

République Algérienne Démocratique et Populaire  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

UNIVERSITE MOULOUD MAMMARI DE TIZI-OUZOU



FACULTE DU GENIE ELECTRIQUE ET D' INFORMATIQUE  
DÉPARTEMENT D'AUTOMATIQUE

## Mémoire de Fin d'Études de MASTER ACADÉMIQUE

Domaine : Sciences et Technologies

Filière : Automatique

Spécialité : Automatique et Systèmes

*Présenté par*

Anaïs LAMARA MOHAMMED

Sabrina LAMARA MOHAMMED

Thème

## Paramétrisation d'un problème de commande optimale basée sur la conversion Bolza-Mayer

*Mémoire soutenu publiquement le 24/06/ 2024 devant le jury composé de :*

**M Said DJENNOUNE**

Professeur, UMMTO, Président

**Mme Ghania IDIRI**

MCB, UMMTO, Encadrant

**Mme Ouerdia MEGHERBI**

MCB, UMMTO, Examineur

**Mme Lamia SERSOUR**

MCB, UMMTO, Examineur

# Table des matières

<b>Introduction générale</b>	<b>1</b>
<b>1 Généralités sur la commande optimale</b>	<b>3</b>
1 Introduction . . . . .	3
2 Définition de la commande optimale . . . . .	3
2.1 Le modèle du système . . . . .	4
2.2 Les conditions terminales . . . . .	4
2.3 L'horizon de commande . . . . .	5
2.4 Les contraintes . . . . .	5
2.5 Le critère de performances : . . . . .	6
3 Formulation d'un problème de commande optimale . . . . .	7
4 Exemples de modélisation d'un problème de commande optimale . . . . .	8
5 Résolution d'un problème de commande optimale . . . . .	12
5.1 Méthode de calcul des variations . . . . .	13
5.2 Principe du minimum de Pontryagin . . . . .	18
5.3 Commande linéaire quadratique (LQ) . . . . .	23
6 Conclusion . . . . .	24
<b>2 Optimisation</b>	<b>25</b>
1 Introduction . . . . .	25
2 Définition de la fonction objectif et les variables de décision . . . . .	25
3 Les contraintes . . . . .	25
4 Formulation mathématique d'un problème d'optimisation . . . . .	26
5 Le minimum et le maximum d'une fonction . . . . .	27
5.1 Minimum ou maximum local . . . . .	27
5.2 Minimum ou maximum global . . . . .	27
6 Types de problèmes d'optimisation . . . . .	28
6.1 Problème d'optimisation sans contraintes . . . . .	28
6.2 Problème d'optimisation avec contraintes . . . . .	28
6.3 Problème d'optimisation avec contraintes mixtes . . . . .	28

7	Exemple de modélisation d'un problème d'optimisation . . . . .	29
8	Résolution d'un problème d'optimisation . . . . .	30
8.1	Optimisation sans contraintes . . . . .	30
8.2	Optimisation avec contraintes . . . . .	34
9	Optimisation globale . . . . .	36
9.1	Méthodes heuristiques . . . . .	36
9.2	Méthodes exactes . . . . .	37
9.3	Méthodes hybrides . . . . .	37
9.4	Différence entre les méthodes heuristiques, exactes et hybrides . . . . .	37
10	Principe des algorithmes génétiques (AGs) . . . . .	37
10.1	Notions de la théorie des algorithmes génétiques . . . . .	38
11	Exemple de résolution d'un problème d'optimisation par les algorithmes génétiques	40
12	Avantages et inconvénients des algorithmes génétiques . . . . .	43
13	Conclusion . . . . .	43

### **3 Résolution d'un problème de commande optimale par la méthode de conversion**

	<b>Bolza-Mayer</b>	<b>44</b>
1	Introduction . . . . .	44
2	Conversion Bolza-Mayer . . . . .	44
2.1	Exemple d'application de la méthode de conversion Bolza-Mayer . . . . .	46
3	Paramétrisation du vecteur de commande . . . . .	49
3.1	Principe : . . . . .	49
3.2	Les expressions de paramétrisation de la commande . . . . .	49
3.3	Exemple de paramétrisation d'un vecteur de commande . . . . .	50
4	Synthèse d'un problème de commande optimale basée sur la conversion Bolza-Mayer	58
5	Exemples d'application sur la synthèse d'un problème de commande optimale basée sur la conversion Bolza-Mayer . . . . .	60
5.1	Exemple 1 . . . . .	60
5.2	Exemple 2 . . . . .	64
6	Conclusion . . . . .	67

### **Conclusion générale** **68**

# Liste des tableaux

1.1	Conditions aux limites. . . . .	14
2.1	Tableau représentant le fonctionnement de l'usine. . . . .	29
2.2	La nature du point critique. . . . .	31
2.3	Conditions de Kuhn-Tucker. . . . .	36
2.4	Codage du problème. . . . .	41
2.5	Évaluation de l'adaptation. . . . .	41
2.6	Sélection des parents. . . . .	42
2.7	Recombinaison. . . . .	42
2.8	Seconde génération. . . . .	42
3.1	Résultats de synthèse d'un problème de commande optimale à l'aide de paramétrisation du vecteur de commande. . . . .	58
3.2	Résultats de la paramétrisation de l'exemple 1. . . . .	62
3.3	Résultats de la paramétrisation de l'exemple 2. . . . .	65

# Table des figures

1.1	Schéma d'explication du principe de la commande optimale. . . . .	4
1.2	Système mécanique. . . . .	8
1.3	Système électrique. . . . .	10
1.4	Tracés de $x_1(t)$ , $x_2(t)$ . . . . .	17
1.5	Tracé de $u^*(t)$ . . . . .	17
1.6	Tracés de $x_1(t)$ , $x_2(t)$ . . . . .	22
1.7	Tracé de $u^*(t)$ . . . . .	22
3.1	Chariot sur une voie. . . . .	51
3.2	L'évolution de l'état du système. . . . .	63
3.3	L'évolution de la commande optimale. . . . .	63
3.4	L'évolution de l'état du système. . . . .	66
3.5	L'évolution de la commande optimale. . . . .	67

# Liste des symboles

$x(t)$  : Vecteur d'état

$u(t)$  : Vecteur de commande

$y(t)$  : Vecteur de sortie ou vecteur d'état

$t$  : Variable de temps

$f(x(t), u(t), t)$  : Fonction vectorielle

$h(x(t), u(t), t)$  : Fonction vectorielle

$t_0$  : Instant initial

$t_f$  : Instant final

$g(x(t), u(t), t)$  : Fonction mathématique

$f(t)$  : Force

$J$  : Critère de performance

$Q, R$  : Matrices de pondération

$F(\dot{x}(t), x(t), t)$  : Fonction non linéaire

$g(x(t), \dot{x}(t), t)$  : Fonction non linéaire

$\lambda(t)$  : Multiplicateurs de Lagrange

$u^*(t)$  : Commande optimale

$H(x(t), \lambda(t), u(t), t)$  : Fonction d'Hamilton

$e(t)$  : Erreur de poursuite

$A(t)$  : Matrice d'état

$B(t)$  : Matrice de commande

$y^d(t)$  : Consigne désirée

$K, V, S$  : Matrices

$f(x)$  : Fonction objectif

$x$  : Variables de décision

$g_i(x)$  : Contraintes d'égalité

$h_j(x)$  : Contraintes d'inégalité

$x^*$  : Point critique

$V_\epsilon(x^*)$  : Voisinage autour de  $x^*$

$\epsilon$  : Taille du voisinage autour de  $x^*$  ou précision souhaitée

$\nabla$  : Le gradient

$\nabla^2$  : Le Hessien

$A$  : Matrice symétrique ou matrice de coefficient

$\Delta_k$  : Mineurs principaux

$x^{(k)}$  : L'estimé

$\|\cdot\|$  : Norme euclidienne

$|\cdot|$  : Le module

$P(x)$  : Approximation de la fonction  $f(x)$

$L$  : Le Lagrangien

$\mu_j$  : Multiplicateurs de Lagrange

$c$  : Vecteur de coefficient

$b$  : Vecteur des valeurs contraintes

$P$  : Population

$x^{(i)}$  : Individu

$a_i$  : Coefficients constantes

$v_i$  : Fonctions orthogonales connues

# Introduction générale

En automatique, lorsque le cahier des charges impose l'optimisation d'un critère de performance et le respect d'un ensemble de contraintes, le problème consiste à trouver une commande optimale. La théorie de la commande optimale regroupe l'ensemble des théories et des méthodes utilisées pour concevoir des lois de commande répondant à ces exigences. Une commande optimale, par définition, est celle qui permet de transférer un système dynamique d'un état initial à un état final, tout en respectant un certain nombre de contraintes sur les états et les commandes, et en optimisant un critère de performance [6].

Dans la résolution d'un problème de commande optimale, il existe principalement deux classes de méthodes : les méthodes indirectes et les méthodes directes. Les méthodes indirectes, telles que le Principe du minimum, le calcul des variations, et les équations de Riccati, utilisent des conditions nécessaires d'optimalité pour dériver la loi de commande optimale. Les méthodes directes, quant à elles, transforment le problème de commande optimale en un problème d'optimisation classique à résoudre en appliquant des algorithmes d'optimisation [8].

Dans ce mémoire, nous allons nous intéresser à la résolution des problèmes de commande optimale pour les systèmes dynamiques, en convertissant le problème de commande optimale sous forme Bolza en un problème de commande optimale sous forme Mayer et en transformant ce dernier en un problème d'optimisation en utilisant la méthode de paramétrisation du vecteur de commande. Pour la résolution de ce problème d'optimisation, nous allons appliquer des algorithmes génétiques afin de trouver la solution optimale.

Dans le premier chapitre, nous aborderons les généralités sur la commande optimale et les différentes méthodes utilisées pour résoudre un problème de commande optimale.

Dans le deuxième chapitre, nous présenterons les généralités sur l'optimisation et les diverses méthodes employées pour résoudre un problème d'optimisation.

Dans le dernier chapitre, nous nous concentrerons sur la conversion Bolza-Mayer, la paramétrisation du vecteur de commande, ainsi que sur l'utilisation des algorithmes génétiques.



Et finalement, on termine par une conclusion générale.

# Chapitre 1

## Généralités sur la commande optimale

### 1 Introduction

La théorie de la commande vise à élaborer des règles de contrôle assurant des performances spécifiques. Ce chapitre traite de manière générale des principes liés à la commande optimale.

### 2 Définition de la commande optimale

La commande optimale consiste à trouver une commande qui optimise (minimise ou maximise) un critère de performances donné en respectant certaines conditions [8].

La commande optimale est une méthode largement utilisée dans divers domaines tels que l'aérospatiale, l'automobile, l'énergie, la robotique, la finance, la biologie, la médecine et les télécommunications. Elle permet de concevoir des systèmes de contrôle efficaces en optimisant des critères de performance tout en tenant compte de contraintes et de variables dynamiques spécifiques à chaque domaine.

Le schéma de la figure 1.1 présente les différents éléments d'une commande optimale :

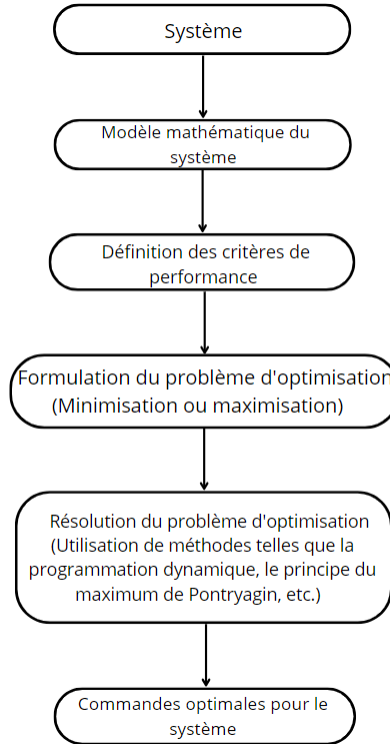


FIGURE 1.1 – Schéma d’explication du principe de la commande optimale.

## 2.1 Le modèle du système

Le modèle du système est écrit sous forme d’état comme suit [8] :

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

$$y(t) = h(x(t), u(t), t) \quad (1.2)$$

Avec :

$x(t) \in \mathbb{R}^n$  : le vecteur d’état,

$u(t) \in \mathbb{R}^m$  : le vecteur de commande,

$y(t) \in \mathbb{R}^r$  : le vecteur de sortie,

$t \in \mathbb{R}$  : la variable de temps,

$f(x(t), u(t), t)$  et  $h(x(t), u(t), t)$  : les fonctions vectorielles.

## 2.2 Les conditions terminales

Les conditions terminales sont l’état initial et l’état final du système, c’est-à-dire l’état du système avant l’application de la commande et après l’application de cette dernière [8].

— L’état initial  $x(t_0)$  : c’est l’état initial du système à l’instant  $t_0$ .

- L'état final  $x(t_f)$  : c'est l'état final du système à l'instant  $t_f$  après l'application de la commande. On distingue deux cas : fixe (variables d'état connues) ou libre (variables d'état inconnues).

## 2.3 L'horizon de commande

l'horizon de commande définit le nombre de pas de temps sur lesquels les actions de commande sont programmées à l'avance.

$$T = [t_0, t_f]$$

## 2.4 Les contraintes

Pour assurer la sécurité du système et de l'environnement, les contraintes doivent être respectées lors de la commande du système. On distingue deux types de contraintes [8] :

- Contraintes instantanées ; qui doivent être respectées à chaque instant  $t \in [t_0, t_f]$ , donnée comme suit :

$$g(x(t), u(t), t) \leq 0 \tag{1.3}$$

Où :  $g(x(t), u(t), t)$  est une fonction mathématique.

### Exemple d'une contrainte instantanée :

La contrainte instantanée peut être illustré comme suit :

Pour une protection d'un ressort dans un système mécanique, la force  $f(t)$  ne doit pas dépasser une force maximale  $f_{max}$ , on écrit :

$$f(t) < f_{max}$$

- Contraintes intégrales : qui doivent être respectée le long de l'horizon de commande, donnée comme suit :

$$\int_{t_0}^{t_f} h(x(t), u(t), t) dt \leq h_{max} \tag{1.4}$$

Avec :  $[t_0, t_f]$  l'horizon de commande.

### Exemple d'une contrainte intégrale :

La contrainte intégrale peut être illustrée par la limitation de la quantité totale d'électricité consommée sur une période donnée pour maximiser les bénéfices.

$$\int_0^T D(t) dt \leq E \tag{1.5}$$

Avec :

$D(t)$  : la fonction de consommation d'électricité.

$E$  : la quantité totale d'électricité disponible sur la période de temps  $[0, T]$ .

## 2.5 Le critère de performances :

Le critère de performances représente les objectifs à atteindre et à optimiser, c'est-à-dire mesurer, améliorer et atteindre les résultats souhaités. Le critère de performances s'écrit généralement sous la forme suivante [4], [8] :

$$J(u(t)) = \Psi(x(t_f), t_f) + \int_{t_0}^{t_f} Q(x(t), u(t), t) dt \quad (1.6)$$

Avec :

$\Psi(x(t_f), t_f)$  : la partie terminale.

$\int_{t_0}^{t_f} Q(x(t), u(t), t) dt$  : la partie intégrale.

Il existe trois problèmes :

**1. Mayer** : contient seulement la partie terminale.

$$J(u(t)) = \Psi(x(t_f), t_f) \quad (1.7)$$

**2. Lagrange** : contient seulement la partie intégrale.

$$J(u(t)) = \int_{t_0}^{t_f} Q(x(t), u(t), t) dt \quad (1.8)$$

**3. Bolza** : contient les deux parties ; terminale et intégrale.

$$J(u(t)) = \Psi(x(t_f), t_f) + \int_{t_0}^{t_f} Q(x(t), u(t), t) dt \quad (1.9)$$

Les principaux critères utilisés sont :

**1. La poursuite** :

La poursuite consiste à trouver la trajectoire d'un système dynamique qui minimise un critère de coût défini, tout en respectant les contraintes.

$$J(u(t)) = \int_{t_0}^{t_f} (x(t)^d - x(t))^T Q (x(t)^d - x(t)) dt \quad (1.10)$$

Avec :  $Q = Q^T \geq 0$  une matrice de pondération.

## 2. La régulation :

La régulation est un cas particulier de la poursuite c'est-à-dire :  $x^d(t) = 0$ .

$$J(u(t)) = \int_{t_0}^{t_f} x(t)^T Q x(t) dt \quad (1.11)$$

## 3. L'énergie minimale :

Consiste à minimiser la consommation d'énergie tout en assurant une performance satisfaisante du système. Dans ce cas le critère s'écrit :

$$J(u(t)) = \int_{t_0}^{t_f} u(t)^T R u(t) dt \quad (1.12)$$

Avec :  $R = R^T > 0$  une matrice de pondération.

## 4. Le temps minimal :

Visé à trouver la trajectoire qui minimise la durée nécessaire pour atteindre un objectif donné.

$$J(u(t)) = \int_{t_0}^{t_f} 1 dt = t_f - t_0 \quad (1.13)$$

## Remarque :

Dans un critère on peut avoir plusieurs objectifs à la fois. Par exemple on prend trois objectifs : La régulation, l'énergie minimale, le temps minimal.

$$J(u(t)) = \int_{t_0}^{t_f} x(t)^T Q x(t) dt + \int_{t_0}^{t_f} u(t)^T R u(t) dt + \int_{t_0}^{t_f} 1 dt \quad (1.14)$$

# 3 Formulation d'un problème de commande optimale

La formulation d'un problème de commande optimale demande la précision [8] :

- du modèle d'état du système,
- des conditions terminales,

- des contraintes,
- du critère de performances à optimiser.

En conclusion, la forme générale d'un problème de commande optimale est donnée comme suit :

$$\begin{aligned}
 \underset{u(t)}{\text{opt}} J(u(t)) &= \Psi(x(t_f), t_f) + \int_{t_0}^{t_f} Q(x(t), u(t), t) dt \\
 \text{Sujet à :} & \\
 \dot{x}(t) &= f(x(t), u(t), t) \\
 x(t_0) &= x_0 \\
 x(t_f) &= x_f \text{ ou libre} \\
 g(x(t), u(t), t) &\leq 0 \\
 \int_{t_0}^{t_f} h(x(t), u(t), t) dt &\leq h_{max}
 \end{aligned} \tag{1.15}$$

## 4 Exemples de modélisation d'un problème de commande optimale

### Exemple 1 :

Soit le système mécanique représenté dans la figure si-dessous :

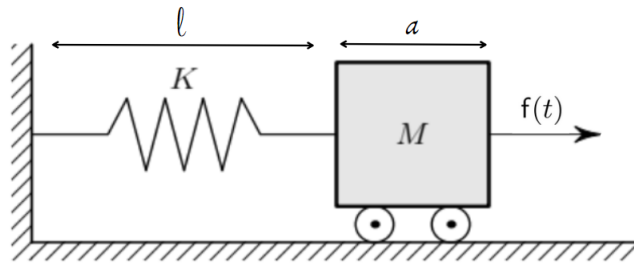


FIGURE 1.2 – Système mécanique.

On désire déterminer la force  $f(t)$  qui permet de transférer la masse  $M$ , en 2 minutes, à une position égale au double de sa position initiale tout en minimisant l'énergie mise en œuvre. Pour protéger le ressort, la force  $f(t)$  ne doit pas dépasser  $f_{max}$ .

L'état du système :

$$x(t) = \begin{pmatrix} y(t) \\ \dot{y}(t) \end{pmatrix} \tag{1.16}$$

Avec :  $y(t)$  la position de la masse et  $\dot{y}(t)$  la vitesse de la masse.

L'état initial :

$$x(0) = \begin{pmatrix} l + \frac{a}{2} \\ 0 \end{pmatrix} \quad (1.17)$$

L'état final :

$$x(t_f) = x(2) = \begin{pmatrix} 2l + a \\ 0 \end{pmatrix} : \text{l'état final est imposé} \quad (1.18)$$

L'horizon de commande :

$$T = [0, 2] \quad (1.19)$$

La contrainte :

$$f(t) < f_{max} : \text{contrainte instantanée.} \quad (1.20)$$

Le critère :

Dans ce problème, on veut minimiser l'énergie donc le critère s'écrit :

$$J(f(t)) = \int_0^2 f^2(t) dt \quad (1.21)$$

Le modèle d'état du système :

On applique la loi de Newton :

$$\sum \vec{F} = m \vec{\gamma} \quad (1.22)$$

Ce qui donne :

$$\Rightarrow f(t) - Ky(t) = M\ddot{y}(t) \quad (1.23)$$

$$x_1(t) = y(t) \Rightarrow \dot{x}_1(t) = \dot{y}(t) = x_2(t) \quad (1.24)$$

$$x_2(t) = \dot{y}(t) \Rightarrow \dot{x}_2(t) = \ddot{y}(t) = \frac{1}{M}f(t) - \frac{K}{M}x_1(t) \quad (1.25)$$



Alors, le modèle du système s'écrit :

$$\begin{cases} \begin{pmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{pmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{K}{M} & 0 \end{bmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{1}{M} \end{pmatrix} f(t) \\ y(t) = \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} \end{cases} \quad (1.26)$$

En conclusion, le problème de commande optimale s'écrit :

$$\begin{aligned} \min_{f(t)} J(f(t)) &= \int_0^2 f^2(t) dt \\ \text{Sujet à :} \\ \dot{x}_1(t) &= x_2(t) \\ \dot{x}_2(t) &= -\frac{K}{M}x_1(t) + \frac{1}{M}f(t) \\ x(0) &= \begin{pmatrix} x_1(0) \\ x_2(0) \end{pmatrix} = \begin{pmatrix} l + \frac{a}{2} \\ 0 \end{pmatrix} \\ x(2) &= \begin{pmatrix} x_1(2) \\ x_2(2) \end{pmatrix} = \begin{pmatrix} 2l + a \\ 0 \end{pmatrix} \\ f(t) &< f_{max} \end{aligned} \quad (1.27)$$

## Exemple 2 :

Soit le système électrique représenté dans la figure si-dessous :

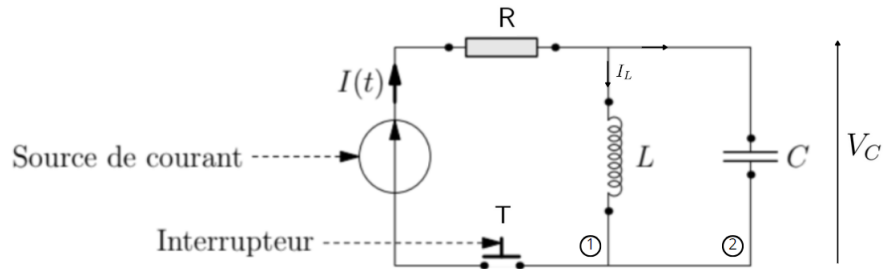


FIGURE 1.3 – Système électrique.

On désire déterminer la commande optimale  $I(t)$  qui permet de transférer rapidement le système de son état initial à un état final caractérisé par une tension aux bornes de la capacité  $C$  de  $1V$

et un courant de  $0A$  dans l'inductance  $L$ . Initialement, le condensateur est déchargé. À l'instant  $t = 0s$ , on ferme l'interrupteur  $T$ . Pour protéger les éléments du circuit, on doit éviter des courants  $I(t)$  qui dépassent la valeur tolérée qui est de  $1,5A$ .

L'état du système :

$$x(t) = \begin{pmatrix} V_C(t) \\ I_L(t) \end{pmatrix} \quad (1.28)$$

L'état initial :

Initialement, le condensateur est déchargé donc :

$$x(0) = \begin{pmatrix} V_C(0) \\ I_L(0) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad (1.29)$$

L'état final :

$$x(t_f) = \begin{pmatrix} 1 \\ 0 \end{pmatrix} ; \text{état final imposé} \quad (1.30)$$

L'horizon de commande :

$$T = [0, t_f] \quad (1.31)$$

La contrainte :

$$I(t) \leq 1.5 ; \text{contrainte instantanée} \quad (1.32)$$

Le critère :

On veut minimiser le temps donc :

$$J(I(t)) = \int_0^{t_f} 1 dt \quad (1.33)$$

Le modèle d'état du système :

Pour la modélisation, on applique la loi des mailles :

Pour la maille (2), on a :

$$\frac{dI_L(t)}{dt} = \frac{1}{L}V_C(t) \quad (1.34)$$

Et on a :

$$V_C(t) = \frac{1}{C} \int I_C(t) dt \quad (1.35)$$

$$I_C(t) = I(t) - I_L(t) \quad (1.36)$$

Donc :

$$\frac{dV_C(t)}{dt} = -\frac{1}{C}I_L(t) + \frac{1}{C}I(t) \quad (1.37)$$

Alors, le modèle du système s'écrit :

$$\begin{cases} \frac{dV_C(t)}{dt} = -\frac{1}{C}I_L(t) + \frac{1}{C}I(t) \\ \frac{dI_L(t)}{dt} = \frac{1}{L}V_C(t) \end{cases} \quad (1.38)$$

En conclusion, le problème de commande optimale s'écrit :

$$\begin{aligned} \text{Sujet à : } \quad & \min_{I(t)} J(I(t)) = \int_0^{t_f} 1 dt \\ & \frac{dV_C(t)}{dt} = -\frac{1}{C}I_L(t) + \frac{1}{C}I(t) \\ & \frac{dI_L(t)}{dt} = \frac{1}{L}V_C(t) \\ & \begin{pmatrix} V_C(0) \\ I_L(0) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \\ & \begin{pmatrix} V_C(t_f) \\ I_L(t_f) \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \\ & I(t) \leq 1,5 \end{aligned} \quad (1.39)$$

## 5 Résolution d'un problème de commande optimale

Il existe plusieurs méthodes pour la résolution d'un problème de commande optimale, voici quelques unes :

## 5.1 Méthode de calcul des variations

Le calcul des variations est une méthode puissante appliquée en commande optimale pour trouver des solutions analytiques ou numériques, en optimisant les performances du système dynamique tout en respectant diverses contraintes.

En effet, la procédure se fait d'abord par la résolution de l'équation du modèle par rapport à  $u(t)$ . Après avoir résolu celle-ci viendra l'application de l'équation d'Euler-Lagrange.

Pour expliquer le principe de la méthode, on considère le problème de commande optimale suivant [8], [9] :

$$\underset{u(t)}{\text{opt}} J(u(t)) = \int_{t_0}^{t_f} Q(x(t), u(t), t) dt \quad (1.40)$$

*Sujet* : à :

$$\dot{x}(t) = f(x(t), u(t), t) \quad (1.41)$$

$$x(t_0) = x_0 \quad (1.42)$$

$$x(t_f) = x_f \text{ ou libre} \quad (1.43)$$

Selon la nature du modèle du système, deux méthodes peuvent être utilisées :

**a. La méthode directe** : elle est appliquée lorsque la résolution de l'équation du modèle par rapport à  $u(t)$  est possible.

1. La première étape consiste à résoudre l'équation de modèle par rapport à  $u(t)$  i.e :

$$\dot{x}(t) = f(x(t), u(t), t) \longrightarrow u(t) = F(\dot{x}(t), x(t), t) \quad (1.44)$$

où :  $F(\dot{x}(t), x(t), t)$  est une fonction généralement non linéaire

2. La deuxième étape est de remplacer l'expression de  $u(t)$  dans l'équation du critère (1.40), ce qui donne :

$$J = \int_{t_0}^{t_f} Q(x(t), F(\dot{x}(t), x(t), t), t) dt = \int_{t_0}^{t_f} g(x(t), \dot{x}(t), t) dt \quad (1.45)$$

Avec :  $g(x(t), \dot{x}(t), t) = Q(x(t), F(\dot{x}(t), x(t), t), t)$ .

3. La méthode de calcul des variations consiste à résoudre l'équation d'Euler-Lagrange selon l'état initial et l'état final, les différents cas existants sont résumés dans le tableau suivant :

	État final imposé	État final libre
État initial imposé	$\frac{\partial g}{\partial x} - \frac{d}{dt}(\frac{\partial g}{\partial \dot{x}}) = 0$ $x(t_0) = x_0$ $x(t_f) = x_f$	$\frac{\partial g}{\partial x} - \frac{d}{dt}(\frac{\partial g}{\partial \dot{x}}) = 0$ $x(t_0) = x_0$ $\frac{\partial g}{\partial \dot{x}} _{t=t_f} = 0$
État initial libre	$\frac{\partial g}{\partial x} - \frac{d}{dt}(\frac{\partial g}{\partial \dot{x}}) = 0$ $\frac{\partial g}{\partial \dot{x}} _{t=t_0} = 0$ $x(t_f) = x_f$	$\frac{\partial g}{\partial x} - \frac{d}{dt}(\frac{\partial g}{\partial \dot{x}}) = 0$ $\frac{\partial g}{\partial \dot{x}} _{t=t_0} = 0$ $\frac{\partial g}{\partial \dot{x}} _{t=t_f} = 0$

TABLE 1.1 – Conditions aux limites.

La résolution de ce système d'équations, nous donnera la solution optimale  $x^*(t)$ .

4. Calcul de la commande optimale :

On remplace l'expression de  $x^*(t)$  dans (1.44), on obtient la commande optimale :

$$u^*(t) = F(\dot{x}^*(t), x^*(t), t) \quad (1.46)$$

**b. La méthode des multiplicateurs de Lagrange :** elle est appliquée lorsque la résolution de l'équation du modèle par rapport à  $u(t)$  est impossible ou lorsque les contraintes ne peuvent pas être insérer dans le critère. Dans ce cas, celle-ci permet de convertir le problème de commande optimale en un problème de calcul des variations à plusieurs variables.

1. Introduire l'équation du modèle ou les contraintes dans le critère en utilisant les multiplicateurs de Lagrange  $\lambda(t)$ , alors (1.40) devient :

$$J(X) = \int_{t_0}^{t_f} Q(x(t), u(t), t) dt + \lambda^T(t) [\dot{x}(t) - f(x(t), u(t), t)] dt \quad (1.47)$$

$$\text{Avec : } X(t) = \begin{bmatrix} x(t) \\ u(t) \\ \lambda(t) \end{bmatrix} \text{ et } \lambda(t) = \begin{bmatrix} \lambda_1(t) \\ \lambda_2(t) \\ \cdot \\ \cdot \\ \lambda_n(t) \end{bmatrix}$$

2. Résoudre le problème de calcul des variations en utilisant l'équation d'Euler-Lagrange :

$$\frac{\partial g}{\partial X} - \frac{d}{dt} \left( \frac{\partial g}{\partial \dot{X}} \right) = 0 \Leftrightarrow \frac{\partial g}{\partial x} - \frac{d}{dt} \left( \frac{\partial g}{\partial \dot{x}} \right) = 0 \quad (1.48)$$

$$\frac{\partial g}{\partial u} - \frac{d}{dt} \left( \frac{\partial g}{\partial \dot{u}} \right) = 0 \quad (1.49)$$

$$\frac{\partial g}{\partial \lambda} - \frac{d}{dt} \left( \frac{\partial g}{\partial \dot{\lambda}} \right) = 0 \quad (1.50)$$

Après la résolution de ce système d'équations, on obtiendra la solution  $x^*(t)$ .

3. Calculer la commande optimale :

$$u^*(t) = F(x^*(t), x^*(t), t) \quad (1.51)$$

## Exemple de résolution d'un problème de commande optimale par la méthode de calcul des variations

Soit un problème de commande optimale suivant :

$$\min_{u(t)} J(u(t)) = \frac{1}{2} \int_0^2 u^2(t) dt \quad (1.52)$$

*Sujet à :*

$$\dot{x}_1(t) = x_2(t) \quad (1.53)$$

$$\dot{x}_2(t) = u(t) \quad (1.54)$$

$$x_1(0) = 1 \quad (1.55)$$

$$x_2(0) = 1 \quad (1.56)$$

$$x_1(2) = 0 \quad (1.57)$$

$$x_2(2) = 0 \quad (1.58)$$

À partir de (1.53) et de (1.54) on aura :

$$x_2(t) - \dot{x}_1(t) = 0 \quad (1.59)$$

$$u(t) - \dot{x}_2(t) = 0 \quad (1.60)$$

Dans ce cas, on utilise la méthode des multiplicateurs de Lagrange.

Alors, on aura :

$$g(x(t), \dot{x}(t), t) = \frac{1}{2}u^2(t) + \lambda_1(t) [x_2(t) - \dot{x}_1(t)] + \lambda_2(t) [u(t) - \dot{x}_2(t)] \quad (1.61)$$

Les équations d'Euler-Lagrange s'écrivent :

$$\frac{\partial g}{\partial x_1} - \frac{d}{dt} \left( \frac{\partial g}{\partial \dot{x}_1} \right) = 0 \Rightarrow \dot{\lambda}_1(t) = 0 \quad (1.62)$$

$$\frac{\partial g}{\partial x_2} - \frac{d}{dt} \left( \frac{\partial g}{\partial \dot{x}_2} \right) = 0 \Rightarrow \lambda_1(t) + \dot{\lambda}_2(t) = 0 \quad (1.63)$$

$$\frac{\partial g}{\partial \lambda_1} - \frac{d}{dt} \left( \frac{\partial g}{\partial \dot{\lambda}_1} \right) = 0 \Rightarrow x_2(t) - \dot{x}_1(t) = 0 \quad (1.64)$$

$$\frac{\partial g}{\partial \lambda_2} - \frac{d}{dt} \left( \frac{\partial g}{\partial \dot{\lambda}_2} \right) = 0 \Rightarrow u(t) - \dot{x}_2(t) = 0 \quad (1.65)$$

$$\frac{\partial g}{\partial u} - \frac{d}{dt} \left( \frac{\partial g}{\partial \dot{u}} \right) = 0 \Rightarrow u(t) + \lambda_2(t) = 0 \quad (1.66)$$

Donc :

$$x_1(t) = -t^3 + \frac{5}{4}t^2 + t + 1 \quad (1.67)$$

$$x_2(t) = -\frac{3}{2}t^2 + \frac{5}{2}t + 1 \quad (1.68)$$

$$\lambda_1(t) = 3 \quad (1.69)$$

$$\lambda_2(t) = -3t + \frac{5}{2} \quad (1.70)$$

Alors, la commande optimale est :

$$u^* = 3t - \frac{5}{2} \quad (1.71)$$

Les tracés de  $x_1(t)$  et  $x_2(t)$  sont présentés par la figure 1.4 :

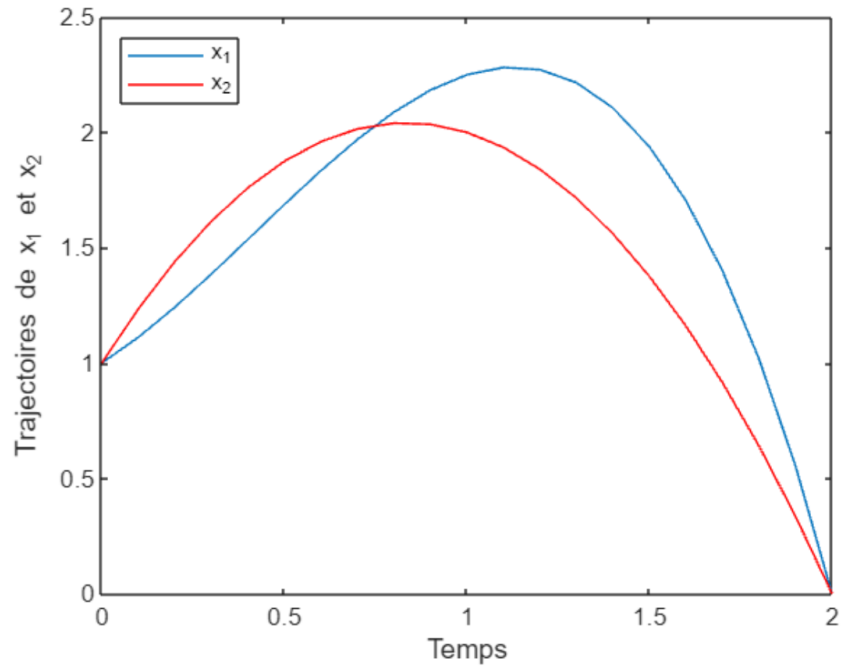


FIGURE 1.4 – Tracés de  $x_1(t)$ ,  $x_2(t)$ .

Le tracé de  $u^*(t)$  est présenté par la figure 1.5 :

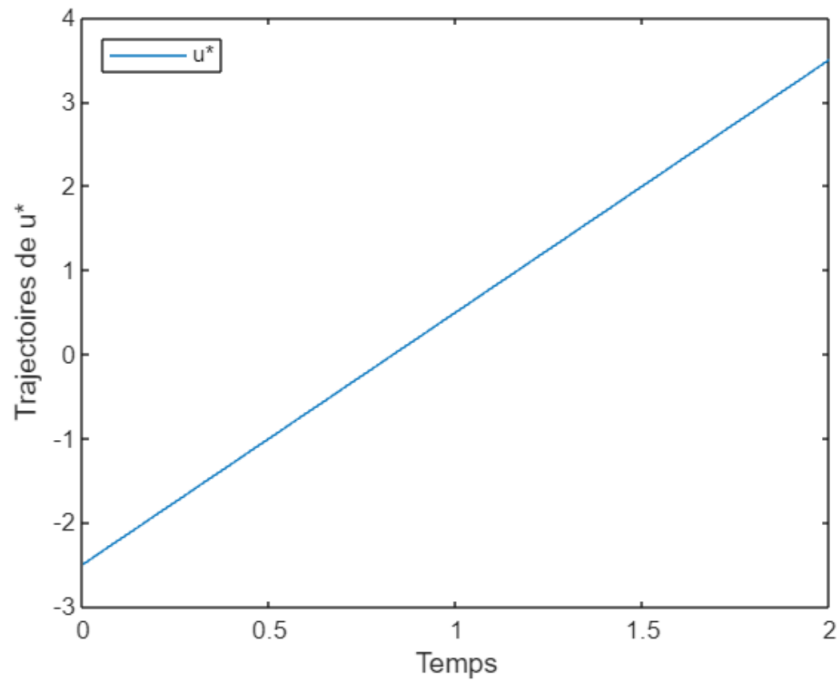


FIGURE 1.5 – Tracé de  $u^*(t)$ .



## 5.2 Principe du minimum de Pontryagin

Le principe du minimum de Pontryagin énonce que dans un problème de commande optimale, le système évolue de manière à minimiser une fonction de coût, en ajustant les variables d'état et de commande selon les équations d'Hamilton [8], [9].

Soit le problème de commande optimale suivant :

$$\begin{aligned} \underset{u(t)}{\text{opt}} J(u(t)) &= \Psi(x(t_f), t_f) + \int_{t_0}^{t_f} Q(x(t), u(t), t) dt \\ \text{Sujet à :} & \\ \dot{x}(t) &= f(x(t), u(t), t) \\ x(t_0) &= x_0 \\ x(t_f) &= x_f \text{ ou libre} \end{aligned} \quad (1.72)$$

En utilisant les multiplicateurs de Lagrange  $\lambda(t)$  et en insérant l'équation du modèle dans celle du critère comme suit :

$$\dot{x}(t) = f(x(t), u(t), t) \Rightarrow f(x(t), u(t), t) - \dot{x}(t) = 0 \quad (1.73)$$

$$J(u(t)) = \Psi(x(t_f), t_f) + \int_{t_0}^{t_f} Q(x(t), u(t), t) + \lambda^T(t) (f(x(t), u(t), t) - \dot{x}(t)) dt \quad (1.74)$$

On aura :

$$J(u(t)) = \Psi(x(t_f), t_f) + \int_{t_0}^{t_f} H(x(t), \lambda(t), u(t), t) - \lambda^T(t) \dot{x}(t) dt \quad (1.75)$$

Avec :  $H(x(t), \lambda(t), u(t), t) = Q(x(t), u(t), t) + \lambda^T(t) f(x(t), u(t), t)$ , est la fonction d'Hamilton.

Pour l'obtention de l'expression de la commande, on suit ces étapes :

1. Construction de la fonction d'Hamilton :

$$H(x(t), \lambda(t), u(t), t) = Q(x(t), u(t), t) + \lambda^T(t) f(x(t), u(t), t) \quad (1.76)$$

2. Détermination de l'expression de la commande  $u(t)$  à partir de  $\frac{\partial H}{\partial u}$  ;

- . Si la résolution de  $\frac{\partial H}{\partial u} = 0$  par rapport à  $u(t)$ , passer à l'étape 3.
- . Sinon la méthode du principe du minimum ne s'applique pas.

3. Évaluation de  $\frac{\partial^2 H}{\partial u^2}$  :

Si  $\frac{\partial^2 H}{\partial u^2} > 0$ , le problème admet un minimum.

Si  $\frac{\partial^2 H}{\partial u^2} < 0$  et le problème admet un maximum.

4. Résolution de l'équation  $\frac{\partial H}{\partial u} = 0$  pour avoir l'expression de la commande optimale, qui aura la forme suivante :

$$u^*(t) = F(x(t), \lambda(t), t) \quad (1.77)$$

5. Détermination de  $H^*(x(t), \lambda(t), u(t), t)$  :

On remplace  $u^*(t)$  dans (1.76), on aura :

$$H^* = H|_{u=u^*} \quad (1.78)$$

Avec :  $u^* = F(x^*(t), \lambda^*(t), t)$

6. Résolution des équations de Hamilton-Pontryagin, qui sont les suivantes :

$$\dot{x}^*(t) = \frac{\partial H^*}{\partial \lambda^*} \quad (1.79)$$

$$\dot{\lambda}^*(t) = -\frac{\partial H^*}{\partial x^*} \quad (1.80)$$

7. Établir les paramètres des limites :

. État final imposé :

On aura ces deux conditions aux limites suivantes :

$$x(t_0) = x_0 \quad (1.81)$$

$$x(t_f) = x_f \quad (1.82)$$

. État final libre :

On aura ces deux conditions aux limites suivantes :

$$x(t_0) = x_0 \quad (1.83)$$

$$\lambda(t_f) = \frac{\partial \psi(x(t_f), t_f)}{\partial x(t_f)} \quad (1.84)$$

8. Calculer l'expression de la commande optimale :

$$u^*(t) = F(x^*(t), \lambda^*(t), t) \quad (1.85)$$

## Exemple de résolution d'un problème de commande optimale par la méthode du principe du minimum de Pontryagin

Prenons l'exemple précédent :

$$\min_{u(t)} J(u(t)) = \frac{1}{2} \int_0^2 u^2(t) dt \quad (1.86)$$

*Sujet à :*

$$\dot{x}_1(t) = x_2(t) \quad (1.87)$$

$$\dot{x}_2(t) = u(t) \quad (1.88)$$

$$x_1(0) = 1 \quad (1.89)$$

$$x_2(0) = 1 \quad (1.90)$$

$$x_1(2) = 0 \quad (1.91)$$

$$x_2(2) = 0 \quad (1.92)$$

On détermine la fonction de Hamilton :

$$H(x(t), \lambda(t), u(t), t) = \frac{1}{2} u^2(t) + \lambda_1(t) x_2(t) + \lambda_2(t) u(t) \quad (1.93)$$

L'expression de la commande  $u(t)$  :

$$\frac{\partial H(x(t), \lambda(t), u(t), t)}{\partial u(t)} = u(t) + \lambda_2(t) = 0 \quad (1.94)$$

$$\Rightarrow u^*(t) = -\lambda_2(t) \quad (1.95)$$

On évalue  $\frac{\partial^2 H}{\partial u^2}$  :

$$\frac{\partial^2 H}{\partial u^2} = 1 > 0 \Rightarrow u^*(t) \text{ est un minimum.}$$

Donc :

$$u^*(t) = -\lambda_2(t) \quad (1.96)$$

Alors :

$$H^* = H|_{u=u^*} = -\frac{1}{2} \lambda_2^2(t) + \lambda_1(t) x_2(t) \quad (1.97)$$

Équations de Hamilton-Pontryagin :

$$\dot{x}_1(t) = \frac{\partial H^*}{\partial \lambda_1} = x_2(t) \quad (1.98)$$

$$\dot{x}_2(t) = \frac{\partial H^*}{\partial \lambda_2} = -\lambda_2(t) \quad (1.99)$$

$$\dot{\lambda}_1(t) = -\frac{\partial H^*}{\partial x_1} = 0 \quad (1.100)$$

$$\dot{\lambda}_2(t) = -\frac{\partial H^*}{\partial x_2} = -\lambda_1(t) \quad (1.101)$$

Donc :

$$x_1(t) = -t^3 + \frac{5}{4}t^2 + t + 1 \quad (1.102)$$

$$x_2(t) = -\frac{3}{2}t^2 + \frac{5}{2}t + 1 \quad (1.103)$$

$$\lambda_1(t) = 3 \quad (1.104)$$

$$\lambda_2(t) = -3t + \frac{5}{2} \quad (1.105)$$

Suite à l'exécution d'un programme Matlab, On obtient :

$$u^* = 3t - \frac{5}{2} \quad (1.106)$$

Les tracés de  $x_1(t)$  et  $x_2(t)$  sont présentés par la figure 1.6 :

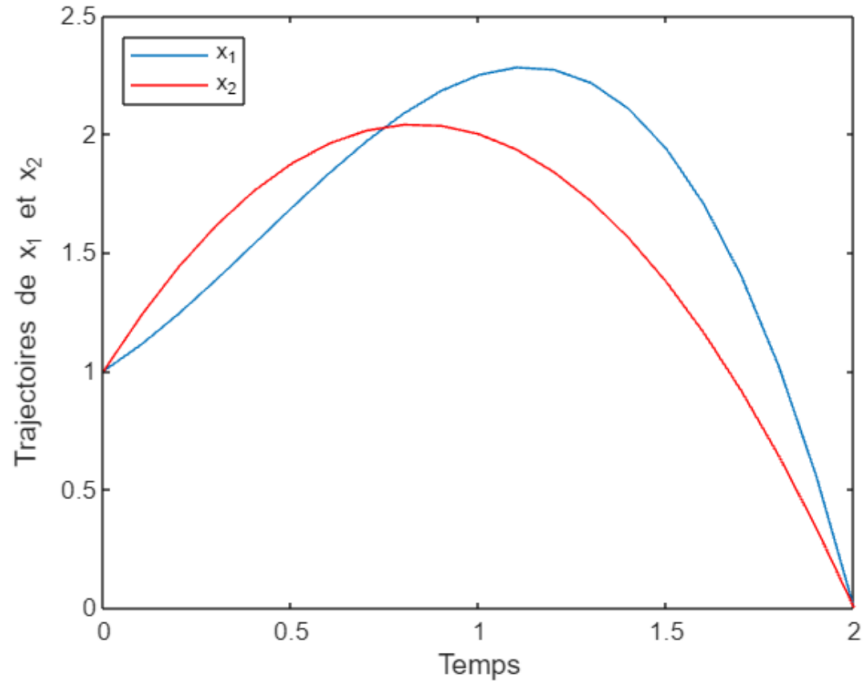


FIGURE 1.6 – Tracés de  $x_1(t)$ ,  $x_2(t)$ .

Le tracé de  $u^*(t)$  est présenté par la figure 1.7 :

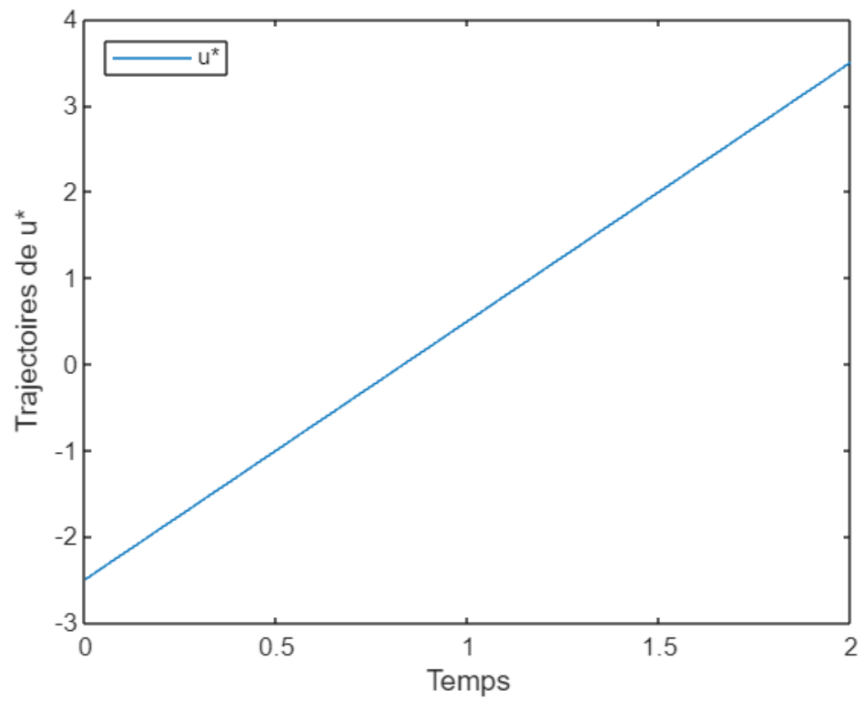


FIGURE 1.7 – Tracé de  $u^*(t)$ .

### 5.3 Commande linéaire quadratique (LQ)

Pour cette méthode, le problème de commande optimale est de la forme suivante [11] :

$$\begin{aligned} \min_{u(t)} J(u(t)) &= \frac{1}{2} e^T(t_f) H e(t_f) + \int_0^{t_f} \frac{1}{2} e^T(t) Q e(t) + \frac{1}{2} u^T(t) R u(t) dt \\ \text{Sujet à :} & \\ \dot{x}(t) &= A(t)x(t) + B(t)u(t) \\ x(0) &= x_0 \end{aligned} \tag{1.107}$$

Le critère est quadratique et le modèle est linéaire.

Avec :

$A(t)$  : la matrice d'état.

$B(t)$  : la matrice de commande.

$H$  : une matrice avec :  $H = H^T \geq 0$  ,

$Q$  et  $R$  : deux matrices de pondération avec :  $Q = Q^T \geq 0$  et  $R^T = R > 0$ .

$e(t) = y^d(t) - y(t)$  : l'erreur de poursuite.

$y^d(t)$  : la consigne désirée.

Pour la résolution du problème, on utilise le principe du minimum :

L'Hamiltonien s'écrit :

$$H(x(t), u(t), \lambda(t)) = \frac{1}{2} \left[ [e(t)]^T Q [e(t)] + [u(t)]^T R [u(t)] \right] + \lambda^T(t) [Ax(t) + Bu(t)] \tag{1.108}$$

L'expression de la commande optimale en dérivant  $H(x(t), u(t), \lambda(t))$  par rapport à  $u(t)$  est sous forme :

$$u^*(t) = -R^{-1} B^T \lambda(t) \tag{1.109}$$

Les équations de Hamilton-Pontryagin sont :

$$\dot{x}^*(t) = \frac{\partial H^*(x(t), u(t), \lambda(t))}{\partial \lambda^*(t)} = -BR^{-1}B^T \lambda^* + Ax^*(t) \tag{1.110}$$

$$\dot{\lambda}^*(t) = -\frac{\partial H^*(x(t), u(t), \lambda(t))}{\partial x^*(t)} = Qx^d(t) - Qx^*(t) + A^T \lambda^*(t) \tag{1.111}$$

Supposons qu'il y a une relation linéaire ente  $\lambda(t)$  et  $x(t)$ , on écrit :

$$\lambda(t) = K(t)x(t) - V(t) \tag{1.112}$$

Avec :  $K(t)$  et  $V(t)$  sont deux matrices.

On dérive (1.112), on obtient :

$$\dot{\lambda}(t) = K(t)\dot{x}(t) + \dot{K}(t)x(t) - \dot{V}(t) \quad (1.113)$$

En remplaçant  $\dot{x}(t)$  et  $\dot{\lambda}(t)$  par leurs équations, on aura :

$$\dot{K}(t) + K(t)A + A^T K(t) - K(t)SK(t) + Q = 0 \quad (1.114)$$

$$\dot{V}(t) + (A^T - K(t)S)V(t) - Qx^d(t) = 0 \quad (1.115)$$

(1.114) est l'équation différentielle de Ricatti.

Avec :  $S = BR^{-1}B^T$

La résolution de (1.114) et (1.115) nous donne l'expression de  $\lambda(t)$ , et en remplaçant  $\lambda(t)$  dans (1.109), on obtient l'expression de la commande optimale comme suit :

$$u^*(t) = -R^{-1}B^T (K(t)x(t) + SV(t)) \quad (1.116)$$

## 6 Conclusion

Les méthodes de résolution des problèmes de commande optimale sont des approches mathématiques et algorithmiques visant à déterminer la meilleure manière de contrôler un système dynamique afin d'optimiser certains critères prédéfinis. Dans ce chapitre nous avons présenté plusieurs méthodes dont la méthode de calcul de variation et la méthode du principe du minimum.

Bien que les méthodes indirectes soient utiles et souvent nécessaires pour résoudre certains problèmes complexes ou mal définis, elles présentent plusieurs inconvénients qui doivent être soigneusement pris en compte lors de leur application, tels que le coût en termes de temps de calcul pour des problèmes plus complexes.

# Chapitre 2

## Optimisation

### 1 Introduction

L'optimisation en automatique se réfère généralement à l'amélioration des performances des systèmes automatiques tels que les contrôleurs, les algorithmes de régulation, etc.

Mathématiquement, elle consiste à rechercher le minimum ou le maximum d'une fonction avec ou sans contraintes [1].

### 2 Définition de la fonction objectif et les variables de décision

En optimisation, la fonction objectif est une fonction mathématique qui exprime quantitativement ce que l'on cherche à maximiser ou minimiser. Elle représente le critère d'optimisation principal dans un problème donné [1].

Les variables de décision sont les paramètres que l'on cherche à ajuster ou à déterminer de manière optimale dans le cadre d'un problème d'optimisation, en tenant compte des objectifs à atteindre et des contraintes à respecter.

Soit un problème d'optimisation avec  $n$  variables de décision  $x_1, x_2, \dots, x_n$ . Chaque  $x_i$  est une variable réelle qui peut être modifiée pour optimiser une fonction objectif  $f(x_1, x_2, \dots, x_n)$  sous certaines contraintes.

### 3 Les contraintes

Soit un problème d'optimisation avec  $n$  variables de décision  $x_1, x_2, \dots, x_n$ . Les contraintes



peuvent être définies mathématiquement comme suit [8] :

### Contraintes d'égalité :

$$g_i(x_1, x_2, \dots, x_n) = 0, \quad i = 1, 2, \dots, m.$$

Ces contraintes expriment des relations d'égalité entre les variables. Par exemple,  $x_1 + x_2 = 10$  est une contrainte d'égalité.

### Contraintes d'inégalité :

$$h_j(x_1, x_2, \dots, x_n) \leq 0, \quad j = 1, 2, \dots, p.$$

Ces contraintes expriment des relations d'inégalité entre les variables. Par exemple,  $x_1 - x_2 \leq 5$  est une contrainte d'inégalité.

## 4 Formulation mathématique d'un problème d'optimisation

Soit  $f(x)$  la fonction objectif.

Mathématiquement, un problème d'optimisation est formulé comme suit [7] :

### a) En cas de maximisation :

$$\begin{aligned} & \max_x f(x) \\ \text{Sous contraintes :} & \\ & g_i(x) = 0 \\ & h_j(x) \leq 0 \end{aligned} \tag{2.1}$$

### b) En cas de minimisation :

$$\begin{aligned} & \min_x f(x) \\ \text{Sous contraintes :} & \\ & g_i(x) = 0 \\ & h_j(x) \leq 0 \end{aligned} \tag{2.2}$$

## Remarques :

1) La recherche d'un maximum peut se ramener à la recherche d'un minimum et vice versa comme suit [1] :

$$\max f(x) = -\min(-f(x)) \text{ et } \min f(x) = -\max(-f(x))$$

2) La relation entre le maximum et le minimum :

Dans un intervalle fermé et borné  $[a, b]$ , le théorème des valeurs extrêmes stipule que si  $f(x)$  est continue sur  $[a, b]$ , alors elle atteint à la fois un maximum et un minimum sur cet intervalle.

Mathématiquement, cela se traduit par :

$$\exists c, d \in [a, b] : \max_{x \in [a, b]} f(x) = f(c) \text{ et } \min_{x \in [a, b]} f(x) = f(d)$$

**Définition d'une fonction continue :** Mathématiquement, une fonction  $f : A \rightarrow \mathbb{R}$  (où  $A$  est un sous-ensemble de  $\mathbb{R}$  est dite continue en un point  $a \in A$  si :

$$\lim_{x \rightarrow a} f(x) = f(a)$$

## 5 Le minimum et le maximum d'une fonction

Soit  $f$  une fonction de  $\mathbb{D} \subset \mathbb{R}^n$  dans  $\mathbb{R}$  ( $f : \mathbb{D} \subset \mathbb{R}^n \rightarrow \mathbb{R}$ ) .

### 5.1 Minimum ou maximum local

$f$  admet un minimum (respectivement maximum) local en  $x^* \in \mathbb{D}$  , si et seulement s'il existe un voisinage  $V_\varepsilon(x^*)$  de  $x^*$ , tel que :  $\forall x \in V_\varepsilon(x^*)$  avec :  $\varepsilon$  la taille du voisinage autour du point  $x^*$ ,  $f(x^*) \leq f(x)$  (respectivement  $f(x^*) \geq f(x)$ ) [1].

On dit alors que  $f(x^*)$  est un minimum local de  $f$  dans  $\mathbb{D}$  et  $x^*$  est un point de minimum local.

### 5.2 Minimum ou maximum global

$f$  admet un minimum (respectivement maximum) global en  $x^* \in \mathbb{D}$  si [1] :

$$\forall x \in \mathbb{D}, f(x^*) \leq f(x) \text{ (respectivement } f(x^*) \geq f(x))$$

## 6 Types de problèmes d'optimisation

On peut trouver trois types de problèmes d'optimisation [3] :

### 6.1 Problème d'optimisation sans contraintes

L'objectif est de maximiser ou de minimiser une fonction objectif  $f(x)$  sans aucune restriction ou contrainte sur les valeurs que peuvent prendre les variables de décision  $x$ .

Mathématiquement, le problème s'écrit :

$$\min_x f(x) \quad (2.3)$$

### 6.2 Problème d'optimisation avec contraintes

L'objet est de maximiser ou de minimiser une fonction objectif  $f(x)$  tout en respectant des contraintes sous forme d'égalités ou d'inégalités sur les variables de décision  $x$ .

a) De type égalité :

$$\begin{aligned} & \min_x f(x) \\ \text{Sous contraintes :} & \\ & g_i(x) = 0 \end{aligned} \quad (2.4)$$

b) De type inégalité :

$$\begin{aligned} & \min_x f(x) \\ \text{Sous contraintes :} & \\ & h_i(x) \leq 0 \end{aligned} \quad (2.5)$$

### 6.3 Problème d'optimisation avec contraintes mixtes

L'objectif est de maximiser ou de minimiser une fonction objectif  $f(x)$  tout en respectant à la fois des contraintes sur les variables de décision  $x$  et des contraintes sans variables de décision (contraintes fixes).

Le problème s'écrit :

$$\begin{aligned} & \min_x f(x) \\ \text{Sous contraintes :} & \\ & g_i(x) = 0 \\ & h_i(x) \leq 0 \end{aligned} \quad (2.6)$$

## 7 Exemple de modélisation d'un problème d'optimisation

Une firme fabrique deux produits  $P_1$  et  $P_2$  à l'aide des matières premières  $M_1, M_2$  et  $M_3$ . Le fonctionnement de l'usine est représenté par le tableau suivant :

	$P_1$	$P_2$
$M_1$	2	1
$M_2$	4	2
$M_3$	0	1

TABLE 2.1 – Tableau représentant le fonctionnement de l'usine.

La direction de la firme dispose des matières premières  $M_1, M_2$  et  $M_3$  en quantités respectives 800, 700 et 300. Le profit dû à la fabrication d'une unité de  $P_1$  est égal à 5 euros, et celui d'une unité de  $P_2$  est égal à 6 euros. La tâche de la direction est de faire fonctionner cette usine de manière optimale, c'est-à-dire de rendre le profit maximum tout en respectant les contraintes sur les matières premières.

Le problème de fabrication des deux produits est modélisé comme suit :

Choix des variables de décision.

Soient :

$x_1$  : la quantité de produit  $P_1$

$x_2$  : la quantité de produit  $P_2$

On a le profit de la production d'une unité de  $P_1$  est de 5 euros, et celui de  $P_2$  est de 6 euros, et à partir du tableau (2.1) on a les contraintes sur les matières premières à respecter.

Alors, le problème d'optimisation mathématiquement se présente :

$$\begin{aligned} & \text{Max } 5x_1 + 6x_2 \\ \text{Sous contraintes :} & \\ & 2x_1 + x_2 \leq 800 \\ & 4x_1 + 2x_2 \leq 700 \\ & x_2 \leq 300 \\ & x_1 \geq 0, x_2 \geq 0 \end{aligned} \tag{2.7}$$

# 8 Résolution d'un problème d'optimisation

## 8.1 Optimisation sans contraintes

### Méthode analytique

Considérons le problème d'optimisation sans contraintes suivant [1] :

$$\begin{aligned} & \text{Min ou Max } f(x) \\ & x \in \mathbb{R}^n \end{aligned} \tag{2.8}$$

où :  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  est une fonction donnée.

La méthode analytique est basée sur le calcul des points critiques et l'analyse de la matrice hessienne aux points critiques. Les étapes de cette méthode sont :

#### a) Recherche des points critiques :

Un point  $x^*$  vérifiant  $\nabla f(x^*) = 0$  est appelé point stationnaire ou point critique de  $f$ .

**Le gradient d'une fonction :** Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$ , le gradient de  $f$  noté  $\nabla f(x)$  est donné par :

$$\nabla f(x) = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \\ \cdot \\ \cdot \\ \frac{\partial f}{\partial x_n} \end{bmatrix} \tag{2.9}$$

#### b) La nature du point critique $x^*$ :

La nature du point critique  $x^*$  dépend du hessien de la fonction  $f$ .

Le hessien de  $f$  est défini comme suit :

$$\nabla^2 f(x) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdot & \cdot & \cdot & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdot & \cdot & \cdot & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdot & \cdot & \cdot & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix} \quad (2.10)$$

Alors :

$\nabla^2 f(x)$	$x^*$
Définie positive	Est un point de minimum local
Définie négative	Est un point de maximum local
Indéfinie (les valeurs propres de $\nabla^2 f(x)$ sont $> 0$ et $< 0$ )	Est un point de selle ou point de col ( $x^*$ n'est pas un point d'extremum)
Semi-définie positive	Est un point singulier
Semi-définie négative	Est un point singulier

TABLE 2.2 – La nature du point critique.

### c) Les mineurs principaux :

Une autre méthode pour savoir la nature de la matrice  $\nabla^2 f(x)$  et celle des mineurs principaux. Soit  $A$  une matrice symétrique qui s'écrit comme suit :

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdot & \cdot & \cdot & a_{1n} \\ a_{21} & a_{22} & \cdot & \cdot & \cdot & a_{2n} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ a_{1n} & a_{2n} & \cdot & \cdot & \cdot & a_{nn} \end{bmatrix}$$

Alors :  $\Delta_1 = a_{11}$ ,  $\Delta_2 = \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix}, \dots, \Delta_n = \det(A)$  sont les mineurs principaux de  $A$ . On a :

- Si  $\Delta_k > 0$ ,  $k = 1, \dots, n$  alors  $A$  est définie positive.
- Si  $(-1)^k \Delta_k < 0$ ,  $k = 1, \dots, n$ , alors  $A$  est définie négative.
- Si  $\Delta_k > 0$ ,  $k = 1, \dots, n-1$  et  $\Delta_n = 0$  alors  $A$  est semi-définie positive.
- Si  $(-1)^k \Delta_k > 0$ ,  $k = 1, \dots, n-1$  et  $\Delta_n = 0$  alors  $A$  est semi-définie négative.
- Si  $\Delta_n < 0$  et si  $n$  est pair alors  $A$  est indéfinie.

### Exemple :

Considérons le problème d'optimisation suivant :

$$\min_x f(x) = x_1^2 + x_2^3 + 3x_1x_2 \quad (2.11)$$

Calcul du gradient :

$$\nabla_x f(x_1, x_2) = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \end{bmatrix} = \begin{bmatrix} 2x_1 + 3x_2 \\ 3x_1 + 3x_2^2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (2.12)$$

Alors les solutions sont :  $x_1 = -\frac{9}{4}, x_2 = \frac{3}{2}$  ou  $x_1 = 0, x_2 = 0$

Calcul de la matrice hessienne :

$$\nabla_x^2 f(x_1, x_2) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} \end{bmatrix} = \begin{bmatrix} 2 & 3 \\ 3 & 6x_2 \end{bmatrix} \quad (2.13)$$

Calcul des mineurs principaux de  $\nabla_x^2 f(x_1, x_2)$  :

Avec le point critique  $x_1 = -\frac{9}{4}, x_2 = \frac{3}{2}$  on aura :

$$\nabla_x^2 f\left(-\frac{9}{4}, \frac{3}{2}\right) = \begin{bmatrix} 2 & 3 \\ 3 & 9 \end{bmatrix} \quad (2.14)$$

Alors ces mineurs principaux sont :

$$\Delta_1 = 2 > 0$$

$$\Delta_2 = \det(\nabla_x^2 f(x_1, x_2)) = 9 > 0$$

Donc la matrice hessienne est définie positive, alors ce point critique est un minimum.

Avec le point critique  $x_1 = 0, x_2 = 0$  on aura :

$$\nabla_x^2 f(0, 0) = \begin{bmatrix} 2 & 3 \\ 3 & 0 \end{bmatrix} \quad (2.15)$$

Alors ces mineurs principaux sont :

$$\Delta_1 = 2 > 0$$

$$\Delta_2 = \det(\nabla_x^2 f(x_1, x_2)) = -9 < 0$$

Donc la matrice hessienne est indéfinie, alors ce point critique est un point de selle.

## Méthodes numériques

### Méthode du gradient :

Dans cette méthode, on passe d'un estimé  $x^{(k)}$  de la solution  $x^*$  à l'estimé suivant  $x^{(k+1)}$  par l'algorithme [8] :

$$x^{(k+1)} = x^{(k)} - \alpha^{(k)} \nabla_x f(x^{(k)}) \quad (2.16)$$

Pour  $k = 0, 1, 2, \dots$  le nombre des itérations jusqu'à ce que  $\|\nabla_x f(x^{(k+1)})\| < \varepsilon$  avec :  $\varepsilon$  est la précision souhaitée pour l'algorithme d'optimisation et  $\|\cdot\|$  est la norme Euclidienne.

Si  $x = (x_1, x_2, \dots, x_n)^T$ , alors  $\|x\|_2 = (\sum_{i=1}^n |x_i|^2)^{\frac{1}{2}}$  où  $|\cdot|$  est le module.

Deux techniques existe pour déterminer le scalaire  $\alpha^{(k)}$  appelé aussi le pas.

1. Pas de descente : on choisit le pas  $\alpha^{(k)}$  tel que :  $f(x^{(k+1)}) < f(x^{(k)})$ .
2. Pas de descente optimal : on détermine  $\alpha^{(k)}$  minimisant  $f(x^{(k+1)})$ .

### Méthode de Newton :

Soit :  $f : \mathbb{R}^n \rightarrow \mathbb{R}, f \in \mathbb{C}^2$

Le principe de la méthode de Newton est de calculer une approximation  $p(x)$  de  $f(x)$  autour d'un point  $x^k$  par son développement de Taylor du second ordre, comme suit [1] :

$$f(x) \approx p(x) = f(x^{(k)}) + (x - x^{(k)})^T \nabla f(x^{(k)}) + \frac{1}{2} (x - x^{(k)})^T \nabla^2 f(x^{(k)}) (x - x^{(k)}) \quad (2.17)$$

On calcule alors un nouveau point  $x^{(k+1)}$  qui minimise  $p(x)$ , comme suit :

$$\nabla p(x) = \nabla f(x^{(k)}) + \nabla^2 f(x^{(k)}) (x - x^{(k)}) = 0 \quad (2.18)$$



Si la matrice hessienne  $\nabla^2 f(x^k)$  est inversible alors l'algorithme est le suivant :

$$x^{(k+1)} = x^{(k)} - (\nabla^2 f(x^{(k)}))^{-1} \nabla f(x^{(k)}) \quad (2.19)$$

Les méthodes numériques en optimisation, telles que la méthode du gradient et la méthode de Newton, offrent des outils puissants pour résoudre une variété de problèmes d'optimisation. Cependant, elles présentent également des inconvénients qu'il est important de prendre en compte. Ces inconvénients incluent la dépendance à la précision numérique, la sensibilité aux points de départ, la convergence lente dans certains cas. Ce qui peut conduire à des résultats incorrects si la précision n'est pas suffisante.

## 8.2 Optimisation avec contraintes

**Problèmes avec contraintes égalité :**

**Méthode de multiplicateurs de Lagrange :** Soit le problème d'optimisation avec contrainte égalité suivant [1] :

$$\begin{aligned} & \text{Min } f(x) \text{ (ou Max } f(x)) \\ & \text{sous contraintes :} \\ & g_i(x) = 0, i = 1, \dots, m; m \leq n \\ & x \in \mathbb{R}^n \end{aligned} \quad (2.20)$$

avec :  $m$  le nombre de contraintes et  $n$  le nombre de variables de décision.

Soit la fonction de Lagrange (ou le Lagrangien) de ce problème :

$$L(x, \lambda) = f(x) + \sum_{i=1}^m \lambda_i g_i(x) \quad (2.21)$$

où :  $\lambda_i$  sont appelés les multiplicateurs de Lagrange. Le problème revient à trouver le minimum de  $L(x, \lambda)$ .

$$\text{Soient : } g(x) = \begin{bmatrix} g_1(x) \\ \cdot \\ \cdot \\ g_m(x) \end{bmatrix} \text{ et } \nabla_{(x,\lambda)} L(x, \lambda) = \begin{bmatrix} \frac{\partial L}{\partial x_1} \\ \frac{\partial L}{\partial x_2} \\ \cdot \\ \frac{\partial L}{\partial x_n} \\ \frac{\partial L}{\partial \lambda_1} \\ \frac{\partial L}{\partial \lambda_2} \\ \cdot \\ \frac{\partial L}{\partial \lambda_n} \end{bmatrix}, \text{ avec : } \nabla_{(x,\lambda)} L(x, \lambda) \text{ est le gradient de}$$

$L(x, \lambda)$  par rapport à  $x$  et  $\lambda$ .

### Problème avec contraintes inégalité :

**Conditions de Kuhn-Tucker :** Le problème d'optimisation avec contraintes inégalité s'écrit [8] :

$$\begin{aligned} & \text{Min } f(x) \text{ (ou Max } f(x)) \\ & \text{sous contraintes :} \\ & h_j(x) \leq 0, j = 1, \dots, p \\ & x \in \mathbb{R}^n \end{aligned} \tag{2.22}$$

Pour la résolution de ce type de problème, on utilise les conditions de Kuhn-Tucker, dont son principe est donné comme suit :

Soit la fonction de Lagrange :

$$L(x, \mu) = f(x) + \sum_{j=1}^p \mu_j h_j(x) \tag{2.23}$$

où :  $\mu_j$  sont appelés les multiplicateurs de Lagrange.

$$\text{Soient : } h(x) = \begin{bmatrix} h_1(x) \\ \cdot \\ \cdot \\ h_p(x) \end{bmatrix} \text{ et } \mu = \begin{bmatrix} \mu_1 \\ \cdot \\ \cdot \\ \mu_p \end{bmatrix}$$

Les conditions de Kuhn-Tucker à vérifier pour résoudre ce problème sont présentés dans le tableau (2.3) :

Contraintes	Problème	
	Minimum	Maximum
$h_j(x) \leq 0$	$\nabla_x L(x, \lambda) = 0$ $\nabla_\lambda L(x, \lambda) \leq 0$ $\mu_j h_j(x) = 0$ $\mu_j \geq 0$	$\nabla_x L(x, \lambda) = 0$ $\nabla_\lambda L(x, \lambda) \leq 0$ $\mu_j h_j(x) = 0$ $\mu_j \leq 0$
$h_j(x) \geq 0$	$\nabla_x L(x, \lambda) = 0$ $\nabla_\lambda L(x, \lambda) \geq 0$ $\mu_j h_j(x) = 0$ $\mu_j \leq 0$	$\nabla_x L(x, \lambda) = 0$ $\nabla_\lambda L(x, \lambda) \geq 0$ $\mu_j h_j(x) = 0$ $\mu_j \geq 0$

TABLE 2.3 – Conditions de Kuhn-Tucker.

## 9 Optimisation globale

L'optimisation globale cherche les meilleures solutions pour des problèmes complexes en explorant diverses configurations. Utilisée dans des domaines comme l'ingénierie et la finance, elle emploie des algorithmes avancés pour maximiser les performances ou minimiser les coûts, contribuant ainsi à l'efficacité et à l'innovation.

L'optimisation globale en automatique est utilisée dans divers domaines tels que le contrôle des processus industriels, la robotique, les systèmes embarqués, les réseaux de communication, les systèmes de contrôle de l'énergie et l'environnement, la finance et l'économie, ainsi que la santé et la biotechnologie. Ses applications incluent l'optimisation des performances, la réduction de la consommation d'énergie, la gestion des ressources, la planification stratégique et la résolution de problèmes complexes [5].

### 9.1 Méthodes heuristiques

Ces méthodes exigent des caractéristiques mathématiques spécifiques de la fonction objectif, telles que la continuité, la différentiabilité et la convexité. Elles sont conçues pour localiser la solution avec une très grande précision, souvent au détriment d'un nombre d'itérations élevé. Parmi les méthodes heuristiques, on trouve le recuit simulé, les essaims de particules, la recherche de voisinage, l'algorithme des fourmi, l'algorithme tabou et les algorithmes génétiques.

## 9.2 Méthodes exactes

Les méthodes exactes en optimisation visent à trouver la solution optimale d'un problème tout en garantissant la précision mathématique. Ces approches reposent souvent sur des propriétés formelles de la fonction objectif et des contraintes. Une méthode exacte classique est la programmation linéaire, qui cherche à optimiser une fonction linéaire sous des contraintes linéaires. La formulation mathématique générale est [5] :

$$\begin{aligned} & \text{Min } c^T x \\ & \text{sous la contrainte :} \\ & Ax \leq b \\ & x \geq 0 \end{aligned} \tag{2.24}$$

## 9.3 Méthodes hybrides

Les méthodes d'optimisation globale hybrides combinent différentes approches ou techniques pour améliorer l'efficacité et la robustesse de la résolution des problèmes. En intégrant des méthodes heuristiques, exactes ou d'apprentissage machine, ces approches visent à exploiter les avantages de chaque méthode, surmontant ainsi les limitations individuelles. Les combinaisons hybrides sont souvent adaptées de manière spécifique aux caractéristiques du problème, permettant une recherche plus efficace de solutions de haute qualité dans des contextes complexes [5].

## 9.4 Différence entre les méthodes heuristiques, exactes et hybrides

les méthodes heuristiques offrent une solution rapide mais non garantie, les méthodes exactes garantissent la meilleure solution mais peuvent être très coûteuses en termes de temps de calcul, et les méthodes hybrides cherchent un compromis entre rapidité et qualité de la solution en combinant différentes approches.

Dans ce travail, on propose d'utiliser la méthodes des algorithmes génétiques pour la résolution du problème d'optimisation.

# 10 Principe des algorithmes génétiques (AGs)

Les algorithmes génétiques ont la structure suivante [8] :

1. Initialiser le temps ;
2. Créer une population initiale ;
3. Calculer l'adaptation de chaque individu ;
4. Tant que (il n'y a pas de solution satisfaisante) et (le temps est inférieur au temps limite) faire :

- Incrémenter le temps ;
- Sélectionner les parents ;
- Déterminer les gènes des nouveau-nés par recombinaison des gènes parentaux ;
- Faire subir des mutations aléatoires à la population ;
- Évaluer l'adaptation de chaque individu ;
- Sélectionner les survivants.

## 10.1 Notions de la théorie des algorithmes génétiques

### Chromosome et schéma de représentation

Les algorithmes génétiques utilisent une représentation codée des variables de décision plutôt que les valeurs directes. Cette représentation prend la forme de chaînes de symboles appelées chromosomes, où chaque variable est codée avec une taille de chromosome uniforme dans la population. Ces chromosomes sont constitués de symboles provenant d'un ensemble appelé alphabet, avec l'alphabet binaire ( $\{0, 1\}$ ) étant un choix courant. Ainsi, un chromosome est une séquence de caractères composée de 1 et de 0. Par exemple, pour une fonction à une seule variable, chaque chromosome représente une valeur de cette variable sous une forme codée.

$$x = 3 \Rightarrow \underbrace{\begin{array}{|c|c|c|c|c|c|c|c|} \hline 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ \hline \end{array}}_{\text{Chromosome de taille 8}}$$

Dans le cas d'une fonction à plusieurs variables par exemple :  $x = (x_1, x_2) = (2, 1)$

$$\underbrace{\begin{array}{|c|c|c|c|c|c|c|c|} \hline 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ \hline \end{array}}_{\text{Variable } x_1 \text{ de taille 8}} \underbrace{\begin{array}{|c|c|c|c|c|c|c|c|} \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ \hline \end{array}}_{\text{Variable } x_2 \text{ de taille 8}} \\ \text{Chromosome de vecteur } x \text{ de taille 16}$$

### Création de la population initiale

Lorsqu'on ne connaît pas la solution au problème, la population est générée de manière aléatoire, tandis que si l'on a déjà des informations sur les solutions, on crée des individus représentant ces solutions connues. Cependant, le défi principal réside dans le choix de la taille de la population, notée  $m$ . Les biologistes ont introduit le concept de diversité requise pour refléter le fait qu'une espèce doit avoir une variabilité suffisante pour survivre. Cependant, maintenir une population trop grande peut rallonger le temps de calcul. Ainsi, il est crucial de trouver un compromis optimal entre la diversité nécessaire et la taille de population pour des performances de calcul efficaces.

## Évaluation de l'adaptation

Pour évaluer les performances individuelles, on utilise une fonction d'adaptation qui mesure l'utilité de chaque solution par rapport au problème donné. Cette fonction, également appelée fonction objectif, est basée sur le profit ou le bénéfice que la solution apporte.

## Sélection des parents

Afin d'améliorer les performances de la génération suivante, il est essentiel de permettre l'accouplement des individus les plus performants. Chaque individu a donc une probabilité proportionnelle à son niveau d'adaptation de devenir parent. Pour ce faire, on sélectionne un nombre  $\alpha \in [0, 1]$ , puis pour chaque individu  $x^{(i)}$  de la population, on calcule la quantité suivante :

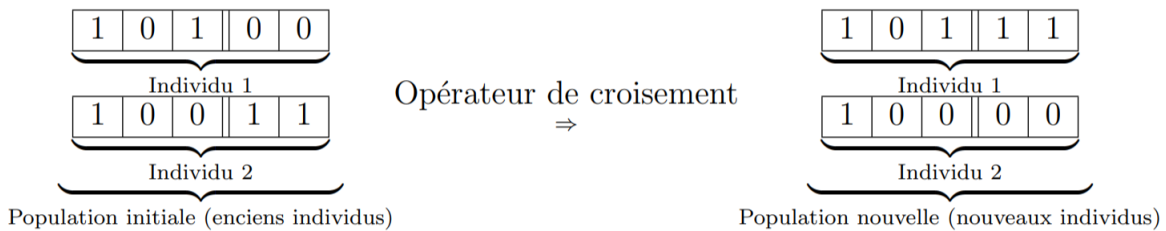
$$P_i = \frac{f(x^{(i)})}{\sum_{j=1}^p f(x^{(j)})} \quad (2.25)$$

Si  $P_i > \alpha$ , sélectionner  $x^{(i)}$ . À partir des éléments sélectionnés, on forme une nouvelle population dont la taille est la même que celle de la population initiale ( $m$ ).

## Opérateurs génétiques

### Recombinaison (crossover)

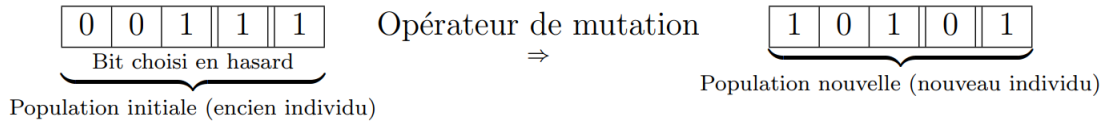
Pour créer un nouvel individu, il ne faut pas simplement copier le génome d'un seul parent, mais plutôt prendre aléatoirement des gènes de chacun des parents. Ce processus, similaire à celui observé dans la nature, est appelé crossover. Il est fondamental car il permet d'explorer diverses combinaisons génétiques et d'élargir ainsi l'ensemble des solutions potentielles.



## Mutation

Les mutations sont des altérations aléatoires du génome. Cependant, il n'est pas souhaitable de muter tous les gènes d'un individu, car cela le rendrait entièrement déterminé de manière aléatoire. Au contraire, il est préférable de modifier une petite partie du génome pour apporter une nouveauté

à l'individu. La mutation est considérée comme ayant un rôle secondaire par rapport à la recombinaison génétique, et elle est généralement associée à une faible probabilité, typiquement de l'ordre de 0,001.



### Sélection des survivants

Cette étape vise à conserver uniquement les solutions les plus prometteuses tout en maintenant une population suffisamment grande et variée. En général, on décide de maintenir la taille de la population d'une génération à l'autre, ce qui implique qu'il y aura autant de décès que de nouveaux-nés. Parfois, on préfère ne conserver que les descendants, ce qui garantit à la fois la diversité nécessaire et l'évolution de la population.

## 11 Exemple de résolution d'un problème d'optimisation par les algorithmes génétiques

Soit le problème de maximisation suivant :

$$\begin{aligned} \max_x f(x) &= x^2 \\ \text{Sous contraintes :} & \\ & 1 \leq x \leq 31 \end{aligned} \tag{2.26}$$

### a) Codage du problème

On choisit ici une population de 4 individus. On détermine la population initiale en tirant au hasard la valeur de chaque allèle. Supposons que l'on obtienne les individus suivants :

Numéro de l'individu	Génotype					Phénotype
1	0	1	1	0	1	13
2	1	1	0	0	0	24
3	0	1	0	0	0	8
4	1	0	0	1	1	19

TABLE 2.4 – Codage du problème.

### b) Évaluation de l'adaptation

Dans ce cas, la fonction d'adaptation (objectif) est simple : il s'agit de la fonction  $f(x) = x^2$ . Ainsi pour calculer l'adaptation du premier individu, il suffit de calculer le carré de 13. On obtient ainsi :

Numéro de l'individu	Génotype					Phénotype	Adaptation	$P_i (\alpha = 0, 1)$
1	0	1	1	0	1	13	169	0,1444
2	1	1	0	0	0	24	576	0,4923
3	0	1	0	0	0	8	64	0,0547
4	1	0	0	1	1	19	361	0,3085

TABLE 2.5 – Évaluation de l'adaptation.

### c) Sélection des parents

Il suffit de tirer au hasard 4 individus parmi la population en tenant compte de leurs adaptations respectives. On obtient, ainsi, une nouvelle population comprenant par exemple :

- Une copie de l'individu 1.
- Une copie de l'individu 4.
- Deux copies de l'individu 2.
- Aucune copie de l'individu 3.

C'est-à-dire :



Numéro de l'individu	Génotype					Phénotype	Adaptation
1	0	1	1	0	1	13	169
2	1	1	0	0	0	24	576
2	1	1	0	0	0	24	576
4	1	0	0	1	1	19	361

TABLE 2.6 – Sélection des parents.

#### d) Recombinaison

On choisit ici, pour le croisement de déterminer au hasard les deux parents, et de couper les chromosomes au hasard : la première partie du chromosome ira au premier descendant, alors que la seconde ira à l'autre. Supposons que l'individu 1 se recombine avec l'individu 2 et l'individu 3 avec l'individu 4. (on note ||| la cassure).

Numéro de l'individu	Avant					Après				
1	0	1	1	0	1	0	1	1	0	0
2	1	1	0	0	0	1	1	0	0	1
2	1	1	0	0	0	1	1	0	1	1
4	1	0	0	1	1	1	0	0	0	0

TABLE 2.7 – Recombinaison.

#### e) Seconde génération

Dans cet exemple, on choisit de limiter la durée de vie de chaque individu à une génération. La nouvelle génération sera donc composée exclusivement des enfants :

Numéro de l'individu	Génotype					Phénotype	Adaptation	$P_i (\alpha = 0, 1)$
1	0	1	1	0	0	12	144	0,0821
2	1	1	0	0	1	25	625	0,3563
3	1	1	0	1	1	27	729	0,4156
4	1	0	0	0	0	16	256	0,1459

TABLE 2.8 – Seconde génération.

Si on calcule la moyenne des valeurs d'adaptation, on choisit 293 pour la génération initiale, contre 439 pour la seconde. Si on réitère le processus, on obtiendra des valeurs de plus en plus grandes, jusqu'à l'obtention de la solution.

## 12 Avantages et inconvénients des algorithmes génétiques

Les algorithmes génétiques offrent un avantage majeur : ils n'exigent pas la connaissance préalable des caractéristiques de la solution recherchée, mais simplement la capacité à déterminer laquelle des deux solutions est meilleure. Toutefois, ces algorithmes sont complexes à programmer (les paramètres tels que la taille de la population et la fonction d'évaluation sont délicats à définir), et ils ont une probabilité très faible, voire nulle, de trouver la solution idéale ; ils se contentent de s'en approcher.

## 13 Conclusion

L'optimisation locale et l'optimisation globale sont des concepts essentiels dans de nombreux domaines, de l'ingénierie à l'économie en passant par les sciences naturelles. L'optimisation locale se concentre sur l'amélioration des solutions dans un contexte donné, tandis que l'optimisation globale vise à trouver la meilleure solution possible dans un espace de recherche plus vaste.

# Chapitre 3

## Résolution d'un problème de commande optimale par la méthode de conversion Bolza-Mayer

### 1 Introduction

Dans ce chapitre, on s'intéresse à la résolution d'un problème de commande optimale de type Bolza. L'idée consiste à convertir ce dernier en un problème de commande optimale de type Mayer, puis à transformer ce dernier en un problème d'optimisation en utilisant la méthode de paramétrisation du vecteur de commande. Ensuite, pour la résolution du problème d'optimisation on applique les algorithmes génétiques pour trouver la solution optimale.

### 2 Conversion Bolza-Mayer

La conversion Bolza-Mayer est une méthode utilisée en optimisation pour transformer un problème de commande optimale donné sous forme de Lagrangien en un problème d'intégrale de coût à minimiser. Cela permet souvent de simplifier le traitement mathématique du problème et de le rendre plus accessible à des méthodes de résolution standard [10].

Pour expliquer ce principe, on considère le problème de commande optimale suivant :

$$\begin{aligned}
 \min_{u(t)} J(u(t)) &= \Psi(x(t_f), t_f) + \int_{t_0}^{t_f} Q(x(t), u(t), t) dt \\
 \text{Sujet à :} & \\
 \dot{x}(t) &= f(x(t), u(t), t) \\
 x(t_0) &= x_0 \\
 x(t_f) &\text{ est libre}
 \end{aligned} \tag{3.1}$$

Dans ce problème de commande optimale (3.1), on a le critère sous la forme Bolza. Pour avoir le critère sous la forme Mayer, on applique la méthode de conversion Bolza-Mayer. Son principe est le suivant :

On suppose un vecteur  $y = \begin{pmatrix} y_1 & y_2 & \dots & y_n & y_{n+1} \end{pmatrix}$ , avec  $n$  est le nombre de variables d'états.

$$i = 1, \dots, n : y_i = x_i \Rightarrow y = \begin{pmatrix} x_1 & x_2 & \dots & x_n & y_{n+1} \end{pmatrix}.$$

Pour calculer la variable  $y_{n+1}$ , la principe de la méthode consiste à supposer la relation suivante :

$$\begin{cases} \dot{y}_{n+1}(t) = Q(x(t), u(t), t) + \frac{\partial \Psi(x(t), t)}{\partial x(t)} f(x(t), u(t), t) + \frac{\partial \Psi(x(t_f), t_f)}{\partial t} \\ y_{n+1}(0) = \Psi(x_0, 0) \end{cases} \tag{3.2}$$

On intègre la première équation de (3.2), on aura :

$$\int_0^{t_f} \dot{y}_{n+1} dt = \int_0^{t_f} Q(x(t), u(t), t) dt + \int_0^{t_f} \frac{\partial \Psi(x(t), t)}{\partial x(t)} f(x(t), u(t), t) dt + \int_0^{t_f} \frac{\partial \Psi(x(t_f), t_f)}{\partial t} dt \tag{3.3}$$

$$\text{Avec : } f(x(t), u(t), t) = \dot{x}(t) = \frac{\partial x(t)}{\partial t}$$

Alors :

$$y_{n+1}(t_f) - y_{n+1}(0) = \int_0^{t_f} Q(x(t), u(t), t) dt + \Psi(x(t_f), t_f) - \Psi(x_0, 0) + t_f \tag{3.4}$$

Avec :

$\Psi(x_0, 0) = y_{n+1}(0)$  et  $t_f$  négligeable ( $\min_x f(x) + a = \min_x f(x)$ ;  $a$  une constante), on obtient :

$$y_{n+1}(t_f) = \Psi(x(t_f), t_f) + \int_{t_0}^{t_f} Q(x(t), u(t), t) dt \quad (3.5)$$

$$= J(u(t)) \quad (3.6)$$

Soit :  $c = (0, 0, \dots, 0, 1)$ , alors :

$$J(u(t)) = y_{n+1}(t_f) = c \cdot y(t_f) \quad (3.7)$$

Alors, le problème de commande optimale (3.1) s'écrit :

$$\begin{aligned} & \min_{u(t)} y_{n+1}(t_f) \\ \text{Sujet à :} & \\ & \dot{x}(t) = f(x(t), u(t), t) \\ & \dot{y}_{n+1}(t) = Q(x(t), u(t), t) + \frac{\partial \Psi(x(t), t)}{\partial x(t)} f(x(t), u(t), t) + \frac{\partial \Psi(x(t_f), t_f)}{\partial t} \\ & x(t_0) = x_0 \\ & y_{n+1}(0) = \Psi(x_0, 0) \end{aligned} \quad (3.8)$$

On voit bien qu'en appliquant la méthode de conversion Bolza-Mayer, le problème de commande optimale Bolza (3.1) devient un problème de commande optimale Mayer (3.8).

## 2.1 Exemple d'application de la méthode de conversion Bolza-Mayer

**Exemple 1 :**

Soit le problème de commande optimale suivant :

$$\begin{aligned} & \max_{u(t)} J(u(t)) = \int_0^{t_f} \left( x(t) - \frac{u^2(t)}{2} \right) dt + \frac{1}{4} [x(t_f)]^2 \\ \text{Sujet à :} & \\ & \dot{x}(t) = u(t) \\ & x(0) = x_0 \end{aligned} \quad (3.9)$$

Dans cet exemple on a un seul état ( $n = 1$ ) donc, on va calculer la deuxième variable ( $y_2$ ).

Nous utilisons (3.2) pour introduire la variable d'état supplémentaire  $y_2(t)$  comme suit :

$$\dot{y}_2(t) = x(t) - \frac{u^2(t)}{2} + \frac{1}{2}x(t)u(t), \quad y_2(0) = \frac{1}{4}x_0^2 \quad (3.10)$$

Alors :

$$y_2(t_f) = y_2(0) + \int_0^{t_f} \left( x(t) - \frac{u^2(t)}{2} + \frac{1}{2}x(t)u(t) \right) dt \quad (3.11)$$

$$= \int_0^{t_f} \left( x(t) - \frac{u^2(t)}{2} \right) dt + \int_0^{t_f} \left( \frac{1}{2}x(t)\dot{x}(t) \right) dt + y_2(0) \quad (3.12)$$

$$= \int_0^{t_f} \left( x(t) - \frac{u^2(t)}{2} \right) dt + \int_0^{t_f} \left( \frac{1}{2}x(t)\frac{dx(t)}{dt} \right) dt + y_2(0) \quad (3.13)$$

$$= \int_0^{t_f} \left( x(t) - \frac{u^2(t)}{2} \right) dt + \int_0^{t_f} \frac{1}{2}x(t)dx + y_2(0) \quad (3.14)$$

$$= \int_0^{t_f} \left( x(t) - \frac{u^2(t)}{2} \right) dt + \left[ \frac{1}{4}x^2(t) \right]_0^{t_f} + y_2(0) \quad (3.15)$$

$$= \int_0^{t_f} \left( x(t) - \frac{u^2(t)}{2} \right) dt + \frac{1}{4}x^2(t_f) - \frac{1}{4}x_0^2 + y_2(0) \quad (3.16)$$

$$= \int_0^{t_f} \left( x(t) - \frac{u^2(t)}{2} \right) dt + \frac{1}{4}x^2(t_f), \quad (x_0 = 0) \quad (3.17)$$

$$= J(u(t)) \quad (3.18)$$

Donc, le problème de commande optimale (3.9) s'écrit comme suit :

$$\begin{aligned} & \max_{u(t)} y_2(t_f) \\ \text{Sujet à :} & \\ & \dot{x}(t) = u(t) \\ & \dot{y}_2(t) = x(t) - \frac{u^2(t)}{2} + \frac{1}{2}x(t)u(t) \\ & x(0) = x_0 \\ & y_2(0) = \frac{1}{4}x_0^2 \end{aligned} \quad (3.19)$$

Pour  $x(t) = y_1(t)$  le problème s'écrit :

$$\begin{aligned} & \max_{u(t)} y_2(t_f) \\ \text{Sujet à :} & \\ & \dot{y}_1(t) = u(t) \\ & \dot{y}_2(t) = y_1(t) - \frac{u^2(t)}{2} + \frac{1}{2}y_1(t)u(t) \\ & y_1(0) = x_0, y_1(t_f) \text{ est libre} \\ & y_2(0) = \frac{1}{4}x_0^2 y_2(t_f) \text{ est libre} \end{aligned} \quad (3.20)$$

## Exemple 2

Considérons le problème de commande optimale sous forme Bolza :

$$\min_{u(t)} J(u(t)) = \frac{1}{2} \int_0^1 (x^2(t) + u^2(t)) dt + \frac{1}{2}x^2(1) \quad (3.21)$$

Sujet à :

$$\dot{x}(t) = -2x(t) + u(t) \quad (3.22)$$

$$x(0) = x_0 \quad (3.23)$$

$$x(1) \text{ est libre} \quad (3.24)$$

Dans cet exemple on a un seul état ( $n = 1$ ) donc, on va calculer la deuxième variable  $y_2$ .

On pose :  $x(t) = y_1(t)$ .

Nous utilisons (3.2) pour introduire la variable d'état supplémentaire  $y_2(t)$  comme suit :

$$\dot{y}_2(t) = \frac{1}{2} (y_1^2(t) + u^2(t)) + y_1(t)(-2y_1(t) + u(t)), y_2(0) = \frac{1}{2}x_0^2 \quad (3.25)$$

Alors :

$$\int_0^1 \dot{y}_2(t) dt = \int_0^1 \frac{1}{2} (y_1^2(t) + u^2(t)) dt + \int_0^{t_f} y_1(t)(-2y_1(t) + u(t)) dt \quad (3.26)$$

À partir de (3.22), on a :

$$u(t) = \dot{y}_1(t) + 2y_1(t) \quad (3.27)$$

Suite aux calculs, on aura :

$$y_2(1) = \int_0^1 \frac{1}{2} (y_1^2(t) + u^2(t)) dt + \frac{1}{2}y_1^2(1) \quad (3.28)$$

La conversion Bolza-Mayer nous conduit au problème de commande optimale sous forme Mayer :

$$\begin{aligned}
 & \min_{u(t)} J(u(t)) = y_2(1) \\
 & \text{Sujet à :} \\
 & \dot{y}_1(t) = -2y_1(t) + u(t) \\
 & \dot{y}_2(1) = \frac{1}{2}(y_1^2(t) + u^2(t)) + y_1(t)(-2y_1(t) + u(t)) \\
 & y_1(0) = x_0, y_1(1) \text{ est libre} \\
 & y_2(0) = \frac{1}{2}x_0^2, y_2(1) \text{ est libre}
 \end{aligned} \tag{3.29}$$

### 3 Paramétrisation du vecteur de commande

#### 3.1 Principe :

La paramétrisation du vecteur de commande consiste à approximer la commande par des combinaisons linéaires ou par des fonctions réelles comme suit [11] :

$$u(t) = \sum_{i=0}^{N_p} a_i v_i(t) \tag{3.30}$$

Avec :  $a_i$  ce sont des coefficients constants,  
 et  $v_i(t)$  des fonctions orthogonales connues.

**Définition d'une fonction orthogonale :** deux fonctions  $f$  et  $g$  sont dites orthogonales sur un intervalle  $[a, b]$  si l'intégrale de leur produit sur cet intervalle est nulle :

$$\int_b^a f(x)g(x)dx = 0$$

Ainsi, le problème de détermination de la commande  $u(t)$  revient à déterminer les paramètres  $a_i$  avec  $i = 0, \dots, N_p$ .

C'est le principe de la commande par paramétrisation du vecteur de commande.

#### 3.2 Les expressions de paramétrisation de la commande

Pour la paramétrisation de la commande, on peut approximer cette dernière comme suit [11] :



## Approximation à l'aide d'un polynôme

$u(t)$  est une constante :

$$u(t) = a_0 \quad (3.31)$$

$u(t)$  est un polynôme d'ordre 1 :

$$u(t) = a_0 + a_1 t \quad (3.32)$$

$u(t)$  est polynôme d'ordre 2 :

$$u(t) = a_0 + a_1 t + a_2 t^2 \quad (3.33)$$

$u(t)$  est polynôme d'ordre  $N_p$  :

$$u(t) = a_0 + a_1 t + a_2 t^2 + \dots + a_{N_p} t^{N_p} \quad (3.34)$$

Avec :  $a_i, i = 0, \dots, N_p$  sont des réels.

## Approximation à l'aide des fonctions trigonométriques

$$u(t) = a \cdot \sin \pi \left( \frac{t}{t_f - t_0} \right) \quad (3.35)$$

Avec :  $a \in \mathbb{R}$ .

## Approximation à l'aide d'une série de Fourier

$$u(t) = \sum_{i=1}^N a_{1i} \sin \left( \frac{it}{t_f - t_0} \right) + a_{2i} \cos \left( \frac{it}{t_f - t_0} \right) \quad (3.36)$$

Avec :  $a_{1i}$  et  $a_{2i}, i = 1, \dots, N_p$  sont des réels.

### 3.3 Exemple de paramétrisation d'un vecteur de commande

Un chariot est libre de se déplacer sur une voie sous l'influence d'une force de commande constante  $f$ . Nous voulons que le chariot se déplace de  $100\text{ m}$  en  $10\text{ s}$ , mais nous voulons également utiliser le moins de force possible. La vitesse initiale  $v$  est nulle, la vitesse finale est libre, la masse du chariot est  $m$  et toutes les forces autres que la commande sont nulles. Quelle est la valeur appropriée de  $f$  ?

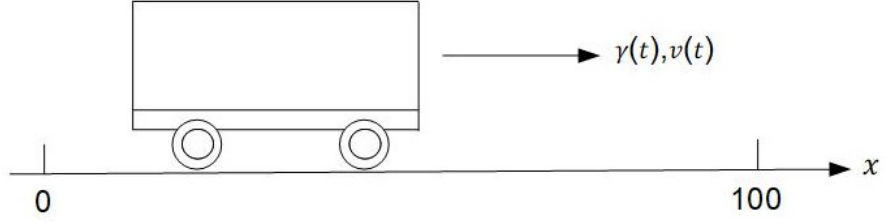


FIGURE 3.1 – Chariot sur une voie.

Le modèle d'état :

Pour la modélisation du système, on applique la loi de Newton :

$$\sum \vec{F}(t) = m\vec{\gamma}(t) \quad (3.37)$$

Avec :  $\gamma(t)$  est l'accélération.

Alors :

$$\mathbf{f} = m.\dot{v} \Rightarrow \dot{v} = \frac{\mathbf{f}}{m} \quad (3.38)$$

et :

$$v = \dot{x}(t) \quad (3.39)$$

On procède à un changement de variables :  $x_1(t) = x(t)$  et  $x_2(t) = \dot{x}(t)$ , alors :

$$\begin{cases} \dot{x}_1(t) = x_2(t) \\ \dot{x}_2(t) = \frac{f(t)}{m} = u(t) \end{cases} \quad (3.40)$$

Les états initiaux :

$$x_1(0) = 0 \quad (3.41)$$

$$x_2(0) = 0 \quad (3.42)$$

Les états finaux :

$$x_1(t_f) \text{ est libre} \quad (3.43)$$

$$x_2(t_f) \text{ est libre} \quad (3.44)$$

Le critère à minimiser :

On veut atteindre un déplacement égale à 100 en un temps  $t_f$  avec l'énergie minimale, alors :

$$J(u(t)) = (x_1(t_f) - 100)^2 + \int_0^{10} u^2(t)dt \quad (3.45)$$

Alors, on aura le problème de commande optimale comme suit :

$$\begin{aligned} \min_{u(t)} J(u(t)) &= (x_1(t_f) - 100)^2 + \int_0^{10} u^2(t)dt \\ \text{Sujet à :} & \\ \dot{x}_1(t) &= x_2(t) \\ \dot{x}_2(t) &= u(t) \\ x_1(0) &= 0, x_2(0) = 0 \\ x_1(t_f) &\text{ est libre, } x_2(t_f) \text{ est libre} \end{aligned} \quad (3.46)$$

Pour expliquer le principe de la technique de la paramétrisation, on propose de résoudre le problème de commande optimale (3.46) en appliquant cette technique afin de le transformer en un problème d'optimisation plus facile à résoudre :

### 1) Approximation à l'aide du polynôme $u(t) = a_0$ :

En remplaçant l'expression de  $u(t)$  dans le problème de commande optimale, on aura :

$$\begin{aligned} \min_{a_0} J(a_0) &= (x_1(t_f) - 100)^2 + \int_0^{10} a_0^2 dt \\ \text{Sujet à :} & \\ \dot{x}_1(t) &= x_2(t) \\ \dot{x}_2(t) &= a_0 \\ x_1(0) &= 0, x_2(0) = 0 \\ x_1(t_f) &\text{ est libre, } x_2(t_f) \text{ est libre} \end{aligned} \quad (3.47)$$

On a :  $\dot{x}_2(t) = a_0$

$$\Rightarrow x_2(t) = \int_0^t a_0 dt = a_0 \cdot t \Big|_0^t + c \quad (3.48)$$

$$\Rightarrow x_2(t) = a_0 t \quad (3.49)$$

avec :  $c = 0$  car les conditions initiales sont nulles.

On a :  $\dot{x}_1(t) = x_2(t)$

$$\Rightarrow \dot{x}_1(t) = a_0.t \quad (3.50)$$

$$\Rightarrow x_1(t) = \int_0^t a_0.t dt \quad (3.51)$$

$$\Rightarrow x_1(t) = \frac{1}{2}a_0t^2 \Big|_0^t = \frac{1}{2}a_0t^2 \quad (3.52)$$

Alors, le critère prend la forme suivante :

$$J(a_0) = \left( \frac{1}{2}a_0t_f^2 - 100 \right)^2 + \int_0^{t_f} a_0^2 dt \quad (3.53)$$

On a :  $t_f = 10$ , en évaluant a partie intégrale, on aura :

$$J(a_0) = (50a_0 - 100)^2 + 10a_0^2 \quad (3.54)$$

Ainsi, le problème de commande optimale devient un problème d'optimisation, dont la variable de décision est  $a_0$  :

Pour la résolution, on va utiliser la méthode analytique,

$$\frac{\partial J}{\partial a_0} = 0 \quad (3.55)$$

Ce qui donne :

$$5020a_0 - 10000 = 0 \quad (3.56)$$

On trouve :

$$a_0 = 1.992 \quad (3.57)$$

Alors, la commande optimale est la suivante :

$$u^*(t) = 1.992 \quad (3.58)$$

et la valeur optimale du critère est :

$$J^* = 39.84 \quad (3.59)$$

**2) Approximation à l'aide du polynôme  $u(t) = a_0 + a_1t$  :**

En remplaçant l'expression de  $u(t)$  dans le problème de commande optimale, on aura :

$$\begin{aligned} \min_a J(a) &= (x_1(t_f) - 100)^2 + \int_0^{10} (a_0 + a_1t)^2 dt \\ \text{Sujet à :} \\ \dot{x}_1(t) &= x_2(t) \\ \dot{x}_2(t) &= a_0 + a_1t \\ x_1(0) &= 0, x_2(0) = 0 \\ x_1(t_f) &\text{ est libre, } x_2(t_f) \text{ est librelibre} \end{aligned} \tag{3.60}$$

$$\text{Avec : } a = \begin{pmatrix} a_0 \\ a_1 \end{pmatrix}.$$

$$\text{On a : } \dot{x}_2(t) = a_0 + a_1t$$

$$\Rightarrow x_2(t) = \int_0^t a_0 + a_1t dt = a_0t \Big|_0^t + a_1 \frac{t^2}{2} \Big|_0^t \tag{3.61}$$

$$\Rightarrow x_2(t) = a_0t + a_1 \frac{t^2}{2} \tag{3.62}$$

$$\text{On a : } \dot{x}_1(t) = x_2(t)$$

$$\Rightarrow \dot{x}_1(t) = a_0t + a_1 \frac{t^2}{2} \tag{3.63}$$

$$\Rightarrow x_1(t) = \int_0^t a_0t + a_1 \frac{t^2}{2} dt \tag{3.64}$$

$$\Rightarrow x_1(t) = a_0 \frac{t^2}{2} \Big|_0^t + a_1 \frac{t^3}{6} \Big|_0^t \tag{3.65}$$

$$\Rightarrow x_1(t) = \frac{1}{2}a_0t^2 + \frac{1}{6}a_1t^3 \tag{3.66}$$

Alors, le critère devient :

$$J(a) = \left( \frac{1}{2}a_0t_f^2 + \frac{1}{6}a_1t_f^3 - 100 \right)^2 + \int_0^{t_f} (a_0 + a_1t)^2 dt \tag{3.67}$$

On a :  $t_f = 10$ , en évaluant le terme intégral, on aura :

$$J(a) = (50a_0 + 166.67a_1 - 100)^2 + 10a_0^2 + 100a_0a_1 + 333.33a_1^2 \tag{3.68}$$

Ainsi, le problème de commande optimale devient un problème d'optimisation, dont les variables de décision sont  $a_0$  et  $a_1$  :

On résout l'équation suivante :

$$\nabla_a J = 0 \quad (3.69)$$

On aura :

$$\frac{\partial J}{\partial a_0} = 5020a_0 + 16767a_1 - 10000 = 0 \quad (3.70)$$

$$\frac{\partial J}{\partial a_1} = 16767a_0 + 56224a_1 - 33334 = 0 \quad (3.71)$$

On l'écrit sous la forme matricielle :

$$\begin{bmatrix} 5020 & 16767 \\ 16767 & 56224 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \end{bmatrix} = \begin{bmatrix} 10000 \\ 33334 \end{bmatrix} \quad (3.72)$$

Alors :

$$\begin{bmatrix} a_0 \\ a_1 \end{bmatrix} = \begin{bmatrix} 5020 & 16767 \\ 16767 & 56224 \end{bmatrix}^{-1} \begin{bmatrix} 10000 \\ 33334 \end{bmatrix} \quad (3.73)$$

On trouve :

$$a_0 = 2.9930 \quad (3.74)$$

$$a_1 = -0.2997 \quad (3.75)$$

Alors, l'expression de la commande optimale est :

$$u^*(t) = 2.9930 - 0.2997t \quad (3.76)$$

et la valeur optimale du critère est :

$$J^* = 29.9106 \quad (3.77)$$

**3) Approximation à l'aide du polynôme  $u(t) = a_0 + a_1t + a_2t^2$  :**

En remplaçant l'expression de  $u(t)$  dans le problème de commande optimale, on aura :

$$\begin{aligned} \min_a J(a) &= (x_1(t_f) - 100)^2 + \int_0^{10} (a_0 + a_1t + a_2t^2)^2 dt \\ \text{Sujet à :} \\ \dot{x}_1(t) &= x_2(t) \\ \dot{x}_2(t) &= a_0 + a_1t + a_2t^2 \\ x_1(0) &= 0, x_2(0) = 0 \\ x_1(t_f) &\text{ est libre, } x_2(t_f) \text{ est libre} \end{aligned} \quad (3.78)$$

$$\text{Avec : } a = \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix}.$$

$$\text{On a : } \dot{x}_2(t) = a_0 + a_1t + a_2t^2$$

$$\Rightarrow x_2(t) = \int_0^t a_0 + a_1t + a_2t^2 dt = a_0t \Big|_0^t + a_1 \frac{t^2}{2} \Big|_0^t + a_2 \frac{t^3}{3} \Big|_0^t \quad (3.79)$$

$$\Rightarrow x_2(t) = a_0t + a_1 \frac{t^2}{2} + a_2 \frac{t^3}{3} \quad (3.80)$$

$$\text{On a : } \dot{x}_1(t) = x_2(t)$$

$$\Rightarrow \dot{x}_1(t) = a_0t + a_1 \frac{t^2}{2} + a_2 \frac{t^3}{3} \quad (3.81)$$

$$\Rightarrow x_1(t) = \int_0^t a_0t + a_1 \frac{t^2}{2} + a_2 \frac{t^3}{3} dt \quad (3.82)$$

$$\Rightarrow x_1(t) = a_0 \frac{t^2}{2} \Big|_0^t + a_1 \frac{t^3}{6} \Big|_0^t + a_2 \frac{t^4}{12} \Big|_0^t \quad (3.83)$$

$$\Rightarrow x_1(t) = \frac{1}{2}a_0t^2 + \frac{1}{6}a_1t^3 + \frac{1}{12}a_2t^4 \quad (3.84)$$

Alors, le critère devient :

$$J(a) = \left( \frac{1}{2}a_0t_f^2 + \frac{1}{6}a_1t_f^3 + \frac{1}{12}a_2t_f^4 - 100 \right)^2 + \int_0^{t_f} (a_0 + a_1t + a_2t^2)^2 dt \quad (3.85)$$

On a :  $t_f = 10$ , en évaluant la partie intégrale, on aura :

$$\begin{aligned} J(a) &= (50a_0 + 166.67a_1 + 833.33a_2 - 100)^2 + 10a_0^2 + 100a_0a_1 \\ &\quad + 666.66a_0a_2 + 333.33a_1^2 + 5000a_1a_2 + 20000a_2^2 \end{aligned} \quad (3.86)$$

Ainsi, le problème de commande optimale devient un problème d'optimisation, dont les variables de décision sont  $a_0$ ,  $a_1$  et  $a_2$  :

On résout l'équation suivante :

$$\nabla_a J = 0 \quad (3.87)$$

On aura :

$$\frac{\partial J}{\partial a_0} = 5020a_0 + 16767a_1 + 83999.66a_2 - 10000 = 0 \quad (3.88)$$

$$\frac{\partial J}{\partial a_1} = 16767a_0 + 56224.43a_1 + 282782.22a_2 - 33334 = 0 \quad (3.89)$$

$$\frac{\partial J}{\partial a_2} = 83999.66a_0 + 282782.22a_1 + 1428873.33a_2 - 166666 = 0 \quad (3.90)$$

On l'écrit sous la forme matricielle :

$$\begin{bmatrix} 5020 & 16767 & 83999.66 \\ 16767 & 56224.43 & 282782.22 \\ 83999.66 & 282782.22 & 1428873.33 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} 10000 \\ 33334 \\ 166666 \end{bmatrix} \quad (3.91)$$

Alors :

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} 5020 & 16767 & 83999.66 \\ 16767 & 56224.43 & 282782.22 \\ 83999.66 & 282782.22 & 1428873.33 \end{bmatrix}^{-1} \begin{bmatrix} 10000 \\ 33334 \\ 166666 \end{bmatrix} \quad (3.92)$$

On trouve :

$$a_0 = 2.9917 \quad (3.93)$$

$$a_1 = -0.2995 \quad (3.94)$$

$$a_2 = 0 \quad (3.95)$$

Alors, l'expression de la commande optimale est :

$$u^*(t) = 2.9917 + -0.2995t \quad (3.96)$$

et la valeur optimale du critère est :

$$J^* = 29.9117 \quad (3.97)$$

Résumons les résultats dans le tableau (3.1) ci-dessous :



Paramétrisation	L'expression de la commande	Critère optimal
$u(t) = a_0$	$u(t) = 1.992$	39.84
$u(t) = a_0 + a_1t$	$u(t) = 2.9930 - 0.2997t$	29.9106
$u(t) = a_0 + a_1t + a_2t^2$	$u(t) = 2.9917 - 0.2995t$	29.9117

TABLE 3.1 – Résultats de synthèse d'un problème de commande optimale à l'aide de paramétrisation du vecteur de commande.

D'après les résultats de simulation, on a constaté que même en augmentant l'ordre du polynôme, la valeur du critère ne s'améliore pas, donc la solution optimale est le polynôme d'ordre 1, c'est-à-dire :

$$u^*(t) = 2.9930 - 0.2997t \quad (3.98)$$

## 4 Synthèse d'un problème de commande optimale basée sur la conversion Bolza-Mayer

Dans cette section, on s'intéresse à la résolution d'un problème de commande optimale à l'aide de la paramétrisation basée sur la conversion Bolza-Mayer .

Soit un problème de commande optimale sous forme Bolza :

$$\begin{aligned}
 \max_{u(t)} J(u(t)) &= \Psi(x(t_f), t_f) + \int_{t_0}^{t_f} \varphi(x(t), u(t), t) dt \\
 \text{Sujet à :} & \\
 \dot{x}(t) &= f(x(t), u(t), t) \\
 x(t_0) &= x_0 \\
 x(t_f) &\text{ est libre}
 \end{aligned} \quad (3.99)$$

Pour la résolution de ce problème de commande optimale, on va suivre les étapes suivantes :

**Étape 1 :** La conversion Bolza-Mayer, nous conduit au problème de commande optimale sous forme Mayer :

$$\begin{aligned}
 & \underset{u(t)}{\max} y_{n+1}(t_f) \\
 \text{Sujet à :} & \\
 & \dot{x}(t) = f(x(t), u(t), t) \\
 & \dot{y}_{n+1}(t) = \varphi(x(t), u(t), t) + \frac{\partial \psi(x(t), t)}{\partial x(t)} f(x(t), u(t), t) + \frac{\partial \psi(x(t_f), t_f)}{\partial t} \\
 & x(t_0) = x_0 \\
 & y_{n+1}(0) = \psi(x_0, 0)
 \end{aligned} \tag{3.100}$$

**Étape 2 :** La paramétrisation de la commande optimale devient :

$$u(t) = \sum_{i=0}^{N_p} a_i v_i(t) \tag{3.101}$$

Avec :  $a_i$  ce sont des coefficients constantes,  
et  $v_i(t)$  des fonctions orthogonales connues.

Alors, on remplace l'expression (3.101) dans le problème (3.100).

$$\begin{aligned}
 & \underset{u(t)}{\max} y_{n+1}(t_f) \\
 \text{Sujet à :} & \\
 \text{labelsom} & \\
 & \dot{x}(t) = f(x(t), \sum_{i=0}^{N_p} a_i v_i(t), t) \\
 & \dot{y}_{n+1} = \varphi(x(t), \sum_{i=0}^{N_p} a_i v_i(t), t) + \frac{\partial \psi(x(t), t)}{\partial x(t)} f(x(t), \sum_{i=0}^{N_p} a_i v_i(t), t) + \frac{\partial \psi(x(t_f), t_f)}{\partial t} \\
 & x(t_0) = x_0 \\
 & y_{n+1}(0) = \psi(x_0, 0)
 \end{aligned} \tag{3.102}$$

**Étape 3 :** La résolution des équations différentielles (le modèle d'état) nous donne la solution de l'état :

$$x(t) = F(a_i, t) \tag{3.103}$$

Et en remplaçant (3.103) dans (??), le problème d'optimisation s'écrit :

$$\underset{a_0, a_1, \dots, a_i}{\max} Z(a_0, a_1, \dots, a_i) \tag{3.104}$$

**Étape 4 :** Application des algorithmes génétiques pour la résolution du problème (3.104).

Avec : la syntaxe Matlab "ga" :

`x = ga(fun,nvars)`

`x = ga(fun,nvars,A,b)`

`x = ga(fun,nvars,A,b,Aeq,beq)`

`x = ga(fun,nvars,A,b,Aeq,beq,lb,ub)`

`x = ga(fun,nvars,A,b,Aeq,beq,lb,ub,nonlcon)`

`x = ga(fun,nvars,A,b,Aeq,beq,lb,ub,nonlcon,options)`

`x = ga(fun,nvars,A,b,Aeq,beq,lb,ub,nonlcon,intcon)`

`x = ga(fun,nvars,A,b,Aeq,beq,lb,ub,nonlcon,intcon,options)`

`x = ga(problem) [x,fval] = ga(_____)`

`[x,fval,exitflag,output] = ga(_____)`

`[x,fval,exitflag,output,population,scores] = ga(_____)`

## 5 Exemples d'application sur la synthèse d'un problème de commande optimale basée sur la conversion Bolza-Mayer

### 5.1 Exemple 1

Soit le problème de commande optimale sous forme Bolza :

$$\begin{aligned} \underset{u(t)}{\max} J(u(t)) &= \int_0^{t_f} \left( x(t) - \frac{u^2(t)}{2} \right) dt + \frac{1}{4}[x(t_f)^2] \\ \text{Sujet à :} & \\ \dot{x}(t) &= u(t) \\ x(0) &= x_0 \end{aligned} \tag{3.105}$$

Pour la résolution, on va suivre les étapes de la section précédente.

**Étape 1 :** Après la conversion Bolza-Mayer vu dans le problème (3.9) et en posant  $x(t) = y_1(t)$  et  $y_1(0) = x_0 = 0$ , avec  $t_f = 1$ , on obtient le problème de commande optimale sous forme de Mayer :

$$\max_{u(t)} J(u(t)) = y_2(t_f)$$

*Sujet à :*

$$\dot{y}_1(t) = u(t) \tag{3.106}$$

$$\dot{y}_2(t) = y_1(t) - \frac{u^2(t)}{2} + \frac{1}{2}y_1(t)u(t)$$

$$y_1(0) = 0, y_1(t_f) \text{ est libre}$$

$$y_2(0) = 0, y_2(t_f) \text{ est libre}$$

**Étape 2 :** Ensuite, on utilise la méthode de la paramétrisation du vecteur de commande pour avoir ce problème sous forme d'un problème d'optimisation.

Pour la paramétrisation, on commence par un polynôme 0 i.e  $u(t) = a_0$  :

$$\max_a J(a) = y_2(t_f)$$

*Sujet à :*

$$\dot{y}_1(t) = a_0 \tag{3.107}$$

$$\dot{y}_2(t) = y_1(t) - \frac{a_0^2}{2} + \frac{1}{2}y_1(t)a_0$$

$$y_1(0) = 0, y_2(0) = 0$$

$$y_1(t_f) \text{ est libre}, y_2(t_f) \text{ est libre}$$

**Étape 3 :** Après la résolution des équations différentielles, on aura :

$$y_1(t) = a_0 t \tag{3.108}$$

$$y_2(t_f) = \left(\frac{1}{2}a_0 + \frac{1}{4}a_0^2\right)t_f^2 - \frac{1}{2}a_0^2 t_f \tag{3.109}$$

En remplaçant (3.109) dans l'expression du critère, le problème de commande optimale devient un problème d'optimisation qui s'écrit :

$$\max_{a_0} J(a_0) = \frac{1}{2}a_0 - \frac{1}{4}a_0^2 \tag{3.110}$$

**Étape 5 :** En appliquant les algorithmes génétiques, on obtient :

$$a_0 = 1.0009 \tag{3.111}$$

Donc :

$$u^*(t) = 1.0009 \tag{3.112}$$

Et :

$$J^* = 0.25 \quad (3.113)$$

Les autres cas de paramétrisation de la loi de commande sont résumés dans le tableau (3.2) ci-dessous :

Paramétrisation	L'expression de la commande	Critère optimal
$u(t) = a_0$	$u(t) = 1.0009$	0.25
$u(t) = a_0 + a_1t$	$u(t) = 26.6085 + 12.5642t$	-261.6290
$u(t) = a_0 + a_1t + a_2t^2$	$u(t) = 23.9937 + 9.7706t + 8.4902t^2$	-251.9936

TABLE 3.2 – Résultats de la paramétrisation de l'exemple 1.

Dans ce problème de commande optimale, on cherche à maximiser le critère et d'après les résultats de simulation, on a constaté que la solution optimale est :

$$u^*(t) = 26.6085 + 12.5642t \quad (3.114)$$

La figure 3.2 présente l'évolution de l'état du système  $x(t)$  :

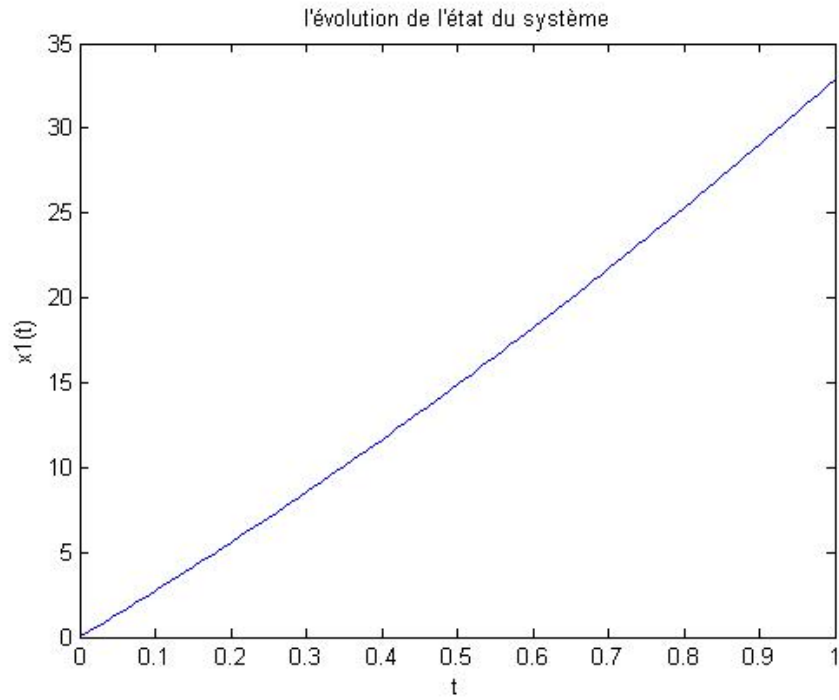


FIGURE 3.2 – L'évolution de l'état du système.

La figure 3.3 présente l'évolution de la commande optimale  $u^*(t)$  :

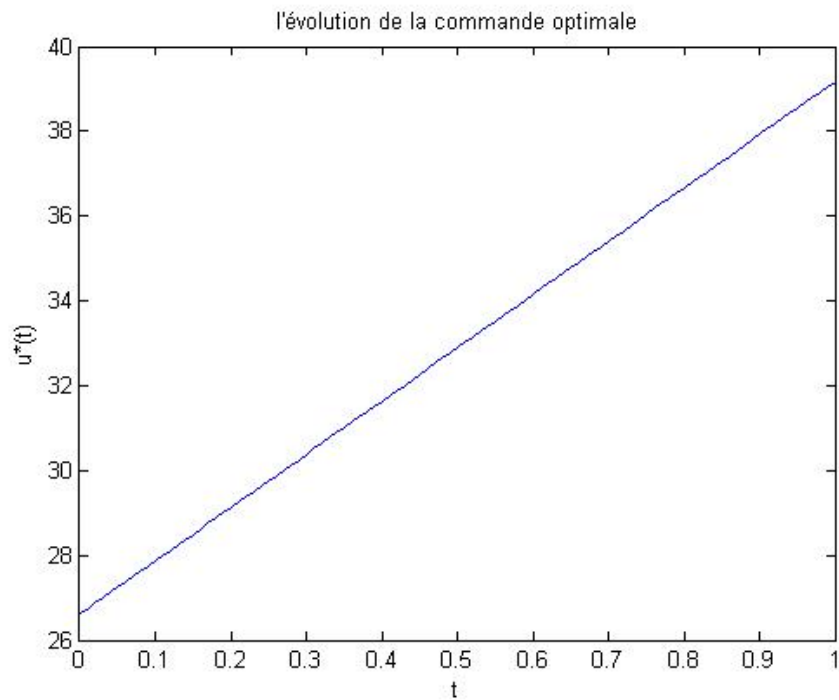


FIGURE 3.3 – L'évolution de la commande optimale.

## 5.2 Exemple 2

Soit un problème de commande optimale sous forme Bolza :

$$\begin{aligned} \min_{u(t)} J(u(t)) &= \frac{1}{2} \int_0^1 (u^2(t) + x^2(t)) dt + \frac{1}{2} x^2(1) \\ \text{Sujet à :} & \\ \dot{x}(t) &= -2x(t) + u(t) \\ x(0) &= x_0, x(1) \text{ est libre} \end{aligned} \tag{3.115}$$

On pose :  $x(t) = y_1(t)$ ,  $x_0 = 0$ .

La conversion Bolza-Mayer nous conduit au problème de commande optimale sous forme Mayer :

$$\begin{aligned} \min_{u(t)} J(u(t)) &= y_2(1) \\ \text{Sujet à :} & \\ \dot{y}_1(t) &= -2y_1(t) + u(t) \\ \dot{y}_2(t) &= \frac{1}{2} (u^2(t) + y_1^2(t)) + y_1(t)(-2y_1(t) + u(t)) \\ y_1(0) &= x_0, y_1(1) \text{ est libre} \\ y_2(0) &= \frac{1}{2} x_0^2, y_2(1) \text{ est libre} \end{aligned} \tag{3.116}$$

Pour la paramétrisation, on approxime la commande d'abord à l'aide d'une constante :  $u(t) = a_0$  :

Dans ce cas, on aura :

$$\begin{aligned} \min_{u(t)} J(u(t)) &= y_2(1) \\ \text{Sujet à :} & \\ \dot{y}_1(t) &= -2y_1(t) + a_0 \\ \dot{y}_2(t) &= \frac{1}{2} (a_0^2 + y_1^2(t)) + y_1(t)(-2y_1(t) + a_0) \\ y_1(0) &= y_{10}, y_1(1) \text{ est libre} \\ y_2(0) &= \frac{1}{2} x_0^2, y_2(1) \text{ est libre} \end{aligned} \tag{3.117}$$

Après la résolution des équations différentielles, on aura :

$$y_1(t) = -\frac{a}{2(e^{2t} - 1)} \quad (3.118)$$

$$y_2(t_f) = \frac{1}{32} \left( 20t_f - \frac{4}{e^{2t_f}} + \frac{3}{e^{4t_f}} + 1 \right) a_0^2 \quad (3.119)$$

En remplaçant (3.119) dans l'expression du critère, le problème de commande optimale devient un problème d'optimisation qui s'écrit :

$$\min_{a_0} J(a_0) = -0.6411a_0^2 \quad (3.120)$$

On résout ce problème d'optimisation par les algorithmes génétiques :

$$a_0 = -0.1258 \quad (3.121)$$

Donc :

$$J^* = 0.1229 \quad (3.122)$$

Et :

$$u^* = -0.1258 \quad (3.123)$$

Les différents résultats de paramétrisation de  $u(t)$  sont résumés dans le tableau (3.3) ci-dessous :

Paramétrisation	L'expression de la commande	Critère optimal
$u(t) = a_0$	$u(t) = -0.1258$	0.1229
$u(t) = a_0 + a_1t$	$u(t) = -12.0319 + 16.9338t$	19.2470
$u(t) = a_0 + a_1t + a_2t^2$	$u(t) = -6.9181 - 5.4381t + 18.8163t^2$	14.9436

TABLE 3.3 – Résultats de la paramétrisation de l'exemple 2.



D'après les résultats du tableau la valeur minimale correspond au premier cas, c'est-à-dire :

$$u^*(t) = -0.1258 \quad (3.124)$$

La figure 3.4 présente l'évolution de l'état du système  $x(t)$  :

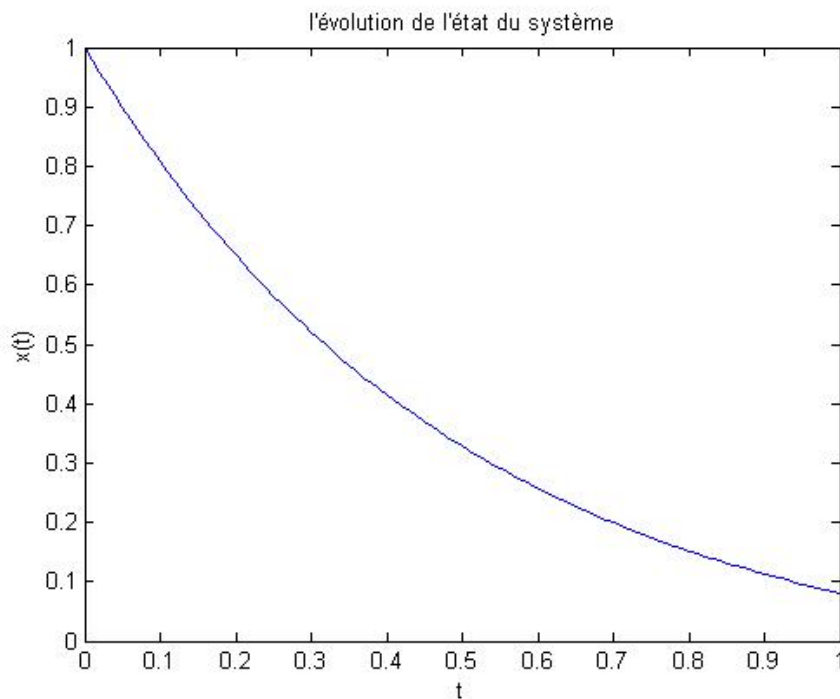


FIGURE 3.4 – L'évolution de l'état du système.

La figure 3.5 présente l'évolution de la commande optimale  $u^*(t)$  :

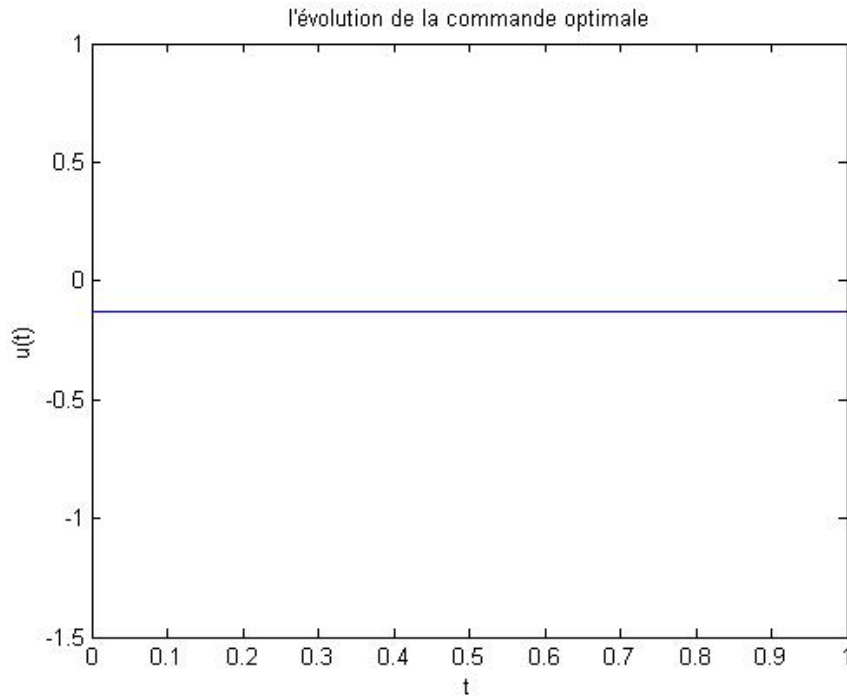


FIGURE 3.5 – L'évolution de la commande optimale.

## 6 Conclusion

Ce chapitre a présenté le principe de la conversion Bolza-Mayer et la technique de paramétrisation du vecteur de commande pour simplifier les problèmes de commande optimale, les transformant en problèmes d'optimisation. Nous avons démontré l'application de ces techniques à travers des exemples pratiques et montré l'efficacité des algorithmes génétiques pour obtenir des solutions optimales.

# Conclusion générale

Le travail présenté dans ce mémoire, porte sur la résolution d'un problème de commande optimale à l'aide de la paramétrisation du vecteur de commande basée sur la conversion Bolza-Mayer.

Au premier lieu, on a abordé les principes fondamentaux de la commande optimale. Cette théorie permet de trouver la meilleure façon de contrôler un système dynamique en optimisant un critère de performance. Nous avons examiné les éléments clés tels que le modèle du système, les conditions initiale et finale, les contraintes et les objectifs à optimiser. Ensuite, nous avons présenté les différentes méthodes de résolution, comme le calcul des variations, le principe de Pontryagin et la commande linéaire quadratique. Ces méthodes fournissent des outils efficaces pour concevoir des systèmes de contrôle performants dans divers domaines.

Ensuite, en optimisation, on cherche à améliorer les performances des systèmes automatiques en trouvant le meilleur compromis entre différentes variables sous contraintes. On utilise des méthodes mathématiques et des algorithmes comme les algorithmes génétiques, pour résoudre ces problèmes de manière efficace et précise. En explorant diverses approches, on peut trouver des solutions optimales pour différents types de problèmes, ce qui contribue à l'efficacité et à l'innovation dans de nombreux domaines.

Enfin, nous avons vu que la conversion Bolza-Mayer rend les problèmes de commande optimale plus simples. En utilisant la paramétrisation du vecteur de commande on transforme ces problèmes de commande optimale en problèmes d'optimisation. Alors, on peut efficacement résoudre ces problèmes, par exemple en utilisant les algorithmes génétiques. C'est ce qu'on a démontré par des exemples d'application et prouvé l'efficacité de cette approche par les résultats obtenus.

En conclusion, la résolution des problèmes de la commande optimale basée sur la conversion Bolza-Mayer et la paramétrisation du vecteur de commande offre la perspective d'étendre cette approche pour la résolution des problèmes de commande optimale des systèmes d'ordre fractionnaire.

# Bibliographie

- [1] N.DJEGHALI. Cours d'optimisation, Département Automatique, UMMTO, 2020-2021.
- [2] N.DJENNANE. *Résolution d'un problème de commande optimale par la méthode de l'itération sur le vecteur de commande*, Mémoire de fin d'études, Département Automatique, UMMTO, 2017.
- [3] Y.DODGE. *Optimisation appliquée*, Springer-Verlag France, 2005.
- [4] F.FRAH et F.ABRIKA. *Synthèse d'une commande optimale par paramétrisation du vecteur de commande*, Mémoire de fin d'études, Département Automatique, UMMTO, 2016.
- [5] G.IDIRI. *Commande prédictive des systèmes non linéaires dynamiques*, Mémoire de Magister, Département Automatique, UMMTO, 2011.
- [6] G.IDIRI. *Développement de méthodes de résolution de problèmes de contrôle optimale des systèmes d'ordre fractionnaire*, Thèse de doctorat, Département Automatique, UMMTO, 2017.
- [7] S.LOUNES et S.OULD MAHAMMED. *Identification des Systèmes Dynamiques Par l'Optimisation Globale*, Mémoire de fin d'études, Département Automatique, UMMTO, 2019.
- [8] A.MAIDI. Cours d'optimisation et commande optimale, Département Automatique, UMMTO, 2005.
- [9] D.S.NAIDU. *Optimal control systems*, Idaho.USA, 2003.
- [10] P.SETHI. *Optimal Control theory : Application to Management Science and Economics*. Springer, Switzerland, 2021.
- [11] R.F.STENGEL. *Optimal Control and Estimation*. Dover Publications, New York, 1994.

## **Mots-clefs :**

Commande optimale, optimisation statique et globale, algorithmes génétiques, conversion Bolza-Mayer, paramétrisation du vecteur de commande.

## **Résumé :**

Dans ce mémoire, nous intéresserons à la résolution d'un problème de commande optimale pour les systèmes dynamiques, en convertissant un problème de commande optimale sous forme Bolza en un problème de commande optimale sous forme Mayer et en transformant ce dernier en un problème d'optimisation en utilisant la méthode de paramétrisation du vecteur de commande. Pour la résolution de ce problème d'optimisation, nous allons appliquer les algorithmes génétiques.