

UNIVERSITÉ MOULOUD MAMMARI DE TIZI-OUZOU
FACULTÉ DES SCIENCES
DÉPARTEMENT DE MATHÉMATIQUES



MÉMOIRE DE MASTER

En Mathématiques

Option : Recherche Opérationnelle

THÈME

Analyse et résolution des problèmes de contrôle optimal bi niveaux : Méthodes et applications.

PRÉSENTÉ PAR :

M.ZAZI Mohamed

DEVANT LE JURY COMPOSÉ DE :

M. AIDENE	Mohamed	Professeur	UMMTO	Président
Mme. HARRACHE	Fazia	M. C. B	UMMTO	Encadrante
M. AOUANE	Mohouhand	M. A. A.	UMMTO	Examinateur
Mme. AKLI	Meriem	M. A. A	UMBB	Invitée

Juillet 2025

Table des matières

1	Concepts fondamentaux de la programmation mathématique :	6
1.1	Convexité :	6
1.1.1	Ensembles convexes :	6
1.1.2	Fonctions convexes :	9
1.2	Programmation Mathématique :	10
1.2.1	Optimisation sous contraintes :	10
2	Programmation mathématique bi-niveaux :	21
2.1	Programmation mathématique bi-niveaux :	21
2.1.1	Formulation du problème bi-niveaux :	21
2.1.2	Étude du problème bi-niveaux :	23
2.1.3	Multiplicité des solutions optimales de second niveau :	24
2.1.4	Programmation linéaire bi-niveaux :	27
2.1.5	Condition d'optimalité :	35
2.1.6	Méthode de résolution :	41
3	Contrôle optimal bi niveaux :	44
3.1	Rappel sur le Contrôle optimal :	44
3.1.1	Système dynamique :	45
3.1.2	Système de contrôle :	46
3.1.3	Contrôlabilité :	51
3.1.4	Principe du maximum de Pontryagin (P.M.P) :	52
3.2	Problème du Contrôle optimale bi niveaux :	54
3.2.1	Analyse d'un exemple :	57

Remerciements :

Je remercie en premier lieu Dieu le tout puissant qui m'a donné la santé et la volonté pour réaliser ce modeste travail.

Je tiens à remercier ma promotrice Mme. *Harrache Fazia* pour m'avoir proposer ce sujet de mémoire, pour ses précieux conseils, et l'énorme soutien qu'elle m'a offert tout au long de ce travail.

Je remercie également M. *Aidene Mohamed* qui me fait l'honneur de présider le jury de ma soutenance, ainsi que M. *Aouane Mohouhand* et Mme. *Akli Meriem* qui me font l'honneur d'examiner ce travail.

Je tiens aussi à remercier tous les enseignants de l'université *Mouloud Mammeri* de Tizi-Ouzou, en particulier les enseignants de la faculté des sciences, pour l'enseignement exemplaire qu'il m'ont fourni lors de mes études universitaires, ce qui m'a énormément aidé pour arriver là où je suis aujourd'hui.

Je remercie ma famille pour le soutien inconditionnel qu'il m'ont apporté à travers mon éducation et mes études au fil des ans.

Je remercie finalement mes amis et mes camarades pour leur soutien moral et intellectuel qu'ils m'ont apporté.

Introduction :

L'optimisation mathématique est un domaine central de la programmation mathématique. Elle vise à exploiter divers outils mathématiques pour obtenir la meilleure solution possible d'un modèle mathématique quelconque ou bien à s'en approcher. Ces modèles représentent souvent des problématiques rencontrées dans le monde réel. Leurs solutions peuvent nous faire gagner une quantité considérable de ressources ainsi que du temps [1, 2, 3, 4].

L'histoire de l'optimisation mathématique remonte aux travaux fondateurs de *Lagrange* (1736-1813) sur les multiplicateurs pour les problèmes avec contraintes, et à ceux de *Cauchy* (1789-1857) qui proposa les premières méthodes itératives de descente. Cependant, c'est avec la formalisation de la programmation linéaire par *Dantzig* en 1947 (algorithme du simplexe) et les développements ultérieurs en théorie de la dualité (*Kuhn et Tucker*, 1951) que ce domaine prend son essor moderne [4, 5].

Dans les années 1950-1960, le domaine de l'optimisation connaît d'importants développements avec l'apparition de nouvelles méthodes. La programmation non linéaire permet alors de résoudre des problèmes plus complexes que les modèles linéaires. Parallèlement, la programmation dynamique est introduite pour traiter les décisions s'étalant sur plusieurs étapes. On voit aussi émerger les premiers modèles hiérarchiques, inspirés par la théorie des jeux qui prennent en compte les interactions entre différents décideurs. Ces avancées trouvent rapidement des applications pratiques, notamment en économie ou en ingénierie.

Un problème est dit hiérarchique lorsqu'il modélise une situation à plusieurs décideurs ordonnés dans une structure hiérarchique. Ces décideurs visent tous à améliorer leurs bénéfices, sachant que leurs décisions influencent celles des autres décideurs et sont influencées par celles-ci. Pour atteindre cet objectif ces décideurs doivent contrôler toutes variables de décisions tout en tenant compte des interactions entre eux. Un problème à plusieurs niveaux est un problème hiérarchique dans lequel chaque décideur possède un niveau de décision distinct. Dans le modèle mathématique d'un tel problème, la décision optimale du décideur de plus bas niveau sert de contrainte au niveau supérieur, ainsi toutes les décisions appartenant aux différents niveaux de décision influencent le résultat global de la résolution du problème.

L'optimisation de problème à plusieurs niveaux est utilisée dans plusieurs domaines

tels que l'économie, la recherche opérationnelle, la conception de système en ingénierie, la science de l'environnement, et le comportement dans une organisation [1].

Dans le cas d'un problème à plusieurs niveaux avec deux décideurs - c'est-à-dire deux niveaux hiérarchiques distincts dont l'un est supérieur à l'autre - on parle de problème de programmation bi-niveaux. Ce type de problème a été formulé pour la première fois en 1934 par Heinrich von Stackelberg dans un ouvrage sur l'économie de marché. Par ailleurs, un problème de programmation mathématique à deux niveaux, étudié pendant des années en théorie des jeux, porte le nom de *jeu de Stackelberg*. En effet, Stackelberg analyse un problème de marché économique où deux niveaux de décision peuvent être modélisés de telle sorte que la décision de chaque décideur dépend de celle de l'autre. Dans ce cadre, le leader est capable d'influencer les décisions du follower, mais doit également tenir compte de la réaction de ce dernier. Il s'agit donc d'un duopole où les deux acteurs, tout en étant en concurrence, coopèrent pour obtenir le profit maximal [1, 6].

La théorie du contrôle étudie les systèmes dynamiques régis par des commandes externes, dont le comportement peut être influencé par un paramètre de contrôle. Son objectif principal consiste à déterminer comment transférer un système d'un état initial spécifié vers un état final désiré, tout en satisfaisant des critères de performance spécifiques.

D'un point de vue mathématique, ces systèmes se modélisent à l'aide d'équations différentielles (ordinaires ou aux dérivées partielles), de formulations intégrales ou d'approches stochastiques, ce qui place naturellement cette théorie à l'intersection de plusieurs domaines mathématiques. Apparue après la Seconde Guerre mondiale pour répondre aux besoins croissants en navigation aérienne, la théorie du contrôle optimal s'inscrit dans la continuité du calcul des variations et s'appuie sur les principes variationnels de la mécanique classique (équations d'Euler-Lagrange). Elle a connu des développements majeurs avec le principe du maximum de Pontryagin (1956) - fournissant des conditions nécessaires d'optimalité pour déterminer des trajectoires optimales - ainsi qu'avec l'avènement de la programmation dynamique, l'introduction de l'analyse fonctionnelle et l'établissement de liens profonds avec la théorie de la stabilité de Lyapunov. Ces avancées en font aujourd'hui un outil fondamental pour l'optimisation des systèmes dynamiques commandés.

Notre étude se concentre sur les problèmes bi-niveaux et leur application au contrôle optimal. Ces problèmes mettent en jeu deux décideurs en interdépendance hiérarchique. Le premier est appelé *leader* ou décideur de niveau supérieur, le second est appelé *follower* ou décideur de niveau inférieur. Le contrôle des variables de décision est partagé entre les deux décideurs.

Dans le cas linéaire (fonctions objectives et contraintes linéaires), la solution se trouve parmi les points extrêmes du polyèdre des contraintes. Bien que structurellement simple, ce problème présente une complexité algorithmique difficile. Les principales méthodes de résolution exploitent des reformulations en problèmes à un seul niveau.

Ce présent travail est divisé en trois chapitres, organisé de la manière suivante :
Le premier chapitre introduit les notions de base qui seront utilisées dans tout ce travail. Nous présentons d'abord les outils essentiels d'analyse convexe, particulièrement utiles pour les problèmes d'optimisation mathématique. Ensuite, nous abordons deux concepts fondamentaux : l'optimisation avec contraintes et l'optimisation paramétrique, qui sont à la base de l'étude des problèmes bi-niveaux.

Le deuxième chapitre aborde les problèmes de programmation bi-niveaux selon deux perspectives distinctes. La première partie traite des approches optimiste et pessimiste, qui dépendent de la nature de la relation entre le Leader et le Suiveur - coopérative dans le premier cas, conflictuelle dans le second. La seconde partie se concentre sur le cas particulier des problèmes bi-niveaux linéaires, où les fonctions objectifs et les contraintes sont linéaires. Ces problèmes présentent une complexité algorithmique NP-difficile, comme l'a démontré Jeroslow dans ses travaux. Cette double analyse permet de couvrir à la fois les aspects relationnels et structurels des problèmes bi-niveaux.

Le troisième et dernier chapitre étudie les problèmes de contrôle optimal bi-niveaux. Nous commençons par rappeler les concepts clés du contrôle optimal (contrôlabilité, stabilité), puis nous présentons la formulation mathématique du problème bi-niveau [35, 36] en précisant nos hypothèses de travail. Notre approche consiste à transformer ce problème à deux niveaux en un problème à un niveau en utilisant la fonction valeur du niveau inférieur, ce qui permet d'obtenir des conditions nécessaires d'optimalité. Pour illustrer ces concepts, nous concluons par un exemple concret appliqué à une problématique combinant enjeux économiques et environnementaux.

Ce travail se termine par une conclusion.

Chapitre 1

Concepts fondamentaux de la programmation mathématique :

Dans cette partie, on rappelle les concepts essentiels de la programmation mathématique.

On considère U un ouvert dans Ω , où Ω est un ouvert de \mathbb{R}^n . La norme euclidienne sur \mathbb{R}^n est donnée par :

$$\|x\| = \left(\sum_{i=1}^n x_i^2 \right)^{\frac{1}{2}}, \quad \forall x = (x_1, \dots, x_n) \in \mathbb{R}^n.$$

1.1 Convexité :

1.1.1 Ensembles convexes :

Définition 1.1.

On dit qu'un ensemble $U \subset \mathbb{R}^n$ est *convexe* si :

$$\forall \lambda \in [0, 1], \forall x, y \in U \text{ on a } \lambda x + (1 - \lambda)y \in U.$$

Cela signifie que si deux points x et y sont dans U , alors le segment $[x, y]$, qui relie ces points, est contenu dans U .

Exemple : Dans la figure qui suit on a deux ensembles E_1 étant convexe et E_2 non-convexe.

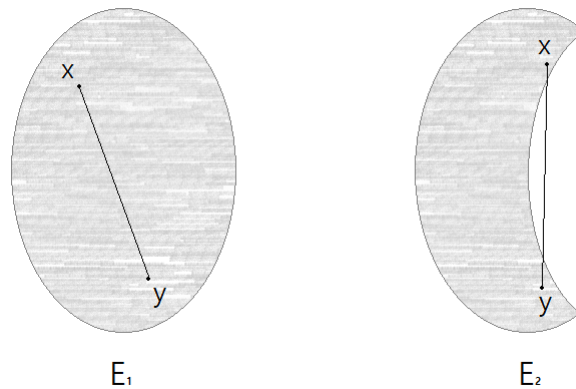


FIGURE 1.1 – Ensembles convexe et non-convexe.

Définition 1.2. (Combinaison convexe)

Un vecteur x de \mathbb{R}^n est appelé *combinaison convexe* des n vecteurs de \mathbb{R}^n , s'il existe des constantes positives $\lambda_1, \lambda_2, \dots, \lambda_n$ avec $\sum_{i=1}^n \lambda_i = 1$ telles que $x = \sum_{i=1}^n \lambda_i x_i$.

Définition 1.3. :(Enveloppe convexe)

On appelle *enveloppe convexe* d'un sous-ensemble quelconque C de \mathbb{R}^n qu'on note $\mathbf{conv}(C)$ l'ensemble de toutes les combinaisons convexes des points de C :

$$\mathbf{conv}(C) = \left\{ \sum_{i=1}^n \lambda_i x_i : x_i \in C, \lambda_i \geq 0, i = 1, \dots, n, \sum_{i=1}^n \lambda_i = 1 \right\}$$

L'enveloppe convexe $\mathbf{conv}(C)$ est toujours convexe, c'est le plus petit ensemble convexe qui contient C .

Exemple : On a à droite de la figure 1.2 qui suit les enveloppes convexes des ensemble qui sont à gauche.

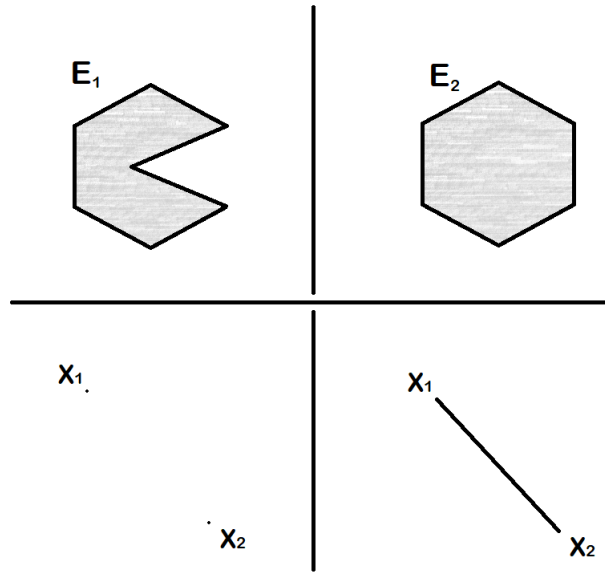


FIGURE 1.2 – Enveloppes convexes.

Définition 1.4. (Point extrême)

Soit U une partie convexe de \mathbb{R}^n . Un point $x \in U$ est un *point extrême* de U si et seulement si $\forall x_1, x_2 \in U$ et $\forall \lambda \in [0, 1] : x = \lambda x_1 + (1 - \lambda)x_2 \Rightarrow x = x_1 = x_2$.

Autrement dit, il n'existe pas de combinaisons convexe l'exprimant.

Définition 1.5. (Ensemble polyédral)

L'ensemble $U \subset \mathbb{R}^n$ est appelé *ensemble polyédral* s'il est une intersection d'un nombre fini de demi-espaces fermés de \mathbb{R}^n i.e :

$$\exists k \in \mathbb{N}, \exists \alpha_i, \exists p_i \in \mathbb{R}^n, p_i \neq 0, i = 1, \dots, k : U = \bigcap_{i=1}^k \{x | p_i^t x \leq \alpha_i\}.$$

Définition 1.6. (Polytope convexe)

Soit $x_1, \dots, x_k, x_{k+1} \in \mathbb{R}^n$, $\text{conv}\{x_1, \dots, x_{k+1}\}$ est appelé un *polytope*.

- Tout polytope est un ensemble polyédral.
- Un polyèdre est tout ensemble convexe de la forme :

$$X = \{x | Ax = b, x \geq 0\}.$$

- Un polyèdre borné est appelé polytope convexe.

Définition 1.7. (Cône)

Un ensemble C de \mathbb{R}^n est appelé *cône* si $\forall x \in C$ et $\forall \lambda \geq 0 \Rightarrow \lambda x \in C$.

Définition 1.8. (Cône convexe)

Un ensemble C est un *cône convexe* s'il est un ensemble convexe et un cône simultanément, c.à.d :

$$\forall x_1, x_2 \in C \text{ et } \forall \lambda_1, \lambda_2 \geq 0 \Rightarrow \lambda_1 x_1 + \lambda_2 x_2 \in C.$$

1.1.2 Fonctions convexes :

Définition 1.9. On suppose que U est un sous ensemble convexe. On dit que :

l'application $J : U \rightarrow \mathbb{R}$ est convexe sur U si :

$$\forall \lambda \in [0, 1], \forall x, y \in U \implies J(\lambda x + (1 - \lambda)y) \leq \lambda J(x) + (1 - \lambda)J(y). \quad (1.1)$$

l'application $J : U \rightarrow \mathbb{R}$ est strictement convexe sur U si :

$$\forall \lambda \in [0, 1], \forall x, y \in U, x \neq y \implies J(\lambda x + (1 - \lambda)y) < \lambda J(x) + (1 - \lambda)J(y).$$

l'application $J : U \rightarrow \mathbb{R}$ est fortement convexe sur U (pour $\alpha > 0$), si :

$$\forall \lambda \in [0, 1], \forall x, y \in U \implies J(\lambda x + (1 - \lambda)y) \leq \lambda J(x) + (1 - \lambda)J(y) - \alpha \lambda(1 - \lambda) \|x - y\|^2.$$

Définition 1.10. (différentiabilité)

On dit qu'une fonction $f : U \subseteq \mathbb{R}^n \rightarrow \mathbb{R}$ est différentiable au point a , s'il existe une application linéaire continue $L : \mathbb{R}^n \rightarrow \mathbb{R}$ telle que :

$$f(a + h) - f(a) = L(h) + \|h\| \varepsilon(h)$$

avec

$$\lim_{h \rightarrow 0} \varepsilon(h) = 0$$

L'application linéaire s'appelle la différentielle de f au point a . Elle sera noté $df(a)$ ou simplement df s'il n'y a pas d'ambiguïté. h est un élément de l'ensemble U et sert à définir la "direction" de dérivation (pour la différentiabilité, on les prend toutes en considérations).

Proposition 1. [4]

1. Caractérisation des fonctions convexes :

On suppose que f est une fonction différentiable en tout point de U (de classe $C^1(U)$).

a) Les assertions suivantes sont équivalentes :

- f est convexe sur U .
- $\forall x \in U, \forall y \in U$, on a $f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle$.
- $\nabla f(x)$ est monotone sur U c'est à dire :
 $\forall x \in U, \forall y \in U$ alors $\langle \nabla f(y) - \nabla f(x), y - x \rangle \geq 0$.

b) Si de plus f est deux fois différentiable en tout point de U (de classe $C^2(U)$)

$$\iff f \text{ est convexe sur } U \iff \forall x \in U, \forall y \in U \text{ on a } \langle \nabla^2 f(x)(y - x), (y - x) \rangle \geq 0.$$

2. Caractérisation des fonctions strictement convexes :

On suppose que f est une fonction différentiable en tout point de U (de classe $C^1(U)$).

a) On a équivalence entre :

- f est strictement convexe sur U .
- $\forall x \in U, \forall y \in U, x \neq y, f(y) > f(x) + \langle \nabla f(x), y - x \rangle$.
- $\nabla f(x)$ est strictement monotone sur U c'est à dire :
 $\forall x \in U, \forall y \in U, x \neq y, \langle \nabla f(y) - \nabla f(x), y - x \rangle > 0$.

b) Si de plus f est deux fois différentiable en tout point de U (de classe $C^2(U)$) \Rightarrow f est strictement convexe sur $U \Leftrightarrow \forall x \in U, \forall y \in U$ avec $x \neq y : \langle \nabla^2 f(x)(y - x), (y - x) \rangle > 0$.

Théorème 1.1. [4]

Si U est un sous-ensemble convexe de \mathbb{R}^n et $f : U \rightarrow]-\infty, +\infty[$ est une fonction convexe, alors un minimum local de f sur U est aussi un minimum global de f sur U .
 Si de plus f est strictement convexe, alors il existe au plus un minimum global de f sur U .

1.2 Programmation Mathématique :

1.2.1 Optimisation sous contraintes :

Un problème d'optimisation avec contraintes est formulé comme suit :

$$\begin{aligned} \min f(x) \\ \text{s.c} \begin{cases} g(x) \leq 0 \\ h(x) = 0 \\ x \in \mathbb{R}^n \end{cases} \end{aligned} \tag{1.2}$$

où $f : \mathbb{R}^n \rightarrow \mathbb{R}$ est la *fonction objectif*, $g : \mathbb{R}^n \rightarrow \mathbb{R}^{m-p}$ avec : $g(x) = (g_1(x), \dots, g_{m-p}(x))$ est appelée *fonction contrainte d'inégalités*, et $h : \mathbb{R}^n \rightarrow \mathbb{R}^p$ avec : $h(x) = (h_1(x), \dots, h_p(x))$ appelée *fonction contrainte d'égalités* du problème.

Définition 1.11.

Soit $x^* \in S$, où $S = \{x \in \mathbb{R}^n / g_i(x) \leq 0, h(x) = 0, \forall i = 1, \dots, m\}$ l'espace des contrainte du problème (1.1). Le problème admet :

- (i) Un minimum global sur S au point x^* , si : $\forall x \in S, f(x^*) \leq f(x)$.
- (ii) Un minimum local sur S au point x^* , si : $\exists V$ voisinage de x^* tq : $\forall x \in S \cap V, f(x^*) \leq f(x)$.
- (iii) Un maximum global sur S au point x^* , si : $\forall x \in S, f(x) \leq f(x^*)$.
- (iv) Un maximum local sur S au point x^* , si : $\exists V$ voisinage de x^* tq : $\forall x \in S \cap V, f(x) \leq f(x^*)$.

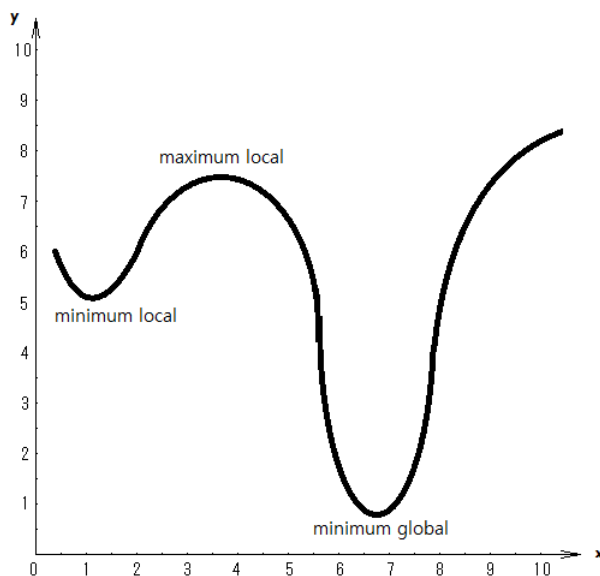


FIGURE 1.3 – Représentation graphique des extremums.

Définition 1.12. (Contrainte active)

Pour un x point réalisable du problème (1.2) et $g_i(x) = 0$ on dit que la i -ème contrainte $g_i(x) \leq 0$ est active en x . Si $g_i(x) < 0$, on dit que la i -ème contrainte $g_i(x) \leq 0$ est inactive.

Les contraintes d'égalités $h(x) = 0$ sont toutes actives.

Définition 1.13.

Soit $x \in S$, $I(x)$ l'ensemble des indices des contraintes actives est définie par $I(x) = \{i | g_i(x) = 0, h(x) = 0\}$.

Définition 1.14. (Régularité)

- Un point $x^* \in \tilde{S}$ avec $\tilde{S} = \{x \in \mathbb{R}^n | g_i(x) = 0, h(x) = 0, \forall i = 1, \dots, m\}$ est dit régulier si la famille de vecteurs $\{\nabla g_i(x^*)\}_{i=1, \dots, m}$ est une famille libre de \mathbb{R}^n .
- Un point $x^* \in S$ avec $S = \{x \in \mathbb{R}^n | g_i(x) \leq 0, h(x) = 0, \forall i = 1, \dots, m\}$ est dit régulier si :
 - 1 Soit $I(x^*) = \emptyset$,
 - 2 Soit x^* est régulier pour \bar{S} , avec $\bar{S} = \{x \in \mathbb{R}^n | g_i(x) = 0, h(x) = 0, \forall i \in I(x^*)\}$.

Définition 1.15. (Qualification)

Soit S ensembles de contraintes, On dira que les contraintes de S sont qualifiées en un point x^* si :

- 1 Soit $I(x^*) = \emptyset$,
- 2 Soit $\exists w \in \mathbb{R}^n$ t.q : $\langle \nabla g_i(x^*), w \rangle \leq 0, \forall i \in I(x^*)$ si g_i est affine. et $\langle \nabla g_i(x^*), w \rangle < 0, \forall i \in I(x^*)$ si g_i est non affine.

Proposition 2. Si $x^* \in S$ est point régulier de $S \Rightarrow$ les contraintes de S sont qualifiées.

Remarque 1.1. L'ensemble des solutions optimales est noté $\arg \min_{x \in S} f(x)$. On écrit alors

$$x^* = \arg \min_{x \in S} f(x).$$

Définition 1.16. (Solution de base dégénérée)

Soit le problème suivant :

$$\begin{aligned} & \min f(x) \\ & \text{s.c} \begin{cases} Ax = b \\ x \geq 0 \end{cases} \end{aligned} \quad (1.3)$$

Où $x \in \mathbb{R}^n$, $A \in \mathcal{M}_{n \times m}(\mathbb{R})$ est de rang plein, $b \in \mathbb{R}^m$ et $n \geq m$.

1. Une solution de base est tout vecteur $x \in \mathbb{R}^n$ satisfaisant les contraintes $Ax = b$ et $x \geq 0$, composé de m contraintes qui constituent une base.
2. Une solution de base $x \in \mathbb{R}^n$ est dite dégénérée si plus de $n - m$ contraintes sont actives en x , c'est-à-dire si plus de $n - m$ composantes de x sont nulles.
3. Une solution de base $x \in \mathbb{R}^n$ est dite non dégénérée si exactement $n - m$ contraintes sont actives en x .

Théorème 1.2. [7, 4] (Théorème de Weierstrass)

Soit S un ensemble non-vide compact de \mathbb{R}^n , et $f : S \rightarrow \mathbb{R}$ une fonction réelle continue dans S , alors le problème d'optimisation :

$$\begin{cases} \min f(x) \\ x \in S \end{cases}$$

admet au moins une solution optimale $x^* \in S$.

Démonstration :

Soit $m = \inf_{x \in S} f(x)$ (c'est à dire $\forall x \in S : m \leq f(x)$). On note que m peut valoir $-\infty$ si l'ensemble $f(S)$ n'est pas minoré. On peut alors construire une suite infinie $\{x^k\}$ d'éléments de S telle que $f(x^k) \rightarrow m$.

Comme S est compact (fermé et borné), il existe une sous-suite infinie $\{x^l\}_{l \in L} (L \subset \mathbb{N})$ convergeant vers $x^* \in S$.

Comme f est continue on a : $f(x^l) \rightarrow f(x^*)$ et :

$$m = \lim_{k \rightarrow \infty} f(x^k) = \lim_{l \rightarrow \infty} f(x^l) = f(x^*), l \in L.$$

Comme $f(x^*) > -\infty$, on a $m > -\infty$ et $\forall x \in S : f(x^*) = m \leq f(x)$ donc $x^* \in S$ est une solution optimal du problème posé. Si f n'est pas continue mais seulement semi continue inférieurement sur S , alors il suffit dans la relation ci dessus, de remplacer l'égalité par l'inégalité :

$$\lim_{l \rightarrow \infty} f(x^l) \geq f(x^*), l \in L.$$

et la même conclusion sera obtenue.

Lagrangien et conditions de Karush-Khun-Tuker (KKT) :

Définition 1.17. (Lagrangien)

Pour le problème d'optimisation sous contraintes (1.2) avec $g(x) = (g_1(x), \dots, g_{m-p}(x))$ les contraintes d'inégalités et $h(x) = (h_1(x), \dots, h_p(x))$ les contraintes d'égalités du problème.

Le lagrangien associé a ce problème est définie par $\mathcal{L} : \mathbb{R}^n \times \mathbb{R}^{m-p} \times \mathbb{R}^p \rightarrow \mathbb{R}$ et on le note :

$$\mathcal{L}(x, \lambda, \mu) = f(x) + \sum_{i=1}^{m-p} \lambda_i g_i(x) + \sum_{j=1}^p \mu_j h_j(x)$$

Où $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_{m-p}) \in \mathbb{R}^{m-p}$, λ_i les multiplicateurs de Lagrange associés aux contraintes $g_i(x)$ avec $i = 1, \dots, m-p$, et $\mu = (\mu_1, \mu_2, \dots, \mu_p) \in \mathbb{R}^p$, μ_j les multiplicateurs de Lagrange associés aux contraintes $h_j(x)$ avec $j = 1, \dots, p$.

Définition 1.18. (Problème dual)

Dans le cas du problème d'optimisation linéaire :

$$\begin{aligned} & \min_x C^t x \\ & \text{s.c} \begin{cases} Ax \leq b \\ x \geq 0 \end{cases} \end{aligned} \quad (1.4)$$

Où $x \in \mathbb{R}^n$, $C \in \mathbb{R}^n$, $A \in \mathcal{M}_{n \times m}(\mathbb{R})$, $b \in \mathbb{R}^m$.

On calculant :

$$\theta(\lambda, \mu) = \inf_{x \geq 0} \mathcal{L}(x, \lambda, \mu) = \inf_{x \geq 0} [C^t x + \lambda^t (b - Ax)]$$

On obtient le problème dual de (1.4) :

$$\begin{aligned} & \max_{\lambda} \lambda^t b \\ & \text{s.c} \begin{cases} C^t - \lambda^t A \geq 0 \\ \lambda \in (\mathbb{R}_+)^m \end{cases} \end{aligned} \quad (1.5)$$

Théorème 1.3. (Conditions nécessaires de Karush-Khun-Tuker (KKT))[5]

Soit $x^* \in S$ tel que les contraintes de S sont qualifiées en x^* et x^* est un extrémum de f sur S alors $\exists \lambda^* = (\lambda_1^*, \lambda_2^*, \dots, \lambda_m^*) \in (\mathbb{R}_+)^{m-p}$ et $\exists \mu^* = (\mu_1^*, \mu_2^*, \dots, \mu_p^*) \in \mathbb{R}^p$ tel que :

$$S_{KKT} \begin{cases} \nabla_x \mathcal{L}(x^*, \lambda^*, \mu^*) = \nabla_x f(x^*) + \sum_{i=1}^{m-p} \lambda_i^* \nabla_x g_i(x^*) + \sum_{j=1}^p \mu_j \nabla_x h_j(x) = 0_{\mathbb{R}^n} \\ \lambda_i^* g_i(x^*) = 0, \forall i = 1, \dots, m-p \\ h_j(x) = 0, \forall j = 1, \dots, p \\ \lambda^* \in (\mathbb{R}_+)^{m-p} \\ \mu^* \in \mathbb{R}^p \end{cases}$$

Remarque 1.2. Les conditions nécessaires de KKT deviennent des conditions suffisantes dans le cas où le problème est convexe, c'est à dire sa fonction objectif et ses contraintes sont convexes.

Théorème 1.4. (Conditions suffisantes d'optimalité)[8]

Si les conditions suivantes sont satisfaites pour $x^* \in S$:

1) Soit $x^* \in S$ régulier tel qu'il satisfait les conditions de KKT :

$$S_{KKT} \begin{cases} \nabla_x \mathcal{L}(x^*, \lambda^*, \mu^*) = \nabla_x f(x^*) + \sum_{i=1}^{m-p} \lambda_i^* \nabla_x g_i(x^*) + \sum_{j=1}^p \mu_j \nabla_x h_j(x) = 0_{\mathbb{R}^n} \\ \lambda_i^* g_i(x^*) = 0, \forall i = 1, \dots, m-p \\ h_j(x) = 0, \forall j = 1, \dots, p \\ \lambda^* \in (\mathbb{R}_+)^{m-p} \\ \mu^* \in \mathbb{R}^p \end{cases}$$

2) Soit $x^* \in S$ et $\exists \lambda^* \in (\mathbb{R}_+)^m$ tel que :

$$\langle \nabla_x^2 \mathcal{L}(x, \lambda, \mu) d, d \rangle > 0, \forall d \in T_{(S, x^*)} \setminus \{0\}$$

où $T_{(S, x^*)}$ cône tangent et noté :

$$T_{(S, x^*)} = \{d \in \mathbb{R}^n \mid \langle \nabla_x g_i(x^*), d \rangle \leq 0, \forall i \text{ t.q } g_i(x^*) = 0 \text{ et } \langle \nabla_x h_j(x^*), d \rangle = 0, \forall j = 1, \dots, p\}$$

Alors $x^* \in S$ est un minimum local de f sur S .

Équations paramétriques et Théorème des Fonctions Implicite :

On commence par l'exemple suivant :

En résolvant pour $x \in \mathbb{R}$, l'équation non paramétrique : $f(x) := x^2 - 2x - 1 = 0$ qui est équivalente à $(x - 1)^2 - 2 = 0$ on obtient les solutions $x_1 = 1 + \sqrt{2}$, et $x_2 = 1 - \sqrt{2}$.

La version paramétrique de l'équation est la suivante : Pour le paramètre $\delta \in \mathbb{R}$ on cherche une solution $x = x(\delta)$ de $f(x, \delta) := x^2 - 2\delta^2x - \delta^4 = 0$ qui est équivalente à $(x - \delta^2)^2 - 2\delta^4 = 0$ on obtient les solutions $x_1(\delta) = \delta^2 + \delta^2\sqrt{2} = \delta^2(1 + \sqrt{2})$, et $x_2(\delta) = \delta^2 - \delta^2\sqrt{2} = \delta^2(1 - \sqrt{2})$.

Evidemment, les courbes de solutions se rencontrent à $(x, \delta) = (0, 0)$ (figure 1.4). À ce stade, on trouve la dérivée partielle de f :

$$\nabla_x f(x, \delta) = 2x - 2\delta^2 = 0.$$

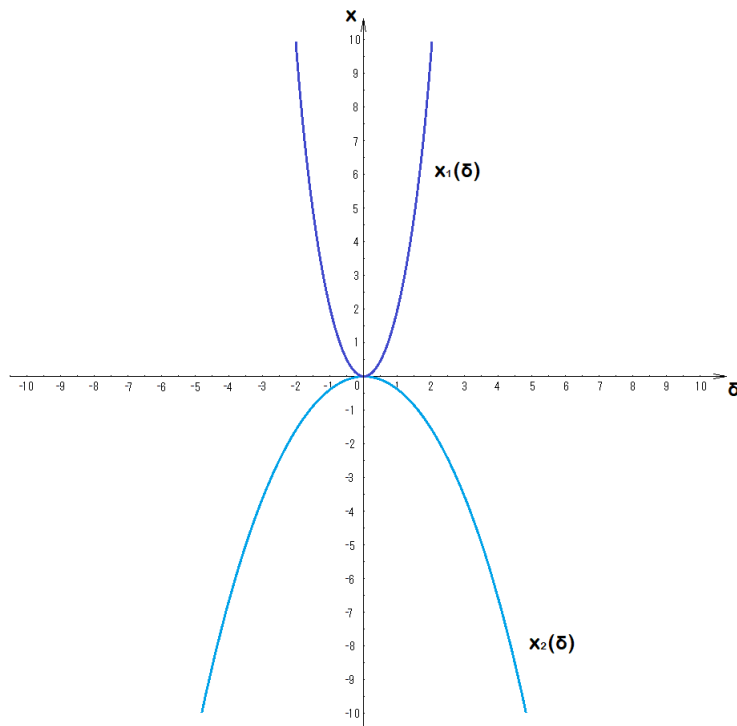


FIGURE 1.4 – Les courbes des solutions de l'équation $(x - \delta^2)^2 - 2\delta^4 = 0$

Remarque 1.3. Pour une fonction $f : \mathbb{R}^2 \rightarrow \mathbb{R}$ de classe C^1 , l'ensemble de solutions d'une équation paramétrique $f(x, \delta) = 0$ est donné localement par une courbe de solutions $(x(\delta), \delta)$ unidimensionnelle de classe C^1 . Toutefois, aux points (x^*, δ^*) où $\nabla_x f(x^*, \delta^*) = 0$, l'ensemble de solutions peut indiquer une singularité (telle qu'une intersection ou non-régularité)

Théorème 1.5. (Théorème des fonctions implicites, version générale)[9] .

Soit U un ouvert de $\mathbb{R}^n \times \mathbb{R}^p$ et $f : U \rightarrow \mathbb{R}^n$ une application de classe C^k , avec $k \geq 1$. Soit $(a, b) \in \mathbb{R}^n \times \mathbb{R}^p$ tel que $f(a, b) = 0$ et la différentielle partielle $\nabla_x f(a, b)$ est inversible. Alors il existe un voisinage V de b dans \mathbb{R}^p et une application $\phi : V \rightarrow \mathbb{R}^n$ de classe C^k tels que

$$\forall \delta \in V, f(x, \delta) = 0 \Leftrightarrow x = \phi(\delta).$$

En outre on peut choisir V de sorte que la différentielle $\nabla_x f(x, \delta)$ est inversible pour tout $(x, \delta) \in \mathbb{R}^n \times V$ et

$$\nabla \phi(\delta) = -[\nabla_x f(\phi(\delta), \delta)]^{-1} \nabla_\delta f(\phi(\delta), \delta)$$

Ici $\nabla_x f(a, b)$ est la différentielle de l'application $x \in \mathbb{R}^n \rightarrow f(x, b) \in \mathbb{R}^n$ au point a . Au départ f est une fonction de $n + p$ variables à valeurs dans \mathbb{R}^n . Si on fixe p variables, on obtient une fonction de n variables à valeurs dans \mathbb{R}^n . La différentielle partielle $\nabla_x f(a, b)$ est alors la différentielle de cette fonction au point a , les p premières variables étant fixées à $b = (b_1, \dots, b_p)$. Sa matrice dans la base canonique de \mathbb{R}^n est

$$Jac_x f(a, b) = \begin{pmatrix} \frac{\partial f_1}{\partial x_{p+1}}(a, b) & \dots & \frac{\partial f_1}{\partial x_{p+n}}(a, b) \\ \vdots & & \vdots \\ \frac{\partial f_n}{\partial x_{p+1}}(a, b) & \dots & \frac{\partial f_n}{\partial x_{p+n}}(a, b) \end{pmatrix} \in M_n(\mathbb{R})$$

On discute comment une courbe de solution $(x(\delta), \delta)$ d'une équation paramétrique $f(x, \delta) = 0$ $f : \mathbb{R}_n \times \mathbb{R} \rightarrow \mathbb{R}_n$, $\delta \in \mathbb{R}$, peut être suivie numériquement.

L'idée de base est d'utiliser une sorte de procédure de Newton. Les itérations d'une procédure de Newton pour, calculer une solution, commencent par sélectionner un point de départ (approprié) x^0 et d'ensuite itérer selon :

$$x^{k+1} = x^k - [\nabla f(x^k)]^{-1} f(x^k), \quad k = 0, 1, \dots$$

Il est bien connue que cette itération converge de façon quadratique vers une solution x^* de $f(x^*) = 0$ si

- x^0 est choisi assez près de x^* .
- $\nabla f(x^*)$ est une matrice inversible.

Le moyen le plus simple de suivre approximativement une courbe de solution $x(\delta)$ de $f(x, \delta) = 0$, sur un intervalle $t \in [a, b]$ est de discrétiser $[a, b]$ par

$$\delta_l = a + l \frac{b-a}{N}, \quad l = 0, \dots, N,$$

(pour certains $N \in \mathbb{N}$) et pour calculer pour tout $l = 0, \dots, N$, une solution $x_l = x(\delta_l)$ de $f(x, \delta_l) = 0$ par une itération de Newton,

$$x_l^{k+1} = x_l^k - [\nabla_x f(x_l^k, \delta_l)]^{-1} f(x_l^k, \delta_l), \quad k = 0, 1, \dots$$

en commençant par $x_l^0 = x_{l-1} + \frac{b-a}{N} x'(\delta_{l-1})$, (pour $l \geq 1$). La dérivé $x'(\delta_{l-1})$ peut être calculé avec la formule dans le théorème des fonctions implicites[8].

$$x'(\delta_{l-1}) = -[\nabla_x f(x(\delta_{l-1}), \delta_{l-1})]^{-1} \nabla_{\delta_{l-1}} f(x(\delta_{l-1}), \delta_{l-1})$$

Optimisation paramétrique :

L'optimisation paramétrique étudie les problèmes d'optimisation dépendant d'un paramètre $\delta \in \mathbb{R}^q$, à considérer lors de la recherche d'un optimum local $x^* \in S$. Dans un tel problème, la valeur de x dépend de celle du paramètre δ , on note : $x = x(\delta)$. La fonction objectif, notée $v(\delta) = f(x, \delta)$, dépend explicitement de ce paramètre. Formellement, un tel problème s'écrit :

$$\begin{aligned} \min_x & f(x, \delta) \\ \text{s.c} & \begin{cases} g(x, \delta) \leq 0 \\ x \in \mathbb{R}^n \end{cases} \end{aligned} \quad (1.6)$$

Où $\delta \in \mathbb{R}^q$ un paramètre, $f : \mathbb{R}^n \rightarrow \mathbb{R}$ la fonction objectif et $g : \mathbb{R}^n \rightarrow \mathbb{R}^m$ avec : $g(x, \delta) = (g_1(x, \delta), \dots, g_m(x, \delta))$ la fonction des contraintes.

Remarque 1.4. Si on fixe le paramètre $\delta \in \mathbb{R}^q$ en une valeur $\bar{\delta}$, le problème d'optimisation sous contraintes paramétrique devient un problème d'optimisation sous contraintes simple.

Exemple : Ci-dessous un problème d'optimisation paramétrique :

$$\begin{aligned} \min & f(x, \delta) = (1 - 3\delta)x_1 + (5 + \delta)x_2 \\ \text{s.c} & \begin{cases} -9x_1 + 3x_2 \leq 0 \\ 3x_1 - 9x_2 \leq 0 \\ x_1 + x_2 \leq 4 \\ x \in \mathbb{R}^2 \end{cases} \end{aligned} \quad (1.7)$$

Où $\delta \in \mathbb{R}$, $f : \mathbb{R}^2 \rightarrow \mathbb{R}$ et $g : \mathbb{R}^2 \rightarrow \mathbb{R}^3$ avec : $g(x, \delta) = (-9x_1 + 3x_2, 3x_1 - 9x_2, x_1 + x_2 - 4)$.

Pour un paramètre $\bar{\delta} = 0$ fixé, le problème (1.7) devient un problème d'optimisation classique comme suit :

$$\begin{aligned} \min & f(x) = x_1 + 5x_2 \\ \text{s.c} & \begin{cases} -9x_1 + 3x_2 \leq 0 \\ 3x_1 - 9x_2 \leq 0 \\ x_1 + x_2 \leq 4 \\ x \in \mathbb{R}^2 \end{cases} \end{aligned} \quad (1.8)$$

Le problème (1.8) a pour solution $x^* = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$, avec $f^* = f \begin{pmatrix} 0 \\ 0 \end{pmatrix} = 0$.

Pour tout $\delta \in \mathbb{R}$ la valeur du minimum $x^*(\delta)$ et la fonction qui lui correspondant $v^*(\delta)$ du problème (1.7) sont données comme suit :

$$x(\delta) = \begin{cases} (0, 0)^t & \text{si } \delta < 1 \\ \lambda(0, 0)^t + (1 - \lambda)(3, 1)^t & \text{si } \delta = 1 \\ (3, 1)^t & \text{si } \delta > 1 \end{cases} \quad v(\delta) = \begin{cases} 0 & \text{si } \delta \leq 1 \\ 8 - 8\delta & \text{si } \delta > 1 \end{cases}$$

Dans les figures 1.5 et 1.6 qui suivent on voit bien qu'en fonction du paramètre δ la valeur optimale de la fonction objectif de notre problème $v(\delta) = f(x, \delta)$ se comporte comme une fonction linéaire par morceaux.

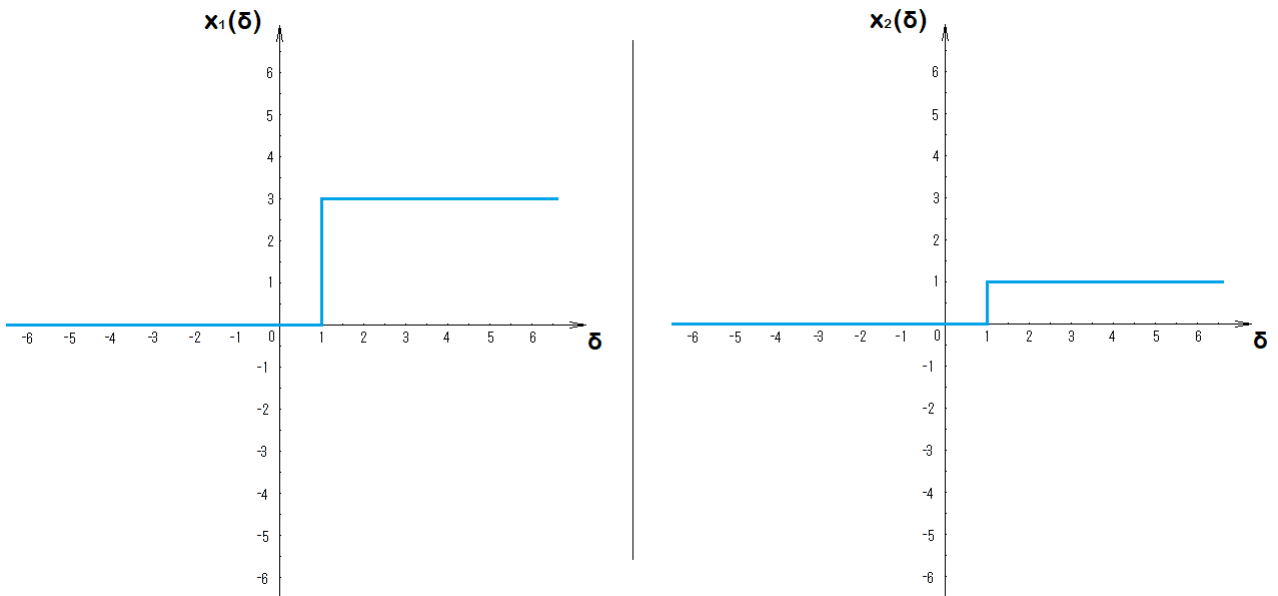


FIGURE 1.5 – Représentation de $x_1(\delta)$ et $x_2(\delta)$ (respectivement à gauche et à droite)

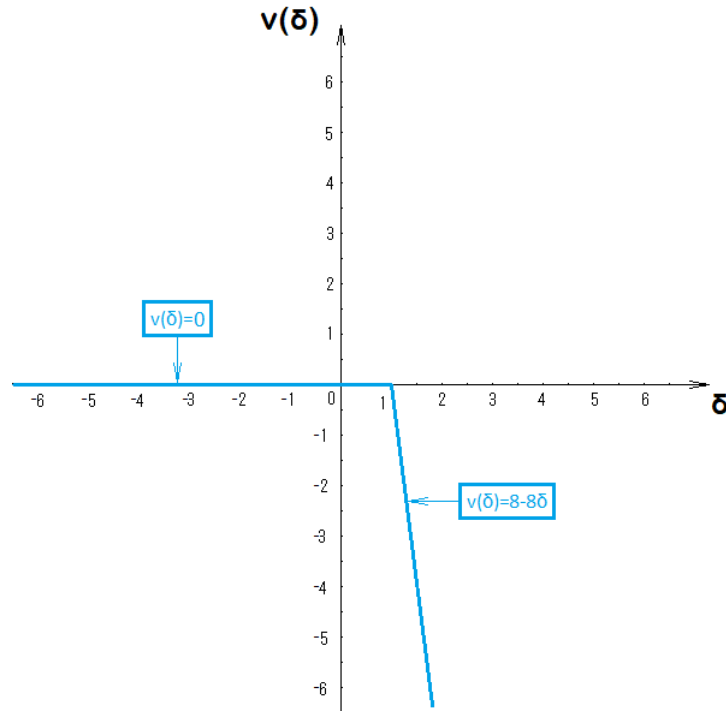


FIGURE 1.6 – Représentation de $v(\delta) = f(x, \delta)$

Programmation linéaire paramétrique :

Dans un programme linéaire paramétrique, étant donné une fonction matricielle de classe C^2 $A(\delta) : \mathbb{R}^p \rightarrow \mathbb{R}^{m \times n}$ avec m lignes $a_j^t(\delta)$, $j \in J := \{1, \dots, m\}$, des fonctions vectoriel de classe C^2 : $b(\delta) : \mathbb{R}^p \rightarrow \mathbb{R}^m$, $C(\delta) : \mathbb{R}^p \rightarrow \mathbb{R}^n$ et un espace paramétrique ouvert $\Delta \subset \mathbb{R}^p$:

Pour tout $\delta \in \Delta$, on veut résoudre le programme primal P

$$P : \begin{aligned} & \min C(\delta)^t x \\ & \text{s.c} \begin{cases} A(\delta)x \leq b(\delta) \\ x \in \mathbb{R}^n \end{cases} \end{aligned} \quad (1.9)$$

Avec $S_p(\delta) = \{x | A(\delta)x \leq b(\delta), x \in \mathbb{R}^n\}$.

On obtiens le problème dual correspondant :

$$D : \begin{aligned} & \max b(\delta)^t y \\ & \text{s.c} \begin{cases} A(\delta)^t y = C(\delta) \\ y \geq 0 \\ y \in \mathbb{R}^m \end{cases} \end{aligned} \quad (1.10)$$

Avec $S_d(\delta) = \{y | A(\delta)^t y = C(\delta), y \geq 0, y \in \mathbb{R}^m\}$.

Pour $\delta^* \in \Delta$ et $x^* \in S_p^* = S_p(\delta^*)$ l'ensemble des indice des contraintes actives est $J_0(x^*, \delta^*) = \{j \in J | a_j^t(\delta^*)x^* = b_j(\delta^*)\}$.

Supposons pour $\delta^* \in \Delta$ le point $x^* \in S_p^*$ est un sommet optimal de P . Pour trouver pour δ près de δ^* des solutions $x(\delta)$ de P , on doit trouver des solutions réalisables x et y du système de conditions d'optimalité,

pour

$$P : A_{J_0(x^*, \delta^*)}(\delta)x = b_{J_0(x^*, \delta^*)}(\delta) \quad (1.11)$$

pour

$$D : A_{J_0(x^*, \delta^*)}(\delta)^t y_{J_0(x^*, \delta^*)} = C(\delta), \quad y_{J_0(x^*, \delta^*)} \geq 0 \quad (1.12)$$

Si x^* est un sommet non dégénéré de P , pour $\delta^* \in \Delta$, c'est à dire si $A_{J_0(x^*, \delta^*)}(\delta^*)$ est inversible, et le minimum y^* de D satisfait la condition de complémentarité stricte : $y^* \geq 0, J_0(x^*, \delta^*)$, cela est possible en appliquant le théorème des fonctions implicites à ces systèmes.

Théorème 1.6 (Résultat de la stabilité locale). [8]

Soit $x^* \in S_p^*$ un sommet optimal de P , pour $\delta^* \in \Delta$, avec une solution dual correspondante $y_{J_0(x^*, \delta^*)}^*$ telle que :

- x est un sommet non dégénéré,
- $y_j^* > 0, \forall j \in J_0(x^*, \delta^*)$.

(Selon le théorème 1.3, x^* est un maximum de P , pour $\delta^* \in \Delta$, de l'ordre $s = 1$.) Il existe alors un voisinage V et des fonctions $x : V \rightarrow \mathbb{R}^n$ et $y_{J_0(x^*, \delta^*)} : V \rightarrow \mathbb{R}^{|J_0(x^*, \delta^*)|}$ de classe C^1 , telle que $x(\delta^*) = x^*, y_{J_0(x^*, \delta^*)}(\delta^*) = y_{J_0(x^*, \delta^*)}^*$ et pour tout $\delta \in V$ le point $x(\delta)$ est un sommet optimal de P (d'ordre $s = 1$) avec le multiplicateur correspondant $y_{J_0(x^*, \delta^*)}(\delta)$. De plus, pour $\delta \in V$ les dérivés de $x(\delta)$ et la fonction valeur $v(\delta) = C^t(\delta)x(\delta)$ sont donnés par

$$\nabla x(\delta) = [A_{J_0(x^*, \delta^*)}(\delta)]^{-1}(\nabla b_{J_0(x^*, \delta^*)}(\delta) - \nabla A_{J_0(x^*, \delta^*)}(\delta)x(\delta))$$

et

$$\nabla v(\delta) = \nabla C^t(\delta)x(\delta) + [y_{J_0(x^*, \delta^*)}(\delta)]^t [\nabla b_{J_0(x^*, \delta^*)}(\delta) - \nabla A_{J_0(x^*, \delta^*)}(\delta)x(\delta)]$$

Preuve.

Étant donné que le sommet x^* est non dégénéré, la matrice $A_{J_0(x^*, \delta^*)}(\delta^*)$ est inversible. Donc, le théorème des fonctions implicites peut être appliqué aux systèmes (1.11), (1.12) dans un voisinage V des sommet optimal $x(\delta)$ de P et $y_{J_0(x^*, \delta^*)}(\delta) > 0$ (puisque $y_{J_0(x^*, \delta^*)}^* = y_{J_0(x^*, \delta^*)}(\delta^*) > 0$) de D . En dérivant la relation

$$A_{J_0(x^*, \delta^*)}(\delta)x(\delta) = b_{J_0(x^*, \delta^*)}(\delta)$$

sur δ conduit directement aux formules pour $\nabla x(\delta)$ et $\nabla v(\delta)$.

Chapitre 2

Programmation mathématique bi-niveaux :

2.1 Programmation mathématique bi-niveaux :

Un problème de programmation bi-niveaux est un problème d'optimisation dont lequel un second problème d'optimisation est inclus dans ses contraintes. C'est un problème mathématique hiérarchique représentant deux niveaux de décision, le premier est appelé *leader*, qui représente l'objectif du décideur avec le plus haut niveau de décision, c'est le problème principal suite auquel les contraintes sont régies. Le second est le *suiveur* qui représente l'objectif du décideur qui dispose d'un niveau de décision inférieur au précédent, c'est celui qui est inclus dans les contraintes du premier problème.

Ces fonctions objectifs représentent l'impact des décisions de chacun des décideurs sur la solution du problème en main, que ces derniers agissent de façon coopérative dans laquelle le suiveur prend une décision avantageuse pour le leader (leurs objectifs ne se contredisent pas) ou de façon non coopérative, dans ce cas le leader doit se méfier des décisions du suiveur et agir en conséquence (leurs objectifs se contredisent).

2.1.1 Formulation du problème bi-niveaux :

Un problème de programmation bi-niveaux se formule de la même façon qu'un problème d'optimisation sous contraintes simple. En premier lieu, on aura une fonction objectif $f^1(x, y)$ à minimiser qui est la représentation de l'objectif du leader, c'est le premier niveau du problème. Ensuite, on aura des contraintes liées à cet objectif $g^1(x, y) = (g_1^1(x, y), \dots, g_i^1(x, y))$, i représente le nombre de ces contraintes, parmi elles on trouve une contrainte qui est à son tour un problème d'optimisation sous contraintes appelé suiveur qui est quant à elle le deuxième niveau du problème, il a comme fonction objectif $f^2(x, y)$ avec ses propres contraintes $g^2(x, y) = (g_1^2(x, y), \dots, g_j^2(x, y))$, et j représente le nombre de ces contraintes, ce qui caractérise des problèmes d'optimisations sous contraintes classiques. Les variables d'un problème de programmation bi-niveaux sont données par le couple (x, y) tel que x le vecteur des variables du premier niveau et y le vecteur des variables du deuxième

niveau.

Le problème est donc formulé comme suit :

$$\begin{array}{l} \min_{x,y} f^1(x,y) \\ \left\{ \begin{array}{l} g^1(x,y) \leq 0 \\ x \in \mathbb{R}^n \end{array} \right. \\ \text{s.c} \left\{ \begin{array}{l} \min_y f^2(x,y) \\ \text{s.c} \left\{ \begin{array}{l} g^2(x,y) \leq 0 \\ y \in \mathbb{R}^m \end{array} \right. \end{array} \right. \end{array} \quad (2.1)$$

Où $f^l : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}$ fonctions objectif du problème de l^{eme} niveau, et $g^l : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^p$ contraintes du problème de l^{eme} niveau, avec $l = 1, 2$.

Exemple : [6] On a le problème suivant :

$$\begin{array}{l} \min_x f^1(x,y) = x^2 + y^2 \\ \left\{ \begin{array}{l} -1 \leq y \leq 1 \\ \min_y f^2(x,y) = -xy \\ \text{s.c} \left\{ \begin{array}{l} 0 \leq y \leq 1 \end{array} \right. \end{array} \right. \end{array} \quad (2.2)$$

où $f^i : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$, $i = 1, 2$. Pour tout $x \in \mathbb{R}$,

$$M(x) = \arg \min_y \{-xy : 0 \leq y \leq 1\}.$$

Le problème précédent peut s'écrire alors

$$\left\{ \begin{array}{l} \min_x f^1(x,y) = x^2 + y^2, \\ y \in M(x), \\ -1 \leq y \leq 1, \end{array} \right.$$

On obtient les résultats suivants en évaluant le problème du second niveau, et en portant sa solution optimale dans la fonction objectif du niveau supérieur :

$$M(x) = \begin{cases} \{0\} & \text{si } x < 0 \\ \{1\} & \text{si } x > 0 \\ [0,1] & \text{si } x = 0 \end{cases}$$

$$f^1(x,y) = \begin{cases} x^2 & \text{si } x < 0 \\ 1 + x^2 & \text{si } x > 0 \\ [0,1] & \text{si } x = 0 \end{cases}$$

Voici une représentation graphique des résultats obtenus :

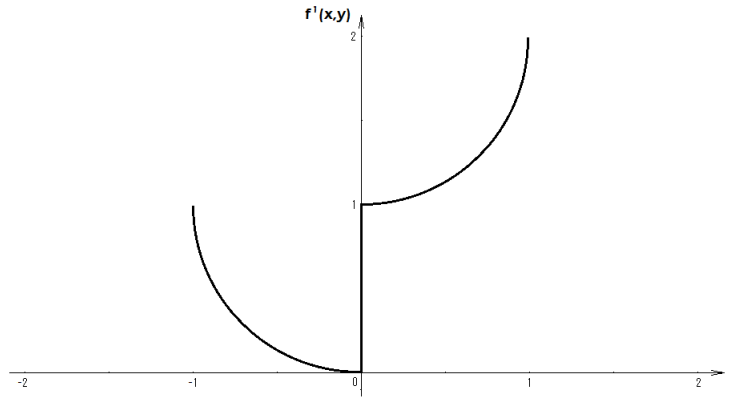


FIGURE 2.1 – Représentation graphique des résultats de l'exemple.

2.1.2 Étude du problème bi-niveaux :

Un problème de programmation bi-niveaux est considéré être difficile à résoudre, ceci étant le cas même pour un problème bi-niveaux linéaire. Il a été déjà prouvé que ce dernier est NP-difficile par plusieurs auteurs (Jeroslow, Bard, Ben-Ayed et Blair).[13, 2, 15]

Pour étudier un problème de programmation bi-niveaux, sa reformulation en problème d'optimisation classique peut être une bonne avenue à suivre. Pour ce faire, on peut transformer la fonction objectif du second niveau en une contrainte du premier niveau tel que [10] :

$$\begin{aligned} & \min_{x,y} f^1(x,y) \\ & \text{s.c} \begin{cases} g^1(x,y) \leq 0 \\ g^2(x,y) \leq 0 \\ f^2(x,y) \leq \Psi(x) \\ x \in \mathbb{R}^n \\ y \in \mathbb{R}^m \end{cases} \end{aligned} \quad (2.3)$$

Où $\Psi(x) = \min_{y \in Y} \{f^2(x,y)/g^2(x,y) \leq 0\}$.

On peut aussi, dans le cas où toutes les fonctions du problème de second niveau sont convexes, utiliser les conditions nécessaires de KKT du suiveur comme contraintes, pour transformer le problème bi-niveaux en un problème à un niveau [10] :

$$\begin{array}{l}
\min_{x,y} f^1(x,y) \\
s.c \left\{ \begin{array}{l}
g^1(x,y) \leq 0 \\
\nabla_y f^2(x,y) + \lambda^t \nabla_y g^2(x,y) = 0 \\
g^2(x,y) \leq 0 \\
\lambda^t g^2(x,y) = 0 \\
x \in \mathbb{R}^n \\
y \in \mathbb{R}^m \\
\lambda \in \mathbb{R}_+^m
\end{array} \right.
\end{array} \quad (2.4)$$

2.1.3 Multiplicité des solutions optimales de second niveau :

La fonction multivoque $M(x)$ représente l'ensemble des solutions optimales du problème du second niveau. Le problème linéaire bi-niveaux peut être formulé comme suit[11] :

$$\begin{array}{l}
\min_{x,y} f^1(x,y) \\
s.c \left\{ \begin{array}{l}
g^1(x,y) \leq 0 \\
x \in \mathbb{R}^n \\
y \in M(x)
\end{array} \right.
\end{array} \quad (2.5)$$

Dans le cas où l'ensemble $M(x)$ est constitué de plus d'un élément (multiplicité de solutions optimales du suiveur) pour un $x \in \mathbb{R}^n$ fixé, il existe dans la littérature deux approches principales pour formuler la solution du problème de programmation bi-niveaux : l'approche pessimiste (forte) et l'approche optimiste (faible)[12, 13].

Dans un tel problème, le leader ne peut pas forcer le suiveur à prendre une décision spécifique, puisque la solution du problème de second niveau qui sera choisie par le décideur(suiveur) est inconnue. Donc, lors de la résolution on doit suivre l'une des approches ci-dessous.[3]

Approche Optimiste

Dans cette approche le premier niveau peut supposer la coopération du second niveau, c'est à dire ce dernier va choisir à chaque fois une solution $y^* \in M(x)$ qui est la meilleure du point de vue du leader[12, 1]. La formulation dans ce cas ne change pas :

$$\begin{array}{l}
\min_{x,y} f^1(x,y) \\
s.c \left\{ \begin{array}{l}
g^1(x,y) \leq 0 \\
x \in \mathbb{R}^n \\
y \in M(x)
\end{array} \right.
\end{array} \quad (2.6)$$

On désigne par :

$$m^0(x) = \min_y \{f^1(x,y) : y \in M(x)\} \quad (2.7)$$

la valeur optimiste de la fonction objectif du premier niveau. Alors, l'approche optimiste du problème de programmation bi-niveaux est réduite à :

$$\min \{m^0(x) : g^1(x,y) \leq 0\} \quad (2.8)$$

Un couple $(\bar{x}, y(\bar{x}))$ tel que \bar{x} est solution de (2.8), et $y(\bar{x})$ solution de (2.7) pour $x = \bar{x}$, est également une solution optimale pour le problème :

$$\min_{x,y} \{f^1(x, y) : g^1(x, y) \leq 0\} \quad (2.9)$$

Remarque 2.1. la plupart des contributions au problème de programmation bi-niveaux sont consacrées à ce problème. Une des raisons de cela est le fait que ce problème a une solution optimale sous des hypothèses raisonnables.

L'existence de la solution optimale du problème (2.6) dépend du théorème ci-dessous :

Théorème 2.1. (Dempe, 2002).

Si l'ensemble $\{(x, y) : g^1(x, y) \leq 0, g^2(x, y) \leq 0\}$ est non vide et compact, et l'hypothèse de qualification des contraintes de Mangasarian-Fromowitz est satisfaite pour x tel que $g^1(x, y) \leq 0$, alors le problème (2.6) a une solution.

Approche Pessimiste

Dans cette approche la coopération entre le premier et le second niveau n'est pas autorisée, c'est à dire le premier niveau se protège en limitant le dommage résultant d'une sélection indésirable du second niveau tout en respectant son objectif[12, 1].

La formulation dans ce cas devient :

$$\begin{aligned} & \min_{x,y} \max_y f^1(x, y) \\ & \text{s.c} \begin{cases} g^1(x, y) \leq 0 \\ x \in \mathbb{R}^n \\ y \in M(x) \end{cases} \end{aligned} \quad (2.10)$$

où $m^p(x) = \max_y \{f^1(x, y); y \in M(x)\}$ représente la pire valeur de la fonction objectif du premier niveau qui peut être atteinte sur l'ensemble des solutions du problème de second niveau.

Théorème 2.2. (Dempe, 2002).

Supposant que l'application multivoque $M(\cdot)$ est semi-continue inférieurement en tout point x tel que $g^1(x, y) \leq 0$, et que l'hypothèse de qualification des contraintes de Mangasarian-Fromowitz est satisfaite en chacun de ces points, alors le problème (2.10) a une solution.

Solution Affaiblie :

Si le problème de second niveau est non convexe (pour une valeur fixée du paramètre x), le calcul d'une solution optimale globale pour le problème de second niveau peut être intraitable (spécialement retracer la solution globale de l'application multivoque pour différentes valeurs du paramètre). Dans ce cas, il pourrait être utile de modifier le problème bi-niveaux tel qu'une solution optimale locale pour le problème du second niveau soit recherchée, au

lieu d'une solution optimale globale. Cependant, ceci pourrait hypothéquer complètement l'existence d'une solution optimale du problème de programmation bi-niveaux, comme l'illustre l'exemple ci-après.

Exemple : (Vogel (2002))[1]

Considérons le problème bi-niveaux :

$$\begin{aligned} \min_x f^1(x, y) &= (y + 1)^2 \\ \text{s.c} \left\{ \begin{array}{l} -3 \leq x \leq 2 \\ y \in M_s(x) \end{array} \right. \end{aligned} \quad (2.11)$$

Avec le problème de second niveau :

$$\begin{aligned} \min_y f^2(x, y) &= y^3 - 3y \\ \text{s.c} \left\{ \begin{array}{l} y \geq x \end{array} \right. \end{aligned} \quad (2.12)$$

Si $M_s(x)$ désigne l'ensemble des solutions optimales globales du dernier problème dans une approche optimiste, alors une solution optimale est $x^* = -2$, il n'existe pas de solutions optimales dans un contexte pessimiste. Mais, si $M_s(x)$ désigne par contre l'ensemble des solutions locales du problème de second niveau, on a les résultats suivants :

$$m_s^o(x) = \min\{f^1(x, y) : y \in M_s(x)\} = \begin{cases} (x + 1)^2 & \text{si } x \in [-3, -1] \cup [1, 2] \\ 4 & \text{si } x \in [-1, 1[\end{cases}$$

et

$$m_s^p(x) = \min\{f^1(x, y) : y \in M_s(x)\} = \begin{cases} (x + 1)^2 & \text{si } x \in]1, 2] \\ 4 & \text{si } x \in [-3, 1] \end{cases}$$

Alors, $\inf m_s^o(x) = 0$ et une solution optimale de ce problème n'existe pas. D'autre part, $\inf m_s^p(x) = 4$ et tous points $x \in [-3, 1]$ sont solutions optimales. Cette situation est due au fait que l'application multivoque sur les solutions optimales locales d'un problème d'optimisation paramétrique n'est pas semi continue supérieurement en général, si l'hypothèse de convexité est ôtée du théorème 2.4.

Afin de contrôler cette situation déplaisante, Vogel (2002) a introduit une condition faible pour une solution optimiste ou pessimiste. Pour cela, considérons l'application multivoque $\bar{M}_s := cl(M_s)$ définie via la fermeture du graphe de l'application multivoque M_s :

$$gr(\bar{M}_s) := cl(gr(M_s))$$

Considérons le problème :

$$\begin{aligned} \min_x f^1(x, y) \\ \text{s.c} \left\{ \begin{array}{l} g^1(x, y) \leq 0 \\ y \in M_s(x) \end{array} \right. \end{aligned} \quad (2.13)$$

et définissons :

$$\begin{aligned}
\bar{m}^o(x) &:= \inf_{y \in M_s(x)} f^1(x, y) \\
\bar{f}_o^1 &:= \inf_{x: g^1(x, y) \leq 0} \bar{m}^o(x) \\
\bar{m}^p(x) &:= \sup_{y \in M_s(x)} f^1(x, y) \\
\bar{f}_p^1 &:= \sup_{x: g^1(x, y) \leq 0} \bar{m}^o(x)
\end{aligned}$$

où M_s est une application multivoque définie par les solutions du problème du second niveau (par exemple, des solutions optimales locales ou globales ou encore des points stationnaires).

Définition 2.1. (Vogel(2002))

Considérons le problème définie par (2.13).

- 1) un point \bar{x} avec $g^1(\bar{x}, \bar{y}) \leq 0$ est une solution optimiste faible du problème de programmation bi-niveaux s'il y a un certain $\bar{y} \in \bar{M}_s(\bar{x})$ tel que $f^1(\bar{x}, \bar{y}) = \bar{f}_o^1$.
- 2) un point \bar{x} avec $g^1(\bar{x}, \bar{y}) \leq 0$ est une solution pessimiste faible du problème de programmation bi-niveaux s'il y a un certain $\bar{y} \in \bar{M}_s(\bar{x})$ et une séquence $\{x^k\}_{k=1}^\infty$ tels que $\lim_{x \rightarrow \infty} x^k = \bar{x}$ et $\lim_{x \rightarrow \infty} \bar{m}_p(x^k) = \bar{f}_p^1$ aussi bien que $f^1(\bar{x}, \bar{y}) = \bar{f}_o^1$.

Les définitions de solution optimiste faible ou de solution pessimiste faible sont différentes, puisque s'il existe une solution optimiste faible, on peu montrer que la propriété additionnelle similaire à celle utiliser pour la solution solution pessimiste est satisfaite. Le théorème suivant est nécessaire pour assuré qu'une telle solution n'est pas une solution isolée, située loin de toutes solutions réalisables pour des problème légèrement perturbés.

Théorème 2.3. (Vogel(2002))

Supposons que l'ensemble $\{(x, y) : y \in M_s(x), g^1(x, y) \leq 0\}$ est non vide et bornée. Alors, le problème de programmation bi-niveaux (2.13) comporte une solution optimiste faible et une solution pessimiste faible.

2.1.4 Programmation linéaire bi-niveaux :

Dans un problème de programmation linéaire bi-niveaux les fonctions objectifs du premier et du deuxième niveaux sont linéaires ainsi que toutes leurs contraintes. Ci-dessous on a une formulation générale d'un problème de programmation linéaire bi-niveaux :

$$\begin{aligned}
& \min_{x, y} C_1^t x + D_1^t y \\
& \left\{ \begin{array}{l} A_1 x + B_1 y \leq b_1 \\ x \geq 0 \end{array} \right. \\
& \text{s.c } \left\{ \begin{array}{l} \min_y C_2^t x + D_2^t y \\ \text{s.c } \left\{ \begin{array}{l} A_2 x + B_2 y \leq b_2 \\ y \geq 0 \end{array} \right. \end{array} \right. \quad (2.14)
\end{aligned}$$

Où $x \in \mathbb{R}^n$, $y \in \mathbb{R}^m$, $C_l \in \mathbb{R}^n$, $D_l \in \mathbb{R}^m$, $A_l \in \mathbb{R}^n \times \mathbb{R}^m$, $B_l \in \mathbb{R}^n \times \mathbb{R}^m$, $b_l \in \mathbb{R}^m$, $l = 1, 2$

Définition 2.2. Les définitions ci-dessous caractérisent les solutions d'un problème de programmation linéaire bi-niveaux :

- a) Le domaine Ω des contraintes du problème de programmation linéaire bi-niveaux est défini par :

$$\Omega = \{x \geq 0, y \geq 0 : A_1x + B_1y \leq b_1, A_2x + B_2y \leq b_2\}.$$
- b) L'ensemble des solutions réalisables du second niveau (suiveur) pour un x fixé est noté par $\Omega_y(x)$ et est défini par : $\Omega_y(x) = \{y \geq 0 : B_2y \leq b_2 - A_2x\}.$
- c) La trace de Ω sur l'ensemble des décisions du leader est définie par :

$$\Omega_x^2 = \{x \geq 0 : \Omega_y(x) \neq \emptyset\}.$$
- d) L'ensemble des solutions optimales du second niveau (réactions rationnelles du suiveur) pour $x \in \Omega_x^2$, est donné par :

$$M(x) = \{y : y = \operatorname{argmax}[C_2^t x + D_2^t y : y \in \Omega_y(x)]\}.$$
- e) La fonction valeur : $V(x) = \{C_2^t x + D_2^t y, y \in M(x)\}$
- f) Le domaine induit (ou ensemble induit) est défini par : $DI = \{(x, y) \in \Omega, y \in M(x)\}$

L'union de toutes les décisions que le leader peut prendre (c.à.d toutes les solutions du premier niveau) ainsi que les réactions rationnelles correspondantes du suiveur forment le domaine induit DI . [1, 14]

Le caractère général d'un problème de programmation linéaire bi-niveaux inclue des contraintes du premier niveau ($A_1x + B_1y \leq b_1$, et $x \geq 0$). Malgré cela, la plupart des chercheurs travaillant dans ce domaine utilisent une formulation sans contraintes du premier niveau. [12]

Exemple : Dans ce qui suit, un exemple d'un problème de programmation linéaire bi-niveaux est présenté :

$$\begin{array}{l} \min_{x,y} -8x + 16y \\ \left. \begin{array}{l} x \geq 0 \\ \min_y -y \\ s.c \left\{ \begin{array}{l} 3x + 2y \leq 22 \\ -x + y \leq 1 \\ -2x - y \leq -4 \\ 2x - 3y \leq 6 \\ y \geq 0 \end{array} \right. \end{array} \right\} \end{array} \quad (2.15)$$

Le problème de second niveau ci-dessus a pour fonction objectif $f^2(x, y) = y$ et 4 contraintes linéaires qui forment un polyèdre présenté dans les figures qui suivent :

La figure 2.2, ci-dessous, représente le domaine induit du problème de second niveau.

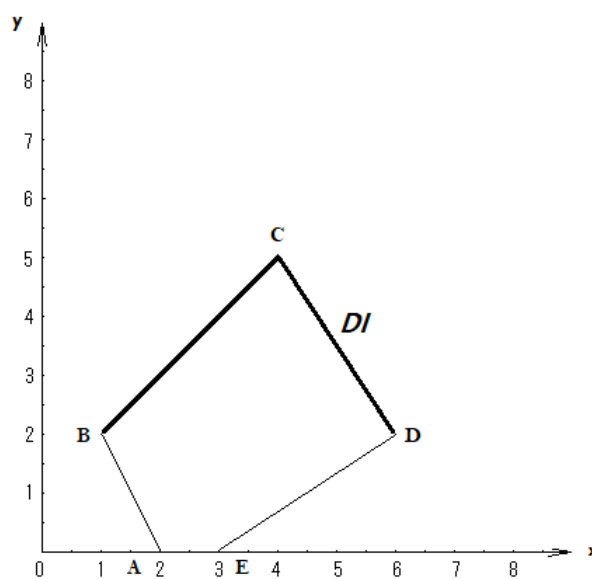


FIGURE 2.2 – domaine induit du problème de second niveau

Où :

$$y \in M(x) = \begin{cases} 11 - (3/2)x & \text{si } 6 \geq x \geq 4 \\ 1 + x & \text{si } 4 \geq x \geq 1 \end{cases}$$

La figure 2.3, ci-dessous, représente la solution du problème de premier niveau en tenant compte du domaine induit qui est atteint au point $D = (6, 2)$.

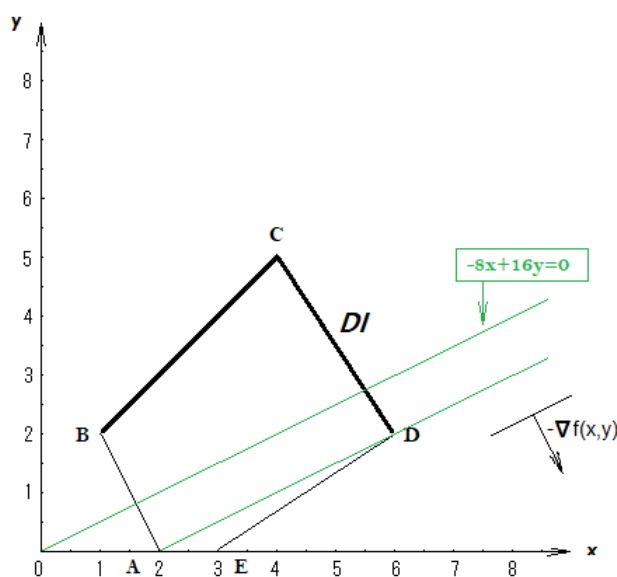


FIGURE 2.3 – Solution graphique du problème de premier niveau

Remarque 2.2. la solution du problème se réalise au point $D = (6, 2)$.

Exemple 2 : Dans ce qui suit, le problème de programmation linéaire bi-niveaux obtenue en passant la première contrainte du problème du second niveau du problème de l'exemple 1 au niveau supérieur.

$$\begin{array}{l} \min_{x,y} -8x + 16y \\ \text{s.c} \left\{ \begin{array}{l} x \geq 0 \\ 3x + 2y \leq 22 \\ \min_y -y \\ \text{s.c} \left\{ \begin{array}{l} -x + y \leq 1 \\ -2x - y \leq -4 \\ 2x - 3y \leq 6 \\ y \geq 0 \end{array} \right. \end{array} \right. \end{array} \quad (2.16)$$

La figure 2.4, ci-dessous, représente le domaine induit du problème de second niveau.

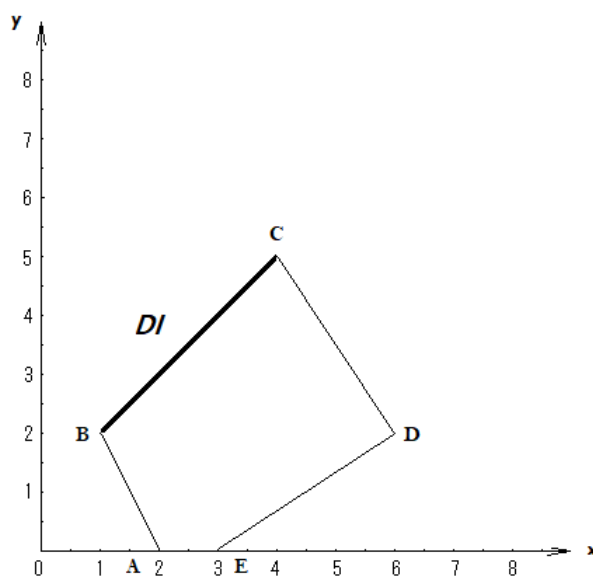


FIGURE 2.4 – domaine induit du problème de second niveau

Où :

$$y \in M(x) = \{1 + x \text{ si } 4 \geq x \geq 1\}$$

La figure 2.5, ci-dessous, représente la solution du problème de premier niveau en tenant compte du domaine induit qui est atteint au point $D = (6, 2)$.

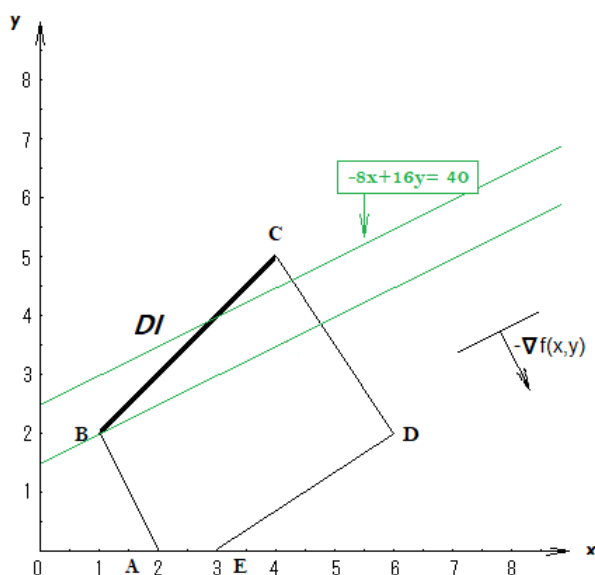


FIGURE 2.5 – Solution graphique du problème de premier niveau

Remarque 2.3. la solution du problème se réalise au point $B = (1, 2)$.

Complexité d'un problème de programmation linéaire bi-niveaux :

On rappelle quelques définitions de base dont on a besoin pour étudier la complexité du problème de programmation linéaire bi-niveaux :

Définition 2.3. (Opération élémentaire)

Une opération élémentaire est une opération qui prend un temps constant d'exécution quel que soit la donnée. Par exemples : les opérations $+$, $*$, $-$, et les comparaisons $>$, $<$.

Définition 2.4. (Problème de décision)

Un problème de décision Q est un problème pour lequel tout ses solutions sont incluses dans l'ensemble $I_Q = \{Oui, Non\}$.

Définition 2.5. (Réduction) Soient Q et Q' deux problèmes de décision avec x instance de Q et y instance de Q' . Une réduction polynomiale du problème Q vers le problème Q' est une application f définie :

$$f : I_Q \rightarrow I_{Q'} \\ x \mapsto y = f(x)$$

tel que :

- f est calculable d'une manière polynomiale.
- $\forall x \in I_Q$ est vrai $\Leftrightarrow f$ est vrai.

On écrit donc :

$$Q \leq Q'.$$

Définition 2.6. (Classe P)

La classe P est la classe des problèmes de décision qui admettent un algorithme de complexité polynomiale (il admet donc un nombre d'opérations élémentaires polynomiales).

Définition 2.7. (Classe NP)

La classe NP est formée des problèmes de décision Q dont chaque instance peut être vérifiée en un temps polynomiale.

Définition 2.8. (NP-C)

Un problème Q est NP-complet si

- 1) Q est NP.
- 2) tout autre problème Q' de NP $Q' \leq Q$.

Remarque 2.4. Si le problème Q ne satisfait que la condition 2) il est dit NP-difficile.

Remarque 2.5. NP ne veut pas dire non-polynomial, il signifie non-déterministe polynomiale.

Remarque 2.6. Intuitivement on définit : $P \subseteq NP$ comme représenté dans la figure 2.6 ci-dessus. Cependant, l'idée que $P=NP$ n'est pas hors de question, si elle est démontrée la théorie de la complexité algorithmique serait mise en question.

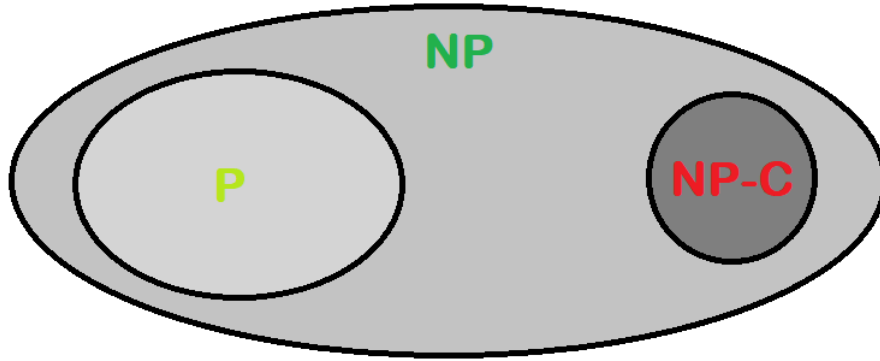


FIGURE 2.6 – Relation entre les différentes classes de problèmes

On montre d'abord que la version problème de décision du problème linéaire bi-niveaux appartient au NP. Ci-dessous le problème (2.14), dans le cas optimiste, est écrit sous forme d'un problème de décision :

Données : $C_l \in \mathbb{R}^n$, $D_l \in \mathbb{R}^m$, $A_l \in \mathbb{R}^n \times \mathbb{R}^m$, $B_l \in \mathbb{R}^n \times \mathbb{R}^m$, $b_l \in \mathbb{R}^m$, $l = 1, 2$ et une constante $\alpha \in \mathbb{R}$.

Question : On cherche si une solution $x^* \geq 0$ existe tel que le problème de second niveau a une solution optimale, et au moins une solution optimal y^* du problème de second niveau satisfais $A_1 x^* + B_1 y^* \leq b_1$, et $C_1^t x^* + D_1^t y^* \leq \alpha$.

On note que cette version du problème linéaire bi-niveaux, dans le cas optimiste, est équivalente au problème qui cherche à savoir si le problème (2.14) admet une solution réalisable, car la contrainte $C_1^t x + D_1^t y \leq \alpha$ joue le rôle d'une contrainte de couplage supplémentaire.

Théorème 2.4. [16]

la version problème de décision du problème linéaire bi-niveaux dans le cas optimiste est NP.

Ci-dessous le problème (2.14), dans le cas pessimiste, est écrit sous forme d'un problème de décision :

Données : $C_l \in \mathbb{R}^n$, $D_l \in \mathbb{R}^m$, $A_l \in \mathbb{R}^n \times \mathbb{R}^m$, $B_l \in \mathbb{R}^n \times \mathbb{R}^m$, $b_l \in \mathbb{R}^m$, $l = 1, 2$ et une constante $\alpha \in \mathbb{R}$.

Question : On cherche si une solution $x^* \geq 0$ existe telle que le problème de second niveau a une solution optimale, et toutes solutions optimales y^* du problème de second niveau satisfait $A_1 x^* + B_1 y^* \leq b_1$, et $C_1^t x^* + D_1^t y^* \leq \alpha$.

Dans cette formulation on rencontre une certaine difficulté supplémentaire : pour une instance positive, nous devons nous assurer que x^* est choisi de telle sorte que toutes les solutions optimales du problème de niveau inférieur répondent aux contraintes de couplage et $C_1^t x^* + D_1^t y^* \leq \alpha$.

Théorème 2.5. [16] La version problème de décision du problème linéaire bi-niveaux dans le cas pessimiste est NP.

2.1.5 Condition d'optimalité :

Cas linéaire :

Pour le problème de programmation linéaire bi-niveaux, bien que le DI soit non convexe en général, il possède certaines propriétés des ensemble convexes. Les plus importantes étant les suivantes :[1]

Propriété. [17]

Tout point extrême du domaine induit DI est aussi point extrême du domaine réalisable Ω du problème linéaire bi-niveaux.

Propriété. [18, 19]

Lorsqu'un problème linéaire bi-niveaux admet une solution optimale, cette solution est atteinte en un point extrême du domaine induit DI .

Calvette et Galé (1998) ont généralisé ce résultat lorsque les fonctions économiques du première et du second niveau sous la forme d'une minimisation sont quasi concaves.

Utilisation des conditions de KKT :

Comme noté ci-dessus, le problème bi-niveaux peut être reformulé en un problème à un niveau en remplaçant le problème suiveur par les conditions de KKT associées, on obtient le problème (2.4).

Il est possible d'obtenir des conditions nécessaire d'optimalité pour le problème bi-niveaux avec cette formulation tant que le problème de second niveau est un problème paramétrique convexe (seulement dans l'approche optimiste). Cependant, même dans ce cas, ceci n'est pas facile, puisque il a été établi que le problème (2.4) ne satisfait pas aux hypothèse de régularité classiques (voir Scheel et Scholres, 2000).[1]

Ci-dessous un problème relaxé du problème (2.4) :

$$\begin{aligned} & \min_{x,y} f^1(x,y) \\ & \text{s.c} \left\{ \begin{array}{l} \nabla_y f^2(x,y) + \lambda^t \nabla_y g^2(x,y) = 0 \\ g_i^2(x,y) = 0, \text{ for } g_i^2(x^*, y^*) = 0 \\ \lambda_i = 0, \text{ for } \lambda_i^* = 0 \\ g_i^2(x,y) \leq 0, \text{ for } g_i^2(x^*, y^*) < 0 \\ \lambda_i \geq 0, \text{ for } \lambda_i^* > 0 \\ g^1(x,y) \leq 0 \end{array} \right. \end{aligned} \quad (2.17)$$

Dans le théorème qui suis on a besoin de l'hypothèse de qualification des contraintes de Mangasarian-Fromovitz *stricte* :

Proposition 3. (MFCQS) [3]

l'hypothèse de qualification des contraintes de Mangasarian-Fromovitz stricte est satisfaite au point x^* pour le problème (1.2), s'il existe des multiplicateur de Lagrange (λ, μ) ,

$$\lambda \geq 0, \lambda^t g(x^*) = 0, \nabla f(x^*) + \lambda^t \nabla g(x^*) + \mu^t \nabla h(x^*) = 0$$

et une direction d satisfaisant

$$\begin{aligned} \nabla f_i(x^*)d &< 0, \text{ pour chaque } i \text{ avec } g_i(x^*) = \lambda_i = 0, \\ \nabla g_i(x^*)d &= 0, \text{ pour chaque } i \text{ avec } \lambda_i > 0, \\ \nabla h_j(x^*)d &= 0, \text{ pour chaque } j \end{aligned}$$

et $\{\nabla g_i(x^*) : \lambda_i > 0\} \cup \{\nabla_x h_j(x^*) : j = 1, \dots, p\}$ sont linéairement indépendant.

Théorème 2.6. [3]

Soient (x^*, y^*, λ^*) un minimum local du problème (2.4) et utilisant $z^* = (x^*, y^*)$.

- Si l'hypothèse de qualification des contraintes de Mangasarian-Fromovitz (MFCQ) est valide pour le problème (2.17) au point (x^*, y^*, λ^*) , donc il existe des multiplieurs $(\kappa, \omega, \zeta, \xi)$ satisfaisant :

$$\left\{ \begin{array}{l} \nabla f^1(z^*) + \kappa^t(0, \nabla_y g^1(y^*)) + \nabla(\nabla_x \mathcal{L}(z^*, \lambda^*)\omega) + \zeta^t \nabla g_2(z^*) = 0 \\ \nabla_x g_2(z^*)\omega - \xi = 0 \\ (g_2)_i(z^*)\zeta_i = 0, \forall i \\ \lambda_i^* \xi_i = 0, \forall i \\ \zeta_i \xi_i \geq 0, i \in K \\ \kappa^t g^1(y^*) = 0 \\ \kappa \geq 0 \end{array} \right. \quad (2.18)$$

Où $K = \{i : (g_2)_i(x^*, y^*) = \lambda^* = 0\}$.

- Si l'hypothèse de qualification des contraintes de Mangasarian-Fromovitz stricte (MFCQS) est satisfaite pour le problème (2.17) donc il existe des multiplicateurs $(\kappa, \omega, \zeta, \xi)$ résolvant le système d'équations (2.18) avec l'équation $\zeta_i \xi_i \geq 0, i \in K$ remplacé par

$$\zeta_i \geq 0, \xi_i \geq 0, i \in K$$

Approche au dual de Fenchel-Lagrange d'un problème bi-niveaux avec fonction de valeur :

Soit le problème suivant :

$$\begin{array}{l} \min_x f^1(x, v(x)) \\ \text{s.c } \left\{ \begin{array}{l} x \in X \\ \min_y f^2(x, y) \\ \text{s.c } \{ y \in Y \end{array} \right. \end{array} \quad (2.19)$$

Où $v(x)$ est la valeur optimale du problème de niveau inférieur, X, Y deux sous-ensembles compacts convexes de \mathbb{R}^n et \mathbb{R}^m (respectivement), $f^1 : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}$, et $f^2 : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}$. Le problème (2.19) est appelé un problème bi-niveaux avec la fonction de valeur extrémal $v(\cdot)$.

On définit un dual du problème (2.19) impliquant des fonctions conjuguées, appelées Dual de Fenchel-Lagrange de (2.19). Cette dualité a été initialement introduite dans [16] pour des problèmes de programmation convexe ordinaires et ensuite utilisé dans plusieurs œuvres

Soit $h : \mathbb{R}^n \rightarrow \mathbb{R}^{n+1}$ une fonction définie par

$$h(x) := (h_1(x), \dots, h_n(x), h_{n+1}(x))^t = (x, v(x))^t,$$

c'est à dire pour $x = (x_1, \dots, x_n)^t$,

$$\begin{cases} h_i(x) = x_i, & \text{for } i = 1, \dots, n, \\ h_i(x) = v(x), & \text{for } i = n + 1, \end{cases}$$

Ensuite, le problème (2.19) peut être écrit comme suit

$$\begin{aligned} & \min_x f^1(h(x)) \\ & \text{s.c } \begin{cases} x \in X \\ \min_y f^2(x, y) \\ \text{s.c } \{ y \in Y \end{cases} \end{aligned} \quad (2.20)$$

qui est un problème de programmation composé.

Dans ce qui suite, afin de donner le dual du problème (2.19), on adopte la procédure donnée par boç et al. (2006) pour les problèmes de programmation composés. Ensuite, on commence l'approche dual en tenant compte du problème de minimisation suivant

$$\begin{aligned} & \min f^1(y) \\ & \text{s.c } \begin{cases} x \in X \\ y \in \mathbb{R}^{n+1} \\ h(x) \leq y \end{cases} \end{aligned} \quad (2.21)$$

On fait les hypothèses suivantes :

- i) La fonction f^1 est croissante sur \mathbb{R}_+^{n+1} .
- ii) La fonction f^1 est convexe sur \mathbb{R}^{n+1} .
- iii) La fonction f^2 est convexe sur $\mathbb{R}^n \times \mathbb{R}^m$.

Soient $\text{Inf}(P_{2.19})$ et $\text{Inf}(P_{2.21})$ désignant les bornes inférieures des problèmes (2.19) et (2.21), respectivement. On a ensuite le résultat suivant.

Proposition 4. [20]

Soient les hypothèses **i)**, **ii)**, **iii)** vérifiées. Alors

- Le problème (2.19) a au moins une solution,
- $\text{Inf}(P_{2.19}) = \text{Inf}(P_{2.21})$.

Soit le problème dual de Lagrange associé a (2.21),

$$\begin{aligned}
& \text{Max}_{\alpha} \inf_{x,y} \{f^1(y) + \langle \alpha, h(x) - y \rangle\} \\
& \text{s.c.} \begin{cases} \alpha \geq 0, \alpha \in \mathbb{R}^{n+1} \\ x \in X \\ y \in \mathbb{R}^{n+1} \end{cases}
\end{aligned} \tag{2.22}$$

et considérons le problème de maximisation suivant dans lequel la fonction objectif est exprimée en termes de fonctions conjugué de f^1 , $\alpha^t h$ et ψ_X

$$\begin{aligned}
& \text{Max}_{\alpha,\beta} \{-(f^1)^*(\alpha) - (\alpha^t h)^*(\beta) - \psi_X^*(-\beta)\} \\
& \text{s.c.} \begin{cases} \alpha \in \mathbb{R}_+^{n+1} \\ \beta \in \mathbb{R}^n \end{cases}
\end{aligned} \tag{2.23}$$

où pour $\alpha = (\alpha, \dots, \alpha_{n+1})^t \in \mathbb{R}^{n+1}$, $\alpha^t h$ désigne la fonction définie sur \mathbb{R}^n dans \mathbb{R} , par

$$\alpha^t h(x) = \langle \alpha, h(x) \rangle = \sum_{i=1}^{n+1} \alpha_i^t h_i(x) = \sum_{i=1}^n \alpha_i x_i + \alpha_{n+1} v(x),$$

avec $x = (x_1, \dots, x_n)^t$. En utilisant le fait que $\psi_X^* = \sigma_X$, le problème dual (2.23) peut également être écrit sous la forme

$$\begin{aligned}
& \text{Max}_{\alpha,\beta} \{-(f^1)^*(\alpha) - (\alpha^t h)^*(\beta) - \sigma_X(-\beta)\} \\
& \text{s.c.} \begin{cases} \alpha \in \mathbb{R}_+^{n+1} \\ \beta \in \mathbb{R}^n \end{cases}
\end{aligned} \tag{2.24}$$

(2.24) est appeller le problème dual de Fenchel-Lagrange par rapport à (2.19)

Théorème 2.7. [20]

Soient les hypothèses **i)**, **ii)**, **iii)** vérifiées. Alors il y a une forte dualité entre (2.19) et (2.24), c'est-à-dire

- $\text{Inf}(P_{2.19}) = \text{Sup}(D_{2.24})$.
- le dual (2.24) admet des solutions.

Conditions d'optimalité pour (2.19) et (2.24)

Les théorèmes suivants donnent les conditions d'optimalité nécessaire et suffisamment respectivement pour le problème (2.19) et son dual.

Théorème 2.8. (Conditions nécessaires d'optimalité)[20]

Soient les hypothèses **i)**, **ii)**, **iii)** vérifiées. Soient $\bar{x} \in X$ et $\bar{\alpha}, \bar{\beta} \in \mathbb{R}_+^{n+1} \times \mathbb{R}^n$ des solutions de problèmes (2.19) et (2.24), respectivement. Alors, \bar{x} et $\bar{\alpha}, \bar{\beta}$ vérifient les conditions d'optimalité nécessaires suivantes :

$$\begin{aligned}
f^1(h(\bar{x})) + (f^1)^*(\bar{\alpha}) &= \langle \bar{\alpha}, h(\bar{x}) \rangle, \\
\langle \bar{\alpha}, h(\bar{x}) \rangle + (\bar{\alpha}^t h)^*(\bar{\beta}) &= \langle \bar{\beta}, \bar{x} \rangle, \\
\sigma_X(-\bar{\beta}) + \langle \bar{\beta}, \bar{x} \rangle &= 0.
\end{aligned}$$

Théorème 2.9. (conditions suffisantes d'optimalité)[20]

Soient les hypothèses **i)**, **ii)**, **iii)** vérifiées. Soient \bar{x} et $\bar{\alpha}, \bar{\beta}$ des points réalisable de (2.19) et (2.24), respectivement. Supposons qu'ils satisfont les conditions cités dans le théorème 2.8. Alors, \bar{x} est une solution du problème (2.19) et $\bar{\alpha}, \bar{\beta}$ sont des solutions du problème (2.24),

Problème biniveaux simple :

Soit le problème d'optimisation bi-niveau convexe suivant :

$$\begin{aligned} & \min f^1(x) \\ & \text{s.c} \left\{ \begin{array}{l} \min f^2(x) \\ \text{s.c} \{ x \in X \end{array} \right. \end{aligned} \quad (2.25)$$

Où X un ensemble convexe fermé dans \mathbb{R}^n , $f^1, f^2 : \mathbb{R}^n \rightarrow \mathbb{R}$ fonction objectif convexe du problème de premier niveau et de second niveau. On peut également supposer que $f^1, f^2 : \mathbb{R}^n \rightarrow \bar{\mathbb{R}}$ où $\bar{\mathbb{R}} = \mathbb{R} \cup \{-\infty, +\infty\}$ sont des fonctions convexes et semi-continues inférieurement. Le problème indiqué ci-dessus est appelé problème de programmation bi-niveaux simple. On appelle ce problème simple car il est représenté par une seule variable de décision contrairement à la représentation à deux variables dans le problème bi-niveaux standard. Tout problème d'optimisation convexe standard peut être exprimé en tant que problème de programmation bi-niveaux simple.[21]

D'abord on reformule le problème un problème équivalent à un niveau. Supposons que le problème de niveau inférieur dans (2.25) a une borne inférieure finie. Soit $\alpha = \inf_{x \in X} f^2(x)$, on reformule le problème donc comme suit :

$$\begin{aligned} & \min f^1(x) \\ & \text{s.c} \left\{ \begin{array}{l} f^2(x) \leq \alpha \\ x \in X \end{array} \right. \end{aligned} \quad (2.26)$$

Les conditions de KKT pour le (2.25) est définie comme les conditions de KKT pour le problème reformulé (2.26). Les conditions de KKT pour n'importe quel x réalisable de (2.26) supposent l'existence de $\lambda \geq 0$, telle que

- 1) $0 \in \partial f^1(x) + \lambda \partial f^2(x) + N_X(x)$
- 2) $\lambda(f^2(x) - \alpha) = 0$

avec $N_X(x)$ équivalent au produit des dérivés des contraintes du problème, engendré de X , et des multiplicateurs correspondants.

Si x^* est la solution de (2.25), alors $f^2(x^*) - \alpha = 0$ et donc **2)** est automatiquement vérifié. Ainsi, si x^* résout (2.25), on a

$$0 \in \partial f^1(x^*) + \lambda \partial f^2(x) + N_X(x)$$

comme condition de KKT requise. Cela doit être à la fois nécessaire et suffisant[21].

Dans le cas où le problème de niveau inférieur est un problème de programmation linéaire, c'est à dire on étudie le problème suivant :

$$\begin{aligned} & \min f(x) \\ & \text{s.c} \left\{ \begin{array}{l} \min C^t x \\ \text{s.c} \{ Ax = b \} \end{array} \right. \end{aligned} \quad (2.27)$$

Où $C \in \mathbb{R}^n$, A est une matrice $m \times n$ et $b \in \mathbb{R}^m$. En raison de la simplicité, supposons que f est différentiable. Supposons que

$$\alpha = \inf\{C^t x : Ax = b\}$$

Ensuite, on peut le reformuler de la même façon que le problème (2.26) :

$$\begin{aligned} & \min f(x) \\ & \text{s.c} \left\{ \begin{array}{l} C^t x \leq \alpha \\ Ax = b \end{array} \right. \end{aligned} \quad (2.28)$$

Étant donné que (2.28) a des contraintes linéaires, la qualification de contrainte n'est pas nécessaire pour que les conditions de KKT soient vérifiées.

Soit x^* une solution de (2.27). Ensuite, il existe $\lambda \geq 0$ et $\mu \in \mathbb{R}^m$ telle que

$$\begin{aligned} 0 &= \nabla f(x^*) + \lambda C - A^t \mu \\ \nabla f(x^*) &= A^t \mu - \lambda C \end{aligned} \quad (2.29)$$

Soit x^* un point réalisable de (2.27) et supposer qu'il existe $\mu \in \mathbb{R}^m$ et $\lambda \geq 0$, de telle sorte que 2.29 est vérifié. Pour vérifier si x^* est une solution de (2.27), il est important de noter que x^* est également réalisable pour (2.28) car $C^t x^* = \alpha$. Pour tout x point réalisable de (2.27), On a

$$\begin{aligned} f(x) - f(x^*) &\geq \langle \nabla f(x^*), x - x^* \rangle, \\ C^t x - C^t x^* &= C^t (x - x^*). \end{aligned}$$

Ces deux ensemble impliquent que

$$f(x) - f(x^*) + \lambda C^t x - \lambda C^t x^* \geq \langle \nabla f(x^*) + \lambda C, x - x^* \rangle.$$

Avec $C^t x = C^t x^* = \alpha$, en utilisant 2.29, On a

$$f(x) - f(x^*) \geq \langle A^t \mu, x - x^* \rangle,$$

Cela implique que

$$f(x) - f(x^*) \geq \langle \mu, Ax - Ax^* \rangle,$$

Comme $Ax = Ax^* = b$, on obtien $f(x) - f(x^*) \geq 0$. Par conséquent,

$$f(x) \geq f(x^*)$$

ce qui implique que x^* est une solution du problème (2.27).

En particulier, Soit $f(x) = \frac{1}{2}\|x\|^2$ et $\alpha = \inf\{C^t x : Ax = b\}$. donc x^* est une solution du problème de solution norme minimum du problème de programmation linéaire.

$$\begin{aligned} \min C^t x \\ \text{s.c } \{ Ax = b \end{aligned} \tag{2.30}$$

si et seulement s'il existe $\mu \in \mathbb{R}^m$ et $\lambda \geq 0$, tel que

$$x^* = A^t \mu - \lambda C A x^* = b \quad C^t x^* = \alpha \tag{2.31}$$

Donc, l'idée de clé est de résoudre d'abord le problème de programmation linéaire et de trouver α , puis de résoudre le système d'équations ci-dessus pour trouver la solution de norme minimale.

2.1.6 Méthode de résolution :

De nombreux algorithmes ont été mis au point dans le but de résoudre les problèmes de programmations bi niveaux, surtout dans le cas linéaire. On peut classer la plupart des algorithmes de résolution de ces problèmes en trois catégories principales : Les algorithmes d'explorations de points extrêmes, dans le cas de problèmes de programmations bi niveaux linéaire en exploitant la propriété selon la quel une solution optimale de tout problème linéaire est atteinte en un point extrême de l'ensemble de solutions réalisables Ω . Ensuite, les algorithmes basés sur la formulation KKT : basée sur la reformulation du problèmes de programmations bi niveaux en un problème à un seul niveau en utilisant les conditions de KKT du problème de second niveau. Enfin, les algorithmes de descente qui exploitent des formulations du problèmes de programmations bi niveaux.

Algorithmes d'explorations de points extrêmes :

Parmi les algorithmes de cette catégorie le meilleur est l'un des algorithmes les plus connus pour la résolution de problèmes bi niveaux linéaire (2.14). Sachant que pour $\mathfrak{D} = D_1 \cap D_2$, D_1 polyèdre définie par les contraintes du leader et D_2 polyèdre définie par les contraintes du suiveur, donc il existe un point extrême de \mathfrak{D} qui résout le problème. Ensuite, il suffira d'examiner tous les points extrêmes de ce polyèdre pour trouver la solution optimale du problème en un nombre fini d'étapes. Cela n'est toutefois pas suffisant, car le nombre de points extrêmes de \mathfrak{D} est généralement très grand.

Bialas et Karwan (1982) traitent le problème de LB sans contraintes de couplage en supposant que D_2 est compacte et propose le meilleur algorithme pour le résoudre. L'idée est de sélectionner le meilleur point extrême de D_2 par rapport à la fonction objectif f^1 du premier niveau qui est une solution réalisable du problème bi niveaux.

Voici le problème relaxé de notre problème bi niveaux linéaire (2.14) qui servira à sa résolution :

$$\begin{aligned} & \min_{x,y} C_1^t x + D_1^t y \\ & \text{s.c} \left\{ \begin{array}{l} A_1 x + B_1 y \leq b_1 \\ A_2 x + B_2 y \leq b_2 \\ x \geq 0 \\ y \geq 0 \end{array} \right. \end{aligned} \quad (2.32)$$

Par conséquent, une solution optimale au problème relaxe (2.33) ci dessus (x_r^*, y_r^*) est d'abord envisagée. S'il s'agit d'un point de DI , il s'agit alors d'une solution optimale du problème (2.14). Si ce n'est pas le cas, notez $(x_{[1]}, y_{[1]}) = (x_r^*, y_r^*)$ et calculez l'ensemble de ses points extrêmes adjacents $W_{[1]}$.

Ensuite, le point extrême de $W = W_{[1]}$ qui fournit la meilleure valeur de la fonction objectif du leader est sélectionné pour tester s'il s'agit d'un point de DI . On note ce point extrême $(x_{[2]}, y_{[2]})$. Si il est dans DI , l'algorithme se termine. Si ce n'est pas le cas, le point est éliminé de W et l'ensemble de ses points extrêmes adjacents avec une valeur pire dans la fonction objectif du leader, $W_{[2]}$, est ajouté à W .

L'algorithme se poursuit en sélectionnant le meilleur point extrême de W par rapport à la fonction d'objectif du leader et en répétant le processus. Notez que l'algorithme suit un chemin au bornes de D_2 en commençant de la solution optimale du problème relaxé (x_r^*, y_r^*) à un point extrême de D_2 qui est une solution optimale du problème de (2.14) et un point extrême réalisable.

L'efficacité de la K^{ieme} meilleur algorithme dépend fortement de la "proximité" de la solution optimale du problème relaxé à la solution optimale du problème de (2.14).[22]

Approche Paramétrique :

On note que le problème de second niveau peut être considéré comme un problème d'optimisation paramétrique aux variables du leader. Basé la-dessus, Faísca et al. (2007) a développé une approche globale d'optimisation pour résoudre le problème (2.14) basée sur la théorie de la programmation paramétrique. Les variables x sont considérées comme un paramètre vectoriel du problème de second niveau. Ce problème ce résous par un algorithme de programmation multiparamétrique afin de calculer la valeur d'une fonction $y(x)$ noter comme suit :

$$y(x) = \begin{cases} \alpha^1 + \beta^1 x & \text{si } H^1 x \leq h^1 \\ \vdots \\ \alpha^K + \beta^K x & \text{si } H^K x \leq h^K \end{cases}$$

où α^k et h^k sont de vecteurs réel et β^k et H^k sont des matrices réel. L'ensemble de x admissible est divisé en un ensemble de K régions, chacun avec un ensemble de d'ensembles rationnels. L'expression $y(x)$ dans le problème leader y compris, les K problèmes de programmation linéaire ($k = 1, \dots, K$) suivants sont résolus :

$$\begin{aligned} & \min_x C_1^t x + D_1^t (\alpha^k + \beta^k x_i) \\ & \text{s.c } \{ H^k x \leq h^k \end{aligned} \tag{2.33}$$

Ensuite, la solution optimale qui fournit la valeur minimale de la fonction objectif du premier niveau parmi les K solutions est une solution optimale du problème de (2.14). Comme indiqué par les auteurs (Faisca et al. (2007)), l'efficacité de cette procédure dépend de la performance de l'algorithme de programmation multiparamétrique utilisé.[22]

Les algorithmes de Branch-and-Bound :

Les algorithmes de Branch-and-Bound sont des algorithmes de résolution basé sur la reformulation KKT du problème (2.14). L'idée de cette approche est de supprimer le terme de complémentarité et résoudre le programme linéaire résultant. Bard et Moore (1990) ont développé un des premier algorithme de Branch-and-Bound traitant les problème linéaire bi niveaux. Ces auteurs traitent le problème bi niveaux linéaire quadratique sans contraintes de couplage, mais leur algorithme est considéré dans la littérature comme l'un des premiers algorithmes de Branch-and-Bound pour la résolution de problèmes bi niveaux linéaire.[13]

Chapitre 3

Contrôle optimal bi niveaux :

Dans ce chapitre on étudie le problème de contrôle optimal bi niveaux, ce problème est composé donc d'un problème de premier et de second niveau qui sont tous les deux des problèmes de contrôle optimal. Pour ce faire, on doit d'abord faire un rappel à propos de ce qu'est un problème de contrôle optimal et de ses propriétés pour ensuite les appliquer au problème de contrôle optimal bi niveaux.

3.1 Rappel sur le Contrôle optimal :

La théorie du contrôle s'intéresse aux systèmes dynamiques dépendant d'un paramètre (appelé système de contrôle ou bien commande), sur lequel on peut agir pour amener un système dynamique d'un état initial donné à un certain état final. Un problème de contrôle optimal quant à lui est un problème qui a pour but de déterminer une trajectoire optimale entre un ensemble initial et une cible si celle-ci existe, en agissant sur la commande du système dynamique, tout en optimisant un critère donné.

Le contrôle optimal est une généralisation du calcul variationnel, avec en plus de celui-ci une contrainte sur la dérivée de la fonction sur laquelle on optimise. Le problème de calcul variationnel est un problème d'optimisation dans lequel la variable à optimiser est une fonction. En général, il s'agit d'une fonction définie sur un intervalle, avec la formulation suivante :

$$\begin{cases} J(x(t)) = \int_{t_1}^{t_2} F(t, x(t), \dot{x}(t)) dt \longrightarrow \min_{x(t)} \\ x(t_1) = c_1, x(t_2) = c_2 \end{cases}$$

Où $x(t) \in \mathbb{R}^n$ la fonction recherchée (l'inconnue), $\dot{x}(t) \in \mathbb{R}^n$ la dérivée de la fonction recherchée, $J(x(t))$ la fonctionnelle à minimiser, $x(t_1) = c_1$ point initial (à l'instant t_1), $x(t_2) = c_2$ point final (à l'instant t_2).

Exemple : Ceci est un problème de calcul variationnel :

$$\begin{cases} J(x(t)) = \int_0^1 F(t, x(t), \dot{x}(t)) dt \longrightarrow \min_{x(t)} \\ x(0) = 0, x(1) = 2 \end{cases}$$

Où $F(t, x(t), \dot{x}(t)) = t^2 x(t) + \dot{x}^2(t)$.

Dans le problème de contrôle optimal, on cherche à minimiser (ou maximiser) une quantité de la forme :

$$J(u(t)) = g(x(T)) + \int_{t_0}^T f^0(t, x(t), u(t)) dt$$

avec comme contrainte le couple $(x(t), u(t))$ qui vérifie l'équation différentielle ordinaire (EDO) qui suit :

$$\begin{cases} \dot{x} = f(t, x, u), t \in [0, T] \\ x(t_0) = x_0 \end{cases}$$

3.1.1 Système dynamique :

Un système dynamique est la donnée d'un système et d'une loi décrivant l'évolution de ce système. C'est à dire c'est un système dont l'histoire passée est décrite à travers ses variables d'état et dont l'évolution future dépend aussi bien des entrées futures que de la valeur présente de ses variables d'état.

Un système dynamique simple est noté comme suit :

$$\begin{cases} \dot{x} = f(t, x(t)), t \in [t_0, T] \\ x(t_0) = x_0 \end{cases}$$

Dans ce système le vecteur $x^t = (x_1, \dots, x_n) \in \mathbb{R}^n$ est appelé variable d'état, x_0 l'état initial de cette variable. Le système évolue en fonction la variable réelle $t \in [t_0, T]$, la variable d'état $x(t)$ étant un vecteur de fonctions de t sera gouverné par des relations (souvent différentielles) appelées équations d'états $\dot{x} = f(t, x(t))$ avec f une fonction vectorielle de n composantes $f_i, i = 1, \dots, n$.

Remarque 3.1.

- Si les équations forment un système différentiel du second ordre, on peut se ramener à deux équations différentielles du premier ordre en introduisant une nouvelle variable d'état :

Pour $\ddot{x} = f(t, x(t))$ on pose :

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = f(t, x(t)) \end{cases}$$

- Dans le cas où f ne dépend pas explicitement de t , ($\dot{x}(t) = f(x)$), le système est dit autonome, sinon il est dit non-autonome.

Exemple : On a le système dynamique suivant :

$$\begin{cases} \dot{x} = Ax(t) + B, t \in [0, T] \\ x(0) = x_0 \end{cases}$$

$$\text{Où : } x(t) = \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix}, x_0 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, A = \begin{pmatrix} 1 & 2 \\ 4 & 5 \end{pmatrix}, B = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

C'est un système dynamique autonome car la matrice A et le vecteur B ne dépendent pas de la variable $t \in [0, T]$.

3.1.2 Système de contrôle :

Un système de contrôle est un système dynamique avec un paramètre appelé la variable de contrôle (commande) généralement noté $u(t) \in U$ avec $U \subset \mathbb{R}^m$ qui est une fonction localement intégrable définie sur $[t_0, T]$ qui nous permet d'agir sur le système de façon à atteindre une cible ou un objectif donné.

Le système ci-dessous est donc un système de contrôle :

$$\begin{cases} \dot{x} = f(t, x(t), u(t)), t \in [t_0, T] \\ x(t_0) = x_0 \end{cases} \quad (4)$$

Variable de contrôle :

La variable de contrôle $u(t) = (u_1(t), \dots, u_m(t)) \in U$, avec $U \subset \mathbb{R}^m$ l'ensemble des contrôles admissibles, nous permet d'agir sur le système dynamique et de déterminer la trajectoire à suivre pour qu'il passe d'un état vers l'autre. Il y a plusieurs type de contrôle qui peuvent être utiliser pour influencer un système dynamique comme ce qui suit :

1/ Contrôle borné :

Dans beaucoup de problème de contrôle, on peut minorer ou majorer les commandes $u_j(t)$, $j = 1, \dots, m$ par des constantes, $a_j \leq u_j \leq b_j$, on peut remplacer u_j par v_j en posant $u_j = \frac{1}{2}(a_j + b_j) + \frac{1}{2}(a_j - b_j)v_j$ et ainsi v_j est aussi intégrable et l'on a $-1 \leq v_j \leq 1$. Donc lorsque U est borné, il est toujours pratique de se ramener à des commandes entre -1 et 1 .

2/ Contrôle Bang-Bang :

Un contrôle $u(t) = (u_1(t), \dots, u_m(t)) \in U$ est appelé contrôle bang-bang, si pour chaque instant t et chaque indice $j = 1, \dots, m$ on a : $|u_j(t)| = 1$. En d'autres termes, une commande bang-bang est une commande qui bascule brusquement entre deux valeurs et qui possède au moins un instant de commutation.

3/ Contrôle continu par morceaux :

Cette classe de contrôle est la plus intéressante pour les application pratique de la théorie, bien que l'existence d'un contrôle optimal ne soit pas garantie en général.

Une fonction réelle $u(t)$, $t \in [t_0, T]$, est dite continue par morceaux, s'il existe une partition $t_0 \leq t_1 \leq \dots \leq t_N \leq t_{N+1} = T$ telle que $u(t)$ peut être considérée comme une fonction continue dans $[t_k, t_{k+1}]$ pour $k = 0, 1, \dots, N$.

Trajectoire :

Dans le système de contrôle (4) précédent, on appelle une trajectoire du système toute fonction régulière $t \in [t_0, T] \mapsto (x(t), u(t)) \in \mathbb{R}^n \times \mathbb{R}^m$ qui est vérifiée sur un intervalle $[t_0, T]$ de \mathbb{R} . Il s'agit donc, des solutions du système $x(t)$ dans $t \in [t_0, T]$ associé au contrôle u , on les note $x_u(t)$.

Dans la figure ci-dessous, on remarque l'existence de plusieurs trajectoires reliant x_0 à $x(T)$:

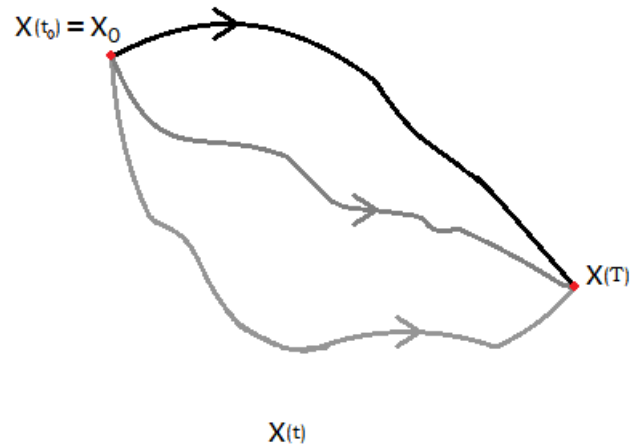


FIGURE 3.1 – Trajectoires reliant x_0 à $x(T)$

Stratégies de contrôle d'un système dynamique :

Ci dessous on discute les deux principal stratégie de contrôle d'un système dynamique :

Stratégie en boucle ouverte :

Dans ce cas la commande est envoyée en entrée sans dépendre de l'état de système. La figure ci-dessous schématise cette stratégie :

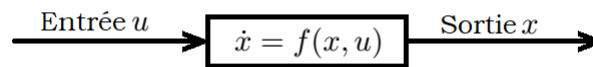


FIGURE 3.2 – Commande en boucle ouverte

Stratégie en boucle fermée :

Dans ce cas la commande est envoyée en entrée tout en prenant en compte l'état de système, c'est à dire le contrôle u dépend de t et de la variable d'état $x(t)$. La figure ci-dessous résume cette stratégie :

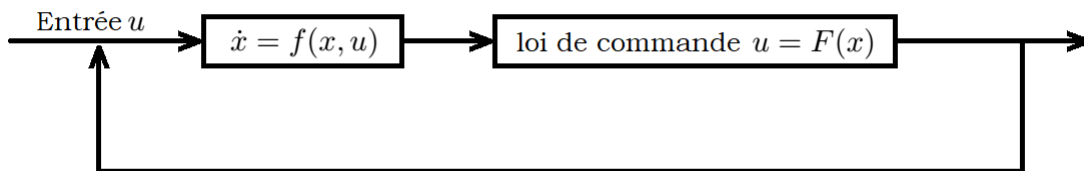


FIGURE 3.3 – Commande en boucle fermée

Problème de contrôle optimal :

Un problème de contrôle optimal est composé d'une fonctionnelle $J(u(t))$ qui représente le coût à minimiser, ou le gain à maximiser du problème, qu'on note :

$$J(u(t)) = g(x(T)) + \int_{t_0}^T f^0(t, x(t), u(t)) dt$$

avec $u(t) \in U$, $t \in [t_0, T]$, le contrôle du problème, en tenant compte d'un système de contrôle qui représente les contraintes du problème :

$$\begin{cases} \dot{x} = f(t, x(t), u(t)), t \in [t_0, T] \\ x(t_0) = x_0 \end{cases} \quad (4)$$

Pour résoudre ce problème de contrôle optimal, il faut tout d'abord vérifier s'il est contrôlable, en d'autres termes, si un contrôle $u(t) \in U$ existe tel que la trajectoire qui lui est associée atteigne la cible désignée. Ensuite, il faut chercher les contrôles $u(t) \in U$ qui réalisent des trajectoires optimales parmi toutes les trajectoires qui existent, c'est à dire, celles qui réalisent le minimum (ou le maximum) de la fonctionnelle du problème.

Les différents types de fonctionnelle d'un problème de contrôle optimal :

Un problème de contrôle optimal peut s'écrire de plusieurs façons différentes selon les fonctions $g : \mathbb{R}^n \rightarrow \mathbb{R}$ et $f^0 : [t_0, T] \times \mathbb{R}^n \times U \rightarrow \mathbb{R}$. Ci-dessous les problèmes les plus connus :

Problème de Lagrange :

Dans le problème de Lagrange, l'expression de la fonctionnelle $J(u(t))$, ne contient que l'intégrale de la fonction $f^0(t, x(t), u(t))$ comme ce qui suit :

$$J(u(t)) = \int_{t_0}^T f^0(t, x(t), u(t)) dt \longrightarrow \min$$
$$s.c \begin{cases} \dot{x}(t) = f(t, x(t), u(t)) \\ x(t_0) \in M_0, x(T) \in M_1 \\ u(t) \in U \subset \mathbb{R}^n, x(t) \in \mathbb{R}^n \end{cases}$$

Le problème de Mayer :

Dans le problème de Mayer, l'expression de la fonctionnelle $J(u(t))$, ne contient que la fonction $g(x(T))$ comme ce qui suit :

$$J(u(t)) = g(x(T)) \longrightarrow \min$$
$$s.c \begin{cases} \dot{x}(t) = f(t, x(t), u(t)) \\ x(t_0) \in M_0, x(T) \in M_1 \\ u(t) \in U \subset \mathbb{R}^n, x(t) \in \mathbb{R}^n \end{cases}$$

Le problème de Bolza(Mayer-Lagrange) :

Le problème de Bolza, regroupe le problème de Lagrange et le problème de Mayer. l'expression de la fonctionnelle $J(u(t))$ contient l'intégrale de la fonction $f^0(t, x(t), u(t))$ et la fonction $g(x(T))$ comme ce qui suit :

$$J(u(t)) = g(x(T)) + \int_{t_0}^T f^0(t, x(t), u(t))dt \longrightarrow \min$$
$$s.c \begin{cases} \dot{x}(t) = f(t, x(t), u(t)) \\ x(t_0) \in M_0, x(T) \in M_1 \\ u(t) \in U \subset \mathbb{R}^n, x(t) \in \mathbb{R}^n \end{cases}$$

Dans les formulations du problème de contrôle optimal précédentes, l'ensemble $M_0 \subset \mathbb{R}^n$ (respectivement $M_1 \subset \mathbb{R}^n$) représente l'ensemble des valeurs possibles de $x(t)$ lorsque $t = t_0$ (respectivement l'ensemble des valeurs possibles de $x(t)$ lorsque $t = T$).

Exemple : [23] (Wagon-fusée).

Soit un wagon poussé par des moteurs-fusées de chaque coté. On introduit les variables suivantes :

$q(t)$: position au temps t.

$v(t) = \dot{q}(t)$: vitesse au temps t.

$u(t)$: la poussée des moteurs-fusées.

Où $-1 \leq u(t) \leq 1$, son signe dépend du moteur-fusée qui est en marche.

On veut trouver comment faire marcher les moteurs-fusées, pour arriver à l'origine 0 avec une vitesse nulle ($v(t) = 0$) en un temps minimum. En supposant que la masse du véhicule $m = 1$, la loi du mouvement est :

$$\ddot{q}(t) = u(t).$$

On écrivant $x(t) = (q(t), v(t))^t$, on a :

$$\begin{cases} \dot{x} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} x(t) + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u(t), t \in [0, T] \\ x(t_0) = x_0 = (q_0, v_0)^t \end{cases}$$

Puisque le but est d'arriver au point (0,0) en un temps minimum, on pose :

$$J(u(t)) = - \int_0^T 1 dt = -T.$$

T = le premier temps pour lequel $q(T) = v(T) = 0$.

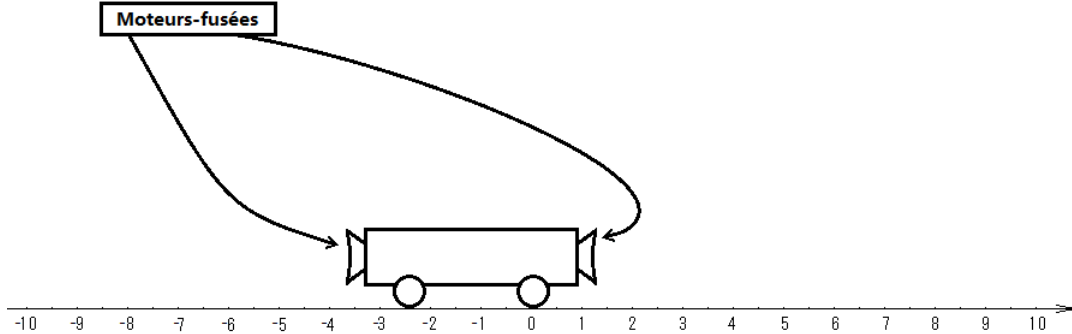


FIGURE 3.4 – Wagon poussé par des moteurs-fusées de chaque coté

3.1.3 Contrôlabilité :

Pour définir le concept de contrôlabilité pour un problème de contrôle optimal, on doit d'abord définir la notion d'ensemble accessible :

Définition 3.1. [24]

L'ensemble accessible en temps T pour le système de contrôle général, noté

$$Acc(x_0, T) = \{x_u(T) / u \in U\}$$

est l'ensemble des points accessibles partant de x_0 en temps T , c'est à dire, l'ensemble des extrémités du système.

Théorème 3.1. [24] :

Considérons le système de contrôle (4) :

$$\begin{cases} \dot{x} = f(t, x(t), u(t)), t \in [t_0, T] \\ x(t_0) = x_0 \end{cases} \quad (4)$$

où la fonction f est de classe C^1 sur \mathbb{R}^{1+n+m} et les contrôles $u \in U$ des fonctions mesurables à valeurs dans un compact $\Omega \subset \mathbb{R}^m$. On suppose que :

- $\exists b$ un réel positif tel que toute trajectoire associée est uniformément bornée par b sur $[t_0, T]$, i.e.

$$\exists b > 0 \mid \forall u \in U \forall t \in [t_0, T] \|x_u(t)\| \leq b.$$

- $\forall (t, x)$ l'ensemble des vecteurs vitesses $V(t, x) = \{f(t, x, u) \mid u \in \Omega\}$ est convexe.

Alors $Acc(x_0, t)$ est compact et varie continûment en t sur $[t_0, T]$.

Définition 3.2. [24]

Le système (4) est dit contrôlable (en temps quelconque) depuis x_0 si $\mathbb{R}^n = \bigcup_{T \geq 0} Acc(x_0, T)$.

Il est dit contrôlable en temps T si $\mathbb{R}^n = Acc(x_0, T)$. C'est à dire $\forall x_0, x(T) \in \mathbb{R}^n, \exists u(t)$ un contrôle tel que la trajectoire associée, $x_u(t)$, relie x_0 à $x(T)$ en temps T .

Proposition 5. [24]

Considérons le système (4) où $f(x_0, u_0) = 0$. Notons $A = \frac{\partial f}{\partial x}(x_0, u_0)$ et $B = \frac{\partial f}{\partial u}(x_0, u_0)$. On suppose que

$$rg(B|AB|\dots|A^{n-1}B) = n.$$

Alors le système est localement contrôlable en x_0 .

3.1.4 Principe du maximum de Pontryagin (P.M.P) :

Soit le problème de contrôle optimal suivant :

$$\begin{aligned} J(u(t)) &= g(x(T)) + \int_{t_0}^T f^0(t, x(t), u(t)) dt \longrightarrow \min \\ \text{s.c} \begin{cases} \dot{x} &= f(t, x(t), u(t)), t \in [t_0, T] \\ x(t_0) &= x_0 \end{cases} \end{aligned} \quad (5)$$

Où $u(t) \in U \subset \mathbb{R}^n, x(t) \in \mathbb{R}^n, g : \mathbb{R}^n \rightarrow \mathbb{R}$ et $f^0 : [t_0, T] \times \mathbb{R}^n \times U \rightarrow \mathbb{R}$.

Définition 3.3. (Hamiltonien)[24]

Le Hamiltonien du système (5) : $H : [t_0, T] \times \mathbb{R}^n \times (\mathbb{R}^n \setminus \{0\}) \times U \rightarrow \mathbb{R}$ est la fonction :

$$H(t, x, u, p, p_0) = \langle p, f(t, x, u) \rangle + p_0 f^0(t, x, u)$$

Avec $p(t)$ vecteur d'état adjoint, et p_0 variable duale du coût.

Souvent $p(t)$ est considéré comme un vecteur ligne, et on écrit :

$$H(t, x, u, p, p_0) = pf(t, x, u) + p_0 f^0(t, x, u).$$

Enoncé général :

Ce théorème est l'énoncé général du principe du maximum de Pontryagin.

Théorème 3.2. (Trélat [2005], Pontryagin [1987])[24]

On considère le système de contrôle dans \mathbb{R}^n :

$$\dot{x}(t) = f(t, x(t), u(t)). \quad (3.1)$$

Où $f : \mathbb{R}^+ \times \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$ est de classe C^1 et où les contrôles sont des applications mesurables et bornées à valeurs dans un compact $U \subset \mathbb{R}^m$. Soient M_0 et M_1 deux sous-ensembles de \mathbb{R}^n .

On définit le coût

$$J(t_f, u) = g(t_f, x(t_f)) + \int_{t_0}^{t_f} f_0(t, x(t), u(t)) dt. \quad (3.2)$$

Où $f_0 : \mathbb{R}^+ \times \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}$ et $g : \mathbb{R}^+ \times \mathbb{R}^n \rightarrow \mathbb{R}$ sont de classe C^1 et $x(\cdot)$ est la trajectoire solution de (3.1) associée au contrôle u .

On considère le problème de contrôle suivant : déterminer une trajectoire reliant M_0 à M_1 en minimisant le coût. Le temps final peut être fixé ou non.

Si le contrôle $u \in U$ associé à la trajectoire $x(\cdot)$ est optimal sur $[t_0, t_f]$, alors il existe une application $p(\cdot) : [t_0, t_f] \rightarrow \mathbb{R}^n$ absolument continue et un réel $p_0 \leq 0$, tels que le couple $(p(\cdot), p_0)$ est non trivial et tel que pour presque tout $t \in [t_0, t_f]$

$$\dot{x} = \frac{\partial H}{\partial p}(t, x(t), p(t), p_0, u(t)),$$

$$\dot{p} = -\frac{\partial H}{\partial x}(t, x(t), p(t), p_0, u(t)),$$

où

$$H(t, x(t), p(t), p_0, u(t)) = p_0 f_0(t, x(t), u(t)) + p(t) f(y, x(t), u(t))$$

est le hamiltonien du système vérifiant la condition de maximisation sur $[t_0, t_f]$:

$$H(t, x(t), p(t), p_0, u(t)) = \max_{v \in U} H(t, x(t), p(t), p_0, v).$$

Si de plus M_0 et M_1 (où juste l'un des deux ensembles) sont des variétés de \mathbb{R}^n ayant des espaces tangents en $x(0) \in M_0$ et $x(t_f) \in M_1$, alors le vecteur adjoint peut être construit de manière à vérifier les conditions de transversalité aux deux extrémités (ou juste l'une des deux).

$$p(t_0) \perp (t_f)_{x(t_0)} M_0. \quad (3.3)$$

Et

$$p(t_f) - p_0 \frac{\partial g}{\partial x}(t_f, x(t_f)) \perp (t_f)_{x(t_f)} M_1. \quad (3.4)$$

Ces conditions sont appelées conditions de transversalité sur le vecteur adjoint.

3.2 Problème du Contrôle optimale bi niveaux :

On étudie ci-dessus un problème du contrôle optimale bi niveaux en s'appuyant sur le principe (P.M.P). Le problème du contrôle optimale du second niveau (3.5) est écrit comme suit :

$$P_2 : J_2(z, u(t)) = G(x(T)) + \int_{t_0}^T g^0(t, x(t), z, u(t))dt \longrightarrow \min$$

$$s.c \begin{cases} \dot{x}(t) = g(t, x(t), z, u(t)) \\ x(t_0) = x_0, x(T) \in M_1 \\ u(t) \in U \subset \mathbb{R}^n, x(t) \in \mathbb{R}^n \end{cases} \quad (3.5)$$

Où $t \in [t_0, T]$, $x(t) \in \mathbb{R}^n$ variable d'état, $z \in Z$ avec $Z \subset \mathbb{R}^m$ la décision du leader et $u(t) \in U$ la fonction de contrôle du suiveur. Dans ce problème on suppose que z est une décision choisie par le leader donnée et $u(t)$ fonction de contrôle du suiveur en réponse à z .

Le problème du contrôle optimale du premier niveau (3.6) compte à lui, ayant des contraintes dépendant du problème précédant, est écrit :

$$P_1 : J_1(z, u_z(t)) = F(x(T)) + \int_{t_0}^T f^0(t, x_z(t), z, u_z(t))dt \longrightarrow \min$$

$$s.c \begin{cases} z \in Z \\ \text{Tout vecteur optimale } (x_z(t), u_z(t)) \text{ du problème } P_2 \end{cases} \quad (3.6)$$

Le problème peut assit être formulé comme suit :

$$J(z, u_z(t)) = F(x(T)) + \int_{t_0}^T f^0(t, x_z(t), z, u_z(t))dt \longrightarrow \min$$

$$s.c \begin{cases} z \in Z \\ (x_z(t), u_z(t)) \in \begin{cases} J(z, u(t)) = G(x(T)) + \int_{t_0}^T g^0(t, x(t), z, u(t))dt \longrightarrow \min \\ s.c \begin{cases} \dot{x}(t) = g(t, x(t), z, u(t)) \\ x(t_0) = x_0, x(T) \in M_1 \\ u(t) \in U \subset \mathbb{R}^n, x(t) \in \mathbb{R}^n \end{cases} \end{cases} \end{cases}$$

Étude du problème :

En utilisant la fonction valeur V du niveau inférieur, le problème de contrôle optimal bi niveaux (3.6) peut être formulé comme un problème de contrôle optimal à un seul niveau suivant :

$$\begin{aligned}
 J_1(z, u_z(t)) &= F(x(T)) + \int_{t_0}^T f^0(t, x_z(t), z, u_z(t)) dt \longrightarrow \min \\
 \text{s.c } \left\{ \begin{array}{l}
 \dot{x}(t) = g(t, x(t), z, u(t)) \\
 x(t_0) = x_0, x(T) \in M_1 \\
 u(t) \in U \subset \mathbb{R}^n, x(t) \in \mathbb{R}^n \\
 V(n, m) \geq G(x(T)) + \int_{t_0}^T g^0(t, x(t), z, u(t)) dt
 \end{array} \right. \quad (3.7)
 \end{aligned}$$

Définition 3.4.

- i- Un arc est une fonction absolument continue.
- ii- Une solution du problème (3.5) est une paire réalisable qui minimise la valeur de la fonction coût $J_2(z, u(t))$ sur l'ensemble de toutes les paires réalisables.
- iii- On appelle stratégie réalisable de (3.6), un couple (z, u_z) où $z \in Z$ et u_z un contrôle optimal de (3.5). La stratégie (z^*, u_z^*) est optimale pour (3.6) si elle minimise la valeur de la fonction coût $J_1(z, u(t))$ sur l'ensemble des stratégies réalisables de (3.6).

Pour ce qui suit, On pose les hypothèses suivantes :

A_1 $Z \subset \mathbb{R}^m$ et M_1 sont fermés.

A_2 $U \in \mathbb{R}^n$ est une multifonction dont l'ensemble des valeurs est non vide. Le graphe de U , c'est à dire l'ensemble $\{(s, r) : s \in [t_0, T], r \in U(s)\}$, qu'on notera par GrU , est $L \times B$ -mesurable, où $L \times B$ est une σ -algèbre de sous-ensembles de $[t_0, T] \times \mathbb{R}^n$ engendrée par $M \times N$ où M est une partie mesurable au sens de Lebesgue de $[t_0, T]$ et N est une partie borélienne de \mathbb{R}^n . vide et compact.

A_3 Il existe une fonction à valeurs réelles k , $L \times B$ -mesurable définie sur GrU telle que : pour tout couple $(t, u) \in GrU$, les fonctions $g(t, \cdot, \cdot, u)$, $F(t, \cdot, \cdot, u)$, $G(t, \cdot, \cdot, u)$ sont localement lipschitziennes de rang $k(t)$, où $k(t)$ est une fonction intégrable. Pour tout couple $(x, z) \in \mathbb{R}^n \times \mathbb{R}^m$, les fonctions suivantes : $g(\cdot, x, y, \cdot) : [t_0, T] \times \mathbb{R}^m \rightarrow \mathbb{R}^n$, $F(\cdot, x, y, \cdot) : [t_0, T] \times \mathbb{R}^m \rightarrow \mathbb{R}$, $G(\cdot, x, y, \cdot) : [t_0, T] \times \mathbb{R}^m \rightarrow \mathbb{R}$ sont $L \times B$ -mesurables.

A_4 es fonctions $f^0, g^0 : \mathbb{R}^m \rightarrow \mathbb{R}$ sont continues, localement lipschitziennes.

A_5 Pour tout $Z \subset \mathbb{R}^m$, le problème (3.5) admet une paire réalisable.

On définit l'hamiltonien du problème (3.5) par la fonction suivante :

$$H_2(t, x, z, u, p_2, \lambda) = p_2 g(t, x, z, u) - \lambda G(t, x, z, u).$$

En appliquant le principe de maximum, on obtient :

$$H_2(t, x, z, u^0, p_2, \lambda) = \sup_{u \in U(t)} H_2(t, x, z, u, p_2, \lambda).$$

Condition nécessaire d'optimalité

Définissons le pseudo-hamiltonien du problème (3.17) comme

$$H_1(t, x, z, p_1, \lambda, r) = p_1(t)g(t, x, z, u) - rG(t, x, z, u) - \lambda F(t, x, z, u)$$

avec $t \in [t_0, T]$, $x \in \mathbb{R}^n$, $p_1 \in \mathbb{R}^m$, $z \in Z$, $r \in \mathbb{R}$.

Théorème 3.3. [35]

On suppose que :

- i- Les hypothèses A_1 à A_4 sont satisfaites.
- ii- $(z, u(t))$ une stratégie optimale du problème (3.6).
- iii- $\hat{\partial}g^0(x(T)) \cap \hat{N}_{M_1}(x(T)) = 0$, et $x(t)$ la trajectoire correspondante.
- iv- La fonction valeur V du problème du niveau inférieur (3.5) est lipschitzienne et continue.

Alors il existe $\lambda \geq 0$, $r \geq 0$ et des arcs p_1, η tels que :

$$\left\{ \begin{array}{l} (\dot{p}_1(t), \dot{\eta}(t)) = \partial_{x,z} H_1(t, x(t), z, p_1(t), u(t), \lambda, r) \text{ p.p., (1e)} \\ \max_{u \in U(t)} H_1(t, x(t), z, p_1(t), u, \lambda, r) = H_1(t, x(t), z, p_1(t), u(t), \lambda, r) \text{ p.p., (2e)} \\ \eta(t_0) = 0, (3e) \\ -\dot{p}_1(T) \in \lambda \hat{\partial}f^0(x(T)) + r \hat{\partial}g^0(x(T)) + \hat{N}_{M_1}(x(T)), (4e) \\ \eta(T) \in r \partial V(z), (5e) \\ \|p_1\|_\infty + \|\eta\|_\infty + \lambda + r > 0, (6e) \end{array} \right. \quad (3.8)$$

avec $\|\eta\|_\infty = \max_{t_0 \leq t \leq T} |\eta|$.

3.2.1 Analyse d'un exemple :

Le problème de la régulation de la pêche est un problème du contrôle optimale bi niveaux qui a été formulé et traité par *Clarke et Murno* [38]. Ce problème modélise la difficulté rencontrée par les états côtiers lorsqu'ils imposent des conditions aux flottes de pêche actives dans leurs eaux, ceci dans le but de préserver leurs ressources halieutiques pour le long terme et de maximiser leur bénéfices au travers de cette initiative.

Dans ce problème bi niveaux les flottes de pêche cherche à maximiser leurs prises ainsi leurs profit tout en tenant compte des contraintes imposées par les états côtiers il s'agit bien du rôle du suiveur dans ce problème à deux niveaux. Tandis que les états côtiers cherche à maximiser leurs bénéfices comme exprimé ci-dessus, ils jouent le rôle de leader dans le problème.

Pour étudier le problème du suiveur, l'une des contraintes auxquelles il doit d'abord faire face, est la préservation de la population de poissons dans sa zone d'activité. Le système régissant cette dernière est noté comme suit :

$$\dot{x}(t) = F(x) - qE(t)x(t),$$

Il s'agit de l'évolution de la population de poisson en un temps t , avec $x(t)$ la taille de cette population correspondant au temps t , $F(x)$ le taux naturel de sa croissance, $E(t)$ la quantité de la pêche en temps t (ce qui s'agit du contrôle dans ce problème) et q est une constante positive appelé coefficient de capture.

Dans la suite il est supposé que $F(x) > 0$ c'est à dire la croissance naturelle de la population est non nul, pour $0 < x < \bar{x}$, $F(0) = F(\bar{x}) = 0$ où $F(0) = 0$ est évident car ça signifie qu'il n'y a pas population de poissons, et \bar{x} est la charge utile de la ressource. Il admet aussi $F''(x) < 0$ pour tout $x > 0$.

On suppose que le contrôle $E(t)$ est borné tel que $0 \leq E(t) \leq E_{max}$, où E_{max} sa borne supérieure.

Supposons que l'état côtier impose la condition que la population de poissons ne devra pas descendre au-dessous du seuil $\tilde{x} \geq 0$ au temps terminal T . Il est établi aussi une stratégie (ensemble de taxes : $n :=$ taxe sur la récolte et $m :=$ taxe sur l'effort) pour laquelle la flotte de pêche doit répondre par une action $E(t)$.

On obtient alors le problème que le suiveur vise à résoudre afin de maximiser son gain :

$$J_2(E(t)) = \int_0^T e^{-\delta t} [(p_0 - n)qx(t) - (c_0 + m)]E(t)dt \longrightarrow \max_E$$

$$s.c \begin{cases} \dot{x}(t) = F(x(t)) - qE(t)x(t) \\ x(0) = x_0, x(T) \geq \tilde{x}, \\ E(t) \in [0, E_{max}], \end{cases} \quad (3.9)$$

où p_0 représente le prix unitaire d'une prise de poissons, c_0 le coût des charges, et $\delta > 0$ le taux d'actualisation.

En se qui concerne le problème du leader celui-ci dépend du problème (3.9) car les bénéfices maximum envisageable pour le leader (état côtier) sont contraint a ceux du suiveur, ce qui fait du problème un problème bi niveaux optimiste.

On note le problème comme suit :

$$J_1(E(t)) = \int_0^T e^{-\delta t} (nqx(t) + m)E(t)dt \longrightarrow \max_{(n,m)} \quad (3.10)$$

s.c { Tout vecteur optimale $(n, m, x(t), E(t))$ du problème (3.9)

On peut aussi, dans le but de facilite les calculs, utilisant une fonction valeur. On définit la fonction valeur par $V(n, m) := \sup(P_{3.9})$, c'est une fonction valeur optimale du problème (3.9).

Sachant que L est la rémunération alternative des autres états côtiers, la participation d'une flotte de pêche est imposée uniquement dans le cas où $V(n, m) \geq L$.

Avec la fonction $V(n, m)$ on obtient le problème d'optimisation dynamique bi-niveaux suivant :

$$J_1(E(t)) = \int_0^T e^{-\delta t} (nqx(t) + m)E(t)dt \longrightarrow \max_{(n,m)}$$

$$s.c \left\{ \begin{array}{l} \dot{x}(t) = F(x(t)) - qE(t)x(t) \\ x(0) = x_0, x(T) \geq \tilde{x}, \\ E(t) \in [0, E_{max}] p.p, \\ V(n, m) \leq \int_0^T e^{-\delta t} [(p_0 - n)qx(t) - (c_0 + m)]E(t)dt \\ V(n, m) \geq L \end{array} \right. \quad (3.11)$$

On commence par étudier le problème (3.9) en faisant appel au principe du maximum de Pontryagin (P.M.P).

Il s'agit d'un problème de contrôle optimal linéaire, donc son hamiltonien est égal à :

$$H_2(t, x, n, m, E, p_2) = p_2(t)[F(x(t)) - qEx(t)] + P_O[e^{-\delta t}[(p_0 - n)qx(t) - (c_0 + m)]E],$$

avec $p_2(T) \geq 0$.

Où $p_2(t)$ le vecteur d'état, P_O variable dual du coût. Puisque il s'agit d'un problème de maximisation alors $P_O = 1$.

Selon le principe du maximum, on aura :

$$\max_{E(t) \in [0, E_{max}]} H_2(t, x, n, m, E, p_2) = H_2(t, x, n, m, E(t), p_2),$$

avec $H_2(t, x, n, m, E(t), p_2) = p_2(t)[F(x(t)) - qE(t)x(t)] + e^{-\delta t}[(p_0 - n)qx(t) - (c_0 + m)]E(t)$, $p_2(T) \geq 0$.

La condition nécessaire pour qu'un couple (x, E) soit solution de (3.9) est l'existence d'un arc p_2 tel que :

$$\dot{p}_2(t) = -\frac{\partial H_2}{\partial x} = -\{p_2(t)[F'(x) - qE] + e^{-\delta t}((p_0 - n)qE)\},$$

d'où

$$-\dot{p}_2(t) = p_2(t)[F'(x) - qE] + e^{-\delta t}(p_0 - n)qE \quad (3.12)$$

Comme on cherche une valeur de $E(t)$ qui maximise l'hamiltonien H_2 , $E(t)$ doit être ou bien un contrôle singulier, ou bien il prend la valeur de l'une de ces borne ($E(t) = 0$ ou $E(t) = E_{max}$). Le contrôle singulier apparaît quand le coefficient de E dans l'hamiltonien est égal à zéro, c'est à dire :

$$H_2(t, x, n, m, E(t), p_2) = E(t)\{-p_2(t)qx(t) + e^{-\delta t}[(p_0 - n)qx(t) - (c_0 + m)]\} + p_2(t)F(x(t)),$$

alors le coefficient de E : $-p_2(t)qx(t) + e^{-\delta t}[(p_0 - n)qx(t) - (c_0 + m)] = 0$, ce qui implique :

$$p_2(t) = e^{-\delta t}\left[(p_0 - n) - \frac{c_0 + m}{qx}\right]. \quad (3.13)$$

En dérivant par rapport à t on obtient :

$$\dot{p}_2(t) = e^{-\delta t}\left[-\delta\left[(p_0 - n) - \frac{c_0 + m}{qx}\right] + \frac{c_0 + m}{qx^2} \frac{dx}{dt}\right]. \quad (3.14)$$

On prend l'équation 3.12 :

$$\dot{p}_2(t) = -p_2[F'(x) - qE] - e^{-\delta t}[(p_0 - n)qE]$$

et on remplace p_2 par ça valeur 3.13 dans l'équation, et on obtient :

$$\dot{p}_2(t) = -e^{-\delta t}\left\{\left[(p_0 - n) - \frac{c_0 + m}{qx}\right][F'(x) - qE] + (p_0 - n)qE\right\},$$

On remplaçant $\dot{p}_2(t)$ dans l'expression précédente avec ca valeur dans 3.14, le contrôle E disparaît et on obtient l'égalité suivante :

$$\delta = F'(x) + \frac{F(x)(c_0 + m)/qx^2}{p_0 - n - (c_0 + m)/qx} \quad (3.15)$$

Pour n et m fixés, cette équation admet une solution unique x_* représentant la biomasse optimale (la masse totale de poissons peuplant le milieu). La trajectoire optimale est celle qui va le plus vite vers la biomasse optimale x_* .

On passe maintenant a l'étude du problème (3.11) :
 Le hamiltonien de ce problème est donné par :

$$H_1(t, x, n, m, E, p_1, \lambda, r) = p_1(t)[F(x(t)) - qEx(t)] + e^{-\delta t}[r[(p_0 - n)qx(t) - (c_0 + m)] + \lambda(nqx(t) + m)]E$$

On vérifie facilement que les conditions du théorème 3.3 sont satisfaites. On note que le problème du suiveur a une solution unique. En vertu de ce théorème on a : si (n, m, x, E) est une solution optimale du problème (3.11), alors il existe des arcs $p_1, p_2, \eta_1, \eta_2, q_1, q_2$ et des scalaires $\lambda \geq 0, r \geq 0, 0 \leq \hat{r} \leq r$ tels que :

$$\left\{ \begin{array}{l}
\dot{p}_1(t) = -\frac{\partial H_1}{\partial x} = -\{p_1(t)[F'(x) - qE] + e^{-\delta t}(r(p_0 - n) + \lambda n)qE\}, \quad (1) \\
\dot{\eta}_1 = \frac{\partial H_1}{\partial n} = (r - \lambda)e^{-\delta t}qx E, \quad (2) \\
\dot{\eta}_2 = \frac{\partial H_1}{\partial m} = (r - \lambda)e^{-\delta t}E, \quad (3) \\
\max_{E \in [0, Emax]} H_1(t, x, n, m, E, p_1, \lambda, r) = H_1(t, x, n, m, E(t), p_1, \lambda, r), \quad (4) \\
(\eta_1, \eta_2)(0) = (0, 0), \quad (5) \\
p_1(T) \geq 0, \quad (6) \\
(\eta_1, \eta_2)(T) = \hat{r}\lambda q(0), \quad (7) \\
-\dot{p}_2 = \frac{\partial H_2}{\partial x} = p_2[F'(x) - qE] + e^{-\delta t}(p_0 - n)qE, \quad (8) \\
\dot{q}_1 = \frac{\partial H_2}{\partial n} = e^{-\delta t}qx E, \quad (9) \\
\dot{q}_2 = \frac{\partial H_2}{\partial m} = e^{-\delta t}E, \quad (10) \\
\max_{E(t) \in [0, Emax]} H_2(t, x, n, m, E, p_2) = H(t, x, n, m, E(t), p_2), \quad (11) \\
(q_1, q_2)(0) = (0, 0), \quad (12) \\
p_2(T) \geq 0, \quad (13) \\
\|p_1\|_\infty + \|\eta\|_\infty + \lambda + r > 0., \quad (14)
\end{array} \right. \quad (3.16)$$

Sans perte de généralité, posons $\lambda = 1$. Des résultats (1) et (4) précédents, de la même façon que pour 3.15, il apparaît que l'état stable (n, m, x_*) du problème (3.11) est une solution de l'équation suivante :

$$\delta = F'(x_*) + \frac{F(x_*)(r(c_0 + m) - m)/qx_*^2}{r(p_0 - n) + n - (r(c_0 + m) - m)/qx_*}. \quad (3.17)$$

Puisque (n, m, x, E) est une solution optimale du problème (3.11), (x, E) est alors la solution optimale du problème (3.9) et x_* est la biomasse optimal associée avec (n, m) est définie par l'équation 3.17. En combinant les équations 3.17 et 3.15 on aura :

$$\begin{aligned}n &= \rho p_0, \\m &= -\rho c_0,\end{aligned}$$

où ρ est une constante qui provient de la résolution du système des deux équations 3.17 et 3.15 aux deux inconnues n et m . *Clarke* et *Murno* ont montré que pour la flotte de pêche, le couple taxe-charge (n, m) optimal doit être atteint pour $V(n, m) = L$, ce qui donne $\rho = (V_0 - L)/V_0$, où V_0 est le revenu global net pour la flotte,

$$V_0 = \max \left(\int_0^{T_1} e^{-\delta t} (p_0 q x(t) - c_0) E(t) dt, \right.$$

et que la politique de pêche optimale $E(t)$ permettra de maximiser le revenu global net pour les flottes de pêche. La condition nécessaire d'optimalité ci-dessus est satisfaite pour $\lambda = 1$, $r = 1$ et $\hat{r} = 0$.

Conclusion :

Dans ce travail on a étudié les problèmes de programmation bi-niveaux, ceux-ci visent à modéliser des problèmes de structure hiérarchique avec deux niveaux, le leader cherche à maximiser ses bénéfices tout en prenant compte des réactions du suiveur. Ce dernier cherche à maximiser son propre gain en tenant compte du niveau supérieur. Un tel problème peut, soit être résolu dans le cas optimiste avec la coopération des deux niveaux, ou au contraire, le leader doit être pessimiste et envisager la non coopération du suiveur.

Le problème de contrôle optimal bi-niveaux permet d'étudier une variété de problèmes, en tenant compte de variables difficiles à contrôler dans une formulation à un niveau. Il est appliqué dans plusieurs domaines, comme par exemples le domaine de l'énergie, où le leader a besoin de gérer la production d'énergie tandis que le suiveur cherche à la distribuer, et le domaine de la gestion des ressources naturelles, où le leader cherche à préserver ces ressources, tandis que le suiveur cherche à maximiser ses bénéfices.

Dans un tel problème, on cherche à trouver un contrôle optimal qui permet au suiveur d'atteindre son objectif dans les conditions souhaitées que ce soit en un temps optimal, en respectant certaines conditions, ou en évitant des situations désagréables. Cela tout en prenant en compte un niveau supérieur qui lui aussi cherche à prendre l'initiative optimale la moins exposée aux risques potentiels.

Résumé :

Dans ce mémoire nous étudions les problèmes de programmation bi-niveaux et leur application au contrôle optimal. Un problème de programmation bi-niveaux est un problème d'optimisation dont lequel un second problème d'optimisation est inclus dans ses contraintes. C'est un problème mathématique hiérarchique représentant deux niveaux de décision, le premier est appelé leader, qui représente l'objectif du décideur avec le plus haut niveau de décision. Le second est le suiveur qui représente l'objectif du décideur qui dispose d'un niveau de décision inférieur au précédent, c'est celui qui est inclus dans les contraintes du premier problème. D'abord, nous introduisons les notions de programmation mathématique de base dont on a besoin. Ensuite, nous abordons les problèmes de programmation bi-niveaux selon leurs différentes caractéristiques, leurs formulations, et les approches dans leurs traitements (optimiste et pessimiste). Puis, nous nous concentrons sur le cas particulier des problèmes bi-niveaux linéaires, et enfin nous parlons des conditions d'optimalité et des méthodes de résolution d'un problème de programmation bi-niveaux. Finalement, on rappelle les concepts clés du contrôle optimal (contrôlabilité, stabilité), pour formuler le problème de contrôle optimal bi-niveaux. L'approche de résolution utilisée consiste à transformer ce problème à deux niveaux en un problème à un niveau en utilisant la fonction valeur, ce qui permet d'obtenir des conditions nécessaires d'optimalité. Nous illustrons ces concepts par un exemple concret appliqué à une problématique combinant enjeux économiques et environnementaux.

Mots clés :

Contrôle optimal, Contrôle optimal bi-niveaux, Problème de programmation bi-niveaux, Fonction valeur.

Bibliographie

- [1] Bosco Etoa Etoa, Jean. Optimisation hiérarchique : théorie, algorithmes et application. Paris : Publibook, 2007. 330 p.
- [2] B. Colson, P. Marcotte, and G. Savard. Bilevel programming : A survey. 4OR A Quarterly Journal of Operations Research, 2007.
- [3] Stephan Dempe. Chapter 1 Bilevel Programming - A Survey. Technical University Bergakademie Freiberg, Germany.
- [4] M. Minoux. Programmation mathématique, théorie et algorithmes. Bordas et C.N.E.T-E.N.S.T, Paris 1983.
- [5] Stephen Boyd (Department of Electrical Engineering, Stanford University), Lieven Vandenbergh (Electrical Engineering Department, University of California). Convex Optimization. Cambridge university press, Los Angeles.
- [6] Dempe, Stephan. Nonconvex optimisation and its applications : Foundations of Bilevel Programming. New York, Boston, Dordrecht, London, Moscow : Kluwer Academic Publishers, 2002. 309 p.
- [7] Pavel Popela. Nonlinear Programming. Czech Republic. January 13, 2003.
- [8] Georg Still. Lectures on Parametric Optimization : An Introduction. University of Twente, The Netherlands. March 29, 2018.
- [9] J. Royer (Université Toulouse 3). Calcul différentiel et intégral - L2 Parcours Spécial - S3. 2015-2016.
- [10] Cerulli, Martina. Bilevel optimization and applications. Operations Research [math.OC]. Institut Polytechnique de Paris, 2021. 156 p.
- [11] Harrache, Fazia. Application de la Programmation bi-niveaux au Problème de Contrôle Optimal, mémoire de magister en mathématiques, option : Recherche Opérationnelle et Optimisation. Tizi-Ouzou, 2013. 96 p.
- [12] Bouibed, Karima. Étude des concepts de solution dans les problèmes bi-niveaux multi-objectifs, thèse de doctorat Spécialité : Mathématiques, Option : Recherche Opérationnelle et Optimisation. Tizi-Ouzou, 2016. 110 p.
- [13] Anzi, Aicha. Résolution d'un problème de programmation bi-niveaux linéaire par la méthode D.C, Mémoire de Magister en Mathématiques Appliquées : Modélisation Mathématique et Techniques de Décision. Béjaia, 2009. 90 p.

- [14] ZEGANE, Koceila. Programmation bi-niveaux stochastique pour la tarification dans l'aviation civile, mémoire en vue de l'obtention du diplôme de Master en Recherche Opérationnelle, Option : Aide à la décision. Tizi-Ouzou, 2019. 84 p.
- [15] O. Ben-Ayed and C.E. Blair. Computational difficulties of bilevel linear programming. *Operations Research*, (38) : 556–560, 1990.
- [16] Christoph Buchheim, Fakultät für Mathematik. Bilevel linear optimization belongs to NP and admits polynomial-size KKT-based reformulations. Technische Universität Dortmund, Germany. 2023.
- [17] Bard, J.F. An efficient point algorithm for a linear two-stage optimisation problem, *Operations Research*. 1983.
- [18] Bard, J.F. Convex two-level optimisation. *Mathematical Programming*. 1988.
- [19] Falk, J.E. A linear min-max problem, *Mathematical Programming*. 1973.
- [20] Abdelmalek Aboussoror, Samir Adly. A Fenchel-Lagrange Duality Approach for a Bilevel Programming Problem with Extremal-Value Function. *Journal of Optimization Theory and Applications*, Springer Verlag, 2011, 149 (2), pp.254-268. 10.1007/s10957-011-9831-5. hal-00683042
- [21] Joydeep Dutta, Tanushree Pandit. Algorithms for Simple Bilevel Programming (S. Dempe, A. Zemkoho (eds.), *Bilevel Optimization, Springer Optimization and Its Applications*). Indian Institute of Technology Kanpur, Kanpur, India. 2020.
- [22] Herminia I. Calvete, Carmen Galé. Algorithms for Linear Bilevel Optimization (S. Dempe, A. Zemkoho (eds.), *Bilevel Optimization, Springer Optimization and Its Applications*). Statistical Methods Department, IUMA, University of Zaragoza, Zaragoza, Spain. 2020.
- [23] Lawrence C. Evans. *An Introduction to Mathematical Optimal Control Theory*. Department of Mathematics, University of California, Berkeley. Printemps, 2024.
- [24] Emmanuel Trélat. *Contrôle optimal : théorie et applications*. Université Pierre et Marie Curie (Paris 6) et Institut Universitaire de France, Laboratoire Jacques-Louis Lions, FRANCE. 2005.
- [25] Pierre Cardaliaguet. *Optimisation et programmation dynamique*, Master mention Mathématiques et Applications, 1^{ère} année Université Paris Dauphine. 16 décembre 2019.
- [26] Amirouche Selma, Ameer Bahia. *Contrôle optimal et application*, Filière : Mathématiques, Spécialité : Recherche opérationnelle. Université Mouloud Mammeri de Tizi Ouzou, Faculté des Sciences, Département de Mathématiques, Laboratoire LMPA, Mémoire de Master. Septembre 2022.

- [27] Leknouché Houda, Belbedroune Fadia. Méthode de support pour la résolution d'un problème de contrôle optimal linéaire-quadratique, Mémoire préparé En vue de l'obtention du diplôme de Master en : Mathématiques, Spécialité : Mathématiques appliquées. Institut des sciences et de la technologie, Département de Mathématiques et Informatique. 2021/2022.
- [28] Mahtar Hassiba, Bourahla Leyla. étude analytique et numérique de quelques problèmes de contrôle optimal, Memoire de master, Recherche opérationnelle. Université Saad Dahleb de Blida, Faculté des Sciences, Département de Mathématiques. 18 juillet 2022.
- [29] Fernando Lobo Pereira, João Borges de Sousa. Introduction to Optimal Control. Université de Porto, 4200-465 Porto, Portugal, C4C Autumn School. 5-8 October 2009.
- [30] Dr M.LAZREG. Contrôle Analogique : Cours destiné aux étudiants de troisième année licence de Génie Industriel. République Algérienne Démocratique et Populaire, Ministère de L'enseignement Supérieur et de la recherche Scientifique, Institut de Maintenance et Sécurité Industrielle IMSI. 2021/2022.
- [31] Sylvie Delabrière, Yves Raynaud, Stanislaw Szarek. Analyse Convexe, Cours M1 (4MA057). Sorbonne Université. 2019-2020.
- [32] S. P. Sethi. Optimal control theory : applications to management sciences and economics, Third edition. Springer Nature, Switzerland. 2019.
- [33] Johanne Cohen. La NP-complétude, PRISM/CNRS, Versailles, France.
- [34] Michel Bierlaire, Optimisation linéaire, EPFL - Laboratoire Transport et Mobilité - ENAC.
- [35] J. J. Ye. *Necessary conditions for bilevel dynamic optimization problems*. Department of Mathematics and Statistics, University of Victoria, Canada. 1993.
- [36] J. J. Ye. *Optimal Strategies for Bilevel Dynamic Problems*. Department of Mathematics and Statistics, University of Victoria, Canada. 1993.
- [37] Pierre Fraigniaud. Notes de cours, Algorithmique avancée, Ecole Centrale, CNRS et Université Paris Diderot. 17 octobre 2017.
- [38] F.H. Clarke et G.R. Murno, Coastal states, distant water fishing nations and extended jurisdiction : a principal-agent analysis, Natural Resource Modelling. 1987.