant présent est suffisante si on dispose en outre des valeurs actuelles de la position et de la vitesse -ce qui traduit l'énergie cinétique du système.*

Exemple 4 *Considérons l'exemple d'un circuit RLC. Si on considère comme entrée du système la tension $u(t)$ délivrée par l'alimentation, et comme sortie le courant $i(t)$ dans le circuit. On peut se convaincre que pour déterminer l'évolution future du courant $i(t)$, il n'est pas nécessaire de connaître depuis l'origine des temps la valeur de la tension $u(t)$ appliquée. On peut remplacer cette information par la valeur de la charge dans la capacité et la valeur du flux dans l'inductance à l'instant présent. Ces valeurs -qui déterminent l'énergie emmagasinée dans le circuit- constituent l'état du système : elles extraient de l'histoire passée du circuit l'information nécessaire pour la détermination du futur.*

Exemple 5 *Considérons l'exemple d'une masse attachée à un ressort et tirée par une force extérieure. Si on considère comme entrée du système la force qui tire la masse. On peut se convaincre que pour déterminer l'évolution de la position de la masse, il n'est pas nécessaire de connaître depuis l'origine des temps la valeur de la force appliquée. On peut remplacer cette information par la valeur de la position de la masse et de sa vitesse à l'instant présent.*

Exemple 6 *Considérons à nouveau l'exemple de la grue portuaire décrit dans l'exemple 1. Pour cet exemple, $u = v_c$ (vitesse du chariot) et $y = v_m$ (vitesse de la masse). On peut deviner que pour déterminer l'évolution de la vitesse de la masse, on peut se contenter de connaître cette vitesse (qui traduit l'énergie cinétique), l'angle θ (qui traduit l'énergie potentielle) et l'évolution future de la vitesse du chariot (et il n'est pas nécessaire de connaître sa valeur depuis l'origine des temps).*

On choisit donc $x = \begin{pmatrix} v_m \\ \theta \end{pmatrix}$. A partir des équations (3.3) et (3.4), on déduit une représentation d'état du système :

$$\begin{cases} \dot{x}(t) &= \begin{pmatrix} 0 & g \\ -\frac{1}{L} & 0 \end{pmatrix} x(t) + \begin{pmatrix} 0 \\ \frac{1}{L} \end{pmatrix} u(t) \\ y(t) &= \begin{pmatrix} 1 & 0 \end{pmatrix} x(t) \end{cases} .$$

À noter 2

- Les paramètres de description "des réservoirs d'énergie" du système constituent en général de bonnes variables d'état de celui-ci.
- Nous expliciterons plus encore dans la suite que le nombre de variables d'état (la dimension du vecteur d'état) correspond à l'ordre du système.

3.2.3 Représentations d'état semblables

Considérons un système représenté par ses équations d'état (3.5) en temps continu. Soit P une matrice carrée inversible⁴, et posons $\chi(t) = P^{-1}x(t)$. Si on remplace $x(t)$ par $P\chi(t)$ dans (3.5), on obtient

$$\begin{cases} P\dot{\chi}(t) &= AP\chi(t) + Bu(t) \\ y(t) &= CP\chi(t) + Du(t) \end{cases} ,$$

c'est-à-dire,

$$\begin{cases} \dot{\chi}(t) &= P^{-1}AP\chi(t) + P^{-1}Bu(t) \\ y(t) &= CP\chi(t) + Du(t) \end{cases} .$$

Si on pose maintenant $A' = P^{-1}AP$, $B' = P^{-1}B$, $C' = CP$ et $D' = D$, le système d'équations précédent s'écrit

4. On parle souvent de *matrice de changement de base* ou *matrice de passage*.

$$\begin{cases} \dot{\chi}(t) &= A'\chi(t) + B'u(t) \\ y(t) &= C'\chi(t) + D'u(t) \end{cases} \quad (3.8)$$

La forme de ces dernières équations est identique à celle adoptée pour une représentation d'état (3.5). En d'autres termes, on a ainsi obtenu une deuxième représentation d'état pour le même système.

À noter 3

- De façon complètement similaire, on peut obtenir une représentation d'état équivalente pour un système en temps discret décrit par (3.6) (les manipulations utilisées -multiplication par une matrice de changement de base et son inverse, changement de variables- sont indépendantes de l'ensemble de définition des variables).
- Un système possède autant de représentations d'état qu'il existe de matrice carrée inversible P .
- Le vecteur $\chi(t)$ est aussi un vecteur d'état du système. Les représentations sont équivalentes, mais les grandeurs physiques que traduisent les variables d'état sont différentes (sauf si P est la égale à la matrice identité). On peut comprendre cette observation à partir de l'exemple 4 : on obtient une représentation d'état équivalente si on choisit comme variables d'état du circuit RLC la charge et le courant plutôt que respectivement la tension au borne du condensateur (proportionnelle à la charge) et le flux dans l'inductance (proportionnel au courant).
- Les transformations d'état (via une matrice de changement de base) sont souvent utilisées pour : faciliter la résolution de certains problèmes ou calculs (par ex., évaluer A^n est plus aisé si A est sous une forme diagonale par blocs) ou favoriser l'étude de propriétés comme la stabilité, la commandabilité ou l'observabilité (que nous étudierons plus loin).

Remarque 1 (Comment savoir si deux représentations d'état sont équivalentes ?)

Supposons que nous disposions de deux représentations d'état données par (3.5) et (3.8) censées représenter un même système. La matrice de changement de base P qui relie ces deux représentations peut être obtenue en résolvant le système linéaire :

$$\begin{cases} PA' &= AP \\ PB' &= B \\ C' &= CP \end{cases}$$

où les n^2 inconnues sont les coefficients P_{ij} de la matrice P . La résolution d'un tel système linéaire est assez aisée, et si aucune solution n'existe, il est clair que les deux représentations ne sont pas équivalentes.

3.3 Représentation d'état des systèmes linéaires en temps continu

3.3.1 D'une représentation d'état vers une représentation entrée-sortie

Le but de ce paragraphe est de montrer comment obtenir la fonction de transfert d'un système linéaire décrit par sa représentation d'état. Dans ce but, appliquons la transformée de Laplace aux équations d'évolution et de sortie matricielles (3.5), ce qui revient à l'appliquer à chacune des équations temporelles, et ensuite écrire à nouveau la forme matricielle. Il vient en considérant que $x(0) = 0$

$$\begin{cases} pX(p) &= AX(p) + BU(p) \\ Y(p) &= CX(p) + DU(p) \end{cases}$$

La première équation peut s'écrire $pX(p) - AX(p) = BU(p)$, ou encore⁵ $(p\mathbf{I} - A)X(p) = BU(p)$ avec \mathbf{I} la matrice identité (matrice de la dimension de A avec des 1 sur la diagonale et des 0 partout ailleurs). Ainsi, on aboutit au système équivalent :

$$\begin{cases} X(p) &= (p\mathbf{I} - A)^{-1}BU(p) \\ Y(p) &= CX(p) + DU(p) \end{cases}$$

La notation $(p\mathbf{I} - A)^{-1}$ désigne la matrice inverse de $p\mathbf{I} - A$. Pour des petites dimensions, on peut envisager de la calculer "à la main" (voir par exemple les rappels d'algèbre linéaire en annexe). On peut également faire appel au logiciel Scilab⁶. On obtient donc

$$Y(p) = [C(p\mathbf{I} - A)^{-1}B + D]U(p).$$

La matrice $C(p\mathbf{I} - A)^{-1}B + D$ est la *matrice de transfert* du système. Dans le cas mono-variable (une seule entrée et une seule sortie), $C(p\mathbf{I} - A)^{-1}B + D$ est la *fonction de transfert*, c'est-à-dire une fraction de polynômes en p . En multipliant chaque membre de la précédente équation par le dénominateur de $C(p\mathbf{I} - A)^{-1}B + D$, puis en effectuant la transformée de Laplace inverse on obtient une équation différentielle entrée-sortie pour notre système.

5. L'écriture $pX(p) - AX(p) = (p - A)X(p)$ n'est pas correcte car p est un scalaire alors que A est une matrice.

6. Logiciel gratuit de calcul numérique qui intègre les opérations courantes du calcul matriciel. Il propose également des fonctionnalités limitées de calcul symbolique, qui peuvent précisément être utilisées pour calculer $(p\mathbf{I} - A)^{-1}$, voir l'exemple 8.

Exemple 7 Considérons à nouveau l'exemple de la grue portuaire décrit dans l'exemple 1 et dont on a donné une représentation d'état dans l'exemple 6. La fonction de transfert peut être obtenue à partir de la représentation d'état en appliquant la formule $H(p) = C(p\mathbf{I} - A)^{-1}B$. On donne plusieurs résultats de calculs intermédiaires pour l'application de cette formule :

$$p\mathbf{I} - A = \begin{pmatrix} p & -g \\ \frac{1}{L} & p \end{pmatrix}, \quad |p\mathbf{I} - A| = p^2 + \frac{g}{L}, \quad \text{com}(p\mathbf{I} - A) = \begin{pmatrix} p & -\frac{1}{L} \\ g & p \end{pmatrix}$$

et

$$(p\mathbf{I} - A)^{-1} = \frac{1}{|p\mathbf{I} - A|} \text{com}(p\mathbf{I} - A)^\top = \frac{1}{p^2 + \frac{g}{L}} \begin{pmatrix} p & g \\ -\frac{1}{L} & p \end{pmatrix}.$$

On obtient alors

$$H(p) = \frac{\frac{g}{L}}{p^2 + \frac{g}{L}} = \frac{1}{1 + \frac{L}{g}p^2}.$$

Exemple 8 On considère un système linéaire décrit par la représentation d'état

$$\begin{cases} \dot{x}(t) &= \begin{pmatrix} 1 & 3 \\ 2 & 0 \end{pmatrix} x(t) + \begin{pmatrix} 1 \\ 1 \end{pmatrix} u(t) \\ y(t) &= (0 \quad 1) x(t) \end{cases}.$$

La matrice $(p\mathbf{I} - A)^{-1}$ peut être calculée à partir de Scilab en tapant les lignes suivantes :

```
A=[1 3;2 0];
I=eye(2,2);
p=poly(0,'p');
(p*I-A)^(-1)
```

On obtient finalement l'équation différentielle entrée-sortie :

$$\ddot{y}(t) - \dot{y}(t) - 6y(t) = \dot{u}(t) + u(t)$$

3.3.2 D'une représentation entrée-sortie vers une représentation d'état

Dans ce paragraphe, nous examinons comment obtenir la représentation d'état à partir d'une représentation entrée-sortie d'un système mono-variable (une seule entrée et une seule sortie). Comme nous avons pu le remarquer dans la section 3.2.3, un système linéaire admet un ensemble de représentations d'état équivalentes. *A fortiori* partant d'une représentation entrée-sortie, il existe plusieurs méthodes qui permettent d'obtenir des représentations d'état équivalentes, chacune ayant une forme particulière. On se concentre dans ce paragraphe sur l'obtention de deux formes qui seront utiles par la suite.

Pour expliciter chacune de ces formes, on considère un système linéaire d'ordre 3 décrit de façon équivalente par l'équation différentielle suivante

$$\frac{d^3 y(t)}{dt^3} + a_2 \frac{d^2 y(t)}{dt^2} + a_1 \frac{dy(t)}{dt} + a_0 y(t) = b_2 \frac{d^2 u(t)}{dt^2} + b_1 \frac{du(t)}{dt} + b_0 u(t), \quad (3.9)$$

ou la fonction de transfert suivante⁷

$$H(p) = \frac{b_2 p^2 + b_1 p + b_0}{p^3 + a_2 p^2 + a_1 p + a_0}, \quad (3.10)$$

avec

$$Y(p) = H(p) \cdot U(p). \quad (3.11)$$

Forme canonique de commande

A partir de la fonction de transfert donnée par (3.10), on peut écrire l'équation (3.11) sous la forme du système d'équations

$$\begin{cases} X(p) &= \frac{1}{p^3 + a_2 p^2 + a_1 p + a_0} U(p) \\ Y(p) &= (b_2 p^2 + b_1 p + b_0) X(p) \end{cases}$$

ou encore

$$\begin{cases} p^3 X(p) + a_2 p^2 X(p) + a_1 p X(p) + a_0 X(p) &= U(p) \\ Y(p) &= b_2 p^2 X(p) + b_1 p X(p) + b_0 X(p) \end{cases}$$

7. Le passage d'une de ces représentations entrée-sortie à l'autre est supposée être maîtrisée par le lecteur.

L'objectif est alors d'introduire des variables supplémentaires de manière à obtenir un système d'équations mettant en jeu des polynômes en p de degrés au plus égaux à 1. Pour cela, posons $X_1(p) = X(p)$, $X_2(p) = pX_1(p)$, $X_3(p) = pX_2(p)$, et on obtient alors

$$\begin{cases} X_2(p) = pX_1(p) \\ X_3(p) = pX_2(p) \\ pX_3(p) + a_2pX_2(p) + a_1pX_1(p) + a_0X_1(p) = U(p) \\ Y(p) = b_2pX_2(p) + b_1pX_1(p) + b_0X_1(p) \end{cases}$$

ou encore

$$\begin{cases} X_2(p) = pX_1(p) \\ X_3(p) = pX_2(p) \\ pX_3(p) = -a_2X_3(p) - a_1X_2(p) - a_0X_1(p) + U(p) \\ Y(p) = b_2X_3(p) + b_1X_2(p) + b_0X_1(p) \end{cases} .$$

Ce système d'équations s'écrit sous la forme matricielle suivante

$$\begin{cases} \begin{pmatrix} pX_1(p) \\ pX_2(p) \\ pX_3(p) \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_0 & -a_1 & -a_2 \end{pmatrix} \cdot \begin{pmatrix} X_1(p) \\ X_2(p) \\ X_3(p) \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \cdot U(p) \\ Y(p) = (b_0 \quad b_1 \quad b_2) \cdot \begin{pmatrix} X_1(p) \\ X_2(p) \\ X_3(p) \end{pmatrix} \end{cases} .$$

La transformée de Laplace inverse des équations nous conduit à la forme, dite *canonique de commande*, de la représentation d'état. A savoir, en posant $x(t) = (x_1(t) \quad x_2(t) \quad x_3(t))^T$,

$$\begin{cases} \dot{x}(t) = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_0 & -a_1 & -a_2 \end{pmatrix} x(t) + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} u(t) \\ y(t) = (b_0 \quad b_1 \quad b_2) x(t) \end{cases} . \quad (3.12)$$

Forme canonique d'observation

A partir de la fonction de transfert donnée par (3.10), on peut écrire l'équation (3.11) sous la forme

$$p^3Y(p) + a_2p^2Y(p) + a_1pY(p) + a_0Y(p) = b_2p^2U(p) + b_1pU(p) + b_0U(p)$$

En divisant chaque membre par p^3 et en isolant $Y(p)$, nous obtenons

$$Y(p) = -\frac{a_2}{p}Y(p) - \frac{a_1}{p^2}Y(p) - \frac{a_0}{p^3}Y(p) + \frac{b_2}{p}U(p) + \frac{b_1}{p^2}U(p) + \frac{b_0}{p^3}U(p) ,$$

c'est-à-dire

$$Y(p) = \frac{1}{p} \left[(b_2U(p) - a_2Y(p)) + \frac{1}{p} \left[(b_1U(p) - a_1Y(p)) + \frac{1}{p} (b_0U(p) - a_0Y(p)) \right] \right] .$$

Posons

$$\begin{aligned} X_3(p) &= Y(p) \\ X_1(p) &= \frac{1}{p}(b_0U(p) - a_0Y(p)) \\ &= \frac{1}{p}(b_0U(p) - a_0X_3(p)) \\ X_2(p) &= \frac{1}{p}(b_1U(p) - a_1Y(p) + X_1(p)) \\ &= \frac{1}{p}((b_1U(p) - a_1X_3(p) + X_1(p)) \end{aligned}$$

On peut alors écrire l'équation précédente sous la forme du système d'équations suivant

$$\begin{cases} X_1(p) = \frac{1}{p}(b_0U(p) - a_0X_3(p)) \\ X_2(p) = \frac{1}{p}((b_1U(p) - a_1X_3(p)) + X_1(p)) \\ X_3(p) = \frac{1}{p}((b_2U(p) - a_2X_3(p)) + X_2(p)) \\ Y(p) = X_3(p) \end{cases} ,$$

c'est-à-dire

$$\begin{cases} pX_1(p) &= b_0U(p) - a_0X_3(p) \\ pX_2(p) &= b_1U(p) - a_1X_3(p) + X_1(p) \\ pX_3(p) &= b_2U(p) - a_2X_3(p) + X_2(p) \\ Y(p) &= X_3(p) \end{cases} .$$

Sous forme matricielle, on obtient

$$\begin{cases} \begin{pmatrix} pX_1(p) \\ pX_2(p) \\ pX_3(p) \end{pmatrix} &= \begin{pmatrix} 0 & 0 & -a_0 \\ 1 & 0 & -a_1 \\ 0 & 1 & -a_2 \end{pmatrix} \cdot \begin{pmatrix} X_1(p) \\ X_2(p) \\ X_3(p) \end{pmatrix} + \begin{pmatrix} b_0 \\ b_1 \\ b_2 \end{pmatrix} \cdot U(p) \\ Y(p) &= (0 \ 0 \ 1) \cdot \begin{pmatrix} X_1(p) \\ X_2(p) \\ X_3(p) \end{pmatrix} \end{cases} .$$

La transformée de Laplace inverse des équations nous conduit à la forme, dite *canonique d'observation*, de la représentation d'état. A savoir, en posant $x(t) = (x_1(t) \ x_2(t) \ x_3(t))^T$,

$$\begin{cases} \dot{x}(t) &= \begin{pmatrix} 0 & 0 & -a_0 \\ 1 & 0 & -a_1 \\ 0 & 1 & -a_2 \end{pmatrix} x(t) + \begin{pmatrix} b_0 \\ b_1 \\ b_2 \end{pmatrix} u(t) \\ y(t) &= (0 \ 0 \ 1) x(t) \end{cases} . \quad (3.13)$$

3.3.3 Simulation à partir de la représentation d'état

Dans ce paragraphe, nous allons présenter la méthode d'Euler pour effectuer une simulation sur ordinateur d'un système décrit par ses équations d'état (3.5). Cette méthode est plutôt approximative mais simple à comprendre et satisfaisante pour décrire les comportements de la plupart des systèmes.

Soit dt un nombre très petit devant les constantes de temps du système et qui correspond à la période d'échantillonnage de la méthode. L'équation d'évolution de (3.5) s'approxime⁸ par

$$\frac{x(t+dt)-x(t)}{dt} \simeq Ax(t) + Bu(t) \quad (3.14)$$

ou encore

$$x(t+dt) \simeq x(t) + Ax(t) \cdot dt + Bu(t) \cdot dt \quad (3.15)$$

On en déduit l'algorithme de simulation suivant :

```
x:=x0; t:=0; dt:=0.01;
répéter
  transmettre (saisir ou affecter) la valeur de u en entrée de l'ordinateur;
  y:=Cx+Du;
  sortir (afficher ou mémoriser) y;
  x:=x+A.x.dt+B.u.dt;
  attendre une interruption de l'échantillonneur;
  t=t+dt;
éternellement
```

L'échantillonneur produit une interruption périodique toutes les dt secondes. Ainsi, si l'ordinateur est suffisamment rapide, la simulation se déroule à la même vitesse que l'évolution de notre système physique. On parlera alors de simulation *temps réel*.

Dans certaines circonstances, ce qui nous intéresse est d'obtenir le résultat de la simulation le plus rapidement possible (par exemple pour prédire comment va se comporter un système dans le futur). On parle alors de simulation en *temps différé*. Dans ce cas, il n'est pas nécessaire de ralentir l'ordinateur pour le synchroniser avec notre temps physique, c'est-à-dire que l'on omet la ligne d'attente d'une interruption dans l'algorithme présenté ci-dessus.

8. Cette approximation peut être interprétée comme un développement de Taylor à l'ordre 1.

3.3.4 Solution des équations d'état

Dans ce paragraphe, nous nous attachons à calculer sous forme analytique la solution aux équations d'état (3.5).

Théorème 2 (Solution des équations d'état) Notons $x(t_0)$ l'état à l'instant initial t_0 . Pour un système linéaire à temps continu représenté par (3.5), l'état à l'instant t est donné par

$$x(t) = e^{A(t-t_0)}x(t_0) + \int_{t_0}^t e^{A(t-\tau)}Bu(\tau)d\tau, \quad (3.16)$$

et la sortie s'exprime

$$y(t) = Ce^{A(t-t_0)}x(t_0) + \int_{t_0}^t Ce^{A(t-\tau)}Bu(\tau)d\tau + Du(t). \quad (3.17)$$

L'équation (3.16) fournit l'état du système pour tout $t \geq t_0$ à partir de l'état initial $x(t_0)$, et de l'entrée $u(t)$ appliquée sur l'intervalle $[t_0, t]$.

— La fonction $Ce^{A(t-t_0)}x(t_0)$ est appelée *solution homogène libre* (ou *transitoire*).

— La fonction $\int_{t_0}^t Ce^{A(t-\tau)}Bu(\tau)d\tau + Du(t)$ est appelée *solution forcée*.

On renvoie le lecteur aux annexes de ce document pour un bref rappel sur les propriétés des exponentielles de matrices, ainsi que pour des méthodes de calcul.

3.3.5 Stabilité

Définition 1 (Stabilité) Un système linéaire est (asymptotiquement) stable si lorsque l'entrée s'annule, son état tend vers 0 lorsque t tend vers l'infini.

Supposons qu'une entrée non nulle soit appliquée à un système représenté par (3.5) depuis l'instant initial t_0 jusqu'à un instant t_1 . L'état à l'instant t_1 peut être évalué à partir de (3.16) :

$$x(t_1) = e^{A(t_1-t_0)}x(t_0) + \int_{t_0}^{t_1} e^{A(t_1-\tau)}Bu(\tau)d\tau.$$

Si l'entrée est nulle après t_1 , l'état pour $t \geq t_1$ s'écrit toujours à partir de (3.16) :

$$\begin{aligned} x(t) &= e^{A(t-t_1)}x(t_1) + \int_{t_1}^t e^{A(t-\tau)}Bu(\tau)d\tau \\ &= e^{A(t-t_1)}x(t_1), \end{aligned}$$

qui s'annule lorsque t tend vers l'infini si

$$\lim_{t \rightarrow \infty} e^{A(t-t_1)} = \mathbf{0}.$$

Une définition équivalente à la définition 1 s'énonce donc comme suit.

Définition 2 Un système linéaire est (asymptotiquement) stable si au bout d'un temps suffisamment long, l'état ne dépend plus des conditions initiales (quelles qu'elles soient).

En choisissant arbitrairement $t_1 = 0$, pour caractériser la stabilité on va chercher les conditions pour que

$$\lim_{t \rightarrow \infty} e^{At} = \mathbf{0}.$$

Théorème 3 (Critère de stabilité) Un système linéaire est stable si, et seulement si, toutes les valeurs propres de sa matrice d'évolution sont à parties réelles strictement négatives.

On renvoie le lecteur aux annexes de ce document pour un bref rappel sur les valeurs propres d'une matrice.

Définition 3 (Polynôme caractéristique) *Le polynôme caractéristique $P(p)$ d'un système linéaire représenté par (3.5) est défini comme étant le polynôme caractéristique de la matrice d'évolution A , c'est-à-dire*

$$P(p) = |p\mathbf{I} - A| \quad (3.18)$$

Les racines de $P(p)$ sont les valeurs propres de A .

Dans le cas mono-variable, on peut remarquer, en se référant au paragraphe 3.3.1, que $P(p)$ constitue le dénominateur de la fonction de transfert du système. Les racines de $P(p)$ correspondent donc alors aux pôles de la fonction de transfert.

Corollaire 1 (Critère de stabilité) *Un système linéaire est stable si, et seulement si, toutes les racines de son polynôme caractéristique sont à parties réelles strictement négatives.*

Exemple 9 *Considérons à nouveau l'exemple de la grue portuaire décrit dans l'exemple 1 et dont on a donné une représentation d'état dans l'exemple 6, à savoir*

$$\begin{cases} \dot{x}(t) &= \begin{pmatrix} 0 & g \\ -\frac{1}{L} & 0 \end{pmatrix} x(t) + \begin{pmatrix} 0 \\ \frac{1}{L} \end{pmatrix} u(t) \\ y(t) &= \begin{pmatrix} 1 & 0 \end{pmatrix} x(t) \end{cases} .$$

A partir de cette représentation d'état, on peut étudier la stabilité du système en calculant les valeurs propres de A qui sont aussi les racines de son polynôme caractéristique :

$$P(p) = |p\mathbf{I} - A| = \begin{vmatrix} p & -g \\ \frac{1}{L} & p \end{vmatrix} = p^2 + \frac{g}{L} .$$

Ses racines sont les imaginaires pures $\pm i\sqrt{\frac{g}{L}}$, et le système est donc instable. Cette conclusion corrobore celle de l'exemple 2.

3.3.6 Commandabilité

Il existe plusieurs définitions équivalentes pour la notion de *commandabilité* d'un système linéaire. Nous retenons, dans ce cours, la suivante.

Définition 4 (Commandabilité) *Un système linéaire représenté par les équations (3.5) est dit commandable si pour tout couple de vecteurs d'état (x_0, x_1) , on peut trouver un temps t_1 et une commande $u(t)$, $t \in [t_0, t_1]$, tel que le système, initialisé en x_0 à l'instant t_0 , atteigne l'état x_1 à l'instant t_1 .*

Notons que dans cette définition, on ne formule aucune hypothèse sur l'amplitude des signaux, et en particulier, la commande u peut être aussi énergétique que nécessaire. En pratique, il faut notamment prendre en compte les saturations des actionneurs (qui délivrent la commande).

Théorème 4 (Critère de commandabilité) *Un système linéaire représenté par les équations (3.5) est commandable si, et seulement si,*

$$\text{rang} \overbrace{(B|AB|A^2B|\dots|A^{n-1}B)}^{\Gamma_{com}} = n , \quad (3.19)$$

où n est la dimension de la matrice d'évolution A (ou, de façon équivalente, le nombre de variables d'état). En d'autres termes, il faut que la matrice Γ_{com} , dite de commandabilité, obtenue en juxtaposant les unes à côté des autres les n matrices $B, AB, \dots, A^{n-1}B$ soit de rang égal à la dimension de la matrice A .

Exemple 10 *Considérons à nouveau l'exemple de la grue portuaire décrit dans l'exemple 1 ayant pour représentation d'état*

$$\begin{cases} \dot{x}(t) &= \begin{pmatrix} 0 & g \\ -\frac{1}{L} & 0 \end{pmatrix} x(t) + \begin{pmatrix} 0 \\ \frac{1}{L} \end{pmatrix} u(t) \\ y(t) &= \begin{pmatrix} 1 & 0 \end{pmatrix} x(t) \end{cases} .$$

On a

$$\Gamma_{com} = (B \quad AB) = \begin{pmatrix} 0 & \frac{g}{L} \\ \frac{1}{L} & 0 \end{pmatrix}$$

Le déterminant de Γ_{com} est non nul, le rang de Γ_{com} est donc égal à 2 (la dimension de A), et on peut en conclure que le système est commandable. Cela signifie que partant d'une valeur quelconque de l'état (les variables d'état sont la vitesse de la masse et l'angle θ), on peut "amener" l'état à toute valeur souhaitée.

3.3.7 Observabilité

Définition 5 (Observabilité) *Un système linéaire représenté par les équations (3.5) est dit observable si la connaissance de $y(t)$ et de $u(t)$ pour $t \in \mathbb{R}$ nous permet de déterminer de façon unique l'état $x(t)$, pour tout t .*

Cette propriété revêt une importance pratique, quand certaines des variables d'état d'un système sont inaccessibles à la mesure. Si le système est observable, alors cela nous garantit l'existence d'un moyen (présenté plus après dans ce support) pour calculer les valeurs de ces variables d'état à partir de mesures de la sortie.

Théorème 5 (Critère d'observabilité) *Un système linéaire représenté par les équations (3.5) est observable si, et seulement si,*

$$\text{rang} \left(\overbrace{\begin{pmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{pmatrix}}^{\Gamma_{obs}} \right) = n, \quad (3.20)$$

où n est la dimension de la matrice d'évolution A (ou, de façon équivalente, le nombre de variables d'état). En d'autres termes, il faut que la matrice Γ_{obs} , dite d'observabilité, obtenue en empilant les unes au dessous des autres les n matrices C, CA, \dots, CA^{n-1} soit de rang égal à la dimension de la matrice A .

Exemple 11 *Considérons à nouveau l'exemple de la grue portuaire. On a*

$$\Gamma_{obs} = \begin{pmatrix} C \\ CA \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & g \end{pmatrix},$$

$|\Gamma_{obs}| = g \neq 0$ et donc $\text{rang}(\Gamma_{obs}) = 2 = \dim(A)$. Le système est donc observable. Cela signifie qu'à partir de la seule connaissance de l'entrée et de la sortie, il existe un moyen pour calculer les valeurs des variables d'état.

3.4 Représentation d'état des systèmes linéaires en temps discret

Dans ce chapitre, on se concentre sur les systèmes linéaires pouvant être représentés par les équations d'état

$$\begin{cases} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k) + Du(k) \end{cases} \quad (3.21)$$

où le temps, noté k , prend des valeurs discrètes (typiquement dans \mathbb{Z}).

3.4.1 Solution des équations d'état

Le théorème suivant fournit la solution analytique aux équations d'état (3.21).

Théorème 6 (Solution des équations d'état) *Notons $x(k_0)$ l'état à l'instant initial k_0 . Pour un système linéaire à temps discret représenté par (3.21), l'état à l'instant k est donné par*

$$x(k) = A^{k-k_0}x(k_0) + \sum_{l=k_0}^{k-1} A^{k-1-l}Bu(l), \quad (3.22)$$

et la sortie s'exprime

$$y(k) = CA^{k-k_0}x(k_0) + \sum_{l=k_0}^{k-1} CA^{k-1-l}Bu(l) + Du(k). \quad (3.23)$$

L'équation (3.22) fournit l'état du système pour tout $k \geq k_0$ à partir de l'état initial $x(k_0)$, et de l'entrée $u(k)$ appliquée sur l'intervalle $[k_0, k]$.

— La fonction $CA^{k-k_0}x(k_0)$ est appelée *solution homogène libre* (ou *transitoire*).

— La fonction $\sum_{l=k_0}^{k-1} CA^{k-1-l}Bu(l) + Du(k)$ est appelée *solution forcée*.

Le théorème 6 se prouve aisément par récurrence.

3.4.2 Stabilité

La définition 1 s'applique aux systèmes en temps discrets (en notant k le temps au lieu de t). Supposons qu'une entrée non nulle soit appliquée à un système représenté par (3.21) depuis l'instant initial k_0 jusqu'à un instant k_1 . L'état à l'instant k_1 peut être évalué à partir de (3.22) :

$$x(k_1) = A^{k_1 - k_0} x(k_0) + \sum_{l=k_0}^{k_1-1} A^{k_1-1-l} B u(l).$$

Si l'entrée est nulle après k_1 , l'état pour $k \geq k_1$ s'écrit toujours à partir de (3.22) :

$$\begin{aligned} x(k) &= A^{k-k_1} x(k_1) + \sum_{l=k_1}^{k-1} A^{k-1-l} B u(l) \\ &= A^{k-k_1} x(k_1) \quad (\text{car } u(l) = 0 \text{ pour } l \geq k_1), \end{aligned}$$

qui s'annule lorsque k tend vers l'infini si

$$\lim_{k \rightarrow \infty} A^{k-k_1} = \mathbf{0}.$$

La définition 2 est donc également équivalente pour la stabilité.

En choisissant arbitrairement $k_1 = 0$, pour caractériser la stabilité on va chercher les conditions pour que

$$\lim_{k \rightarrow \infty} A^k = \mathbf{0}.$$

Théorème 7 (Critère de stabilité) *Un système linéaire est stable si, et seulement si, toutes les valeurs propres de sa matrice d'évolution sont strictement dans le disque unité.*

On renvoie le lecteur aux annexes de ce document pour un bref rappel sur les valeurs propres d'une matrice.

Définition 6 (Polynôme caractéristique) *Le polynôme caractéristique $P(z)$ d'un système linéaire représenté par (3.21) est défini comme étant le polynôme caractéristique de la matrice d'évolution A , c'est-à-dire*

$$P(z) = |z\mathbf{I} - A| \quad (3.24)$$

Les racines de $P(z)$ sont les valeurs propres de A .

Corollaire 2 (Critère de stabilité) *Un système linéaire est stable si, et seulement si, toutes les racines de son polynôme caractéristique sont strictement dans le disque unité.*

3.4.3 Commandabilité

La *commandabilité* d'un système linéaire en temps discret se définit de la même façon que pour un système en temps continu (cf. définition 4 en notant k le temps au lieu de t).

Le critère de commandabilité est également le même : on appliquera donc aussi le théorème 4 pour caractériser la commandabilité d'un système en temps discret.

3.4.4 Observabilité

L'*observabilité* d'un système linéaire en temps discret se définit de la même façon que pour un système en temps continu (cf. définition 5 en notant k le temps au lieu de t).

Le critère d'observabilité est également le même : on appliquera donc aussi le théorème 5 pour caractériser l'observabilité d'un système en temps discret.

3.4.5 D'une représentation entrée-sortie vers une représentation d'état

Alors que les systèmes linéaires en temps continu peuvent être décrits par une équation différentielle entrée-sortie, les systèmes en temps discret admettent une relation entrée-sortie sous la forme d'une équation aux différences. Dans le cas mono-variable, on a une équation du type :

$$y(k+n) + a_{n-1}y(k+n-1) + \dots + a_0y(k) = b_m u(k+m) + b_{m-1}u(k+m-1) + \dots + b_0u(k) \quad (3.25)$$

Dans ce paragraphe, nous examinons comment obtenir la représentation d'état à partir d'une telle représentation entrée-sortie d'un système mono-variable. Comme nous avons pu le remarquer dans la section 3.2.3, un système linéaire admet un ensemble de représentations d'état équivalentes. *A fortiori* partant d'une représentation entrée-sortie, il existe plusieurs méthodes qui permettent d'obtenir des représentations d'état équivalentes, chacune ayant une forme particulière. On se concentre dans ce paragraphe sur l'obtention de deux formes qui seront utiles par la suite.

Forme canonique de commande

Choisissons une variable d'état $x_1(k)$ telle qu'elle vérifie

$$x_1(k+n) + a_{n-1}x_1(k+n-1) + \dots + a_0x_1(k) = u(k) \quad (3.26)$$

Puis on pose,

$$\begin{cases} x_2(k) = x_1(k+1) \\ x_3(k) = x_2(k+1) \\ \vdots \\ x_n(k) = x_{n-1}(k+1) \end{cases}$$

On a $x_n(k+1) = x_1(k+n)$, et à partir de (3.26), on vérifiera qu'alors

$$x_n(k+1) = -a_0x_1(k) - a_1x_2(k) - \dots - a_{n-1}x_n(k) + u(k)$$

Posons

$$x(k) = \begin{pmatrix} x_1(k) \\ x_2(k) \\ \vdots \\ x_n(k) \end{pmatrix}.$$

En écrivant ces équations sous forme matricielle, on aboutit à la représentation d'état :

$$\begin{cases} x(k+1) = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ -a_0 & -a_1 & -a_2 & \dots & -a_{n-1} \end{pmatrix} x(k) + \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix} u(k) \\ y(k) = (c_1 \ c_2 \ \dots \ c_n) x(k) + b_n u(k) \end{cases} \quad (3.27)$$

avec $c_i = b_{i-1} - b_n a_{i-1}$.

Forme canonique d'observation

Choisissons comme variables d'état

$$\begin{cases} x_1(k) = y(k) - b_n u(k) \\ x_2(k) = x_1(k+1) + a_{n-1}y(k) - b_{n-1}u(k) \\ \vdots \\ x_n(k) = x_{n-1}(k+1) + a_1y(k) - b_1u(k) \end{cases}$$

En posant

$$x(k) = \begin{pmatrix} x_1(k) \\ x_2(k) \\ \vdots \\ x_n(k) \end{pmatrix},$$

ceci nous conduit à la représentation d'état :

$$\begin{cases} x(k+1) = \begin{pmatrix} -a_{n-1} & 1 & 0 & \dots & 0 \\ -a_{n-2} & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -a_1 & 0 & 0 & \dots & 1 \\ -a_0 & 0 & 0 & \dots & 0 \end{pmatrix} x(k) + \begin{pmatrix} h_1 \\ h_2 \\ \vdots \\ h_{n-1} \\ h_n \end{pmatrix} u(k) \\ y(k) = (1 \ 0 \ \dots \ 0) x(k) + b_n u(k) \end{cases} \quad (3.28)$$

avec $h_i = b_{n-i} - b_n a_{n-i}$.

3.4.6 Étude de systèmes continus *via* un ordinateur : systèmes échantillonnés

Dans ce paragraphe, on aborde l'étude de processus continus *via* un ordinateur. Plus précisément, on se place dans le contexte suivant :

- On considère des processus dont la nature sous-tend que leurs variables d'état dépendent continûment du temps, et qui sont donc représentés par des équations d'état (3.5). Les signaux mis en jeu dans cette représentation sont fonction d'un temps continu (typiquement dans \mathbb{R}).
- On appréhende (identifie, analyse, commande,...) ces systèmes *via* un ordinateur : l'entrée u est générée par le ordinateur, et la mesure de la sortie y est lue par le ordinateur. Les signaux manipulés par le ordinateur sont par essence en temps discret (typiquement dans \mathbb{Z}).

La chaîne de traitement des signaux est schématisée sur la figure 3.3. Détaillons-la de la gauche vers la droite :

1. Le signal $u(k\Delta)$ est fourni par le ordinateur toutes les Δ secondes, où Δ est appelé la *période d'échantillonnage*. Il s'agit d'un nombre binaire.
2. $u(k\Delta)$ est traduit en une valeur analogique par un *convertisseur numérique-analogique* (CNA).
3. Le *bloqueur d'ordre zéro* reçoit en entrée la valeur analogique délivrée par le CNA. A sa sortie, il maintient cette valeur entre deux instants d'échantillonnage (le signal $u(t)$ a une valeur constante entre deux instants d'échantillonnage).
4. Le *processus continu* est sollicité par ce signal d'entrée $u(t)$ (en forme de marches d'escalier). Sa réponse en sortie est le signal continu $y(t)$ mesuré.
5. Ce signal analogique est échantillonné toutes les Δ secondes (rôle de l'*échantillonneur* ECH).
6. Le signal échantillonné est converti en une valeur binaire par le *convertisseur analogique-numérique* (CAN), qui est lue par le ordinateur.

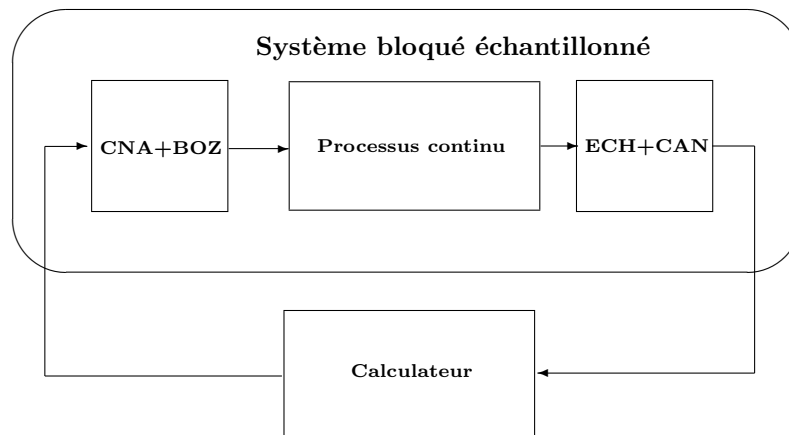


FIGURE 3.3 – Processus continu perçu comme un système échantillonné par le ordinateur

Discretisation d'une équation d'état en temps continu

On se limite à décrire les systèmes échantillonnés aux instants d'échantillonnage, ils se représentent donc comme des systèmes en temps discret. Dans ce paragraphe, on s'intéresse donc au problème suivant :

- On dispose d'une représentation d'état en temps continu du système :

$$\begin{cases} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t) \end{cases} \quad (3.29)$$

- Le système échantillonné peut être représenté aux instants d'échantillonnage $k\Delta$, que l'on note k pour alléger, par des équations d'état :

$$\begin{cases} x(k+1) &= A_{ech}x(k) + B_{ech}u(k) \\ y(k) &= C_{ech}x(k) + D_{ech}u(k) \end{cases} \quad (3.30)$$

- On souhaite obtenir les matrices A_{ech} , B_{ech} , C_{ech} et D_{ech} à partir des matrices connues A , B , C et D .

Partant de (3.29) et connaissant l'état à l'instant d'échantillonnage $t_k = k\Delta$, il est possible à partir de (3.16) d'établir l'état à l'instant d'échantillonnage suivant $t_{k+1} = (k+1)\Delta$:

$$x(t_{k+1}) = e^{A(t_{k+1}-t_k)}x(t_k) + \int_{t_k}^{t_{k+1}} e^{A(t_{k+1}-\tau)}Bu(\tau)d\tau. \quad (3.31)$$

Le BOZ maintient $u(t)$ à la valeur constante $u(t_k) = u(k\Delta)$, notée $u(k)$, pendant l'intervalle $[t_k, t_{k+1}[$, on a donc :

$$u(\tau) = u(k) \text{ pour } \tau \in [t_k, t_{k+1}[.$$

Effectuons le changement de variable $\nu = \tau - t_k$, et comme $\Delta = t_{k+1} - t_k$, l'équation (3.31) devient :

$$x(k+1) = e^{A\Delta}x(k) + \int_0^{\Delta} e^{A(\Delta-\nu)}d\nu Bu(k). \quad (3.32)$$

Par identification entre (3.32) et (3.29), on en déduit :

$$A_{ech} = e^{A\Delta} \quad (3.33)$$

$$B_{ech} = \int_0^{\Delta} e^{A(\Delta-\nu)}d\nu B \quad (3.34)$$

$$C_{ech} = C \quad (3.35)$$

$$D_{ech} = D \quad (3.36)$$

À noter 4

- Les matrices A_{ech} et B_{ech} dépendent de la période d'échantillonnage Δ , et devront donc être ré-évaluées à chaque fois que la période d'échantillonnage est modifiée.
- Le calcul de A_{ech} nécessite d'évaluer l'exponentielle d'une matrice. Des éléments pour ce calcul sont fournis en annexes. Dans la pratique, on pourra faire appel au logiciel Mupad pour un calcul formel ou numérique de A_{ech} . Le logiciel Scilab peut permettre aussi le calcul numérique de A_{ech} .
- Le calcul de B_{ech} nécessite en outre l'intégration d'une expression matricielle. L'intégrande, qui met en jeu une exponentielle de matrice, pourra d'abord être calculée (notamment à l'aide de Mupad ou Scilab). Ces éléments (il s'agit d'une matrice) pourront être intégrés un à un toujours à l'aide de Mupad, voire Scilab.

Choix de la période d'échantillonnage

Pour l'étude des systèmes échantillonnés, le choix de la période d'échantillonnage Δ est crucial car :

- si Δ est trop grand, l'ordinateur peut "rater" des informations importantes sur l'évolution du système. Entre deux échantillons, le système peut, par exemple, avoir oscillé et le calculateur n'aura pas connaissance de ce comportement.
- si Δ est trop petit, le calculateur peut être trop souvent sollicité. Si le système n'évolue que très peu entre deux échantillons successifs, l'information reçue n'apporte pas grand chose et le processeur du calculateur est alors inutilement sollicité.

Le choix de la période d'échantillonnage Δ dépend directement de la dynamique du système. La fréquence d'échantillonnage f_e ($f_e = \frac{1}{\Delta}$) doit en fait respecter le théorème de Shannon, à savoir :

$$f_e \geq 2f_h$$

où f_h est la fréquence la plus élevée à conserver dans le signal. En pratique, on applique souvent la règle suivante en Automatique :

$$5f_h < f_e < 25f_h.$$

D'une part, on "sur-échantillonne" par rapport à la borne que prescrit le théorème de Shannon. D'autre part, on cherche à limiter la fréquence d'échantillonnage pour ne pas solliciter trop souvent le calculateur. On a alors :

	Système du premier ordre	Système du second ordre
	$H(p) = \frac{K}{1+Tp}$	$H(p) = \frac{K}{1+\frac{2\xi}{\omega_n}p+\frac{1}{\omega_n^2}p^2}$
Avec	$f_h \approx \frac{1}{2\pi T}$	$f_h \approx \frac{\omega_n}{2\pi}$
On obtient	$0.25T < \Delta < 1.25T$	$0.25 < \Delta\omega_n < 1.25$

Exemple 12 Considérons à nouveau l'exemple de la grue portuaire décrit dans l'exemple 1 ayant pour représentation d'état

$$\begin{cases} \dot{x}(t) &= \begin{pmatrix} 0 & g \\ -\frac{1}{L} & 0 \end{pmatrix} x(t) + \begin{pmatrix} 0 \\ \frac{1}{L} \end{pmatrix} u(t) \\ y(t) &= \begin{pmatrix} 1 & 0 \end{pmatrix} x(t) \end{cases}.$$

On a d'autre part établi qu'il s'agissait d'un système du deuxième ordre dont la pulsation naturelle est donnée par $\omega_n = \sqrt{\frac{g}{L}}$.

La période d'échantillonnage Δ peut donc être choisie dans l'intervalle $]0.25\sqrt{\frac{L}{g}}, 1.25\sqrt{\frac{L}{g}}[$.

Nous calculons la représentation d'état du système échantillonné à la période Δ :

- On utilise la formule (cf. annexes du support) $e^{At} = \mathcal{L}^{-1} [(p\mathbf{I} - A)^{-1}]$. Notez que l'on a déjà calculé $(p\mathbf{I} - A)^{-1}$ dans l'exemple 9. On a⁹ :

$$e^{At} = \mathcal{L}^{-1} \left[\begin{pmatrix} \frac{p}{p^2+g/L} & \frac{g}{p^2+g/L} \\ -\frac{1/L}{p^2+g/L} & \frac{p}{p^2+g/L} \end{pmatrix} \right] = \begin{pmatrix} \cos(\sqrt{\frac{g}{L}}t) & \sqrt{gL} \sin(\sqrt{\frac{g}{L}}t) \\ -\frac{1}{\sqrt{gL}} \sin(\sqrt{\frac{g}{L}}t) & \cos(\sqrt{\frac{g}{L}}t) \end{pmatrix}$$

D'où

$$A_{ech} = \begin{pmatrix} \cos(\sqrt{\frac{g}{L}}\Delta) & \sqrt{gL} \sin(\sqrt{\frac{g}{L}}\Delta) \\ -\frac{1}{\sqrt{gL}} \sin(\sqrt{\frac{g}{L}}\Delta) & \cos(\sqrt{\frac{g}{L}}\Delta) \end{pmatrix}$$

- On calcule¹⁰ $B_{ech} = \int_0^\Delta e^{A(\Delta-\nu)} d\nu B$:

$$\begin{aligned} B_{ech} &= \int_0^\Delta \begin{pmatrix} \cos(\sqrt{\frac{g}{L}}\Delta - \sqrt{\frac{g}{L}}\nu) & \sqrt{gL} \sin(\sqrt{\frac{g}{L}}\Delta - \sqrt{\frac{g}{L}}\nu) \\ -1/\sqrt{gL} \sin(\sqrt{\frac{g}{L}}\Delta - \sqrt{\frac{g}{L}}\nu) & \cos(\sqrt{\frac{g}{L}}\Delta - \sqrt{\frac{g}{L}}\nu) \end{pmatrix} d\nu B \\ &= \left[\begin{pmatrix} -\sqrt{\frac{g}{L}} \sin(\sqrt{\frac{g}{L}}\Delta - \sqrt{\frac{g}{L}}\nu) & -L \cos(\sqrt{\frac{g}{L}}\Delta - \sqrt{\frac{g}{L}}\nu) \\ \frac{1}{g} \cos(\sqrt{\frac{g}{L}}\Delta - \sqrt{\frac{g}{L}}\nu) & -\sqrt{\frac{g}{L}} \sin(\sqrt{\frac{g}{L}}\Delta - \sqrt{\frac{g}{L}}\nu) \end{pmatrix} \right]_0^\Delta B \\ &= \begin{pmatrix} -1 + \cos(\sqrt{\frac{g}{L}}\Delta) & \\ \frac{1}{\sqrt{(gL)}} \sin(\sqrt{\frac{g}{L}}\Delta) & \end{pmatrix} \end{aligned}$$

- $C_{ech} = C = \begin{pmatrix} 1 & 0 \end{pmatrix}$

9. Pour rappel, la transformée de Laplace de :

- $\sin(\omega t)$ est $\frac{\omega}{p^2+\omega^2}$;
- $\cos(\omega t)$ est $\frac{p}{p^2+\omega^2}$.

10. Pour rappel, la dérivée de :

- $\frac{1}{\omega} \sin(\omega t + \phi)$ est $\cos(\omega t + \phi)$;
- $-\frac{1}{\omega} \cos(\omega t + \phi)$ est $\sin(\omega t + \phi)$.

3.5 Commande par retour d'état

Dans ce chapitre, nous allons étudier la conception de régulateurs pour les systèmes représentés par les équations d'état

$$\begin{cases} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) \end{cases} \quad (3.37)$$

en temps continu, ou

$$\begin{cases} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k) \end{cases} \quad (3.38)$$

en temps discret. Notons que nous considérons ici que la matrice directe D est nulle, ou encore que la sortie n'est pas directement fonction de l'entrée. En pratique, cette situation est courante.

La méthode de commande par retour d'état consiste en l'élaboration d'un signal de commande u à partir des variables d'état x_i , supposées dans un premier temps comme toutes mesurées (on accède en permanence à leurs valeurs). Cette hypothèse est peu réaliste, mais permet d'exposer directement et simplement le principe de la commande. Nous verrons dans le chapitre suivant que si certaines variables d'état ne sont pas accessibles, alors, sous l'hypothèse d'observabilité, il est possible de les déduire de la sortie à l'aide d'un *estimateur* ou *reconstructeur* d'état.

On se place donc dans un premier temps dans le cas où toutes les variables d'état sont accessibles, et on a alors la structure de commande représentée sur la figure 3.4 (on présente le cas en temps discret, on a l'analogie en temps continu) :

- La variable $e(k)$ est la consigne extérieure envoyée au système.
- La loi de commande appliquée au système est donnée par :

$$u(k) = e(k) - L \cdot x(k).$$

Elle tient compte de la consigne extérieure $e(k)$ et du comportement du système *via* $L \cdot x(k)$.

- La matrice L est la *matrice de régulation* que l'on va chercher à calculer.

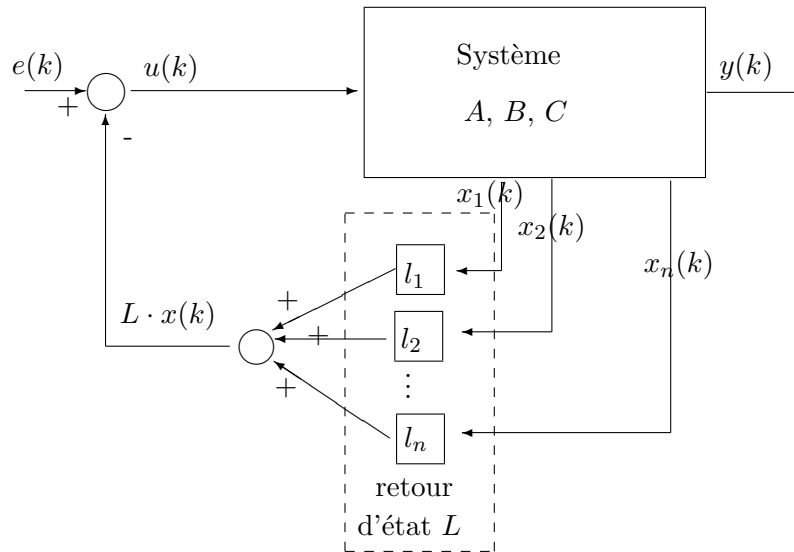


FIGURE 3.4 – Structure d'une commande par retour d'état

3.5.1 Commande par placement des pôles

En temps continu

Avec le retour d'état, la loi de commande appliquée au système est donnée par :

$$u(t) = e(t) - L \cdot x(t).$$

L'équation d'évolution du système bouclé s'écrit alors

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B(e(t) - Lx(t)) \\ &= (A - BL)x(t) + Be(t) \end{aligned} \quad (3.39)$$

Le principe du placement de pôles adopté ici consiste à choisir la matrice de régulation L de façon à imposer les pôles du système bouclé. En d'autres termes, on force la dynamique du système en vue de lui assurer une stabilité et une rapidité choisies.

Fixer les pôles du système bouclé est équivalent à imposer le polynôme caractéristique du système. Soit $P_{des}(p)$ le polynôme caractéristique désiré pour le système bouclé (on le choisit bien sûr de degré n), il nous faut résoudre l'équation polynomiale

$$|p\mathbf{I} - A + BL| = P_{des}(p) \quad (3.40)$$

dite de *placement de pôles*.

Cette équation peut être directement résolue sous Scilab grâce à la fonction `ppol()`.

Pour une réponse plus formelle, on se restreint au cas où le système possède une seule entrée¹¹. L'équation (3.40) peut se traduire en n équations scalaires avec n inconnues, et le problème admet alors une solution unique. Rappelons en effet que deux polynômes de degré n et unitaires

$$p^n + \gamma_{n-1}p^{n-1} + \dots + \gamma_0$$

et

$$p^n + \delta_{n-1}p^{n-1} + \dots + \delta_0$$

sont égaux si, et seulement si, leurs coefficients sont tous égaux, c'est-à-dire ssi

$$\gamma_0 = \delta_0, \gamma_1 = \delta_1, \dots, \gamma_{n-1} = \delta_{n-1}.$$

La résolution du problème est aisée quand la représentation d'état du système est sous la forme canonique de commande¹² (cf. § 3.2.3), c'est-à-dire quand A et B ont les formes suivantes

$$A = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & 1 \\ -a_0 & -a_1 & \dots & \dots & -a_{n-1} \end{pmatrix}, B = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix}.$$

Avec

$$L = (l_0 \quad l_1 \quad l_2 \quad \dots \quad l_{n-1}),$$

on obtient aisément

$$|p\mathbf{I} - A + BL| = p^n + (a_{n-1} + l_{n-1})p^{n-1} + \dots + (a_1 + l_1)p + (a_0 + l_0).$$

Les a_0, a_1, \dots, a_{n-1} sont des coefficients connus du modèle du système. Il suffit alors d'identifier les l_0, l_1, \dots, l_{n-1} pour que

$$|p\mathbf{I} - A + BL| = P_{des}(p),$$

c'est-à-dire

$$p^n + (a_{n-1} + l_{n-1})p^{n-1} + \dots + (a_1 + l_1)p + (a_0 + l_0) = p^n + \alpha_{n-1}p^{n-1} + \dots + \alpha_1p + \alpha_0,$$

où les $\alpha_0, \alpha_1, \dots, \alpha_{n-1}$ sont les coefficients choisis de $P_{des}(p)$. Ceux-ci traduisent les pôles choisis pour le système bouclé en vue de lui assurer des performances voulues. Autrement dit, la commande va permettre d'imposer que le système corrigé ait $P_{des}(p)$ pour dénominateur de sa fonction de transfert.

- Dans le cas d'un système du premier ordre, le dénominateur de la fonction de transfert s'écrit $1 + Tp$ ou encore $1/T + p$, et donc le choix de α_0 dans le polynôme $P_{des}(p)$ permettra de fixer la valeur désirée pour la constante de temps T (qui traduit la "rapidité" du système).
- Dans le cas d'un système du deuxième ordre, le dénominateur de la fonction de transfert s'écrit $1 + \frac{2\xi}{\omega_n}p + \frac{1}{\omega_n^2}p^2$ ou encore $\omega_n^2 + 2\xi\omega_n p + p^2$, et donc le choix de α_0 et α_1 dans le polynôme $P_{des}(p)$ permettra de fixer les valeurs désirées pour la pulsation naturelle ω_n et le coefficient d'amortissement ξ .

Pour que la commande soit physiquement réalisable, il faut que les coefficients l_0 et l_1 soient des nombres réels. Ils faut alors choisir α_0 et α_1 tels qu'ils soient des réels (les racines de $P_{des}(p)$ peuvent être des complexes conjugués).

Pour assurer la stabilité, il faut que ξ soit strictement positif (ce qui revient à choisir des racines à partie réelle

11. On renvoie le lecteur aux ouvrages cités en bibliographie pour le cas multi-entrées (plus complexe).

12. Notons que si le système est commandable, alors il est toujours possible de trouver une représentation d'état semblable (voir les ouvrages cités en bibliographie).

strictement négative pour $P_{des}(p)$. Si on souhaite que le système bouclé se comporte comme un deuxième ordre résonnant amorti ($0 < \xi < 1$), alors la valeur de ξ conditionne l'amplitude du dépassement.

Pour une valeur de ξ fixée, le choix de la pulsation naturelle va permettre de régler la "rapidité" du système (par exemple d'imposer la valeur du temps de réponse à 5%).

- Pour les systèmes d'ordres supérieurs ou égaux à 3, on peut mettre $P_{des}(p)$ sous la forme d'une factorisation de polynômes d'ordres 1 et 2, puis raisonner pour chacun des polynômes comme ci-dessus.

Exemple 13 *Considérons à nouveau l'exemple de la grue portuaire décrit dans l'exemple 1 ayant pour représentation d'état*

$$\begin{cases} \dot{x}(t) &= \begin{pmatrix} 0 & g \\ -\frac{1}{L} & 0 \end{pmatrix} x(t) + \begin{pmatrix} 0 \\ \frac{1}{L} \end{pmatrix} u(t) \\ y(t) &= \begin{pmatrix} 1 & 0 \end{pmatrix} x(t) \end{cases} .$$

Dans l'exemple ??, on a d'autre part établi qu'il s'agissait d'un système du deuxième ordre dont le coefficient d'amortissement est nul et la pulsation naturelle est donnée par $\omega_n = \sqrt{\frac{g}{L}}$. On a montré dans l'exemple 9 que ce système est instable.

On va appliquer une commande par retour d'état en vue d'imposer les pôles du système corrigé. Plus précisément, on souhaite que le système corrigé par retour d'état admette pour polynôme caractéristique :

$$P_{des}(p) = (p + 2)^2$$

On peut justifier ce choix par les explications suivantes.

a) *Les racines du polynôme caractéristique du système corrigé (en l'occurrence la racine double -2) seront alors toutes à partie réelle strictement négative, et on aura donc stabilisé le système.*

b) *Plus précisément, cela revient à imposer $p^2 + 4p + 4$, ou encore $\frac{1}{4}p^2 + p + 1$ comme dénominateur de la fonction de transfert. Par identification, cela revient à imposer :*

- $\xi = 1$ le système asservi sera sans dépassement.
- $\omega_n = 2$ et on peut prévoir le temps de montée pour une entrée de type échelon à l'aide de $t_m = \frac{1}{\omega_n \sqrt{1-\xi^2}} (\pi - \arccos \xi)$.

Déterminons la matrice de régulation L permettant d'obtenir $P_{des}(p)$ comme polynôme caractéristique du système corrigé. On veut résoudre le problème de placement de pôles :

$$|p\mathbf{I} - A + BL| = P_{des}(p).$$

$$p\mathbf{I} - A + BL = \begin{pmatrix} p & -g \\ \frac{1+l_1}{L} & p + \frac{l_2}{L} \end{pmatrix}$$

On obtient $|p\mathbf{I} - A + BL| = p^2 + p\frac{l_2}{L} + \frac{g+l_1g}{L}$, et par identification, nous déduisons

$$\begin{cases} l_1 &= \frac{4L}{g} - 1 \\ l_2 &= 4L \end{cases}$$

Sur la figure 3.5, on représenté la réponse indicielle du système (ensemble chariot-câbles-container) corrigé par ce retour d'état (on a pris $\frac{L}{g} = 0,125$, $L = 1,25$).

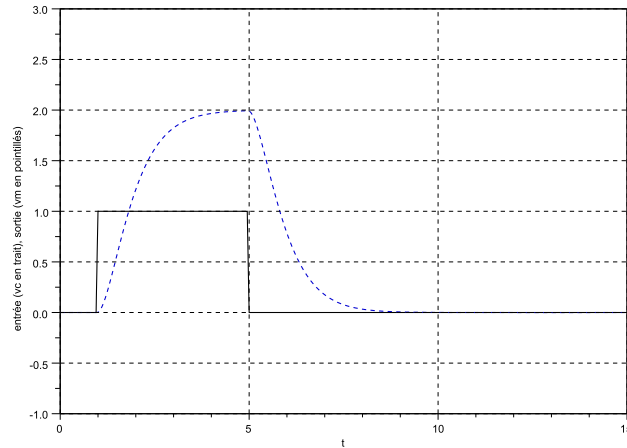


FIGURE 3.5 – Réponse indicielle du système (ensemble chariot-câbles-container) corrigé par retour d'état

En temps discret

La démarche est la même qu'en temps continu. Avec le retour d'état, la loi de commande appliquée au système est donnée par :

$$u(k) = e(k) - L \cdot x(k).$$

L'équation d'évolution du système bouclé s'écrit alors

$$\begin{aligned} x(k+1) &= Ax(k) + B(e(k) - Lx(k)) \\ &= (A - BL)x(k) + Be(k) \end{aligned} \quad (3.41)$$

Le principe du placement de pôles adopté ici consiste à choisir la matrice de régulation L de façon à imposer les pôles du système bouclé. Ce problème est équivalent à imposer le polynôme caractéristique du système. Soit $P_{des}(z)$ le polynôme caractéristique désiré pour le système bouclé (on le choisit bien sur de degré n), il nous faut résoudre l'équation polynomiale

$$|z\mathbf{I} - A + BL| = P_{des}(z) \quad (3.42)$$

dite de *placement de pôles*.

Cette équation peut être directement résolue sous Scilab grâce à la fonction `ppol()`.

Si on se restreint au cas où le système possède une seule entrée¹³, l'équation (3.42) peut se traduire en n équations scalaires avec n inconnues. Il y a alors une unique solution, et, exactement comme en temps continu, on peut l'explicitier directement si la représentation d'état du système est sous la forme canonique de commande (cf. § 3.4.5).

Le choix des coefficients de $P_{des}(z)$ va permettre de fixer le comportement voulu pour le système bouclé.

- Dans le cas d'un système du premier ordre, $P_{des}(z)$ est de la forme

$$P_{des}(z) = z + \alpha_0$$

où $\alpha_0 = -e^{-\frac{\Delta}{T}}$ et permet donc de fixer la valeur désirée pour la constante de temps T (qui traduit la "rapidité" du système).

- Dans le cas d'un système du deuxième ordre, $P_{des}(z)$ est de la forme

$$P_{des}(z) = z^2 + \alpha_1 z + \alpha_0$$

où $\alpha_1 = -2e^{-\xi\omega_n\Delta} \cos(\omega_n\sqrt{1-\xi^2}\Delta)$ et $\alpha_0 = e^{-2\xi\omega_n\Delta}$ pour un coefficient d'amortissement ξ choisi strictement inférieur à 1.

13. On renvoie le lecteur aux ouvrages cités en bibliographie pour le cas multi-entrées (plus complexe).

3.5.2 Régime statique du système bouclé

Dans cette section, on s'intéresse au comportement du système en *régime statique*, c'est-à-dire, lorsque son état n'évolue plus, ou encore lorsqu'une consigne e constante est considérée et que le temps (t ou k) tend vers l'infini. On va d'abord en déduire une expression du gain statique du système bouclé. Dans un deuxième temps, on sera en mesure de proposer une commande asservissant la sortie à une valeur non nulle.

En temps continu

Notons

$$e = \lim_{t \rightarrow \infty} e(t),$$

$$x = \lim_{t \rightarrow \infty} x(t),$$

$$y = \lim_{t \rightarrow \infty} y(t).$$

L'équation d'évolution du système bouclé (3.39) et l'équation de sortie en régime statique s'écrivent :

$$\begin{cases} 0 &= (A - BL)x + Be \\ y &= Cx \end{cases} \quad (3.43)$$

En régime statique, on a effectivement $\dot{x}(t) = 0$. La première équation permet d'obtenir

$$x = -(A - BL)^{-1}Be.$$

En reportant cette expression dans l'équation de sortie, on aboutit à

$$y = -C(A - BL)^{-1}Be. \quad (3.44)$$

En temps discret

L'équation d'évolution du système bouclé en temps discret s'écrit (*cf.* §3.5.1) :

$$x(k+1) = (A - BL)x(k) + Be(k)$$

Un régime statique est atteint dès que pour $k > K$, K correspondant à l'instant de début du régime statique, on a $x(k+1) = x(k)$. Ceci donne

$$x(k) = (A - BL)x(k) + Be(k),$$

ou encore

$$x(k) = (\mathbf{I} - A + BL)^{-1}Be(k).$$

En reportant cette expression dans l'équation de sortie, et en notant y et e les signaux $y(k)$ et $u(k)$ (constants en régime permanent), on aboutit à

$$y = C(\mathbf{I} - A + BL)^{-1}Be. \quad (3.45)$$

Obtention d'un gain unitaire vis-à-vis de la consigne

Des équations (3.44) et (3.45), on peut déduire directement la valeur du gain statique $\frac{y}{e}$, à savoir pour le système bouclé :

$$\begin{array}{ll} -C(A - BL)^{-1}B & \text{en temps continu,} \\ C(\mathbf{I} - A + BL)^{-1}B & \text{en temps discret.} \end{array}$$

Afin d'assurer un gain unitaire entre la consigne e et la sortie y , on peut adjoindre à la structure de commande un gain K_{re} de valeur égale à l'inverse de ces dernières. Celui-ci est placé en amont du retour d'état comme représenté sur la figure 3.6 en temps discret.

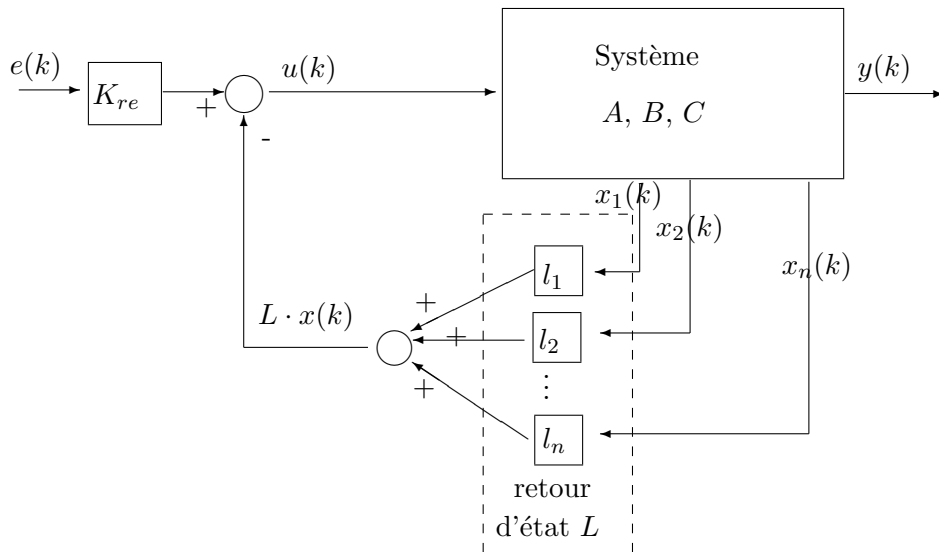


FIGURE 3.6 – Un gain est ajouté en amont du retour d'état

Exemple 14 Considérons à nouveau l'exemple de la grue portuaire décrit dans l'exemple 1 ayant pour représentation d'état

$$\begin{cases} \dot{x}(t) &= \begin{pmatrix} 0 & g \\ -\frac{1}{L} & 0 \end{pmatrix} x(t) + \begin{pmatrix} 0 \\ \frac{1}{L} \end{pmatrix} u(t) \\ y(t) &= \begin{pmatrix} 1 & 0 \end{pmatrix} x(t) \end{cases} .$$

Dans l'exemple 13, on a déterminé la matrice de régulation L permettant d'obtenir $P_{des}(p) = p^2 + 4p + 4$ comme polynôme caractéristique du système corrigé, à savoir :

$$L = \begin{pmatrix} \frac{4L}{g} - 1 & 4L \end{pmatrix}$$

Le gain (en régime statique) du système bouclé est $-C(A - BL)^{-1}B$. On a

$$A - BL = \begin{pmatrix} 0 & g \\ -\frac{4}{g} & -4 \end{pmatrix}, \text{ d'où } (A - BL)^{-1} = \begin{pmatrix} -1 & -\frac{g}{4} \\ \frac{1}{g} & 0 \end{pmatrix}$$

et

$$-C(A - BL)^{-1}B = \frac{g}{4L}.$$

Le gain à ajouter pour obtenir un gain unitaire du système bouclé est donc $K_{re} = \frac{1}{-C(A - BL)^{-1}B} = \frac{4L}{g}$.

Asservissement en sortie à une valeur constante non nulle

Dans le cas (peu réaliste) où le système ne subit aucune perturbation extérieure, l'objectif d'une commande est alors d'amener le système (et notamment ses sorties) à des points d'équilibre (des valeurs souhaitées en sortie, on parle généralement d'*asservissement*). Si l'objectif de l'asservissement est en particulier d'amener la sortie à une valeur constante y_s , le résultat obtenu pour le régime statique fournit directement la consigne à appliquer en entrée du système. En effet :

- en temps continu, on veut $y = \lim_{t \rightarrow \infty} y(t) = y_s$, et on cherche $e(t) = e_s H(t)$ adéquate ($H(t)$ désigne l'échelon d'Heaviside). L'équation (3.44) nous permet d'établir directement l'expression de e_s , à savoir

$$e_s = [-C(A - BL)^{-1}B]^{-1}y_s,$$

- en temps discret, on obtient d'après (3.45)

$$e_s = [C(\mathbf{I} - A + BL)^{-1}B]^{-1}y_s.$$

3.5.3 Ajout d'un effet intégral

En commande analogique classique, l'annulation de l'erreur statique en réponse à un échelon s'effectue *via* un correcteur intégral. Il est possible de mettre en œuvre une correction similaire dans l'espace d'état. On considère le système $\dot{x}(t) = Ax(t) + Bu(t) + p$ où p est un vecteur de perturbation inconnu et constant, censé représenter une perturbation extérieure qui n'a pu être prise en compte dans la modélisation. Un régulateur par retour d'état avec effet intégrale est de la forme

$$u(t) = \underbrace{L_i \int_0^t (e(t) - y(t)) dt}_{\text{effet intégrale}} - \overbrace{Lx(t)}^{\text{retour d'état}},$$

en supposant que le nombre de sortie(s) est égal au nombre d'entrée(s) (y et e sont de mêmes dimensions). Le système en boucle fermée peut donc s'écrire :

$$\begin{cases} \dot{x}(t) &= Ax(t) + Bu(t) + p \\ \dot{z}(t) &= e(t) - y(t) \\ u(t) &= L_i z(t) - Lx(t) \end{cases}.$$

Les équations d'état du système bouclé s'écrivent aussi

$$\begin{cases} \dot{x}(t) &= (A - BL)x(t) + BL_i z(t) + p \\ \dot{z}(t) &= e(t) - y(t) \end{cases}$$

ou, sous forme matricielle :

$$\begin{pmatrix} \dot{x}(t) \\ \dot{z}(t) \end{pmatrix} = \begin{pmatrix} A - BL & BL_i \\ -C & 0 \end{pmatrix} \begin{pmatrix} x(t) \\ z(t) \end{pmatrix} + \begin{pmatrix} p \\ e(t) \end{pmatrix}.$$

Puisque

$$\begin{pmatrix} A - BL & BL_i \\ -C & 0 \end{pmatrix} = \begin{pmatrix} A & 0 \\ -C & 0 \end{pmatrix} - \begin{pmatrix} B \\ 0 \end{pmatrix} (L \quad -L_i),$$

la matrice d'évolution du système bouclé est toujours de la forme $A' - B'L'$ (cf. eq. (3.39)), et le choix de L et L_i se fait alors aussi par placement de pôles (la fonction `ppol()` de Scilab peut notamment être utilisé).

3.6 Reconstruction de l'état

Au chapitre précédent, nous avons supposé que toutes les variables d'état du processus étaient mesurées pour élaborer la commande.

La plupart du temps, soit par impossibilité physique d'introduire un capteur, soit pour des questions de coût, on ne peut pas mesurer tous les états.

Nous allons voir comment on peut, à partir de mesures faites sur l'entrée et la sortie du processus, reconstruire le vecteur d'état complet x , noté alors \hat{x} . Le sous-système qui réalise cette reconstruction est appelé *reconstructeur d'état* (on parle aussi d'*observateur d'état*).

Pour la commande par retour d'état, on se place alors dans le contexte schématisé sur la figure 3.7. C'est l'état reconstruit \hat{x} qui est transmis à la matrice de régulation L pour élaborer le retour d'état, et la loi de commande à appliquer est

$$u = e - L\hat{x}.$$

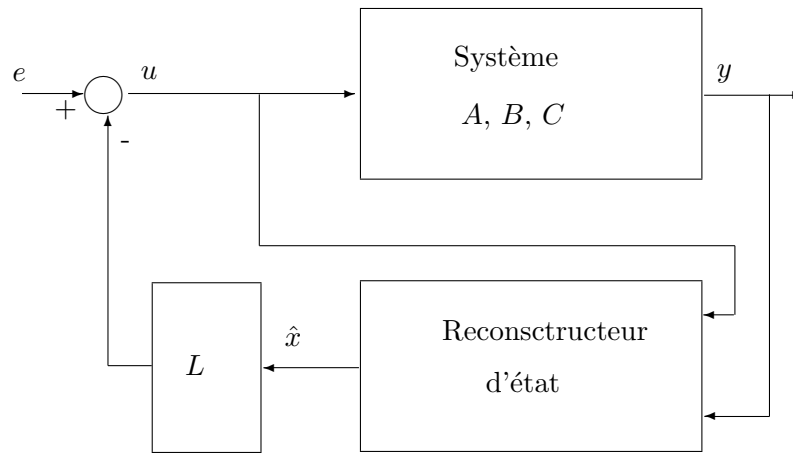


FIGURE 3.7 – Structure d'une commande par retour d'état avec un reconstruteur d'état

3.6.1 Principe du reconstruteur d'état

On considère un processus décrit par les matrices A , B et C de sa représentation d'état.

Si le processus est observable, on peut affirmer, qu'à partir des seules mesures de l'entrée et de la sortie, il est possible de calculer l'état du système (cf. §3.3.7). Le principe de la reconstruction est alors schématisé sur la figure 3.8.

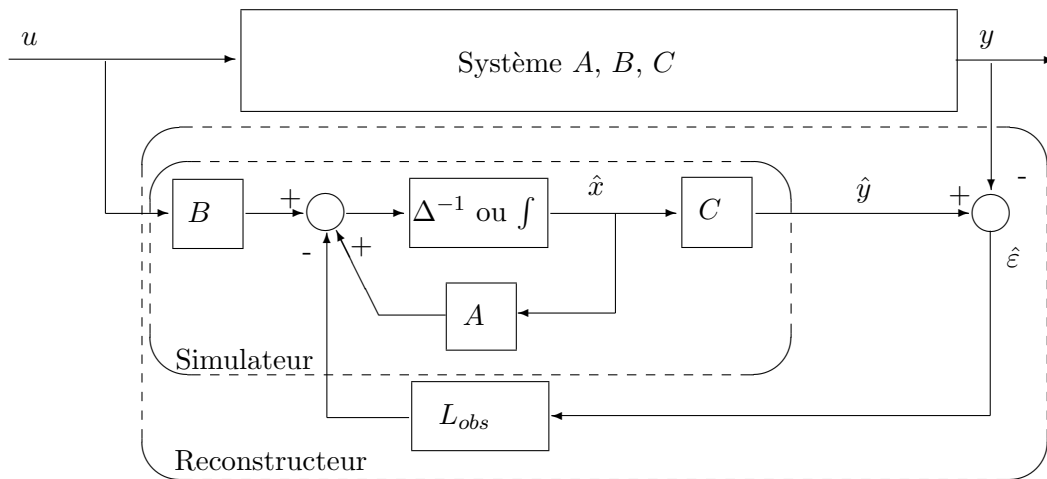


FIGURE 3.8 – Principe du reconstruteur d'état

Le dispositif intègre un *simulateur* de notre système. L'erreur ε entre la sortie du simulateur \hat{y} et la sortie du système y nous permet de venir corriger, par l'intermédiaire d'une matrice de correction L_{obs} , l'évolution de l'état estimé \hat{x} . Le simulateur corrigé s'appelle *reconstruteur*. Son rôle est de nous donner une bonne estimation du vecteur d'état x du système à transmettre à la matrice L de régulation du retour d'état.

3.6.2 Synthèse de la matrice L_{obs}

En temps continu

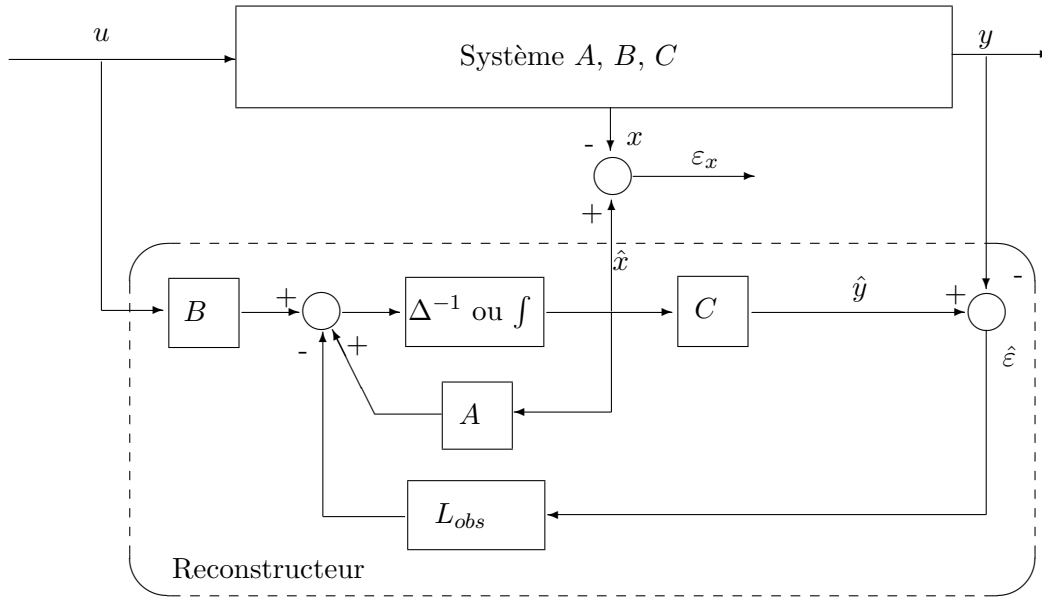


FIGURE 3.9 –

Afin de calculer L_{obs} , extrayons de la figure 3.9 le sous-système d'entrée $x(t)$ et de sortie $\varepsilon_x(t) = \hat{x}(t) - x(t)$. Cette sortie est non mesurable car elle dépend de x mais on l'abstrait pour le besoin des développements suivants. Elle ne servira qu'au calcul de L_{obs} et ne sera pas utilisée par le régulateur.

On a

$$\begin{cases} \dot{x}(t) &= Ax(t) + Bu(t) \\ \frac{d}{dt}(\hat{x}(t)) &= A\hat{x}(t) + Bu(t) - L_{obs}(C\hat{x}(t) - Cx(t)) \end{cases} .$$

d'où

$$\begin{aligned} \dot{\varepsilon}_x(t) = \frac{d}{dt}(\hat{x}(t) - x(t)) &= A\hat{x}(t) + Bu(t) - L_{obs}(C\hat{x}(t) - Cx(t)) - Ax(t) - Bu(t) \\ &= A(\hat{x}(t) - x(t)) - L_{obs}C(\hat{x}(t) - x(t)) \end{aligned} ,$$

ou encore

$$\begin{cases} \dot{x}(t) &= Ax(t) + Bu(t) \\ \dot{\varepsilon}_x(t) &= A\varepsilon_x(t) - L_{obs}C\varepsilon_x(t) \end{cases} .$$

En mettant sous forme matricielle, on aboutit à la représentation d'état suivante pour le système d'entrée $x(t)$ et de sortie $\varepsilon_x(t) = \hat{x}(t) - x(t)$:

$$\begin{cases} \begin{pmatrix} \dot{x}(t) \\ \dot{\varepsilon}_x(t) \end{pmatrix} = \begin{pmatrix} A & 0 \\ 0 & A - L_{obs}C \end{pmatrix} \begin{pmatrix} x(t) \\ \varepsilon_x(t) \end{pmatrix} + \begin{pmatrix} B \\ 0 \end{pmatrix} u(t) \\ \varepsilon_x(t) = (0 \quad \mathbf{I}) \begin{pmatrix} x(t) \\ \varepsilon_x(t) \end{pmatrix} \end{cases} . \quad (3.46)$$

Les pôles de ce système correspondent aux valeurs propres de A et de $A - L_{obs}C$.

La commande $u(t)$ n'intervient pas dans l'évolution du sous vecteur d'état $\varepsilon_x(t)$ (régi par $\dot{\varepsilon}_x(t) = (A - L_{obs}C)\varepsilon_x(t)$). L'erreur de reconstruction $\varepsilon_x(t)$ tend vers zéro si toutes les valeurs propres de $A - L_{obs}C$ sont à parties réelles strictement négatives (voir §3.3.5). Imposer la dynamique de cette erreur (notamment imposer la rapidité de convergence vers zéro) revient à résoudre

$$|p\mathbf{I} - A + L_{obs}C| = P_{obs}(p),$$

où $P_{obs}(p)$ est choisi. Puisque le déterminant d'une matrice est égal à celui de sa transposée, cette équation est équivalente à

$$|p\mathbf{I} - A^T + C^T L_{obs}^T| = P_{obs}(p), \quad (3.47)$$

et on obtient une équation du type placement de pôles (cf. equation (3.40)). La même méthode que celle exposée au paragraphe 3.5.1 s'applique donc pour obtenir L_{obs}^T et donc L_{obs} (on pourra notamment utiliser `ppol()` sous Scilab).

En temps discret

La démarche est identique à celle en temps continu. Pour le sous-système d'entrée $u(k)$ et de sortie $\varepsilon_x(k)$, on peut vérifier que l'on aboutit à la représentation d'état suivante

$$\begin{cases} \begin{pmatrix} x(k+1) \\ \varepsilon_x(k+1) \end{pmatrix} = \begin{pmatrix} A & 0 \\ 0 & A - L_{obs}C \end{pmatrix} \begin{pmatrix} x(k) \\ \varepsilon_x(k) \end{pmatrix} + \begin{pmatrix} B \\ 0 \end{pmatrix} u(k) \\ \varepsilon_x(k) = (0 \quad \mathbf{I}) \begin{pmatrix} x(k) \\ \varepsilon_x(k) \end{pmatrix} \end{cases} \quad (3.48)$$

On peut affirmer que l'erreur de reconstruction $\varepsilon_x(k)$ tend vers zéro si toutes les valeurs propres de $A - L_{obs}C$ sont toutes strictement dans le cercle unité. Là encore, on applique une technique de placement de pôles pour fixer les coefficients de L_{obs} .

3.6.3 Principe de séparation

On peut montrer que les pôles du système en boucle fermée avec un reconstructeur sont d'une part les pôles choisis pour la conception du régulateur, et d'autre part, les pôles choisis pour le reconstructeur. Cette indépendance nous permet bien de déterminer séparément la correction par retour d'état (L) et le reconstructeur (L_{obs}).

Rappels succincts d'algèbre linéaire

On considère une matrice carrée

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix}$$

Calcul du déterminant

Si A est de dimension 2×2

$$A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$$

On a

$$|A| = a_{11}a_{22} - a_{21}a_{12}$$

Pour les dimensions supérieures, on peut appliquer les propriétés suivantes

- on ne modifie pas un déterminant quand on ajoute à une ligne une combinaison linéaire des autres lignes,
 - quand on échange deux lignes d'un déterminant, celui-ci est changé en son opposé,
- pour réécrire le déterminant sous la forme :

$$|A| = \begin{vmatrix} a'_{11} & a'_{12} & \dots & a'_{1n} \\ 0 & a'_{22} & \dots & a'_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & a'_{n2} & \dots & a'_{nn} \end{vmatrix}$$

On a alors

$$|A| = a'_{11} \cdot \begin{vmatrix} a'_{22} & \dots & a'_{2n} \\ \vdots & \ddots & \vdots \\ a'_{n2} & \dots & a'_{nn} \end{vmatrix}$$

De façon plus générale, on peut appliquer un développement selon la colonne j de A comme suit :

$$|A| = \sum_{i=1}^n a_{ij} \text{Cof}_{ij}$$

avec

$$\text{Cof}_{ij} = (-1)^{i+j} M_{ij} \tag{3.49}$$

où M_{ij} représente le déterminant de la matrice résultant de la suppression de la ligne i et de la colonne j dans la matrice A , c'est-à-dire

$$M_{11} = \begin{vmatrix} a_{22} & \dots & a_{2n} \\ \vdots & \ddots & \vdots \\ a_{n2} & \dots & a_{nn} \end{vmatrix}, \dots, M_{1n} = \begin{vmatrix} a_{21} & \dots & a_{2n-1} \\ \vdots & \ddots & \vdots \\ a_{n1} & \dots & a_{nn-1} \end{vmatrix}$$

$$M_{n1} = \begin{vmatrix} a_{12} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n-12} & \dots & a_{n-1n-1} \end{vmatrix}, \dots, M_{nn} = \begin{vmatrix} a_{11} & \dots & a_{1n-1} \\ \vdots & \ddots & \vdots \\ a_{n-11} & \dots & a_{n-1n-1} \end{vmatrix}$$

Calcul de la matrice inverse

- La matrice A est inversible si, et seulement si, $|A| \neq 0$.
- Si A est inversible, alors

$$A^{-1} = \frac{1}{|A|}(\text{Com}A)^{\top}, \text{ avec } (\text{Com}A)_{ij} = \text{Cof}_{ij} \text{ (cf. equ. (3.49)).}$$

Valeurs propres d'une matrice

- On appelle *valeur propre* et *vecteur propre associé* d'une matrice carrée A , un scalaire λ et un vecteur v non nul tels que :

$$A \cdot v = \lambda v.$$

- Les valeurs propres d'une matrice carrée A sont les racines de l'équation :

$$|\lambda \mathbf{I} - A| = 0,$$

appelée *équation caractéristique* de A ; $|\lambda \mathbf{I} - A|$ est le polynôme caractéristique de A .

- Si A et D sont deux matrices telles que $P^{-1}AP = D$ (alors $AP = PD$) et si D est une matrice diagonale, alors multiplier P à droite par D revient à multiplier les colonnes de P par les coefficients diagonaux de D . Notons $P_{.i}$ la i -ième colonne de la matrice P et D_{ii} le i -ème coefficient diagonal de D . Pour tout $i = 1, \dots, n$, on a :

$$AP_{.i} = D_{ii}P_{.i}.$$

$P_{.i}$ est donc un vecteur propre de A associé à la valeur propre D_{ii} .

Exponentielles de matrice

On considère des matrices carrées M et N de dimension n . On note $\mathbf{0}$ et \mathbf{I} les matrices nulle et identité de dimension n .

- L'exponentielle e^M est de la même dimension que la matrice M .
- L'exponentielle d'une matrice répond aux mêmes règles de calcul que la fonction exponentielle, en particulier :

$$e^{\mathbf{0}} = \mathbf{I} \tag{3.50}$$

$$e^M \cdot e^N = e^{M+N} \tag{3.51}$$

$$\frac{d}{dt}(e^{Mt}) = Me^{Mt} \tag{3.52}$$

- L'exponentielle d'une matrice peut être calculée à partir de son développement en séries entières, à savoir :

$$e^M = \mathbf{I} + M + \frac{1}{2!}M^2 + \frac{1}{3!}M^3 + \dots = \sum_{i=0}^{\infty} \frac{1}{i!}M^i. \tag{3.53}$$

Lorsque les puissances M^i deviennent nulles ou négligeables, on peut évaluer l'expression (3.53) par une sommation finie de termes.

Le calcul se trouve simplifié si la matrice M est diagonale, car alors seules subsistent les puissances successives des éléments diagonaux.

- On peut également calculer l'exponentielle d'une matrice à partir d'une transformation de Laplace inverse, plus précisément :

$$e^{Mt} = \mathcal{L}^{-1}[(p\mathbf{I} - M)^{-1}]. \tag{3.54}$$

Cette méthode nécessite le calcul de l'inverse d'une matrice (les logiciels Mupad et Scilab pourront être utilisés) et sa transformation de Laplace inverse. Son avantage est fournir une expression analytique.

Rang d'une matrice et résolution d'un système linéaire

— On dit qu'une matrice carrée est de *rang plein* si son déterminant est non nul.

Le *rang* d'une matrice R , noté $\text{rang}(R)$, est la taille maximale des matrices carrées de rang plein extraites de R .

— Considérons le système

$$Ax = b$$

avec A une matrice carrée et b un vecteur colonne avec autant de lignes que A .

Si A est de rang plein, alors le système admet une unique solution donnée par $x = A^{-1}b$.

— Considérons le système

$$Rx = b$$

avec R une matrice rectangulaire de p lignes et q colonnes, et b un vecteur-colonne de p lignes.

Cas $p < q$: si R est de rang p , alors le système admet une infinité de solutions.

Cas $p > q$: si R est de rang p , alors le système admet au plus une solution. Si cette dernière existe, elle est donnée par $x = (R^T R)^{-1} R^T b$.

Rappels sur les systèmes du deuxième ordre¹⁴

En temps continu

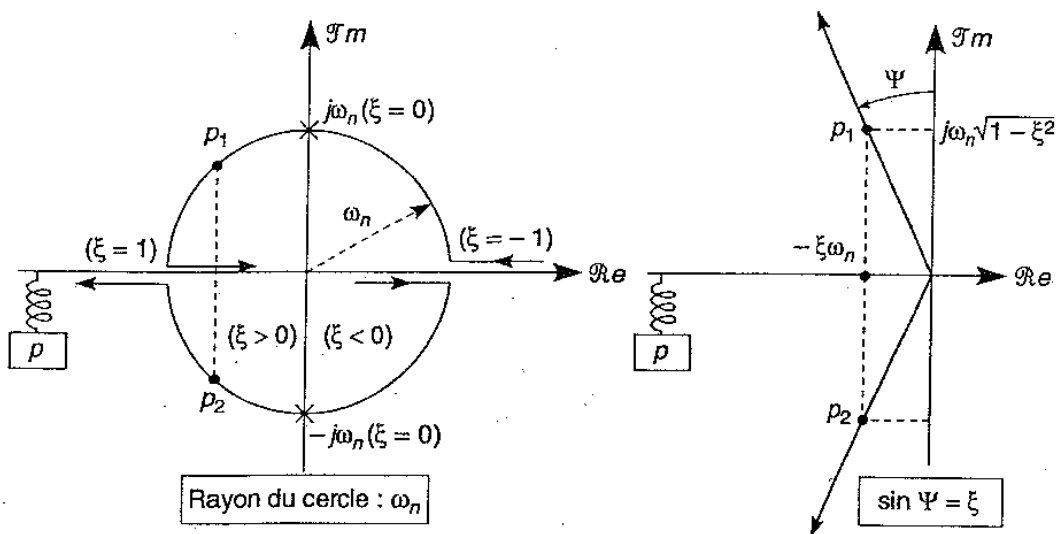


FIGURE 3.10 – Système du deuxième ordre en temps continu : lieu des pôles dans le plan complexe

14. D'après [Rivoire et Ferrier]

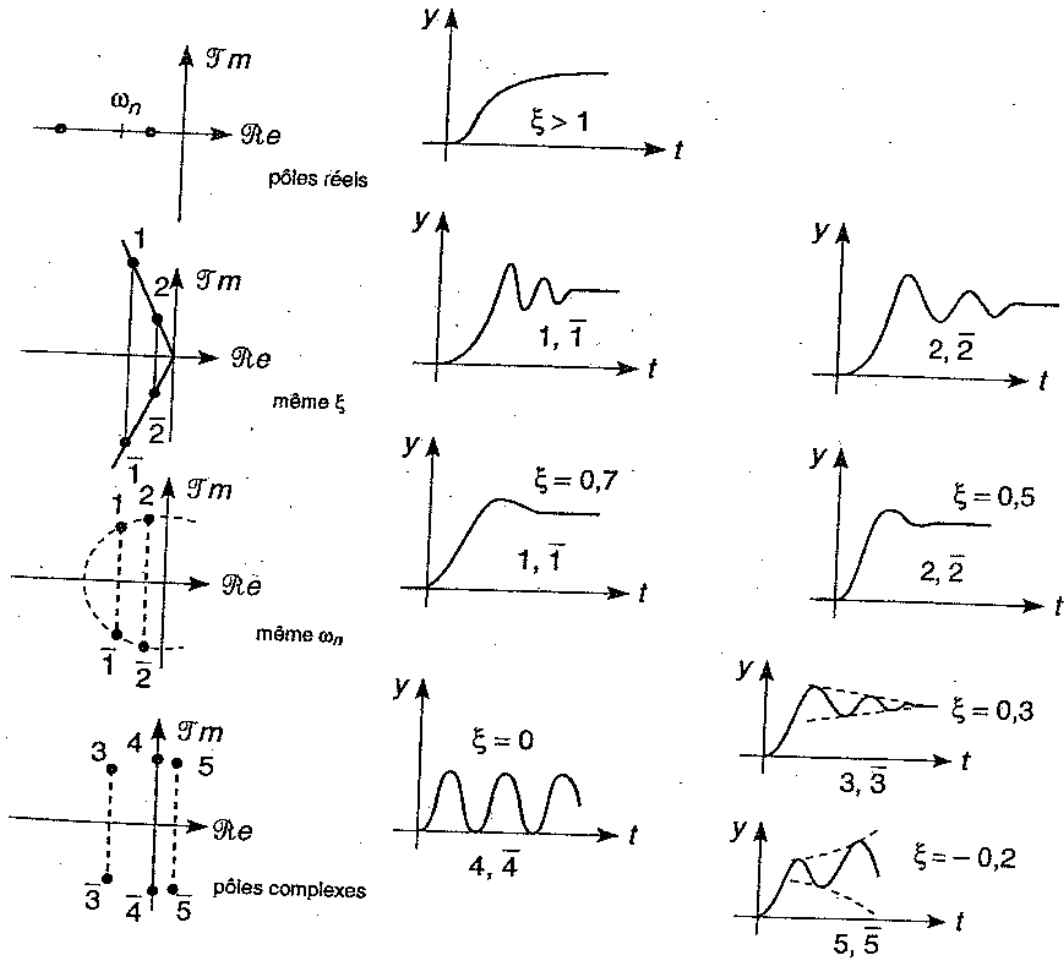


FIGURE 3.11 – Système du deuxième ordre en temps continu : influence de la place des pôles

En temps discret

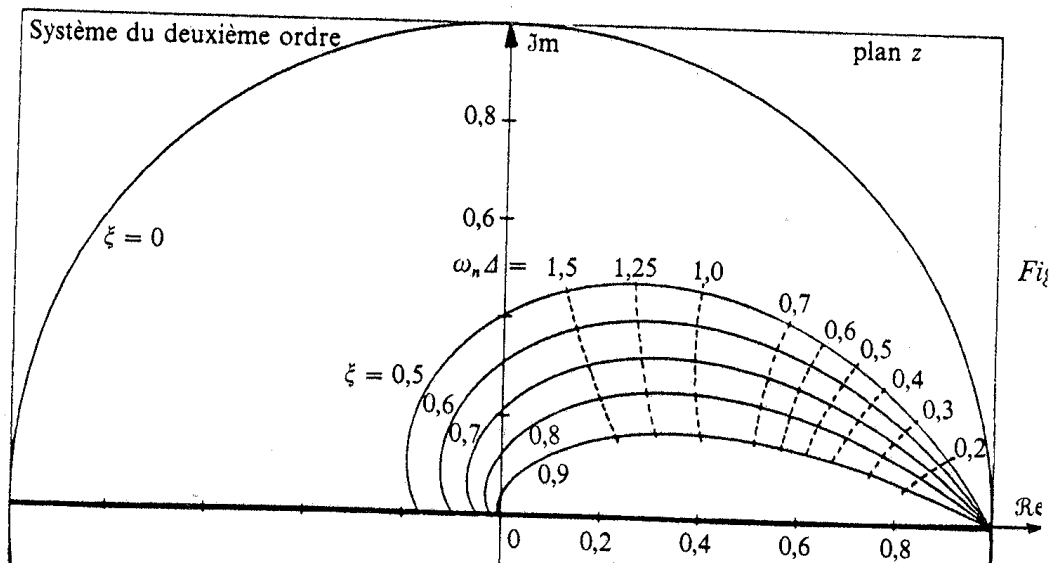


FIGURE 3.12 – Système du deuxième ordre en temps discret : lieu des pôles dans le plan complexe

Bibliographie

- [Isermann] R. Isermann, M. Munchhoff, *Identification of Dynamic systems*, Springer, 2011.
- [Friedland] B. Friedland, *Control System Design : An Introduction to State-Space Methods*, 1986.
- [Rivoire et Ferrier] M. Rivoire et J.L. Ferrier, *Cours et exercices d'Automatique, Tomes 1, 2 et 3*, Eyrolles, Paris, 1995.
- [Ferrier] J.L. Ferrier, *Modèles et systèmes : un tour d'horizon*, Notes de cours, ISTIA.
- [Jaulin] L. Jaulin, *Représentation d'état pour la modélisation et la commande des systèmes*, Eyrolles, Paris, 2005.
- [Jaulin] L. Jaulin, *Commande par espace d'état*, Notes de cours, Ensieteta, 2007 (disponible sur le Web, <http://www.ensieteta.fr/e3i2/Jaulin/polyetat.pdf>).
- [Borne et al.] P. Borne, G. Dauphin-Tanguy, J.P. Richard, F. Rotella et I. Zambettakis, *Modélisation et identification des processus, Tomes 1 et 2*, Technip, Paris, 1992.
- [Alazard] D. Alazard, *Introduction au filtre de Kalman*, Notes de cours, Supaero, 2005.
- [Briere] Y. Briere, *Le filtre de Kalman*, Transparents de cours, ENSICA.