

Université Sétif 1 Ferhat Abbas
Faculté des Sciences
Département de Mathématiques



THÈSE

présentée à la Faculté des Sciences
Département de Mathématiques
pour l'obtention du diplôme de

Doctorat 3^{ème} cycle LMD

Option : Optimisation et Contrôle

Par

Melle. CHENOUF Chahinaz

Thème :

**AMELIORATION DE CERTAINS ALGORITHMES DE
RECOUVREMENT D'OPTIMISATION GLOBALE HÖLDERIENNE**

Thèse soutenue le 13 Janvier 2024 devant le jury composé de :

Président	Mr. Lakhdar CHITER	Prof. Université Sétif 1 Ferhat Abbas
Rapporteur	Mr. Mohamed RAHAL	MCA Université Sétif 1 Ferhat Abbas
Examineurs	Mr. Mohamed ZERGUINE	Prof. Université Batna 2 Mustapha Ben Boulaïd
	Mr. Hamza GUEBBAI	Prof. Université Guelma 8 Mai 1945
	Mr. Taieb HAMAIZIA	Prof. Université Oum El-Bouaghi Larbi Ben M'hidi
Invité	Mr. Salim MESBAHI	Prof. Université Sétif 1 Ferhat Abbas

2023-2024

Remerciements

Avant tout je remercie Allah tout puissant de m'avoir donné le courage et la patience pour terminer ce travail.

*Je remercie particulièrement, mon encadreur **Mohamed Rahal**, Maître de Conférences A à l'université Ferhat Abbas Sétif 1, pour m'avoir encadré et orienté durant mon travail.*

*Je remercie vivement Monsieur **Lakhdar CHITER**, Professeur à l'université Ferhat Abbas Sétif 1, pour l'honneur qu'il me fait en acceptant de présider le Jury de cette thèse.*

*J'adresse mes vifs remerciements à Monsieur **Mohamed ZERGUINE**, Professeur à l'université Mustapha Ben Boulaïd. Batna 2, Monsieur **Hamza GUEBAI**, Professeur à l'université 8 Mai 1945, Guelma, monsieur **Taieb HAMMAIZIA**, Professeur à l'université Larbi Ben M'hidi, Oum El Bouagui et Monsieur **Salim MESBAHI**, Professeur à l'université Ferhat Abbas Sétif 1 d'avoir bien voulu faire parti de ce jury, et d'avoir pris le temps de lire et juger ce travail.*

Notre plus sincère gratitude vont à nos très chers parents pour leur aide, leur soutien et leurs accompagnement.

*Un merci spécial aux professeurs Madame **Zakia KEBBICHE**, Madame **Djaouida Guettaf** et tous les professeurs qui nous ont enseigné tout au long de notre parcours académique et ont illuminé nos vies avec la lumière du savoir.*

في هذه الأطروحة، تمت دراسة طرق التغطية للأمثلية الشاملة التحديدية الهولديرية. تم اقتراح طريقتين جديدتين. الأولى عبارة عن مزيج من مرحلتين متتاليتين كامتداد لطريقة بيافسكي لحل مسائل الأمثلية الشاملة أحادية البعد. تتكون الثانية أيضًا من خطوتين، تعتمد الخطوة الأولى على بناء منحنيات α -كثيفة. هذه الطريقة تحل مسائل الأمثلية الشاملة الهولديرية ذات عدة متغيرات. التجارب العددية للطريقتين على دوال عددية المستمدة من الأدبيات مشجعة.

الكلمات المفتاحية: الأمثلية الشاملة، طرق التغطية، خوارزمية بيافسكي-شوبرت، دوال لهولدر، منحنيات α -كثيفة، التحويل التقلصي.

Résumé:

Dans cette thèse, des méthodes de recouvrement d'optimisation globales déterministes höldériennes ont été étudiées. Deux nouvelles méthodes ont été proposées. La première est une combinaison de deux phases successives comme une extension de la méthode de Piyavskii pour résoudre des problèmes d'optimisation global unidimensionnel. La deuxième est également composée de deux étapes dont la première est basée sur la construction des courbes α -dense. Cette méthode résout des problèmes d'optimisation globale höldériennes multidimensionnel. Les expérimentations numériques des deux méthodes sur des fonctions tests tirées de la littérature sont encourageantes.

Mots clés: Optimisation globale, Méthodes de recouvrement, Algorithme de Piyavskii-Shubert, Fonctions de Hölder, Courbes α -dense, Transformation réductrice.

Abstract:

In this thesis, covering Hölderian deterministic global optimization methods have been studied. Two new methods have been proposed. The first is a combination of two successive phases as an extension of Piyavskii's method to solve one-dimensional global optimization problems. The second is also composed of two steps, the first step is based on the construction of α -dense curves. This method solves multidimensional Hölderian global optimization problems. The numerical experiments of the two methods on test functions drawn from the literature are encouraging.

Keywords: Global optimization, Covering methods, Piyavskii-Shubert's algorithm, Hölder functions, α -dense curves, Reducing transformation.

Table des matières

Introduction	1
1 Optimisation Globale	3
1.1 Définitions et rappels	3
1.1.1 Définition d'un problème d'optimisation globale	3
1.1.2 Classification des problèmes d'optimisation globale	5
1.1.3 Classification des approches d'optimisation globale	5
1.2 Principe général des méthodes de recouvrement	6
1.2.1 L'algorithme d'Evtushenko	8
1.2.2 Méthode de Piyavskii-Shubert	10
1.2.3 Différentes extensions de la méthode de Piyavskii-Shubert	13
1.2.4 La méthode de Brent	13
1.2.5 Méthode Branch and Bound	16
2 Méthodes d'optimisation globale déterministes höldériennes dans $\mathbb{R}$	18
2.1 Introduction	18
2.2 La condition de Hölder	19
2.3 Propriétés des fonctions höldériennes	20
2.4 Méthodes d'optimisation globale höldériennes unidimensionnelles.	24
2.4.1 Expressions explicites du point d'intersection des courbes paraboliques	26
2.4.2 Technique de la sécante	31
2.4.3 Méthode de Piyavskii modifiée <i>MPA</i>	33
2.4.4 Méthode utilisant des fonctions auxiliaires	37

2.4.5	Théorème de convergence	41
2.4.6	Fonctions borne inférieures tangentes par morceaux.	42
2.4.7	Fonctions borne inférieures non-tangentes par morceaux	48
2.4.8	Combinaison efficace des deux techniques KL_α et TPA	51
2.5	Conclusion	53
3	Optimisation globale höldérienne à plusieurs variables	54
3.1	Introduction	54
3.2	Optimisation globale höldérienne sans contraintes	55
3.2.1	Une relaxation des fonctions minorantes paraboliques par morceaux (BB-Piyavskii multidimensionnel)	55
3.2.2	La méthode de la transformation réductrice Alienor	58
3.2.3	Une nouvelle transformation réductrice	59
3.2.4	La méthode mixte Alienor- TPA	63
3.3	Résultats de convergence	65
3.3.1	Convergence de la méthode mixte Alienor- TPA	65
3.4	Optimisation globale höldérienne avec contraintes	66
3.4.1	Préliminaires et notation	68
3.4.2	Génération de courbes α -dense dans un ensemble compact non convexe	69
3.4.3	Construction des courbes α -dense	70
3.4.4	Exemples d'ensembles compacts α -densifiables	73
3.4.5	La méthode de la transformation réductrice combinée avec la méthode TPA	75
3.4.6	Relation entre la densité α et la précision ε	76
3.5	Sur l'estimation de la constante de Hölder h	77
3.6	Conclusion	78
4	Applications numériques	80
4.1	Les testes numériques pour les problèmes unidimensionnelles	80
4.1.1	Discussion et remarques finales.	89

4.2	Les testes numériques pour les problèmes multidimensionnelles sans contraintes	96
4.3	Les testes numériques pour les problèmes multidimensionnelles avec contraintes	98
4.4	Conclusion générale et perspectives	102

Table des figures

1.1	Extremums locaux et globaux d'une fonction à deux variables	4
1.2	Représentation des fonctions minorantes linéaires par morceaux.	11
2.1	Construction des fonctions minorantes paraboliques	30
2.2	Construction des interpolations linéaires liées aux paraboles sur un sous-intervalle [x_{i-1}, x_i]	32
2.3	Le sous-estimateur affine par morceaux de $f(x)$ sur l'intervalle [x_{i-1}, x_i]	35
2.4	Consstrucion des fonctions auxiliaires sur l'intervalle [a, b]	40
2.5	Les fonctions minorantes tangentes par morceaux de f	45
2.6	Les fonctions minorantes non-tangentes par morceaux	51
3.1	Illustration graphique de la transformation d'un problème multidimensionnel en un problème unidimensionnel à l'aide d'une courbe α -dense	62
3.2	La courbe α -dense $\ell(t)$ dans D_1 avec $\alpha = 0.2$	74
3.3	La courbe α -dense $\ell(t)$ dans D_2 avec $\alpha = 0.2$ et $\alpha = 0.01$	74
4.1	Représentation graphique de quelques fonctions tests et des points de la suite obtenus en phase 1	95

Introduction

Les progrès des sciences ont toujours été réalisés grâce à la modélisation qui consiste à représenter les phénomènes du réel par des équations mathématiques. Dans la plupart du temps le modélisateur a aussi, l'idée que l'on pourra ainsi agir sur le phénomène de façon à l'optimiser au sens de certains critères (coût, qualité, temps, etc.).

L'optimisation globale est une branche de mathématiques qui couvre toute la programmation mathématique traditionnelle. L'énorme besoin pratique de résoudre des problèmes d'optimisation globale couplé à une technologie informatique en évolution rapide a permis d'envisager des problèmes qui, il y a à quelques années, auraient été considérés comme insolubles sur le plan informatique. On va devoir chercher des extrema (généralement des minima) d'une fonction à valeurs réelles multiextremale, d'une ou de plusieurs variables. Bien entendu, ce sont les optima globaux, et non pas locaux, qui nous intéressent, d'où l'intérêt de l'optimisation globale ayant pour objectif l'obtention de minima ou de maxima globaux.

Dans cette thèse, on s'intéresse aux méthodes d'optimisation déterministes. Dans le premier chapitre, on commence par rappeler quelques généralités sur l'optimisation globale, ensuite, nous présentons une classe de méthodes déterministes dite de recouvrement qui ont la réputation d'être efficaces en dimension 1. Parmi ces méthodes, il y a celles qui sont basées sur l'utilisation des sous-estimateurs linéaires, d'autre sur le recouvrement du domaine faisable, et d'autres sur les techniques de partition et d'évaluation dans le cas où la fonction objectif est lipschitzienne de constante $L > 0$ définie sur un intervalle $[a, b]$ de $\mathbb{R}$. Le second chapitre est consacré à l'extension de certaines méthodes de recouvrement appliquées aux problèmes de minimisation globale des fonctions höldériennes de constantes $h > 0$ et $0 < \alpha < 1$ définie sur $[a, b] \subset \mathbb{R}$. Dans ce chapitre, nous donnerons deux nouvelles

modifications de la méthode de Piyavskii. La première, noté TPA , consiste à construire une suite croissante de fonctions minorantes tangentes par morceaux aux fonctions paraboliques de la fonction objectif sur l'intervalle $[a, b]$, et nous prouvons un résultat de convergence de l'algorithme TPA . La deuxième modification, noté $KL_\alpha TPA$, est composé de deux phases. La première phase KL_α à pour but de localiser le minimum global avec une tolérance $\delta > 0$, en un nombre d'itérations relativement petit. Après avoir obtenu un intervalle D' donné en phase 1, on applique l'algorithme TPA à la fonction f sur D' pour obtenir le minimum global avec une précision donnée $\varepsilon > 0$. Dans le troisième chapitre, nous étudierons la généralisation de certaines méthodes de recouvrement présentées dans le seconde chapitre, au cas multidimensionnel sans et avec contraintes. Dans la première partie, nous présenterons une généralisation de l'algorithme GPA [21] basée sur une procédure de séparation et évaluation Branch-and-Bound. Aussi, comme deuxième partie du chapitre, on donnera à l'occasion une nouvelle variante de la méthode de la transformation réductrice Alienor appliquées aux fonctions höldériennes. L'idée consiste à approcher la fonction objectif de plusieurs variables définie sur un pavé $\mathbf{P} = \prod_{i=1}^n [a_i, b_i]$ de $\mathbb{R}^n$, par une fonction d'une seule variable en densifiant le pavé $\mathbf{P}$ à l'aide d'une simple courbe assez régulière [65][66][67], pour pouvoir utiliser la méthode unidimensionnelle TPA . Comme troisième partie du chapitre, nous présenterons une méthode mixte $ALienor_TPA_CTs$, pour résoudre un problème d'optimisation globale höldérienne avec contraintes. Nous avons considéré une classe de problèmes lorsque la fonction objectif et les contraintes sont tous höldériennes et non-différentiables.

Enfin, dans le chapitre 4, nous terminerons notre travail par une série d'application numérique des algorithmes vus dans le chapitres 2 et 3 sur des fonctions testes tirées de la littérature. Une étude comparative sur les algorithmes qui existent est suivie par une conclusion générale et quelques perspectives.

1.1 Définitions et rappels

1.1.1 Définition d'un problème d'optimisation globale

Nous définissons un problème d'optimisation globale standard comme suit. Etant donné un ensemble fermé non vide $X \subset \mathbb{R}^n$ et une fonction continue $f : X \rightarrow \mathbb{R}$, trouver au moins un point $x^* \in X$ satisfaisant $f(x^*) \leq f(x)$ pour tout $x \in X$.

Supposons que X un compact de $\mathbb{R}^n$ et f est continue sur X . Alors, clairement, l'existence de x^* est assurée par le théorème bien connu de Weirstrass. Dans d'autres cas importants, on rencontre des ensembles faisables compacts X et des fonctions objectifs f qui sont continues à l'intérieur de X , mais qui ont des discontinuités sur le bord de X . Tout au long de cette thèse, un problème d'optimisation globale est noté par :

$$(P) \left\{ \begin{array}{l} \min f(x) \\ x \in X. \end{array} \right. \quad (1.1)$$

Optimiser la fonction f consiste à déterminer les éléments $x^* \in X$ tels que f atteigne sa valeur minimale (resp. sa valeur maximale) en x^* .

Un point $x^* \in X$ satisfaisant $f(x^*) \leq f(x)$ pour tout $x \in X$ est appelé un minimiseur global de f sur X . La valeur correspondante de f est appelée minimum global (Absolu) de f sur X notée $\min f(X)$ (voir Fig.1.1). L'ensemble de toutes les solutions du problème (1.1) sera noté $\operatorname{argmin} f(X)$. On parle alors de minimisation (nous nous intéressons seulement à la

recherche du minimum d'une fonction, puisqu'un problème de maximisation peut toujours se ramener à un problème de minimisation

$$\max f(x) = -\min(-f(x)).$$

Soit $\varepsilon > 0$, un nombre réel et $V(x^*, \varepsilon)$ un voisinage d'un point x^* de $\mathbb{R}^n$, est défini comme la boule ouverte centrée en x^* de rayon ε .

$$V(x^*, \varepsilon) = \{x \in \mathbb{R}^n : \|x - x^*\| < \varepsilon, \}$$

où $\|\cdot\|$ désigne la norme euclidienne dans $\mathbb{R}^n$.

Un point $x^* \in X$ est appelé minimiseur local de f sur X s'il existe un $\varepsilon > 0$ tel que

$$\forall x \in V(x^*, \varepsilon) \cap X, f(x^*) \leq f(x),$$

$f(x^*)$ est dit minimum local (voir Fig.1.1).

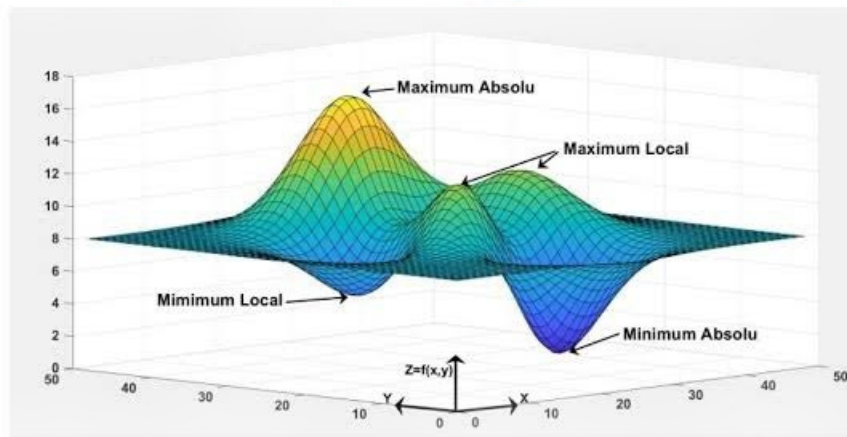


FIGURE 1.1 – Extremums locaux et globaux d'une fonction à deux variables

1.1.2 Classification des problèmes d'optimisation globale

On distingue deux types d'optimisation globale, avec et sans contraintes. Si $X \neq \mathbb{R}^n$, le problème (1.1) est dit problème de minimisation avec contraintes, sinon (1.1) est dit problème de minimisation sans contraintes. Ici, pour que le problème admette un minimum global, nous devons fréquemment exiger des propriétés supplémentaires de f dite condition de croissance à l'infini ($\lim_{\|x\| \rightarrow +\infty} f(x) = +\infty$).

Afin de comprendre les énormes difficultés inhérentes aux problèmes d'optimisation globale et le coût de calcul pour les résoudre, il est important de noter que toutes les techniques standard en optimisation non linéaire ont au plus des minima locaux. De plus, il n'y a pas de critère local pour décider si une solution locale est globale. Par conséquent, les procédés classiques d'optimisation utilisant des outils tels que les dérivées, les gradients, les sous-gradients et similaires, ne sont, en général, pas capables de localiser ou d'identifier un optimum global, et en général un minimum local est différent d'un minimum global. Pour cela, plusieurs chercheurs dans le domaine ont proposé les méthodes d'optimisation globale n'est pas pour trouver seulement les minimums locaux, mais l'essentiel de désigner lequel d'eux est le minimum global, et bien sûr ça va dépendre de la structure du problème donné.

1.1.3 Classification des approches d'optimisation globale

Il existe deux catégories de méthodes utilisées en optimisation globale :

- La première classe est celle des méthodes stochastiques ce sont des méthodes basées sur des tirages aléatoires de points à l'intérieur d'un domaine X . Elles ne nécessitent pas une forte propriété de régularité sur les fonctions qui définissent le problème à traiter. Les méthodes stochastiques permet d'explorer plus facilement un espace des solutions de très grande dimension. Leur inconvénient principal est qu'elles peuvent passer plusieurs fois à côté de la solution globale et la non garantie de la qualité de la solution obtenue. De plus, ces méthodes peuvent s'avérer très coûteuses en temps CPU par rapport aux méthodes déterministes.

- La deuxième classe est les méthodes déterministes globales. Elles garantissent l'obtention de la solution globale du problème. Ces méthodes permettent de bien gérer les contraintes, contrairement aux méthodes stochastiques et peuvent s'appliquer aux problèmes mixtes (va-

riables réelles, entières etc...). Cependant, il faut savoir que les méthodes déterministes globales restent utilisables tant que le nombre de variables ne devient pas trop important. Au delà d'une vingtaine de variables, elles atteignent leurs limites. Pour ces méthodes, l'exploration de l'espace des solutions se fait grâce à des procédures de recherche qui sont élaborées à partir de la constante de Lipschitz, des dérivées ou d'autres informations concernant la fonction à optimiser. Dans ce chapitre, on s'intéresse aux méthodes d'optimisation globale déterministes, parmi elles, les méthodes de recouvrement.

1.2 Principe général des méthodes de recouvrement

Les méthodes de recouvrement sont très utilisées en optimisation globale. Leur intérêt réside dans le fait qu'elles utilisent une hypothèse satisfaite par la plupart des problèmes rencontrés dans la pratique. Ils s'agit de supposer que la fonction objectif est à taux de variation borné. Elles sont appelées les méthodes de recouvrement, car elles génèrent des suites de points contenues dans des sous-domaines dont l'union couvre le domaine faisable. Le principe de ce type de méthodes est le suivant :

D'abord considérons le problème de minimisation globale :

$$(P) \begin{cases} \min f(x) \\ x \in X. \end{cases}$$

D'un point de vue numérique, le calcul d'une solution optimale globale exacte x^* peut être trop coûteuse ou même impossible. Pour cela une solution ne pourrait être obtenue qu'avec une certaine précision ε strictement positive donnée. Généralement, au lieu de résoudre le problème (P) , i.e., trouver un minimiseur global x^* tel que $f^* = f(x^*)$ on résout le problème (P_ε) qui consiste à trouver un point $x_\varepsilon^* \in X_\varepsilon^* = \{x \in X, f(x) - \varepsilon \leq f^*\}$ tel que pour un petit $\varepsilon > 0$ on a

$$f_\varepsilon^* = f(x_\varepsilon^*) \leq f^* + \varepsilon,$$

X_ε^* est dit l'ensemble des solutions ε -approximées.

Donnons d'abord l'idée clé de ces méthodes. Supposons que la fonction f est évaluée aux

points $x_1, x_2, \dots, x_k$. Nous appelons record d'ordre k la valeur suivante.

$$R_k = \min \{f(x_1), f(x_2), \dots, f(x_k)\}, \quad (1.2)$$

et nous appelons point record d'ordre k tout point $x_i, i = 1, \dots, k$ vérifiant : $f(x_i) = R_k$

Définissons l'ensemble suivant :

$$E_k = \{x \in X, R_k - \varepsilon \leq f(x)\}, \quad (1.3)$$

évidemment,

$$R_k - \varepsilon \leq \min_{x \in E_k} f(x),$$

alors :

$$\min_{x \in X} f(x) = \min_{x \in X - E_k} f(x), \text{ si } X \neq E_k.$$

Donc les points de E_k ne peuvent pas améliorer les valeurs de la fonction objectif et ne représentent pas d'intérêt dans la recherche du minimum global et par conséquent il peut être omis de l'ensemble faisable. On continue la recherche du minimum global seulement sur l'ensemble $X - E_k$. En pratique si :

$$X \subset E_k, \quad (1.4)$$

alors le problème initial est résolu et le point record x_k est considéré comme solution approchée (R_k est le minimum global approximé de f), avec la garantie que $x_k \in X$.

Maintenant, si f est évaluée en un nouveau point x_{k+1} , alors un autre record R_{k+1} est obtenu, et par conséquent un ensemble plus grand E_{k+1} peut être omis. La suite réelle (R_k) est décroissante, par contre la suite d'ensembles $\{E_k\}$ est croissante au sens de l'inclusion, i.e.,

$$R_{k+1} \leq R_k, \quad E_k \subset E_{k+1}.$$

Ainsi le problème de minimisation est ramené à la construction d'une suite de points $x_1, x_2, \dots, x_k$ vérifiant (1.4).

Parmi les méthodes de recouvrement, il existe celles qui utilisent des fonctions auxiliaires minorantes de la fonction objectif (des fonctions sous-estimateurs). L'idée commune dans ce type de méthodes est de construire une suite croissante de fonctions minorantes de la fonction objectif de façon que les minima globaux des éléments de cette suite convergent vers un minimum global de la fonction objectif. Généralement, les fonctions rencontrées dans la

plupart des problèmes pratiques, ont un taux de variation borné (hypothèse moins restrictive que la dérivabilité), c'est pourquoi l'hypothèse selon laquelle la fonction objectif f est lipschitzienne est très commune en analyse numérique.

Définition 1.1 Une fonction $f : X \rightarrow \mathbb{R}$ est dite lipschitzienne sur X , s'il existe une constante $L > 0$ telle que

$$|f(x) - f(y)| \leq L \|x - y\|, \forall x, y \in X.$$

Il est bien connu que toutes les fonctions continument différentiable f avec des gradients bornés sur X sont lipschitziennes sur X , où

$$L = \sup \{ \|\nabla f(x)\|, x \in X \},$$

est une constante de Lipschitz ($\|\cdot\|$ désigne la norme euclidienne).

Soit le problème (P_ε) unidimensionnel suivant :

$$(P_\varepsilon) \left\{ \min_{x \in [a, b]} f(x), \right.$$

avec f est une fonction lipschitzienne de constante L définie sur in intervalle $[a, b]$.

Ce problème relativement simple (P_ε) est intéressant, car il répond à de nombreuses applications et aussi par ce que certains algorithmes de résolution du problème (P_ε) peuvent être étendus au cas multidimensionnel. Les problèmes (P) et (P_ε) ont été étudiés par plusieurs auteurs par exemple, Evtushenko [17], Piyavskii [46], Shubert [55], Timonov [59], Pinter [44] et une autre étude est celle de Hansen et Jaumard [26], etc...

1.2.1 L'algorithme d'Evtushenko

L'algorithme d'Evtushenko fait partie de la classe des méthodes de recouvrements itératifs [17]. Dans le cas unidimensionnel, la variante de l'algorithme d'Evtushenko basée sur le théorème suivant :

Théorème 1.1 Soit f une fonction lipschitzienne de constante L , définie sur un intervalle $[a, b]$.

Supposons que f soit évaluée aux points $x_1, x_2, \dots, x_k$ de $[a, b]$. Posons

$R_k = \min \{f(x_1), f(x_2), \dots, f(x_k)\}$, Si on a $[a, b] \subset \bigcup_{i=1}^k I(x_i, r_{ik})$ avec $r_{ik} = \frac{\varepsilon + f(x_i) - R_k}{L}$ alors R_k est le minimum global de f avec la précision ε sur $[a, b]$.

d'après le théorème 1.1, on peut couvrir l'intervalle $[a, b]$ par des sous intervalles $\{I_i\}_{1 \leq i \leq k}$ de centres $\{x_i\}_{1 \leq i \leq k}$ et de rayons $r_{ik} = \frac{\varepsilon + f(x_i) - R_k}{L}$ tels que $[a, b] \subset \bigcup_{i=1}^k I_i$. Puisque le plus petit des rayons r_{ik} est $\frac{\varepsilon}{L}$, on prend $x_1 = a + \frac{\varepsilon}{L}$ qui est le centre de l'intervalle $I_1 = [a, a + \frac{2\varepsilon}{L}]$ et de rayon $r_{11} = \frac{\varepsilon}{L}$, parce que à l'initialisation $f(x_1) = R_1$. Avec cette valeur de x_1 , on gagne du temps et on est sûr de ne pas avoir ignoré le minimum global au voisinage de ce point. En effet si $x^* = \operatorname{argmin}_{x \in [a, b]} f(x) \in [a, a + \frac{\varepsilon}{L}]$ alors

$$|f(x_1) - f(x^*)| \leq Ld(x, x^*) \leq L \frac{\varepsilon}{L}.$$

Pour le point x_2 qui est le centre de l'intervalle I_2 de rayon $r_{22} = \frac{\varepsilon + f(x_2) - R_2}{L}$, on devait prendre $x_2 = x_1 + r_{11} + r_{22}$. Où r_{22} est inconnu mais on sait que $r_{22} > \frac{\varepsilon}{L}$, on prend alors $x_2 = x_1 + r_{11} + \frac{\varepsilon}{L}$.

En général, pour $i \geq 1$, la suite x_{i+1} définie par :

$$x_{i+1} = x_i + \frac{2\varepsilon + f(x_i) - R_k}{L}.$$

Ce choix nous permet de ne pas rater le minimum global de f , car les intervalles I_i se rencontrent. On arrête quand k vérifie $[a, b] \subset \bigcup_{i=1}^k I_i$

Algorithme 1 d'Evtushenko

- 1: Initialisation.
 - 2: $k = 1$, $x_1 = a + \frac{\varepsilon}{L}$, $x_\varepsilon = x_1$, $f_\varepsilon = f(x_\varepsilon)$.
 - 3: Etape $k = 1, 2, \dots$
 - 4: Poser $x_{k+1} = x_k + (f(x_k) - f_\varepsilon + 2\varepsilon)/L$.
 - 5: **if** $x_{k+1} > b - \frac{\varepsilon}{L}$ **then**
 - 6: stop
 - 7: **else**
 - 8: déterminer $f(x_{k+1})$
 - 9: **end if**
 - 10: **if** $f(x_{k+1}) < f_\varepsilon$ **then**
 - 11: $f_\varepsilon = f(x_{k+1})$, et $x_\varepsilon = x_{k+1}$
 - 12: **end if**
 - 13: Poser , $k = k + 1$ et aller à k
-

1.2.2 Méthode de Piyavskii-Shubert

L'algorithme le mieux connu et le plus étudié pour l'optimisation unidimensionnelle lipschitzienne est de Piyavskii [45],[46]. Il a été indépendamment redécouvert par Shubert [55]. C'est un algorithme pour résoudre le problème (P_ε) , à chaque itération elle consiste à construire une fonction minorante F_i de la fonction objectif f sur le domaine faisable $[a, b]$ et évalue f au point qui correspond au minimum de cette fonction minorante (point pic) [7]. L'algorithme s'arrête quand la différence entre la hauteur de la fonction minorante et la valeur imposée ne dépasse pas ε . Cette méthode nécessite la connaissance de la constante de Lipschitz L .

La méthode de Piyavskii-Shubert minimise une fonction lipschitzienne f définie sur l'intervalle $[a, b]$, c'est-à-dire :

$$|f(x) - f(y)| \leq L |x - y|, \forall x, y \in [a, b]. \quad (1.5)$$

Pour construire une fonction minorante de la fonction objectif f sur l'intervalle $[a, b]$ (voir Fig.1.2), on pose $y = \frac{a+b}{2}$ dans l'inégalité (1.5) on obtient la première fonction minorante

$$F_1(x) = f(x_1) - L |x - x_1|.$$

La fonction $F_1(x)$ est minimisée sur $[a, b]$, dans le but d'obtenir le plus petit pic (d'en bas) au point x_2 (voir Fig.1.2) tel que

$$x_2 = \arg \min F_1([a, b]).$$

Maintenant a l'étape k , on prend

$$F_k(x) = \max_{j=1,2,\dots,k} \{f(x_j) - L |x - x_j|\},$$

et

$$x_{k+1} = \arg \min_{x \in [a,b]} F_k(x).$$

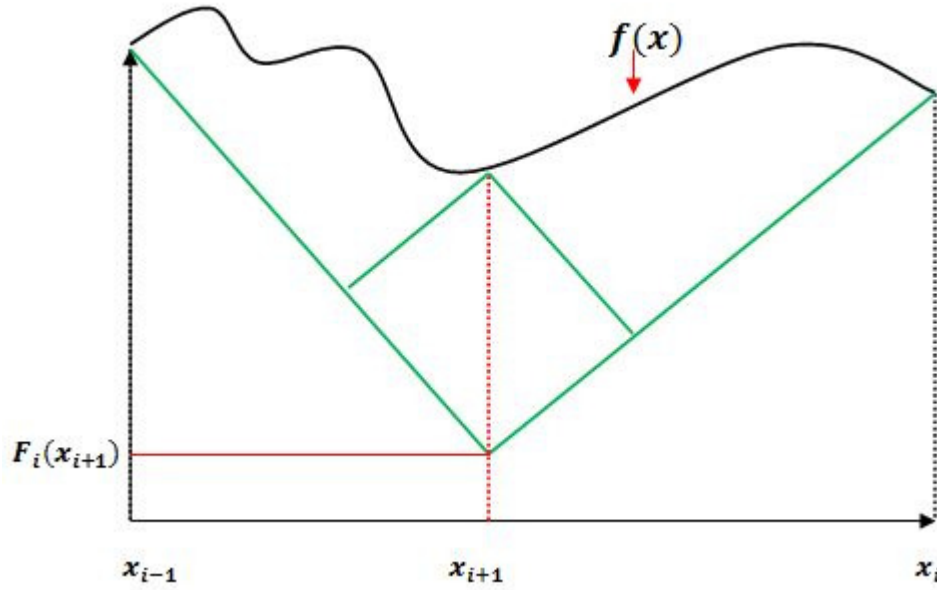


FIGURE 1.2 – Représentation des fonctions minorantes linéaires par morceaux.

Lemme 1.1 Soient f une fonction lipschitzienne sur $[a, b]$ de constante de Lipschitz L et $\tilde{x}$ le point pic donné dans l'intervalle $[a, b]$. Alors $\tilde{x} \in [a, b]$.

Preuve. Nous allons prouver que $a \leq \tilde{x} \leq b$. L'idée est de poser $y = a$ et $y = b$ dans l'inégalité (1.5) pour obtenir les deux inégalités suivantes :

$$f(x) \geq f(a) - L|x - a|, \forall x \in [a, b]$$

$$f(x) \geq f(b) - L|x - b|, \forall x \in [a, b].$$

Alors on obtient la première fonction minorante

$$F_1(x) = \max\{f(a) - L|x - a|, f(b) - L|x - b|\}, \forall x \in [a, b].$$

et

$$\tilde{x} = \underset{[a, b]}{\operatorname{argmin}} F_1(x), \forall x \in [a, b],$$

$$\begin{aligned}
\tilde{x} &= \frac{f(a)-f(b)}{2L} + \frac{a+b}{2} \\
&= -\frac{f(b)-f(a)}{2L} + \frac{L(b-a)}{2L} + a = -\frac{f(b)-f(a)}{2L} - \frac{L(b-a)}{2L} + b \\
&= \underbrace{\frac{L(b-a)-(f(b)-f(a))}{2L}}_{\geq 0} + a = -\underbrace{\frac{L(b-a)-(f(b)-f(a))}{2L}}_{\geq 0} + b.
\end{aligned}$$

(1) (2)

Alors

$$(1) \geq a \text{ et } (2) \leq b.$$

Ce qui achève la preuve.

■

Enfin, on donne l'algorithme de Piyavskii, pour résoudre le problème (P_ε) .

Algorithme 2 Piyavskii_Shubert

- 1: 1. Initialisation.
 - 2: $k = 1$, $x_1 = \frac{a+b}{2}$, $x_\varepsilon = x_1$, $f_\varepsilon = f(x_\varepsilon)$, $F_\varepsilon = f_\varepsilon - \frac{L}{2}(b-a)$,
 - 3: $F_1 = f(x_1) - L|x - x_1|$.
 - 4: 2. Etape $k = 1, 2, \dots$
 - 5: **if** $f_\varepsilon - F_\varepsilon \leq \varepsilon$ **then**
 - 6: Stop.
 - 7: **else**
 - 8: $x_{k+1} = \operatorname{argmin}_{x \in [a, b]} F_k(x)$.
 - 9: **end if**
 - 10: **if** $f(x_{k+1}) \leq f_\varepsilon$ **then**
 - 11: $f_\varepsilon = f(x_{k+1})$, $x_\varepsilon = x_{k+1}$.
 - 12: **end if**
 - 13: Poser $F_{k+1}(x) = \max_{j=1,2,\dots,k+1} \{f(x_j) - L|x - x_j|\}$
 - 14: $F_\varepsilon = \min F_{k+1}([a, b])$.
 - 15: Aller à l'Etape k .
-

Remarque 1.1 : Les algorithmes de résolution du problème (P_ε) discutés jusqu'ici considèrent la constante de Lipschitz L comme connue a priori. Signalons que Strongin [56] propose un

algorithme qui, au lieu d'utiliser L (qui est en général inconnue ou difficile à calculer), utilise une estimation de L qui est un multiple de la plus grande valeur absolue des pentes des droites joignant les points d'évaluation successifs. La convergence vers un minimum global peut être garantie lorsque cette estimation est une borne supérieure suffisamment grande pour la constante de Lipschitz L (pour plus de détails, voir Strongin (1978), Hansen et al. (1989), Hansen et Jaumard (1995)). Aussi la stratégie DIRECT [33] a porté certains remèdes aux difficultés rencontrées dans la méthode de Piyavskii, particulièrement, DIRECT ne dépend pas de la constante de Lipschitz, puisque les points où la fonction est évaluée sont toujours les tiers des sous-intervalles. Par contre l'algorithme de Piyavskii évalue f au point qui correspond au minimum de cette fonction minorant $F_k(x) = \max_{j=1,2,\dots,k} \{f(x_j) - L|x - x_j|\}$.

1.2.3 Différentes extensions de la méthode de Piyavskii-Shubert

Des papiers et livres proposent plusieurs approches pour la résolution numérique du problème (P_ε) et donnent une classification des problèmes et de leurs méthodes de résolution. Par exemple, le livre de Horst et Tuy [29] discute généralement les algorithmes déterministes. L'algorithme de Piyavskii avec diverses extensions semble être l'approche la plus souvent discutée pour le problème (P_ε). Shubert [55], Basso [5][6], Schoen [51], Shen et Zhu [54] et Horst et Tuy [28] proposent une autre formulation de l'algorithme de Piyavskii. Des extensions multidimensionnelle ont été élaborées par Danilin et Piyavskii [14], Mayne et Polak [40], Mladineo [64] et Meewella et Mayne [41]. Evtushenko [16], Galperin [19] et Hansen, Jaumard et Lu [25][26] proposent d'autres algorithmes pour ce type de problèmes où leurs extension au cas multidimensionnel. Hansen et Jaumard [27] résument et discutent les algorithmes proposés et les présentent dans un langage informatique.

1.2.4 La méthode de Brent

L'algorithme proposé par Brent [9][34] est une modification de l'algorithme de Piyavskii. Elle s'applique aux fonctions d'une seule variable et deux fois continûment différentiables dont la dérivée seconde est bornée. L'idée consiste à construire une suite croissante de fonctions $(\varphi_i)_{1 \leq i \leq k}$ paraboliques par morceaux sur l'intervalle $[a, b]$, en se basant sur le résultat

suivant :

Théorème 1.2 Soit f une fonction à une seule variable deux fois continûment dérivable telle que

$$|f''(x)| \leq M, \quad \forall x \in [a, b].$$

Alors $\forall x_1, x_2 \in [a, b]$ vérifiant $x_1 < x_2$, la parabole $\varphi(x)$ définie par :

$$\begin{cases} \varphi''(x) = M, \quad \forall x \in [a, b] \\ \varphi(x_1) = f(x_1) \\ \varphi(x_2) = f(x_2). \end{cases}$$

est un sous estimateur de f sur $[x_1, x_2]$.

Preuve.

Posons $g(x) = \varphi(x) - f(x)$ pour $x \in [a, b]$, alors $g''(x) = \varphi''(x) - f''(x) = M - f''(x) \geq 0$, donc g est convexe et on a : $g(x_1) = g(x_2) = 0$, d'où forcément on a $g(x) \leq 0, \forall x \in [x_1, x_2]$, d'où $\varphi(x) \leq f(x), \forall x \in [x_1, x_2]$. ■

Comme conséquence de la théorème 1.2, si f est évaluée aux points $a = x_0 < x_1 < x_2 < \dots < x_k = b$ de $[a, b]$, on peut construire une fonction parabolique par morceaux sur $[a, b]$. En effet, à partir de chaque couple (x_j, x_{j+1}) de points successifs, on construit une parabole notée $\varphi_{[x_j, x_{j+1}]}$ définie sur l'intervalle $[x_j, x_{j+1}]$ et ceci pour $j = 0, \dots, k-1$. Ensuite on définit la fonction F_k par :

$$\begin{aligned} F_k(x) &: [a, b] \rightarrow \mathbb{R} \\ x &\rightarrow \varphi_{[x_j, x_{j+1}]}(x) \text{ si } x \in [x_j, x_{j+1}], \end{aligned}$$

F_k est une fonction parabolique par morceaux, elle sera notée par :

$$F_k(x) = [\varphi_{[a, x_1]}, \varphi_{[x_1, x_2]}, \dots, \varphi_{[x_{k-1}, x_k]}].$$

Une parabole $\varphi(x)$, vérifiant $\varphi'' = M$ et passant par les deux points $(x_j, f(x_j))$ et $(x_{j+1}, f(x_{j+1}))$ avec $x_j \neq x_{j+1}$, est définie comme suit :

$$\varphi(x) = \frac{M}{2}x^2 + \alpha x + \beta,$$

avec $\alpha = \frac{f(x_j) - f(x_{j+1})}{x_j - x_{j+1}} - \frac{M}{2}(x_j + x_{j+1})$ et $\beta = \frac{x_j f(x_{j+1}) - x_{j+1} f(x_j)}{x_j - x_{j+1}} + \frac{M}{2} x_j x_{j+1}$.

Elle possède le minimiseur global suivant :

$$x_0 = \frac{-\alpha}{M} = \frac{f(x_{j+1}) - f(x_j)}{M(x_j - x_{j+1})} + \frac{x_j + x_{j+1}}{2}.$$

Algorithme 3 Brent

- 1: Initialisation.
 - 2: $k = 1$, $x_1 = \frac{f(a) - f(b)}{M(b - a)} + \frac{a + b}{2}$, $x_\varepsilon = \arg \min \{f(a), f(x_1), f(b)\}$, $f_\varepsilon = f(x_\varepsilon)$.
 - 3: $F_\varepsilon = \varphi_{[a, b]}(x_1)$, $F_1 = [\varphi_{[a, x_1]}, \varphi_{[x_1, b]}]$
 - 4: Etape $k = 1, 2, \dots$
 - 5: **if** $f_\varepsilon - F_\varepsilon \leq \varepsilon$ **then**
 - 6: Stop.
 - 7: **else**
 - 8: $x_{k+1} = \arg \min F_k([a, b])$
 - 9: **end if**
 - 10: **if** $f(x_{k+1}) \leq f_\varepsilon$ **then**
 - 11: $f_\varepsilon = f(x_{k+1})$, $x_\varepsilon = x_{k+1}$
 - 12: **end if**
 - 13: Poser $F_{k+1}(x) = [\varphi_{[a, x_1]}, \varphi_{[x_1, x_2]}, \dots, \varphi_{[x_k, b]}]$ et $F_\varepsilon = \min F_{k+1}([a, b])$
 - 14: $k = k + 1$ et aller à l'Etape k
-

1.2.5 Méthode Branch and Bound

La méthode Branch and Bound (B&B) est une approche d'optimisation globale. Cette méthode basée sur la subdivision du domaine faisable en plusieurs sous ensembles. Pour chaque sous ensemble on construit par exemple une borne supérieure et une borne inférieure pour la fonction objectif afin d'éliminer les parties ne contenant pas l'optimum global et de travailler sur le reste des parties. Noter que cette méthode assez générale pour la résolution des problèmes d'optimisation non convexes, et plus particulièrement des problèmes d'optimisation combinatoire.

Principe de la méthode

Soit le problème d'optimisation globale suivant.

$$\min_{x \in X} f(x). \quad (1.6)$$

L'algorithme (B&B) consiste en la subdivision de l'ensemble faisable X en petites régions X_k où la borne supérieure et inférieure de la fonction objectif sont calculées. Selon ces bornes, chaque région est explorée ou éliminée pour construire l'arbre de Branch and Bound c.-à-d. consiste à engendrer des suites $\{UB_k\}$ et $\{LB_k\}$ des bornes supérieures et inférieures respectivement de la valeur minimale de la fonction objectif de (1.6). La solution globale est obtenue une fois que la courante meilleure borne supérieure est proche de la courante meilleure borne inférieure compte obtenu d'une tolérance spécifiée ε . Soit une région X_k est subdivisée en sous régions X_{ki} . Les bornes supérieures et inférieures à l'itération k sur l'ensemble X_k :

$$\begin{cases} UB_k = \min(UB_{ki}) \\ LB_k = \min(LB_{ki}). \end{cases}$$

Le principe est le meilleur d'abord. Tout sous ensemble sur lequel la borne inférieure dépasse UB_k sera évidemment éliminé.

Algorithme 4 Branch-and-Bound

- 1: 1. Initialisation
 - 2: $k = 0$, choisissez un ensemble X_0 suffisamment grand tel que : $X \subset X_0$
 - 3: Poser : $\Gamma_0 = X_0$.
 - 4: Déterminer : $U_0 = \beta(X_0)$ borne supérieure.
 - 5: Déterminer : $L_0 = \alpha(X_0)$ borne inférieure.
 - 6: 2. Etape
 - 7: **while** $U_k - L_k > \varepsilon$ **do**
 - 8: Supprimer tous les X_i de Γ_k tels que $\alpha(X_i) > U_k$.
 - 9: Choisissez et subdiviser un élément X_i de Γ_k avec $\alpha(X_i) = L_k$.
 - 10: Soit N_k la subdivision de X_i , poser $\Gamma_{k+1} = (\Gamma_k \setminus X_i) \cup N_k$.
 - 11: Poser : $U_{k+1} = \min_{X_i \in \Gamma_{k+1}} \beta(X_i)$ et $L_{k+1} = \min_{X_i \in \Gamma_{k+1}} \alpha(X_i)$.
 - 12: Poser $k = k + 1$.
 - 13: **end while**
 - 14: Considérer U_k comme l'estimation du minimum global de f sur X .
-

Méthodes d'optimisation globale déterministes höldériennes dans $\mathbb{R}$

2.1 Introduction

Considérons le problème d'optimisation globale unidimensionnel sans contraintes suivant, c'est-à-dire trouver au moins un point $x^* \in [a, b]$ et la valeur f^* telle que

$$(P) \begin{cases} f^* = f(x^*) = \min f(x) \\ x \in [a, b], \end{cases} \quad (2.1)$$

où $a, b \in \mathbb{R}$ et f la fonction objectif multi-extrémale unidimensionnel supposée continue et peut être non différentiable sur l'intervalle $[a, b]$. En fait, nous ne chercherons pas la solution exacte x^* de ce problème, nous voulons juste avoir sa valeur approximative x_{opt} . Pour y parvenir, nous développerons un algorithme d'optimisation globale adapté aux fonctions höldériennes, qui donnera une approximation de x^* telle que $|f(x_{opt}) - f(x^*)| \leq \varepsilon$ où ε une précision requise a priori.

Le processus de recherche de l'extrémum global d'une fonction non convexe, non différentiable et multi-extrémale est souvent nécessaire en optimisation globale. Ce dernier est devenu l'un des domaines de recherche les plus actifs ces dernières années. L'irrégularité locale, par exemple, la non-différentiabilité, et l'occurrence d'un grand nombre de minima locaux et globaux de la fonction objectif dans le domaine réalisable sont quelques-uns des problèmes qui rendent l'optimisation globale de certaines classes de fonctions unidimensionnelles non convexes complexes à traiter. Ici, nous abordons le problème de trouver le

minimiseur global d'une fonction f satisfaisant la condition de Hölder.

2.2 La condition de Hölder

Pour tout nombre réel $\alpha \in]0, 1]$, on dit que f est une fonction continue höldérienne d'exposant α et de constante positif h dans l'ensemble $X = [a, b]$, (ou f est (α, h) -höldérienne continue), lorsque f satisfait :

$$h = \sup_{\substack{x, y \in [a, b] \\ x \neq y}} \frac{|f(x) - f(y)|}{|x - y|^\alpha} < +\infty,$$

de manière équivalente, lorsqu'il existe $h > 0$ et $0 < \alpha < 1$, tels que

$$\forall x, y \in [a, b], |f(x) - f(y)| \leq h |x - y|^\alpha. \quad (2.2)$$

On appelle h la constante de Hölder, si $\alpha = 1$ alors h est dite constante de Lipschitz et f fonction lipschitzienne.

Si la fonction objectif f est (α, h) -höldérienne sur $[a, b]$ alors f est aussi (h', α') -höldérienne telle que $h' \geq h$, ou $\alpha' \leq \alpha$. Notons que les fonctions höldériennes telles que α est suffisamment petit sont beaucoup plus irrégulières que celles où α est suffisamment grand. L'interprétation géométrique de (2.2) est la suivante : soit c un point quelconque de $[a, b]$ et posons $y = c$ dans (2.2). Alors $\forall x \in [a, b]$

$$x \in [a, c]: f(c) + h(x - c)^\alpha \leq f(x) \leq f(c) - h(x - c)^\alpha$$

$$x \in [c, b]: f(c) - h(x - c)^\alpha \leq f(x) \leq f(c) + h(x - c)^\alpha.$$

Ces deux inégalités définissent deux cônes paraboliques (au lieu linéaires dans le cas $\alpha = 1$) dont le cône peut être translaté le long du graphe de f de telle sorte que le graphe reste toujours entièrement contenue dans le cône parabolique [18]. La minimisation globale unidimensionnelle des fonctions höldériennes sur un intervalle fermé et borné de $\mathbb{R}$ apparaît dans une variété de situations réelles, par exemple en recherche opérationnelle, le problème de la localisation des points de production dans le cadre d'une politique de prix de livraison unique, voir Hanjoul et al. [24] et problèmes d'optimisation d'horizon infini, voir Kiatsupaibul et al. [35], où les auteurs ont donné une transformation qui réduit le problème de l'horizon

infini en un problème d'optimisation global unidimensionnel équivalent, où la fonction objectif est höldérienne sur un intervalle de $\mathbb{R}$.

2.3 Propriétés des fonctions höldériennes

Proposition 2.1 *On dit que la fonction vectorielle $f = (f_1, f_2, \dots, f_n)$ est (α, h) -höldérienne continue sur X si et seulement si les composantes $f_1, f_2, \dots, f_n$ sont (α, h_i) -höldériennes continues sur X , et pour $i = 1, \dots, n$ on a*

$$h_i \leq h \leq \sqrt{n} \max_{i \in \{1, 2, \dots, n\}} h_i. \quad (2.3)$$

Preuve. Pour $1 \leq i \leq n$, f_i sont des fonctions (α, h_i) -höldériennes continues sur X

$$\begin{aligned} |f_i(x) - f_i(y)| &\leq \|f(x) - f(y)\|, \forall x, y \in X, \\ \|f(x) - f(y)\| &= \sqrt{\sum_{i=1}^n |f_i(x) - f_i(y)|^2} \leq \sqrt{\sum_{i=1}^n h_i^2 |x - y|^{2\alpha}} \\ &\leq \sqrt{n} \max_{i \in \{1, \dots, n\}} h_i |x - y|^\alpha, \forall x, y \in X, \end{aligned}$$

on obtient

$$\left\{ \begin{array}{l} \frac{|f_i(x) - f_i(y)|}{|x - y|^\alpha} \leq \frac{\|f(x) - f(y)\|}{|x - y|^\alpha} \quad (1) \\ \frac{\|f(x) - f(y)\|}{|x - y|^\alpha} \leq \sqrt{n} \max_{i \in \{1, \dots, n\}} h_i \quad (2) \end{array} \right. \quad (2.4)$$

de la seconde inégalité de (2.4) on déduit que

$$h \leq \sqrt{n} \max_{i \in \{1, \dots, n\}} h_i.$$

D'autre part, du première inégalité de (2.4) on déduit que

$$\frac{|f_i(x) - f_i(y)|}{|x - y|^\alpha} \leq h,$$

d'où

$$h_i \leq h.$$

■

Proposition 2.2 *Supposons que X soit borné. Pour tout exposant $\beta < \alpha$, si f est (α, h) -höldérienne continue sur X , alors f est (β, h') -höldérienne continue sur X , et δ le diamètre de X ,*

$$\delta = \text{diam} X = \sup_{x, y \in X} |x - y|,$$

on a alors

$$h' \leq \delta^{1-\beta} h.$$

Preuve. Pour $x, y \in X$, on a

$$|f(x) - f(y)| \leq h |x - y|^\alpha = h |x - y|^{\alpha-\beta} |x - y|^\beta \leq h \delta^{\alpha-\beta} |x - y|^\beta,$$

et comme $\alpha \in]0, 1]$ on a

$$h' \leq \delta^{1-\beta} h,$$

donc f est (β, h') -höldérienne continue sur X . ■

Remarque 2.1 [18] *La réciproque de la proposition 2.2 est fausse en général.*

Proposition 2.3 *Si f est une fonction (α, h) -höldérienne continue sur X , alors f est uniformément continue sur X .*

Preuve. On a f est une fonction (α, h) -höldérienne continue sur X c.-à-d. :

$$|f(x) - f(y)| \leq h |x - y|^\alpha, \quad \forall x, y \in X,$$

pour $\varepsilon > 0$ si $|x - y|$ est plus petit que $\delta = (\frac{\varepsilon}{h})^{\frac{1}{\alpha}}$, alors $|f(x) - f(y)| \leq \varepsilon$. ■

La réciproque de la proposition 2.3 est fausse en général, voir l'exemple suivant.

Exemple 2.1 *Soit*

$$f(x) = \begin{cases} \frac{1}{\log(x)}, & \text{si } x \in]0, \frac{1}{2}] \\ 0, & \text{si } x = 0. \end{cases}$$

f n'est pas höldérienne continue, avec n'importe quel exposant. Supposons, au contraire que f est (α, h) -hölderienne continue. Pour $x, y \in]0, \frac{1}{2}]$ alors la condition (2.2) est satisfaite en particulier, pour $x = 0$, i.e.

$$|f(x)| \leq h x^\alpha, \quad \forall x \in]0, \frac{1}{2}],$$

c.à.d

$$\frac{1}{x^\alpha |\log(x)|} \leq h, \forall x \in \left]0, \frac{1}{2}\right].$$

Absurde, puisque le membre de gauche diverge si $x \rightarrow 0$.

La fonction f est uniformément continue, car sa dérivée

$$f'(x) = -\frac{1}{x \log^2(x)}, \forall x \in \left]0, \frac{1}{2}\right], f'(0) = 0,$$

est continue sur $\left[0, \frac{1}{2}\right]$.

Proposition 2.4 [18] Soit f une fonction définie dans un ensemble ouvert $\Omega \subseteq \mathbb{R}^n$, si f est (α, h) -höldérienne continue dans tout ensemble fermé contenu dans Ω . Alors f n'est pas nécessairement (α, h) -höldérienne continue dans Ω .

Proposition 2.5 Si f est une fonction (α, h) -höldérienne continue et g est (α, h') -höldérienne continue sur X , alors $f + g$ est (α, h'') -höldérienne continue sur X , et

$$h'' \leq h + h'. \quad (2.5)$$

Preuve. Soient $x, y \in X$,

$$\begin{aligned} |(f+g)(x) - (f+g)(y)| &\leq |f(x) - f(y)| + |g(x) - g(y)| \\ &\leq h|x-y|^\alpha + h'|x-y|^\alpha \\ &= (h+h')|x-y|^\alpha. \end{aligned}$$

■

Proposition 2.6 Si les fonctions scalaires f et g sont bornées telles que f est (α, h) -höldérienne continue et g est (α, h') -höldérienne continue sur X , alors fg est (α, h'') -höldérienne continue, et

$$h'' \leq h \sup_X |g| + h' \sup_X |f|. \quad (2.6)$$

Preuve. Soient $x, y \in X$, on a

$$\begin{aligned} |fg(x) - fg(y)| &= |f(x)g(x) - f(y)g(x) + f(y)g(x) - f(y)g(y)| \\ &\leq |g(x)| |f(x) - f(y)| + |f(y)| |g(x) - g(y)| \\ &\leq h \sup_X |g| |x-y|^\alpha + h' \sup_X |f| |x-y|^\alpha. \end{aligned}$$

■

Proposition 2.7 Si les fonctions scalaires $f_1, f_2, \dots, f_k$, $k > 1$, sont bornés et (β, h_i) -höldériennes continues, alors leur produit est $(\beta, \bar{h})$ -höldériennes continue, et

$$C = \max_{i \in \{1, 2, \dots, k\}} \sup |f_i|,$$

et on a

$$\bar{h} \leq k C^{k-1} \max_{i \in \{1, 2, \dots, k\}} h_i. \quad (2.7)$$

Preuve. De (2.6) nous avons

$$h'' \leq C(h_1 + h_2) \leq 2C \max_{i=1,2} h_i,$$

et donc l'assertion (2.7) suit facilement par induction. ■

Proposition 2.8 Si g une fonction (α, h) -höldérienne continue sur X et telle que $\inf |g| > 0$ sur X , alors la fonction $\frac{1}{g}$ est aussi (α, h') -höldérienne continue sur X , et

$$h' \leq \frac{h}{(\inf |g|)^2}. \quad (2.8)$$

Preuve. Pour $x, y \in X$, on a

$$\left| \frac{1}{g}(x) - \frac{1}{g}(y) \right| = \left| \frac{1}{g(x)} - \frac{1}{g(y)} \right| = \left| \frac{g(y) - g(x)}{g(x)g(y)} \right| \leq \frac{h}{(\inf |g|)^2} |x - y|^\alpha.$$

■

Proposition 2.9 Soit f, g deux fonctions bornées et $\inf |g| > 0$. Si f est (α, h) -höldérienne continue et g est (α, h') -höldérienne continue sur X , alors f/g est (α, h'') -höldérienne continue sur X et

$$h'' \leq \frac{h \sup |g| + h' \sup |f|}{\inf |g|^2}. \quad (2.9)$$

Preuve. L'assertion (2.9) découle de la proposition 2.6 et la proposition 2.8 avec g est remplacée par $1/g$. ■

Proposition 2.10 Si les fonctions vectorielles $f = (f_1, \dots, f_n)$, $g = (g_1, \dots, g_n)$ sont bornés, telles que f est (α, h) -höldérienne continue sur X et g est (α, h') -höldérienne continue sur X , alors leur produit scalaire $f \times g$ est (α, h'') -höldérienne continue sur X , et

$$h'' \leq n(h \sup |g| + h' \sup |f|). \quad (2.10)$$

Preuve. D'après la proposition 2.6, pour tout $i \in \{1, \dots, n\}$, la fonction $f_i g_i$ est (α, h_i'') -höldérienne continue sur X , et donc par la proposition 2.5, la fonction $f_1 g_1 + \dots + f_n g_n = f \times g$ est (α, h'') -höldérienne continue. De plus, par (2.5) puis par (2.6) appliqué à $f_i g_i$,

$$h'' = \sum_{i=1}^n h_i'' \leq \sum_{i=1}^n (h_i \sup |g_i| + h_i' \sup |f_i|).$$

À partir de la première inégalité de (2.3) et $|f_i| \leq |f|$, $|g_i| \leq |g|$ on obtient (2.10). ■

Proposition 2.11 *Si g une fonction vectorielle (β, h') -höldérienne continue sur X et f est (α, h) -höldérienne continue sur $Y = g(X)$, alors la composition $f \circ g$ est $(\alpha\beta, h'')$ -höldérienne continue sur X , et*

$$h'' \leq h(h')^\alpha. \quad (2.11)$$

Preuve. Pour $x, y \in X$, nous avons

$$|f \circ g(x) - f \circ g(y)| \leq h |g(x) - g(y)|^\alpha \leq h(h')^\alpha |x - y|^\alpha.$$

■

2.4 Méthodes d'optimisation globale höldériennes unidimensionnelles.

L'idée principale dans la plupart des approches d'optimisation globale déterministe, lors de la minimisation d'une fonction non convexe f est de construire une fonction borne inférieure U à partir d'une fonction objectif f sur un domaine faisable. Les bornes inférieures sont généralement obtenues pour relaxer le problème non convexe d'origine de sorte que le problème relaxé soit convexe. Ces dernières années, le concept de fonctions borne inférieure affines par morceaux a été introduit pour développer des algorithmes d'optimisation globale [62]. Rappelons qu'une fonction borne inférieure d'une fonction f définie sur l'intervalle $[a, b]$, est une fonction $U : [a, b] \rightarrow \mathbb{R}$, telle que

$$U(x) \leq f(x), \quad \text{pour tout } x \in [a, b],$$

avec la possibilité que f n'atteigne U en aucun point de l'intervalle $[a, b]$. Bien qu'il semble très difficile d'obtenir une telle fonction U pour une fonction donnée, nous donnerons quelques procédures pour construire U pour une classe de fonction de Hölder sur l'intervalle $[a, b]$. La procédure reste valable pour les fonctions de Hölder définies sur des rectangles de $\mathbb{R}^n$. On observe qu'à partir de la condition de Hölder (2.2), si un point $y \in [a, b]$ est fixé, alors on a

$$U(x) = f(y) - h |x - y|^\alpha \leq f(x), \quad \forall x \in [a, b],$$

c'est-à-dire que U est une fonction borne inférieure pour f sur $[a, b]$.

La méthode d'optimisation globale déterministe unidimensionnelle que nous allons utiliser pour résoudre le problème (2.1) avec la condition (2.2), est celle de Piyavskii [45] et indépendamment de Shubert [55], qui est la plus connue et la plus discutée dans la littérature parmi les méthodes de recouvrement [29][30][44][60]. Elle s'applique aux fonctions de Lipschitz sur un intervalle fermé et borné de $\mathbb{R}$ dont la constante de Lipschitz est connue [26][45][55]. Une fois la fonction U est construite, une suite de points, $x_1, x_2, \dots, x_k$ de $[a, b]$, convergeant vers le minimiseur global de f sur $[a, b]$, est générée en construisant une suite croissante de fonctions minorantes par morceaux de la fonction objectif f . En effet, à l'initialisation x_1 est choisi arbitrairement dans l'intervalle $[a, b]$. On définit $U_1(x)$, l'approximation convexe par morceaux par :

$$U_1(x) = f(x_1) - h |x - x_1|^\alpha \leq f(x), \quad \forall x \in [a, b].$$

La fonction $U_1(x)$ est représentée par deux courbes paraboliques. Le point suivant x_2 est choisi tel que $x_2 = \operatorname{argmin} U_1([a, b])$. En continuant ainsi, ayant choisi x_k , nous définissons les fonctions borne inférieure convexe par morceaux de $f(x)$ sur $[a, b]$ par

$$U_k(x) = \max_{i=1, \dots, k} \{f(x_i) - h |x - x_i|^\alpha\},$$

et choisissez x_k pour que $x_k = \operatorname{argmin} U_{k-1}([a, b])$. Le critère d'arrêt pour cet algorithme est quand on a $f(x_{opt}) - U(x_{opt}) \leq \varepsilon$, où x_{opt} est le minimiseur global optimal de la fonction borne inférieure par morceaux U de la fonction objectif f sur $[a, b]$ et ε est la précision requise. Il est facile de montrer que dans ces conditions tout point d'accumulation de la suite (x_k) résout le problème (2.1), (2.2). La restriction de la borne inférieure U_k sur chaque sous-intervalle $[x_{i-1}, x_i]$, $i = 2, \dots, k$ de l'intervalle $[a, b]$, peut être donné comme

$$U_i(x) = \max_i \{p_{i-1}(x), p_i(x)\},$$

où

$$p_{i-1}(x) = f(x_{i-1}) - h(x - x_{i-1})^\alpha \quad \text{et} \quad p_i(x) = f(x_i) - h(x_i - x)^\alpha.$$

La fonction $U_i(x)$ est convexe et non différentiable et sa valeur minimale globale peut être calculée en déterminant le point d'intersection des deux courbes paraboliques p_{i-1} et p_i (voir Fig.2.1). Puis à chaque itération on doit résoudre sur $[a, b]$, une équation algébrique non-linéaire :

$$p_{i-1}(x) - p_i(x) = 0. \quad (2.12)$$

Généralement, il est difficile de trouver la solution explicite de l'équation (2.12), en particulier dans le cas où le paramètre α est suffisamment petit ou n'est pas donné comme $\alpha = 1/\beta$, $\beta \in \mathbb{N}^*$, $\beta \neq 1$. On peut même dire que l'équation (2.12) présente la même difficulté qu'un problème d'optimisation locale [39].

L'étude de la maximisation globale des fonctions de Hölder d'une et de plusieurs variables respectivement, sur un intervalle de $\mathbb{R}$ et un rectangle de $\mathbb{R}^n$ a été donnée pour la première fois par Gourdin et al.[21]. Les auteurs étendent l'algorithme initialement conçu par Piyavskii et Shubert aux cas unidimensionnel et multidimensionnel. La technique utilisée dans [21], dans le cas d'une seule variable, est basée sur la résolution à chaque itération de l'algorithme, l'équation non linéaire (2.12) dans le cas où $\alpha = 1/\beta$ et β sont les entiers 2, 3 et 4 et h est donné, car il est facile d'obtenir l'expression analytique du point d'intersection de $p_{i-1}(x)$ et $p_i(x)$ sur $[x_{i-1}, x_i]$. On note *GPA* l'algorithme donné par ces derniers auteurs.

2.4.1 Expressions explicites du point d'intersection des courbes paraboliques

Dans cette sous section, l'algorithme *GPA* nécessite la détermination de l'unique point d'intersection de deux courbes paraboliques sur un intervalle $[x_{i-1}, x_i]$ définies par

$$p_{i-1}(x) \begin{cases} y = f(x_{i-1}) - h(x - x_{i-1})^\alpha \\ x \geq x_{i-1}. \end{cases} \quad \text{et} \quad p_i(x) \begin{cases} y = f(x_i) - h(x_i - x)^\alpha. \\ x \leq x_i. \end{cases}$$

D'après la définition des courbes paraboliques $p_{i-1}(x)$ et $p_i(x)$ sur $[x_{i-1}, x_i]$ et si on pose $\alpha = 1/\beta$, alors

$$\begin{aligned} (f(x_{i-1}) - y)^\beta &= h^\beta (x - x_{i-1}), \text{ Si } y \leq f(x_{i-1}) \\ (f(x_i) - y)^\beta &= h^\beta (x_i - x), \text{ Si } y \leq f(x_i), \end{aligned}$$

donc le point d'intersection de p_{i-1} et p_i vérifie :

$$\begin{cases} (f(x_{i-1}) - y)^\beta = h^\beta (x - x_{i-1}) \\ (f(x_i) - y)^\beta = h^\beta (x_i - x) \\ \text{pour } y \leq \min \{f(x_{i-1}), f(x_i)\}. \end{cases} \quad (2.13)$$

Posons $\mu = \frac{f(x_{i-1}) + f(x_i)}{2} - y$, $\sigma = \frac{f(x_{i-1}) - f(x_i)}{2}$. Le système (2.13) devient :

$$\begin{cases} (\mu + \sigma)^\beta = h^\beta (x - x_{i-1}) \\ (\mu - \sigma)^\beta = h^\beta (x_i - x) \\ \mu \geq |\sigma|. \end{cases} \quad (2.14)$$

On additionnant les deux équations de (2.14), on obtient

$$(\mu + \sigma)^\beta + (\mu - \sigma)^\beta = h^\beta (x_i - x_{i-1}). \quad (2.15)$$

En utilisant la formule de binôme de Newton, l'équation (2.15) peut s'écrire comme suit :

$$\sum_{k=0}^{\beta} C_{\beta}^k \mu^{\beta-k} \sigma^k + \sum_{k=0}^{\beta} C_{\beta}^k \mu^{\beta-k} (-1)^k \sigma^k = h^\beta (x_i - x_{i-1}),$$

d'où

$$\sum_{k=0}^{\beta} C_{\beta}^k \mu^{\beta-k} (1 + (-1)^k) \sigma^k = h^\beta (x_i - x_{i-1}).$$

Comme $1 + (-1)^k = 2$ pour $k = 2i$ et $1 + (-1)^k = 0$ pour $k = 2i + 1$, on a :

$$2 \sum_{i=0}^{Ent(\beta/2)} C_{\beta}^{2i} \mu^{\beta-2i} \sigma^{2i} = h^\beta (x_i - x_{i-1}),$$

$Ent(.)$ désigne la partie entière d'un nombre réel.

En substituant μ et σ par leurs expressions, on obtient

$$2 \sum_{i=0}^{Ent(\beta/2)} C_{\beta}^{2i} \left(\frac{f(x_{i-1}) + f(x_i)}{2} \right)^{\beta-2i} \left(\frac{f(x_{i-1}) - f(x_i)}{2} \right)^{2i} = h^{\beta} (x_i - x_{i-1}). \quad (2.16)$$

De la même façon, les deux équations de (2.14), on tire :

$$\sum_{k=0}^{\beta} C_{\beta}^k \mu^{\beta-k} (1 - (-1)^k) \sigma^k = h^{\beta} (2x - x_i - x_{i-1}).$$

Comme $1 + -(-1)^k = 2$ pour $k = 2i + 1$ et $1 + (-1)^k = 0$ pour $k = 2i$, on a :

$$2 \sum_{i=0}^{Ent(\beta-1/2)} C_{\beta}^{2i+1} \mu^{\beta-2i-1} \sigma^{2i+1} = h^{\beta} (2x - x_i - x_{i-1}).$$

En substituant μ et σ par leurs expressions, on obtient

$$\begin{aligned} 2 \sum_{i=0}^{Ent(\beta-1/2)} C_{\beta}^{2i+1} \left(\frac{f(x_{i-1}) + f(x_i)}{2} - y \right)^{\beta-2i-1} \left(\frac{f(x_{i-1}) - f(x_i)}{2} \right)^{2i+1} \\ = \frac{h^{\beta}}{2} (2x - x_i - x_{i-1}). \end{aligned} \quad (2.17)$$

Alors le point d'intersection des courbes P_{i-1} et P_i vérifie les deux équations (2.16) et (2.17).

Lemme 2.1 Soit μ la solution unique de l'équation :

$$\sum_{i=0}^{Ent(\beta/2)} C_{\beta}^{2i} \mu^{\beta-2i} \sigma^{2i} = h^{\beta} \left(\frac{x_i - x_{i-1}}{2} \right) \text{ telle que } \mu \geq |\sigma|.$$

Alors les coordonnées $\mathbf{x}_i, \mathbf{y}_i$ de l'unique point intersection de P_{i-1} et P_i sont données analytiquement par :

$$\begin{cases} \mathbf{x}_i = \frac{x_i + x_{i-1}}{2} + \frac{1}{h^{\beta}} \sum_{i=0}^{Ent(\beta-1/2)} C_{\beta}^{2i+1} \mu^{\beta-2i-1} \sigma^{2i+1} \\ \mathbf{y}_i = \frac{f(x_i) + f(x_{i-1})}{2} - \mu \end{cases} \quad (2.18)$$

Les résultats suivants, donnent les expressions explicites [21], du point d'intersection $(\mathbf{x}_i, \mathbf{y}_i)$ des courbes $p_{i-1}(x)$ et $p_i(x)$, pour certains cas de $\beta \in \mathbb{N}$. (voir Fig.2.1)

Proposition 2.12 Supposons que la fonction f est höldérienne de constante h et d'exposant $\alpha = 1/\beta$ avec $\beta = 2$, alors le point $(\mathbf{x}_i, \mathbf{y}_i)$ est donné par :

$$\begin{cases} \mathbf{x}_i = \frac{x_{i-1} + x_i}{2} + \frac{f(x_{i-1}) - f(x_i)}{2h^2} \sqrt{2h^2(x_i - x_{i-1}) - (f(x_{i-1}) - f(x_i))^2} \\ \mathbf{y}_i = \frac{f(x_{i-1}) + f(x_i)}{2} + \frac{1}{2} \sqrt{2h^2(x_i - x_{i-1}) - (f(x_{i-1}) - f(x_i))^2}. \end{cases}$$

Proposition 2.13 Pour $\beta = 3$ le point $(\mathbf{x}_i, \mathbf{y}_i)$ est donné par :

$$\begin{cases} \mathbf{x}_i = \frac{x_{i-1} + x_i}{2} + \frac{f(x_{i-1}) - f(x_i)}{8h^3} (3A^{2/3} + 3B^{2/3} - 5(f(x_{i-1}) - f(x_i))^2) \\ \mathbf{y}_i = \frac{f(x_{i-1}) + f(x_i)}{2} + \frac{A^{1/3} + B^{1/3}}{2}, \end{cases}$$

où

$$\begin{cases} A = 2h^3(x_i - x_{i-1}) + \sqrt{(f(x_{i-1}) - f(x_i))^6 + 4h^6(x_i - x_{i-1})^2} \\ B = 2h^3(x_i - x_{i-1}) - \sqrt{(f(x_{i-1}) - f(x_i))^6 + 4h^6(x_i - x_{i-1})^2}. \end{cases}$$

Proposition 2.14 Pour $\beta = 4$ le point $(\mathbf{x}, \mathbf{y})$ est donné par :

$$\begin{cases} \mathbf{x}_i = \frac{x_{i-1} + x_i}{2} + \left(\frac{4\sigma}{h^4}\right) \left(\sqrt{\frac{h^4(x_i - x_{i-1})}{2} + 8\sigma^4} - 2\sigma^2\right) \sqrt{\sqrt{\frac{h^4(x_i - x_{i-1})}{2} + 8\sigma^4} - 3\sigma^2} \\ \mathbf{y}_i = \frac{f(x_{i-1}) + f(x_i)}{2} + \sqrt{\sqrt{\frac{h^4(x_i - x_{i-1})}{2} + 8\sigma^4} - 3\sigma^2}, \end{cases}$$

où

$$\sigma = \frac{f(x_{i-1}) - f(x_i)}{2}.$$

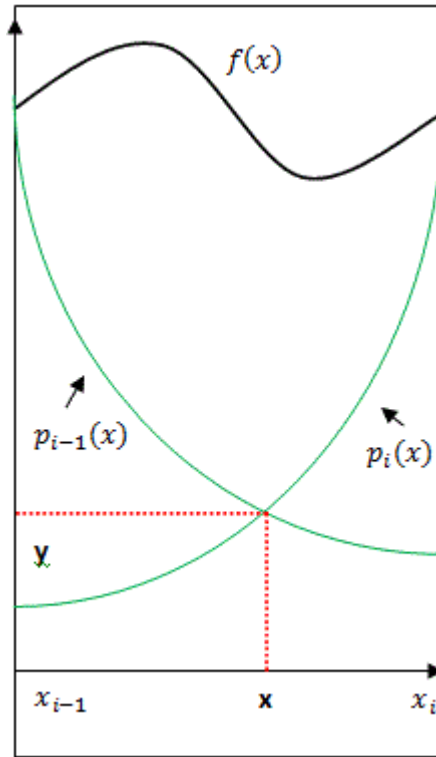


FIGURE 2.1 – Construction des fonctions minorantes paraboliques

Algorithme 5 *GPA*

- 1: Initialisation
 - 2: $k \leftarrow 1$, $x_1 \leftarrow \frac{a+b}{2}$, $x_\varepsilon \leftarrow x_1$, $f_\varepsilon = f(x_\varepsilon)$.
 - 3: $U_\varepsilon = f_\varepsilon - h(\frac{b-a}{2})^\alpha$, $U_1(x) = f(x_1) - h|x - x_1|^\alpha$
 - 4: **while** $f_\varepsilon - U_\varepsilon > \varepsilon$ **do**
 - 5: $x_1, x_2, \dots, x_k$ sont ordonnée $a = x_1 < x_2 < \dots < x_k = b$.
 - 6: $x_{k+1} = \underset{x \in [a, b]}{\operatorname{argmin}} U_k(x)$
 - 7: **if** $f(x_{k+1}) \leq f_\varepsilon$ **then**
 - 8: $x_\varepsilon = x_{k+1}$, $f_\varepsilon = f(x_{k+1})$
 - 9: **end if**
 - 10: Poser : $U_{k+1} = \max_{1 \leq i \leq k+1} (f(x_i) - h|x - x_i|^\alpha)$ et $U_\varepsilon = \min_{x \in [a, b]} U_{k+1}(x)$
 - 11: $k = k + 1$.
 - 12: **end while**
-

2.4.2 Technique de la sécante

Une autre idée a été utilisée par Lera et Sergeyev [36] (notée par *SPA*), en remplaçant le point d'intersection donnée par Gourdin et al.[21] par un point d'intersection y_i de deux interpolations linéaires l_{i-1} et l_i liées respectivement des paraboles p_{i-1} et p_i sur chaque sous-intervalle $[x_{i-1}, x_i]$ de $[a, b]$ (voir Fig.2.2) . On a :

$$\begin{aligned} l_{i-1} &= f(x_{i-1}) - h(x_i - x_{i-1})^{\alpha-1} \cdot x + h(x_i - x_{i-1})^{\alpha-1} \cdot x_{i-1} \\ l_i &= f(x_i) + h(x_i - x_{i-1})^{\alpha-1} \cdot x - h(x_i - x_{i-1})^{\alpha-1} \cdot x_i. \end{aligned}$$

Dans ce cas le point y_i peut être calculé même si α est assez petit ou n'est pas écrit sous la forme $\alpha = 1/\beta$, $\beta = 2, 3$, et 4, par

$$y_i = \frac{x_i + x_{i-1}}{2} - \frac{f(x_i) - f(x_{i-1})}{2h(x_i - x_{i-1})^{\alpha-1}}, \quad 2 \leq i \leq k. \quad (2.19)$$

Ensuite, ils déterminent la valeur minimale $\mathbf{m}_i$ comme borne inférieure constante de la fonction objectif f sur $[x_{i-1}, x_i]$ qui est donnée par

$$\mathbf{m}_i = \min \{p_{i-1}(y_i), p_i(y_i)\}.$$

Proposition 2.15 *Considérons le sous-estimateur $U_i(x)$ et le nombre réel $\mathbf{m}_i$ définie précédemment sur chaque intervalle $[x_{i-1}, x_i]$. Alors*

$$\mathbf{m}_i < f(x), \quad \forall x \in [x_{i-1}, x_i].$$

Preuve. On sait que $U_i(x) = \max \{p_{i-1}(x), p_i(x)\}$ est un sous-estimateur de f sur l'intervalle $[x_{i-1}, x_i]$, donc,

$$U_i(x) \leq f(x), \quad x \in [x_{i-1}, x_i].$$

Puisque la fonction $p_{i-1}(x)$ est strictement décroissante sur $[x_{i-1}, x_i]$ et $p_i(x)$ est strictement croissante sur $[x_{i-1}, x_i]$, il vient,

$$\min \{p_{i-1}(x), p_i(x)\} \leq \min \{U_i(x), x \in [x_{i-1}, x_i]\}, \quad \forall x \in [x_{i-1}, x_i].$$

En particulier, pour $x = y_i$ où y_i est donné par (2.19) on a :

$$\mathbf{m}_i = \min \{p_{i-1}(y_i), p_i(y_i)\} \leq \min \{U_i(y_i), y_i \in [x_{i-1}, x_i]\} < f(x), \quad \forall x \in]x_{i-1}, x_i[.$$

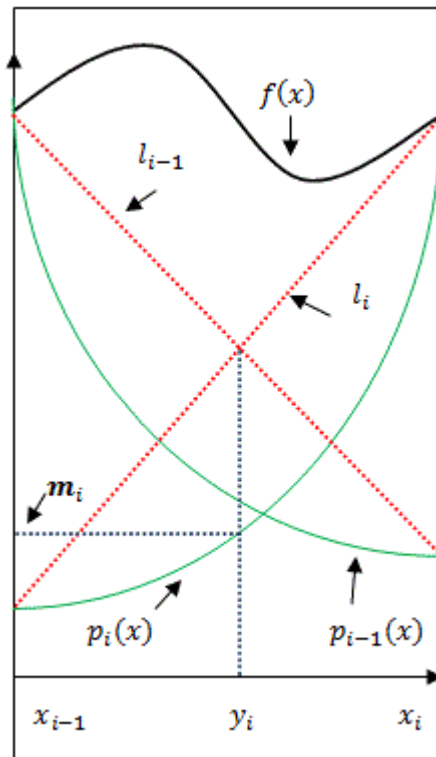


FIGURE 2.2 – Construction des interpolations linéaires liées aux paraboles sur un sous-intervalle $[x_{i-1}, x_i]$

■

Algorithme 6 SPA

```
1: Initialisation
2:  $k \leftarrow 2, x_1 \leftarrow a, x_2 \leftarrow b.$ 
3: Etape  $k$  :  $x_1, x_2, \dots, x_k$  sont ordonné tel que  $a = x_1 < x_2 < \dots < x_k = b.$ 
4: for  $i = 2$  to  $k$  do
5:    $y_i = \frac{x_i + x_{i-1}}{2} - \frac{f(x_i) - f(x_{i-1})}{2h(x_i - x_{i-1})^{\alpha-1}}$ 
6:    $\mathbf{m}_i = \min \{f(x_{i-1}) - h(y_i - x_{i-1})^\alpha, f(x_i) - h(x_i - y_i)^\alpha\}$ 
7: end for
8:  $\mathbf{m}_s = \min \{\mathbf{m}_i, 2 \leq i \leq k\}$ 
9:  $x_s = y_s$ 
10: if  $|x_s - x_{s-1}| > \varepsilon$  then
11:    $x_{k+1} = y_s$ 
12:    $k \leftarrow k + 1$ 
13:   aller à l'Etape  $k$ 
14: else
15:    $f_{opt} = \min \{f(x_i) : 1 \leq i \leq k\}$ 
16:   Stop
17: end if
18: retourner  $f_{opt}$ 
```

Remarque 2.1 Dans [36], les auteurs envisagent également certaines techniques si la constante de Hölder h n'est pas connue a priori.

2.4.3 Méthode de Piyavskii modifiée MPA

Dans [47] et [50], les auteurs ont proposé de nouvelles extensions de l'algorithme de Piyavskii pour minimiser les fonctions de Hölder d'une et plusieurs variables. Ils n'ont pas besoin de résoudre à chaque itération l'équation (2.12). Le principe est de construire une suite de fonctions minorants affines par morceaux de f sur $[a, b]$, similaire à la procédure précédente, même si l'exposant de Hölder α appartient à $]0, 1[\setminus \{1/\beta, \beta \in \mathbb{N}^*\}$. L'idée principale repose essentiellement sur un résultat démontré dans [47] pour des fonctions continues sur un com-

compact C de $\mathbb{R}^n$. Une autre version de ce résultat a été donnée par R. J. Vanderbei dans [61] où l'on suppose cependant que C est convexe ce qui n'est pas nécessaire si C est compact. Mais l'inconvénient de ce résultat est que la constante L_ε dans [47] ou [61], est généralement inconnue. Nous donnons ci-dessous le résultat suivant pour le cas des fonctions höldériennes où L_ε est une constante explicitement donnée par rapport à h, α et ε .

Théorème 2.1 [61] *Une fonction réelle définie sur un ensemble compact C de $\mathbb{R}$ est continue, si et seulement si, pour chaque $\varepsilon > 0$ il existe une constante $L_\varepsilon > 0$ telle que :*

$$|f(x) - f(y)| \leq L_\varepsilon |x - y| + \varepsilon, \quad \forall x, y \in C. \quad (2.20)$$

Théorème 2.2 *Soit f une fonction höldérienne réelle de constantes $h > 0$ et $0 < \alpha < 1$ définie sur l'intervalle $[a, b] \subset \mathbb{R}$, alors pour tout $\varepsilon > 0$, on a :*

$$|f(x) - f(y)| \leq L_\varepsilon |x - y| + \varepsilon, \quad \forall x, y \in [a, b], \quad (2.21)$$

avec

$$L_\varepsilon = \frac{\alpha \varepsilon}{1 - \alpha} \left(\frac{h(1 - \alpha)}{\varepsilon} \right)^{1/\alpha}.$$

Preuve. D'après l'inégalité (2.20) on a :

$$\forall \varepsilon > 0 \text{ et } \forall x, y \in [a, b], \quad L_\varepsilon \geq \frac{|f(x) - f(y)| - \varepsilon}{|x - y|}.$$

Donc on peut prendre :

$$L_\varepsilon = \sup_{x, y \in [a, b]} \frac{h|x - y|^\alpha - \varepsilon}{|x - y|} \geq \sup_{x, y \in [a, b]} \frac{|f(x) - f(y)| - \varepsilon}{|x - y|}.$$

Posons, $|x - y| = t$ et considérons la fonction $\mathbf{g}(t) = \frac{ht^\alpha - \varepsilon}{t}$ avec $t > 0$. L'équation $\mathbf{g}'(t) = 0$ admet une seule solution $t_0 = (\frac{\varepsilon}{h(1 - \alpha)})^{1/\alpha}$ et puisque $\mathbf{g}''(t) < 0$ alors

$$L_\varepsilon = \sup_{t > 0} \frac{ht^\alpha - \varepsilon}{t} = \frac{\varepsilon \alpha}{1 - \alpha} \left(\frac{h(1 - \alpha)}{\varepsilon} \right)^{1/\alpha}.$$

Maintenant, à partir de l'inégalité (2.21), on construit une suite croissante de fonctions minorantes affines par morceaux de $f(x)$ sur l'intervalle $[a, b]$, au lieu paraboliques, définie par

$$\mathbf{L}_k(x) = \max_{i=1, \dots, k} \{f(x_i) - L_\varepsilon |x - x_i| - \varepsilon\}.$$

Sur chaque sous-intervalle $[x_{i-1}, x_i]$ de $[a, b]$, la restriction $\mathbf{L}_i(x)$ de $\mathbf{L}_k(x)$ est une fonction convexe et non différentiable, de pentes L_ε et $-L_\varepsilon$ (voir Fig.2.3). Le minimum global de $\mathbf{L}_i(x)$ sur l'intervalle $[x_{i-1}, x_i]$ est calculé à $\hat{x}_i$ le point d'intersection de deux droites, qui se trouve facilement en résolvant une équation linéaire simple, pour toute valeur de $\alpha \in]0, 1[$.

$$f(x_i) - L_\varepsilon(x_i - x) - \varepsilon = f(x_{i-1}) - L_\varepsilon(x - x_{i-1}) - \varepsilon.$$

Il est donné explicitement pour $i = 2, \dots, k$ par

$$\begin{aligned}\hat{x}_i &= \frac{1}{2L_\varepsilon}(f(x_{i-1}) - f(x_i)) + \frac{1}{2}(x_i + x_{i-1}) \\ \mathbf{L}_i(\hat{x}_i) &= \frac{1}{2}(f(x_{i-1}) + f(x_i)) - \frac{L_\varepsilon}{2}(x_i - x_{i-1}) - \varepsilon.\end{aligned}$$

■

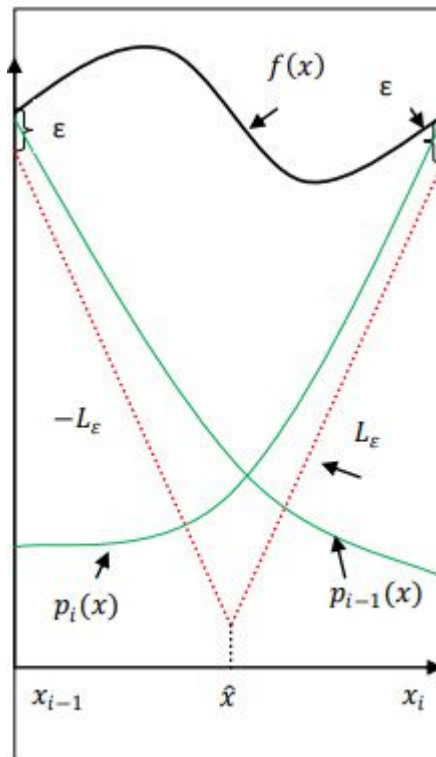


FIGURE 2.3 – Le sous-estimateur affine par morceaux de $f(x)$ sur l'intervalle $[x_{i-1}, x_i]$

Algorithme 7 MPA

```
1: Initialisation
2:  $k \leftarrow 1, x_1 \leftarrow \frac{a+b}{2}, x_\varepsilon \leftarrow x_1, f_\varepsilon = f(x_\varepsilon).$ 
3:  $\mathbf{L}_{opt} = f_\varepsilon - \frac{L_\varepsilon}{2}(b-a) - \varepsilon, \mathbf{L}_1(x) = f(x_1) - L_\varepsilon |x - x_1| - \varepsilon.$ 
4: while  $f_\varepsilon - \mathbf{L}_\varepsilon > \varepsilon$  do
5:    $x_1, x_2, \dots, x_k$  sont ordonnée  $a = x_1 < x_2 < \dots < x_k = b.$ 
6:    $x_{k+1} = \arg \min_{x \in [a, b]} \mathbf{L}_k(x)$ 
7:   if  $f(x_{k+1}) \leq f_\varepsilon$  then
8:      $x_\varepsilon = x_{k+1}, f_\varepsilon = f(x_{k+1})$ 
9:   end if
10:  Poser :  $\mathbf{L}_{k+1} = \max_{1 \leq i \leq k+1} (f(x_i) - L_\varepsilon |x - x_i| - \varepsilon), \mathbf{L}_\varepsilon = \min_{x \in [a, b]} \mathbf{L}_{k+1}(x)$ 
11:   $k = k + 1.$ 
12: end while
```

Convergence de l'algorithme de Piyavskii modifié MPA

Théorème 2.3 *La méthode de Piyavskii modifiée appliquée aux fonctions höldériennes converge en un nombre fini d'itérations vers le minimum global avec une précision inférieure ou égale à ε .*

Preuve. Désignons par f^* le minimum global exact de f sur $[a, b]$ et par f_{opt} le minimum global du problème (2.1) obtenu par la méthode de Piyavskii modifiée, montrons que :

$$f_{opt} - f^* \leq \varepsilon.$$

Supposons que l'algorithme a atteint le minimum global f_{opt} avec une précision ε après k points d'évaluations $x_1, \dots, x_k$. On a donc

$$f_{opt} = \min \{f(x_1), f_2(x), \dots, f_k(x)\},$$

et

$$f_{opt} - \min_{x \in [a, b]} \mathbf{L}_k(x) \leq \varepsilon,$$

avec

$$\mathbf{L}_k(x) = \max_{i=1,\dots,k} \{f(x_i) - L_\varepsilon |x - x_i| - \varepsilon\}.$$

Puisque

$$\mathbf{L}_k(x) \leq f(x), \forall x \in [a, b],$$

alors

$$\min_{x \in [a, b]} \mathbf{L}_k(x) \leq f^* \leq f_{opt},$$

et par conséquent

$$f_{opt} - f^* \leq f_{opt} - \min_{x \in [a, b]} \mathbf{L}_k(x) \leq \varepsilon.$$

■

Remarque 2.2 L'exécution numérique de la modification de l'algorithme de Piyavskii basée sur le résultat du théorème 2.2 dépend essentiellement de la connaissance de la constante L_ε [47]. Malgré l'avantage que fournit ce résultat pour les fonctions de Hölder, le calcul explicite de la constante L_ε en fonction de la précision ε à chaque fois que les constantes h et α sont connues, aussi, il a un inconvénient tel que si la précision ε est suffisamment petite, la constante L_ε est suffisamment grande, alors la convergence de l'algorithme est sûr, mais deviendra lente.

2.4.4 Méthode utilisant des fonctions auxiliaires

Dans cette sous section nous présentons un algorithme pour la minimisation des fonctions höldériennes. Il a été élaboré par A. Yahyaoui et al.[63]. L'idée consiste à introduire une fonction auxiliaire O_i a chaque sous interval $[x_{i-1}, x_i]$ qui permet de donner une valeur approximative de la solution exacte de équation (2.12).

Soit $x_0 = a$, définissons

$$U_0(x) = f(x_0) - h |x - x_0|^\alpha = G_0(x),$$

un sous-estimateur de f sur $[a, b]$.

La courbe G_0 a un sommet $V_1(u = b, \mathbf{H}_0)$ tel que

$$\mathbf{H}_0 = f(x_0) - h |b - x_0|^\alpha = \min_{x \in [a, b]} G_0(x).$$

Posons $x_1 = \underset{x \in [a, b]}{\operatorname{argmin}}(G_0(x)) = b$. A partir du point de coordonnées $(x_1, f(x_1))$, nous traçons la courbe du sous-estimateur :

$$U_1(x) = f(x_1) - h|x - x_1|^\alpha.$$

Définissons :

$$G_1(x) = \max(U_1(x), G_0(x)),$$

et

$$x_2 = \underset{x \in [a, b]}{\operatorname{argmin}}(G_1(x)).$$

Le sommet V_1 est remplacé par un nouveau sommet $V_1'(x_2, G_1(x_2))$, tel que x_2 donné par l'intersection des courbes de G_0 et U_1 , c.-à-d. la solution de l'équation suivante :

$$f(x_0) - h(x - x_0)^\alpha = f(x_1) - h(x_1 - x)^\alpha.$$

Comme on a dit auparavant, en général, il n'est pas facile d'avoir la valeur exacte de la solution de l'équation ci-dessus. Pour cette raison, les auteurs dans [63] introduite une fonction auxiliaire O_1 qui permet de donner une point approché de x_2 que nous notons également x_2 .(voir Fig.2.4)

Le point $V_1'(x_2, G_1(x_2))$ est entre deux points voisins appartenant à la courbe de G_1 , l'un à gauche du point V_1' notée $L(x_0, f(x_0))$ et l'autre à sa droite $R(x_1, f(x_1))$.

On désigne par :

$$\left\{ \begin{array}{l} m_1 = \min(f(x_0), f(x_1)) \\ M_1 = \max(f(x_0), f(x_1)) \\ \mu_1 = \underset{x \in [a, b]}{\operatorname{argmin}}(f(x_0), f(x_1)) \\ \varrho_1 = \underset{x \in [a, b]}{\operatorname{argmax}}(f(x_0), f(x_1)). \end{array} \right.$$

Selon la Fig.2.4 , et dans ce cas, $m_1 = f(x_1)$, $M_1 = f(x_0)$, soit $z_1 \in [x_0, x_1]$, tel que $G_1(z_1) = m_1$, et $z_1 \neq \mu_1$. Cela donne :

$$z_1 = \varrho_1 + \operatorname{sing}(\mu_1 - \varrho_1) \left(\frac{m_1 - M_1}{h} \right)^{\frac{1}{\alpha}}. \quad (2.22)$$

où

$$\text{sing}(x) = \begin{cases} 1 & \text{si } x \geq 0 \\ -1 & \text{si } x < 0 \end{cases}$$

A partir du point $L_1(z_1, m_1)$, on trace la courbe représentative de :

$$O_1(x) = \max(m_1 - h|x - z_1|^\alpha, m_1 - h|x - \mu_1|^\alpha) 1_{j_1(x)}, \quad (2.23)$$

$$\text{où } j_1 = [\min(z_1, \mu_1), \max(z_1, \mu_1)] \text{ et } 1_{j_1(x)} = \begin{cases} 1, & \text{si } x \in j_1 \\ 0, & \text{sinon.} \end{cases}$$

La nouvelle fonction

$$G'_1(x) = G_1(x) 1_{x \notin j_1} + O_1(x).$$

La courbe de G'_1 a un nouveau sommet, donné par la courbe de O_1 , définie par $V''(u_1, H'_1)$ tel que $(u_1 = \frac{z_1 + \mu_1}{2}, \mathbf{H}'_1 = m_1 - h|u_1 - z_1|^\alpha)$.

Par conséquent, le sommet V_1 sera remplacé par V'' . Le nouveau sommet de la courbe de G'_1 , maintenant désigné par V_1 , sera identifié par $(u_1, L_1, R_1, \mathbf{H}_1)$, avec $\mathbf{H}_1 = m_1 - h|u_1 - z_1|^\alpha$, où $L_1(z_1, m_1)$ et $R_1(x_1, m_1)$, sont respectivement, les voisins gauche et droit de V_1 .

Soit $x_2 = \arg\min(G'_1(x))$. Ici, $x_2 = u_1$. Alors, on a :

$$U_2(x) = f(x_2) - h|x - x_2|^\alpha$$

$$G_2(x) = \max(U_2(x), G'_1(x)).$$

Supposons que f est évaluée à $x_0, x_1, \dots, x_n$ et notons :

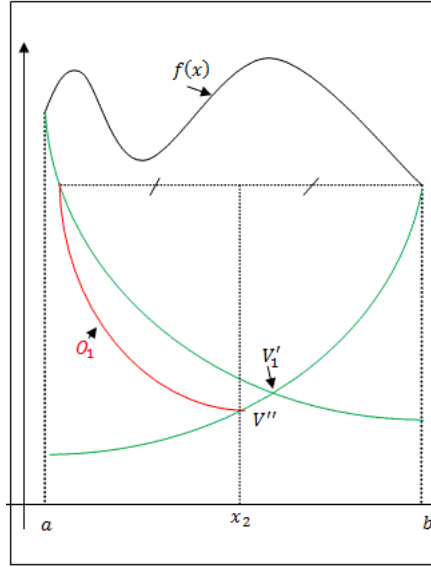


FIGURE 2.4 – Consstrucion des fonctions auxiliaires sur l'intervalle $[a, b]$

$$\varphi_n = \min(f(x_0), f(x_1), \dots, f(x_n)).$$

La courbe G'_n à n sommets : $V_0 \leq V_1 \leq \dots \leq V_n$, de telle sorte que chacun d'eux soit identifié par :

$$\left\{ \begin{array}{l} \text{Le point qui lui est voisin à gauche } L_i(l_i, m_i) \\ \text{Le point qui lui est voisin à droite } R_i(r_i, m_i) \\ \text{son abscisse } u_i = \frac{r_i + l_i}{2} \\ \text{son ordonnée } \mathbf{H}_i = m_i - h |u_i - r_i|^\alpha, \end{array} \right. \quad \text{pour } i = 1, \dots, n. \quad (2.22)$$

soient $x_{n+1} = \arg \min(G'_n(x)) = u_n$, $y_{n+1} = f(x_{n+1})$, $\varphi_{n+1} = \min(\varphi_n, f(x_{n+1}))$ et

$$U_{n+1}(x) = f(x_{n+1}) - h |x - x_{n+1}|^\alpha$$

$$G_{n+1}(x) = \max(G'_n(x), U_{n+1}(x)).$$

Pour la courbe du sous-estimateur G_{n+1} , V_n n'est plus un sommet, mais deux nouveaux sommets apparaissent de chaque côté de V_n : notés V_L (à gauche) et V_R (à droite). Pour les deux sommets V_L et V_R , il n'est pas évident de calculer leurs coordonnées. Chacun d'eux sera remplacé, respectivement, par V'_L et V'_R comme cela a été fait pour G_1 [63].

Algorithme 8 Fonctions Auxiliaires

```
1:  $x_0 = a, x_1 = b, f_{opt} = f(x_1), \mathbf{H}_{opt} = f(x_0) - h|b - a|^\alpha, k = 1.$ 
2: while  $f_{opt} - \mathbf{H}_{opt} \geq \varepsilon$  do
3:   for  $i = 1$  to  $k$  do
4:      $x_0, x_1, \dots, x_k$  sont ordonné tel que  $a = x_0 \leq x_1 \leq \dots \leq x_k = b$ 
5:      $m_i = \min(f(x_{i-1}), f(x_i)), \mu_i = \arg \min(f(x_{i-1}), f(x_i))$ 
6:      $M_i = \max(f(x_{i-1}), f(x_i)), \varrho_i = \arg \max(f(x_{i-1}), f(x_i))$ 
7:      $z_i = \varrho_i + \sin g(\mu_i - \varrho_i) \left( \frac{m_i - M_i}{h} \right)^\alpha$ 
8:      $O_i(x) = \max(m_i - h|x - z_i|^\alpha, m_i - h|x - \mu_i|^\alpha) 1_{j_i(x)}$ , où  $j_1 = [\min(z_i, \mu_i), \max(z_i, \mu_i)]$ 
9:      $u_i = \arg \min O_i(x).$ 
10:     $\mathbf{H}_i = m_i - h|\mu_i - u_i|^\alpha$ 
11:   end for
12:    $\mathbf{H}_{opt} = \min \{\mathbf{H}_i, 1 \leq i \leq k\}, x_{k+1} = u_{opt}$ 
13:   if  $f(x_{k+1}) \leq f_{opt}$  then
14:      $x_{opt} = x_{k+1}, f_{opt} = f(x_{k+1})$ 
15:   end if
16:    $k = k + 1$ 
17: end while
```

2.4.5 Théorème de convergence

Théorème 2.4 Soit f une fonction (h, α) -holdérienne définie sur l'intervalle $[a, b]$. La suite $(\mathbf{H}_n)_{n \in \mathbb{N}}$, définie ci-dessus, accroît au minimum de f .

Preuve. Notons par $\varphi = \min_{x \in [a, b]} f(x)$ et $\Phi = \{x \in [a, b], f(x) = \varphi\}$, $\forall x \in [a, b], G_n(x) \leq f(x)$ pour tout $n \in \mathbb{N}$. Cela implique que :

$$\mathbf{H}_n = \min_{x \in [a, b]} G_n(x) \leq \min_{x \in [a, b]} f(x) = \varphi.$$

Comme $\varphi_{n+1} = \min(f(x_0), \dots, f(x_{n+1}))$ et $m_n = \min(f(x_n), m_{n-1})$, on déduit que

$$\mathbf{H}_{n+1} \leq \varphi_{n+1} \leq m_{n+1},$$

et comme on a $x_{n+1} \in [x_{L_{n+1}}, x_{R_{n+1}}]$ d'où :

$$\begin{aligned} |\mathbf{H}_{n+1} - \varphi_{n+1}| &\leq |\mathbf{H}_{n+1} - m_{n+1}| \leq h |x_{L_{n+1}} - x_{n+1}|^\alpha \leq h \left| \frac{1}{2}(x_{L_{n+1}} - x_{R_{n+1}}) \right|^\alpha \\ &\leq h \left| \frac{1}{2^2}(x_{L_n} - x_{R_n}) \right|^\alpha \leq h \left| \frac{1}{2^n}(x_{L_1} - x_{R_1}) \right|^\alpha, \end{aligned}$$

qui tend vers à 0.

Comme (φ_n) est une suite bornée décroissante, elle converge. Supposons que φ_n converge vers $\mu \neq \varphi$. Comme $f([a, b])$ est un compact, $\mu \in f([a, b])$. Soit $y_n \in [a, b]$, tel que $\varphi_n = f(y_n)$. Comme $[a, b]$ est un compact, la suite (y_n) admet une sous-suite (y_{n_k}) qui converge vers z dans $[a, b]$. La continuité de f implique que $f(z) = \mu$.

Soit $\varepsilon = \mu - \varphi$. Puisque (y_{n_k}) converge vers z , $\exists K, \forall k \geq K, |y_{n_k} - z| < \left(\frac{\varepsilon}{2h}\right)^{\frac{1}{\alpha}}$.

La condition de Hölder implique que :

$$|f(y_{n_k}) - f(z)| \leq h |y_{n_k} - z|^\alpha \leq \frac{\varepsilon}{2},$$

c'est à dire $\forall k \geq K$:

$$\varphi_{n_k} = f(y_{n_k}) \geq f(z) - \frac{\varepsilon}{2} = \mu - \frac{\varepsilon}{2},$$

d'autre part $\forall n \geq n_k$:

$$G_n(x) \geq G_{n_k}(x) = f(y_{n_k}) - h |x - y_{n_k}|^\alpha = \varphi_{n_k} - h |x - y_{n_k}|^\alpha.$$

D'où, $\forall x \in [a, b]$ et $\forall n \geq n_k$ tel que $|x - y_{n_k}| < \left(\frac{\varepsilon}{2h}\right)^{1/\alpha}$, on a

$$G_n(x) \geq \varphi_{n_k} - \frac{\varepsilon}{2} \geq \mu - \frac{\varepsilon}{2} - \frac{\varepsilon}{2} = \varphi.$$

Le nombre réel $z \in]y_{n_k} - \left(\frac{\varepsilon}{2h}\right)^{1/\alpha}, y_{n_k} + \left(\frac{\varepsilon}{2h}\right)^{1/\alpha}[$, alors $\exists j > k$,

$$|y_{n_j} - y_{n_k}| < \left(\frac{\varepsilon}{2h}\right)^{1/\alpha}.$$

Cela signifie que $\mathbf{H}_{n_j} = G_{n_k}(y_{n_k}) > \varphi$. Ceci est absurde. Alors (y_{n_k}) converge vers φ . ■

2.4.6 Fonctions borne inférieures tangentes par morceaux.

Ici, nous donnons d'autres procédures intéressantes pour surmonter certaines difficultés présentées dans la section précédente, et accélérer la convergence de l'algorithme proposé vers le minimum global [10]. En effet, la première idée repose sur l'utilisation des approximations linéaires des fonctions paraboliques $p_{i-1}(x)$ et $p_i(x)$ sur chaque sous-intervalle

$[x_{i-1}, x_i]$ de $[a, b]$ précédemment donné. Il est bien connu que toute fonction différentiable en un point c_i dans un intervalle ouvert $]x_{i-1}, x_i[$, elle peut être approchée près du point c_i par une fonction affine (polynôme de Taylor de degré 1). C'est l'approximation polynomiale de degré 1 au voisinage d'un point et la meilleure fonction affine qui approche le graphe d'une fonction convexe f en tout point est la tangente. Comme les deux fonctions $p_{i-1}(x)$ et $p_i(x)$ sont convexes et différentiables sur l'intervalle $]x_{i-1}, x_i[$, alors chacune des fonctions sera approchée par une fonction affine sur l'intervalle $]x_{i-1}, x_i[$. Soit c_i un point quelconque de $[x_{i-1}, x_i]$, les deux fonctions affines que nous allons utiliser sont les tangentes aux graphes des fonctions $p_{i-1}(x)$ (resp. $p_i(x)$) au même point c_i . L'utilisation des tangentes nous montrera l'efficacité, bien sûr, avec un bon choix de c_i . Soit $c_i = \frac{x_{i-1}+x_i}{2}$ le milieu de l'intervalle $[x_{i-1}, x_i]$, les deux fonctions tangentes de $p_{i-1}(x)$ et $p_i(x)$ au même point milieu c_i seront notés respectivement :

$$\begin{aligned} T_{i-1}(x) &= p'_{i-1}(c_i)(x - c_i) + p_{i-1}(c_i) \\ T_i(x) &= p'_i(c_i)(x - c_i) + p_i(c_i), \end{aligned}$$

où p'_{i-1} et p'_i désignent les fonctions dérivées (voir Fig.2.5). Soit $m_i = \frac{x_i - x_{i-1}}{2}$, en utilisant les expressions de $p_{i-1}(x)$ et $p_i(x)$ pour $x \in [x_{i-1}, x_i]$, on obtient :

$$\begin{cases} T_{i-1}(x) = -ham_i^{\alpha-1}x + hm_i^\alpha(\frac{\alpha c_i}{m_i} - 1) + f(x_{i-1}), \\ T_i(x) = ham_i^{\alpha-1}x - hm_i^\alpha(\frac{\alpha c_i}{m_i} + 1) + f(x_i). \end{cases} \quad (2.23)$$

Proposition 2.16 Soit $T_{i-1}(x)$ et $T_i(x)$ soit les deux fonctions données par (2.23). Définissons une fonction g_i comme suit :

$$g_i(x) = \max\{T_{i-1}(x), T_i(x)\}, \quad i = 1, \dots, n.$$

Alors $g_i(x) < f(x)$ pour $x \in [x_{i-1}, x_i]$ et l'unique minimum global de $g_i(x)$ est donné au point z_i , qui est donné par :

$$z_i = \underset{[x_{i-1}, x_i]}{\operatorname{argmin}}_i(x) = c_i + \frac{t_{i-1} - t_i}{2ham_i^{\alpha-1}}, \quad (2.24)$$

où $t_{i-1} = f(x_{i-1})$ et $t_i = f(x_i)$.

Preuve. Comme les fonctions $p_{i-1}(x)$ et $p_i(x)$ sont convexes sur l'intervalle $[x_{i-1}, x_i]$, les tangentes $T_{i-1}(x)$ et $T_i(x)$ sont respectivement sous les paraboles $p_{i-1}(x)$ et $p_i(x)$, i.e.,

$$T_{i-1}(x) \leq p_{i-1}(x) \leq f(x) \text{ et } T_i(x) \leq p_i(x) \leq f(x) \quad \forall x \in [x_{i-1}, x_i]. \text{ Alors}$$

$$g_i(x) \leq f(x).$$

Comme g_i est une fonction unimodale, alors l'unique minimum global de g_i est atteint au point z_i tel que

$$T_{i-1}(z_i) = T_i(z_i).$$

L'expression (2.24) se déduit facilement de la dernière équation. ■

Proposition 2.17 *Soit f une fonction de Hölder à une seule variable de constante h et d'exposant $\alpha < 1$ définie sur l'intervalle $[a, b]$. Soit la valeur $\mathbf{T}_i = \min \{p_{i-1}(z_i), p_i(z_i)\}$ comme borne inférieure constante de f sur $[x_{i-1}, x_i] \subset [a, b]$. Alors nous avons :*

$$\mathbf{T}_i = \min \left\{ t_{i-1} - h(m_i + \frac{t_{i-1} - t_i}{2h\alpha m_i^{\alpha-1}})^\alpha, t_i - h(m_i + \frac{t_i - t_{i-1}}{2h\alpha m_i^{\alpha-1}})^\alpha \right\},$$

et

$$\mathbf{T}_i < f(x), \forall x \in [x_{i-1}, x_i]. \quad (2.25)$$

Preuve. La valeur $\mathbf{T}_i$ est donnée en remplaçant la variable x dans les deux fonctions $p_{i-1}(x)$, et $p_i(x)$ par l'expression (2.24). Comme nous avons défini la fonction $U_i(x) = \max \{p_{i-1}(x), p_i(x)\}$ et nous avons $T_{i-1}(x) \leq p_{i-1}(x)$ et $T_i(x) \leq p_i(x)$,

$$g_i(x) \leq U_i(x), \forall x \in [x_{i-1}, x_i].$$

De plus, la tangente $T_{i-1}(x)$ est strictement décroissante et $T_i(x)$ est strictement croissante sur l'intervalle $[x_{i-1}, x_i]$. La fonction borne inférieure linéaire par morceaux $g_i(x)$ de la fonction objectif f sur $[x_{i-1}, x_i]$ atteint l'une des fonctions $p_{i-1}(x)$ ou $p_i(x)$ au milieu c_i de l'intervalle $[x_{i-1}, x_i]$, i.e., $g_i(c_i) = p_{i-1}(c_i)$ ou $g_i(c_i) = p_i(c_i)$. Puisque nous avons $U_i(x) < f(x)$, $\forall x \in]x_{i-1}, x_i[$, alors

$$\min \{p_{i-1}(x), p_i(x)\} \leq \min_{[x_{i-1}, x_i]} U_i(x), \forall x \in [x_{i-1}, x_i].$$

En particulier, pour $x = z_i$ il s'ensuit alors :

$$\mathbf{T}_i = \min \{p_{i-1}(z_i), p_i(z_i)\} < f(x), \forall x \in]x_{i-1}, x_i[.$$

■

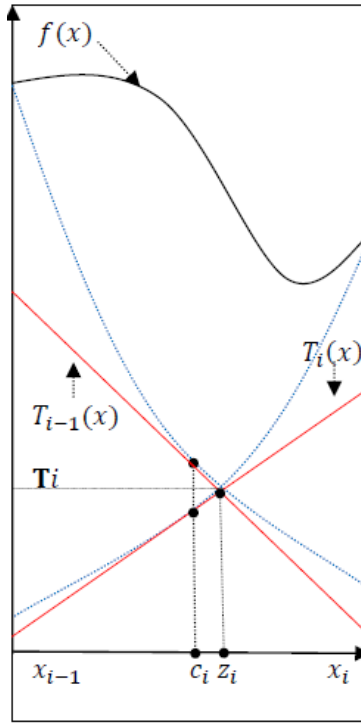


FIGURE 2.5 – Les fonctions minorantes tangentes par morceaux de f

Remarque 2.3 On observe que dans la majorité des cas, la constante $\mathbf{m}_i$ est inférieure à $\mathbf{T}_i$. Notons TPA la version étendue de l'algorithme de Piyavskii's qui n'utilise que les constantes $\mathbf{T}_i$. L'algorithme et la convergence vers la solution optimale du problème (2.1),(2.2) sont donnés ci-dessous et les tests numériques montrent que la technique utilisant les tangentes donne généralement des résultats satisfaisants par rapport aux algorithmes SPA et GPA (voir Tab.4.4).

L'algorithme de TPA est décrit comme suit :

Algorithme 9 *TPA*

1: Initialisation
2: $D = [a, b]$, $k \leftarrow 2$, $x_1 \leftarrow a$, $x_2 \leftarrow b$
3: Etape k : $x_1, x_2, \dots, x_k$ sont ordonné tel que $a = x_1 < x_2 < \dots < x_k = b$.
4: **for** $i = 2$ to k **do**
5: $z_i = c_i + \frac{t_{i-1} - t_i}{2ham_i^{\alpha-1}}$
6: $\mathbf{T}_i = \min \{t_{i-1} - h(z_i - x_{i-1})^\alpha, t_i - h(x_i - z_i)^\alpha\}$
7: ou $t_j = f(x_j)$, $j = 1, \dots, k$
8: **end for**
9:

$$\mathbf{T}_\mu = \min \{\mathbf{T}_i, 2 \leq i \leq k\}. \quad (2.26)$$

10: $x_\mu = z_\mu$
11: **if** $|x_\mu - x_{\mu-1}| > \varepsilon$ **then**
12:

$$x_{k+1} = z_\mu. \quad (2.27)$$

13: $k \leftarrow k + 1$
14: aller à l'Etape k
15: **else**
16: $f_{opt} = \min \{f(x_i) : 1 \leq i \leq k\}$
17: Stop
18: **end if**
19: retourner f_{opt}

Résultat de convergence de l'algorithme TPA

Théorème 2.5 [10] Soit f une fonction réelle définie sur un intervalle fermé $[a, b]$, telle qu'il existe une constante $h > 0$ et $0 < \alpha < 1$ qui vérifie pour tout $x, y \in [a, b]$ la condition (2.2). Soit x^* un minimiseur global de $f(x)$. Alors la suite $(x_k)_{k \geq 1}$ générée par l'algorithme TPA converge

vers x^* , c'est à dire,

$$\lim_{k \rightarrow +\infty} f(x_k) = f(x^*) = \min_{x \in [a, b]} f(x).$$

Preuve. Soit $x_1, x_2, x_3, \dots$ la suite définie et satisfait (2.24), (2.26), (2.27). Si l'on considère que $x_m \neq x_{m'}$ pour tout $m \neq m'$. L'ensemble qui contient les éléments de la suite $(x_k)_{k \geq 1}$ est alors infini et a donc au moins un point limite dans $[a, b]$. Soit $\mathbf{z}$ un point limite quelconque de $(x_k)_{k \geq 1}$ tel que $\mathbf{z} \neq a$, $\mathbf{z} \neq b$, alors la convergence vers $\mathbf{z}$ est bilatérale. Considérons l'intervalle $[x_{\mu-1}, x_\mu]$ déterminé par (2.26) à la $(k+1)$ -ième itération. D'après (2.24), (2.27), on a que le nouveau point x_{k+1} divise l'intervalle $[x_{\mu-1}, x_\mu]$ en sous-intervalles $[x_{\mu-1}, x_{k+1}]$ et $[x_{k+1}, x_\mu]$, donc en déduit

$$\max\{x_{k+1} - x_{\mu-1}, x_\mu - x_{k+1}\} \leq |x_\mu - x_{\mu-1}|. \quad (2.28)$$

Considérons maintenant un intervalle $[x_{\rho(k)-1}, x_{\rho(k)}]$ qui contient $\mathbf{z}$, puisque $\mathbf{z}$ est un point limite de la suite $(x_k)_{k \geq 1}$ et en utilisant (2.24), (2.26), (2.27) et (2.28), on obtient :

$$\lim_{k \rightarrow +\infty} (x_{\rho(k)-1} - x_{\rho(k)}) = 0. \quad (2.29)$$

De plus, la valeur $\mathbf{T}_{\rho(k)}$ qui correspond à $[x_{\rho(k)-1}, x_{\rho(k)}]$, est donnée par

$$\mathbf{T}_{\rho(k)} = \min\{f(x_{\rho(k)-1}) - h(z_\rho - x_{\rho(k)-1})^\alpha, f(x_{\rho(k)}) - h(x_{\rho(k)} - z_\rho)^\alpha\}, \quad (2.30)$$

où z_ρ est obtenu en remplaçant i par ρ dans (2.24). Comme $\mathbf{z} \in [x_{\rho(k)-1}, x_{\rho(k)}]$ et d'après (2.29) alors on a

$$\lim_{k \rightarrow +\infty} \mathbf{T}_{\rho(k)} = f(\mathbf{z}). \quad (2.31)$$

D'autre part d'après (2.25)

$$\mathbf{T}_{j(k)} \leq f(x), \forall x \in [x_{j(k)-1}, x_{j(k)}]. \quad (2.32)$$

à partir de (2.32), $\mathbf{T}_{\rho(k)} = \min\{\mathbf{T}_j, j = 2, \dots, k\}$, alors

$$\mathbf{T}_{\rho(k)} \leq \mathbf{T}_{j(k)}, \forall x \in [x_{j(k)-1}, x_{j(k)}],$$

et puisque $[a, b] = \bigcup_{j=2}^k [x_{j(k)-1}, x_{j(k)}]$, donc

$$\lim_{k \rightarrow +\infty} \mathbf{T}_{\rho(k)} \leq \mathbf{T}_{j(k)}, \forall x \in [a, b]. \quad (2.33)$$

D'où, d'après (2.32), (2.33) on obtient

$$\lim_{k \rightarrow +\infty} \mathbf{T}_{\rho(k)} \leq f(x), \forall x \in [a, b],$$

puisque x^* est le minimiseur global de f sur $[a, b]$

$$\lim_{k \rightarrow +\infty} \mathbf{T}_{\rho(k)} \leq f(x^*) \leq f(\mathbf{z}),$$

d'après (2.32) on a

$$0 \leq f(\mathbf{z}) - f(x^*) \leq f(\mathbf{z}) - \lim_{k \rightarrow +\infty} \mathbf{T}_{\rho(k)} = 0,$$

alors

$$f(\mathbf{z}) = f(x^*).$$

Selon la condition (2.2), la fonction f doit être continue sur $[a, b]$ d'où

$$f(\mathbf{z}) = f(\lim_{k \rightarrow +\infty} x_k) = \lim_{k \rightarrow +\infty} f(x_k),$$

alors

$$f(x^*) = \lim_{k \rightarrow +\infty} f(x_k).$$

■

2.4.7 Fonctions borne inférieures non-tangentes par morceaux

Récemment, le concept de fonctions affines borne inférieures a été introduit pour développer des algorithmes d'optimisation globale [62],[47]. Ici, nous donnons un nouveau résultat pour construire une autre fonction borne inférieure pour la fonction objectif de Hölder f [10]. Commençons par montrer le lemme suivant.

Lemme 2.2 *Soit $\alpha < 1$ et r deux nombres réels positifs et soit x, y deux éléments quelconques d'un intervalle $[a, b]$. Alors nous avons*

$$|x - y|^\alpha \leq \alpha r^{\alpha-1} \cdot |x - y| + (1 - \alpha) r^\alpha, \forall x, y \in [a, b]. \quad (2.34)$$

Preuve. Considérons la fonction $g : \mathbb{R}^+ \rightarrow \mathbb{R}$ définie par

$$g(t) = t^\alpha - \alpha r^{\alpha-1} \cdot t + (\alpha - 1) r^\alpha.$$

Nous avons

$$g'(t) = \alpha(t^{\alpha-1} - r^{\alpha-1}) = 0 \Leftrightarrow t = r.$$

Il est clair que la fonction $g(t)$ est strictement croissante sur l'intervalle $[0, r[$ et strictement décroissante sur l'intervalle $]r, +\infty[$ et puisque $g(r) = 0$ alors on en déduit que

$$g(t) \leq 0, \forall t \in]0, +\infty[.$$

Maintenant si on pose $t = |x - y| > 0$, pour tout $x \neq y$ le résultat du lemme est alors prouvé.

■

Théorème 2.6 Soient $\alpha < 1$ et r deux nombres réels positifs et f une fonction de Hölder de constante h et d'exposant α définie sur l'intervalle $[a, b]$ alors :

i) pour tout $x, y \in [a, b]$ on a

$$|f(x) - f(y)| \leq K_\alpha |x - y| + L_\alpha, \forall x, y \in [a, b] \quad (2.35)$$

avec

$$K_\alpha = h\alpha r^{\alpha-1} \quad \text{et} \quad L_\alpha = h(1-\alpha)r^\alpha.$$

ii) pour une valeur fixée $y = x_i \in [a, b]$, la fonction définie par

$$\varphi_\alpha(x) = f(x_i) - K_\alpha |x - x_i| - L_\alpha,$$

est Lipschitzienne de constante K_α sur $[a, b]$.

Preuve. Pour i) Si on a une fonction de Hölder de constante h et d'exposant $0 < \alpha < 1$ définie sur un intervalle de $\mathbb{R}$ et de la condition (2.2) et (2.34), l'inégalité (2.35) est immédiatement obtenue pour tout $x, y \in [a, b]$.

Pour ii) soit $x, y \in [a, b]$,

$$\begin{aligned} |\varphi_\alpha(x) - \varphi_\alpha(y)| &= |f(x_i) - K_\alpha |x - x_i| - L_\alpha - f(x_i) + K_\alpha |y - x_i| + L_\alpha| \\ &= K_\alpha ||y - x_i| - |x - x_i|| = K_\alpha ||y - x_i + x - x_i| - |x - x_i|| \\ &= K_\alpha ||y - x + x - x_i| - |x - x_i|| \leq K_\alpha |y - x|. \end{aligned}$$

Ce qu'il fallait prouver. ■

D'après ce dernier théorème, on construit une autre suite croissante de fonctions borne inférieures affines par morceaux de la fonction objectif f sur l'intervalle $[a, b]$ (Fig.2.6), avec

l'avantage que les deux constantes K_α et L_α sont indépendantes de la précision ε .

De l'inégalité (2.35) on a

$$f(y) - K_\alpha |x - y| - L_\alpha \leq f(x), \forall x, y \in [a, b].$$

Si toute valeur de $y \in [a, b]$ est fixé, alors on peut considérer la fonction concave $f(y) - K_\alpha |x - y| - L_\alpha$ comme une fonction borne inférieure de f sur $[a, b]$. Ainsi, nous construisons une suite croissante de fonctions borne inférieure affines par morceaux sur $[a, b]$ de f , similaire et différente de la précédente. Soit en effet :

$$B_k(x) = \max_{i=1, \dots, k} \{f(x_i) - K_\alpha |x - x_i| - L_\alpha\} < f(x), \forall x \in [a, b].$$

Sur chaque sous-intervalle $[x_{i-1}, x_i]$, $i = 2, \dots, k$ de $[a, b]$, la restriction de B_k peut être donnée par

$$B_i(x) = \max_i \{f(x_{i-1}) - K_\alpha (x - x_{i-1}) - L_\alpha, f(x_i) - K_\alpha (x_i - x) - L_\alpha\}.$$

Le minimiseur global $\tilde{y}_i$ de la fonction $B_i(x)$ sur $[x_{i-1}, x_i]$ et la valeur $B_i(\tilde{y}_i)$ sont donnés par

$$\begin{cases} \tilde{y}_i = \frac{1}{2} \left(\frac{f(x_{i-1}) - f(x_i)}{K_\alpha} + x_{i-1} + x_i \right), & i = 2, \dots, k \\ B_i(\tilde{y}_i) = \frac{1}{2} (f(x_{i-1}) + f(x_i) - K_\alpha (x_i - x_{i-1}) - 2L_\alpha). \end{cases} \quad (2.36)$$

Remarque 2.4 La fonction borne inférieure $B_k(x)$ n'atteint la fonction objectif f en aucun point de l'intervalle $[a, b]$. Par contre, si par exemple le paramètre $r = 1$, la constante K_α est généralement plus petite que la constante L_ε donnée dans le théorème 2.2. On peut montrer que si la constante L_α est plus petite que la précision ε , alors l'algorithme de Piyavskii modifié KL_α converge vers la solution optimale et l'approximation est meilleure avec la pente K_α car $K_\alpha < L_\varepsilon$. (voir [47]).

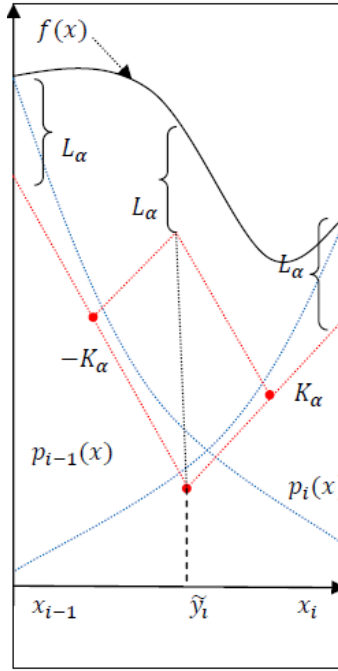


FIGURE 2.6 – Les fonctions minorantes non-tangentes par morceaux

2.4.8 Combinaison efficace des deux techniques KL_α et TPA

La modification de la méthode de Piyavskii appliquée aux fonctions de Hölder définies sur l'intervalle $[a, b]$ de $\mathbb{R}$, noté $KL_\alpha\text{-}TPA$ est basée sur la combinaison de deux phases successives. La phase 1 est capable d'obtenir relativement avec peu d'itérations en utilisant la technique KL_α , un sous-intervalle D' qui contient souvent le point du minimum global de la fonction objectif f en utilisant la fonction borne inférieure affine K_α -Lipschitz de f sur $[a, b]$ avec une tolérance δ beaucoup plus grande que la précision donnée ε . Lorsque l'algorithme KL_α s'arrête, une grande partie de l'intervalle $[a, b]$ est supprimée. La phase 2 intervient l'algorithme TPA comme une version étendue de la méthode de Piyavskii et qui est basé sur la construction d'une suite croissante de fonctions tangentes par morceaux au milieu des sous-intervalles obtenus à chaque itération, aux fonctions paraboliques borne inférieures par morceaux de f . Une recherche globale plus fine sur l'intervalle D' restant dans l'intervalle $[a, b]$, contenant éventuellement la solution du problème avec la précision ε donnée. L'algorithme de $KL_\alpha\text{-}TPA$ est décrit ci-dessous :

Algorithme 10 L'algorithme combiné les deux techniques KL_α et TPA

```

1: Phase 1  $KL_\alpha$ 
2:  $D = [a, b], k \leftarrow 2, x_1 \leftarrow \frac{a+b}{2}, x_{opt} \leftarrow x_1, f_{opt} \leftarrow f(x_{opt}), B_{opt} \leftarrow f_{opt} - k_\alpha(\frac{b-a}{2}) - L_\alpha,$ 
3: while  $f_{opt} - B_{opt} > L_\alpha + \delta$  do
4:    $x_1, x_2, \dots, x_k$  sont ordonnés de telle sorte que  $a = x_1 < x_2 < \dots < x_k = b$ .
5:   for  $i = 2$  to  $k$  do
6:      $\tilde{y}_i = \frac{1}{2}(\frac{f(x_{i-1}) - f(x_i)}{K_\alpha} + x_{i-1} + x_i),$ 
7:      $B_i(\tilde{y}_i) = \frac{1}{2}(f(x_{i-1}) + f(x_i) - K_\alpha(x_i - x_{i-1}) - 2L_\alpha)$ 
8:   end for
9:   trouver l'intervalle  $[x_{v-1}, x_v]$  correspondant à la valeur minimale de  $B_i$ .
10:   $B_v = \min \{B_i : 2 \leq i \leq k\}$  et poser  $x_{k+1} = \tilde{y}_v,$ 
11:  if  $f(x_{k+1}) < f_{opt}$  then
12:     $f_{opt} \leftarrow f(x_{k+1}), x_{opt} \leftarrow x_{k+1}, B_{opt} \leftarrow B_v$  et  $D' = [x_{v-1}, x_v].$ 
13:  end if
14:   $k \leftarrow k + 1$ 
15: end while
16: Phase 2  $TPA$ 
17: Initialisation  $D \leftarrow D', k \leftarrow 2, x_1 \leftarrow x_{v-1}, x_2 \leftarrow x_v$ 
18: Etape  $k$  :  $x_1, x_2, \dots, x_k$  sont ordonnés de telle sorte que  $x_{v-1} = x_1 < \dots < x_k = x_v.$ 
19: for  $i = 2$  to  $k$  do
20:   $z_i = c_i + \frac{t_{i-1} - t_i}{2ham_i^{a-1}}$ 
21:   $\mathbf{T}_i = \min \{t_{i-1} - h(m_i + \frac{t_{i-1} - t_i}{2(ham_i)^{a-1}}), t_i - h(m_i + \frac{t_i - t_{i-1}}{2(ham_i)^{a-1}})\},$  où  $t_j = f(x_j), i = 1, \dots, k$ 
22: end for
23:  $\mathbf{T}_\mu = \min \{\mathbf{T}_i, 2 \leq i \leq k\}, x_\mu = z_\mu.$ 
24: if  $|x_\mu - x_{\mu-1}| > \varepsilon$  then
25:   $x_{k+1} = z_\mu, k \leftarrow k + 1$  et aller à l'Etape  $k$ 
26: else
27:   $f_{opt} = \min \{f(x_i) : 1 \leq i \leq k\}$  et Stop
28: end if

```

2.5 Conclusion

Dans ce chapitre, une méthode de résolution d'un problème d'optimisation globale d'une seule variable non convexe est présentée où la fonction objectif est höldérienne sur un intervalle de $\mathbb{R}$ avec la connaissance a priori de ses paramètres h et α . Récemment, Gourdin et al.[21] et Lera et al.[36] ont introduit une extension de la méthode de Piyavskii aux problèmes d'optimisation de Hölder. De nouvelles techniques intéressantes, appelées TPA et $KL_\alpha TPA$ sont proposées. L'idée est essentiellement basée sur la construction d'une séquence croissante d'une fonction borne inférieure affine par morceaux [10]. Pour le premier algorithme TPA qui utilise les tangentes au milieu des sous-intervalles lors de la recherche de fonctions de cônes paraboliques par morceaux. Il évite la résolution à chaque itération d'équation algébrique non linéaire. La convergence de TPA vers le minimum global de f est montrée et les tests numériques prouvent que cet algorithme améliore généralement les résultats existants notamment en temps de calcul. Pour le deuxième algorithme $KL_\alpha TPA$ qui est composé de deux phases, la première phase KL_α vise à réduire considérablement l'intervalle faisable $[a, b]$ à un intervalle D' qui contient souvent l'optimum global avec quelques itérations, puis utilise l'algorithme TPA sur D' . L'algorithme combiné $KL_\alpha TPA$ a été testé et les résultats ont été satisfaisants pour la majorité des problèmes, que ce soit en termes de temps de calcul ou de nombre de points d'évaluation. Enfin, nous souhaitons étendre ces algorithmes lorsque la constante de Hölder h n'est pas connue a priori et au cas de la fonction de Hölder à plusieurs variables. Dans le troisième chapitre, on proposera quelques techniques de généralisation de certains algorithmes de recouvrement au cas des fonctions höldériennes à plusieurs variables.

Optimisation globale höldérienne à plusieurs variables

3.1 Introduction

L'optimisation globale à plusieurs variables est un domaine de recherche active. C'est le processus de trouver l'extremum global d'une fonction de n variables, avec la possibilité d'être soumis à certaines contraintes. Son importance est due aux besoins croissants dans de nombreuses applications en sciences et en ingénierie. Au cours des dernières années, de nombreux travaux ont été réalisés concernant les fonctions de Lipschitz [30], [29], [44], [60]. Mladinéo [43] fait la généralisation directe de la méthode de Piyavskii au cas multivariables. L'idée est basée sur la détermination du minimum local des fonctions sous-estimateurs. À chaque itération, on a besoin de résoudre un système de n équations quadratiques. Plusieurs techniques ont permis d'améliorer cette approche, Mayne et Polak [40], Jaumard et al. [32], Gergel [20]. On peut voir aussi une autre idée qu'est ramenée la minimisation d'une fonction à plusieurs variables à une succession de minimisation de fonctions à une seule variable où on peut appliquer une méthode de recouvrement ou autre méthode unidimensionnelle. Plus récemment, des travaux abordant des fonctions moins régulières sont apparus [21], [36], [31], [47]. La maximisation globale des fonctions höldériennes à plusieurs variables sur un hyper-rectangle de $\mathbb{R}^n$ a été étudiée pour la première fois par Gourdin et al. [21]. Dans [21], les auteurs ont utilisé la généralisation de l'algorithme de Piyavskii et ont proposé une procédure pour partitionner et éliminer les hyper-rectangles (Branch-and-Bound) sans in-térêt en construisant un sur-estimateur constant par morceaux [21]. Une autre technique de

minimisation d'une fonction à plusieurs variables est ramenée à une minimisation de fonctions à une seule variable où on peut appliquer une méthode unidimensionnelle. Dans ce chapitre, on va étudier la généralisation de certaines méthodes mentionnées dans les chapitres précédentes au cas où la fonction objectif est höldérienne à plusieurs variables.

3.2 Optimisation globale höldérienne sans contraintes

3.2.1 Une relaxation des fonctions minorantes paraboliques par morceaux (BB-Piyavskii multidimensionnel)

Pour le cas multidimensionnel, la généralisation directe de l'algorithme de Piyavskii est très difficile, car la détermination des minima locaux d'une fonction sous-estimateur de la fonction objectif nécessite la détermination de l'intersection de plusieurs hypersurfaces paraboliques. Gourdin et al.[21] ont proposé une procédure d'élimination des régions sans intérêt en construisant des sous-estimateurs constants par morceaux.

Principe de la procédure

Considérons le problème de minimisation globale

$$\min_{x \in \mathbf{P} = \prod_{i=1}^n [a_i, b_i]} f(x) \quad (3.1)$$

que l'on veut résoudre avec une précision exigée $\varepsilon > 0$, avec f une fonction höldérienne de constante $h > 0$ et d'exposant $(1/\beta, \beta > 1)$ définie sur un hyper-rectangle $\mathbf{P} = \prod_{i=1}^n [a_i, b_i]$ de $\mathbb{R}^n$ et à valeurs dans $\mathbb{R}$. C'est-à-dire, une fonction telle que :

$$|f(x) - f(y)| \leq h \|x - y\|^{1/\beta}, \quad \forall x, y \in \mathbf{P} \quad (3.2)$$

où $\|\cdot\|$ désigne la norme euclidienne. D'après (3.2) le meilleur sous-estimateur de la fonction objectif f est $f_i(x) = f(x_i) - h \|x - x_i\|^{1/\beta}$ tel que x_i est un point du pavé $\mathbf{P}$.

Si $x = \frac{a+b}{2} = c$ est le centre du pavé $\mathbf{P}$, la constante sous-estimateur de f est

$$m = \min_{x \in \mathbf{P}} f_i(x) = f(c) - h \left(\frac{\|b - a\|}{2} \right)^{1/\beta}.$$

En effet, le minimum de $f_i(x)$ est évalué en un point e , où la distance entre e et le centre c est maximale. Comme c est le centre du pavé $\mathbf{P}$, alors

$$\|e - c\| = \frac{\|b - a\|}{2}$$

où $\|b - a\|$ est la diagonale de $\mathbf{P}$. Soient $(x_i)_{1 \leq i \leq k}$, k points du pavé $\mathbf{P}$, le sous estimateur de la fonction f est :

$$F_{1k}(x) = \max_{1 \leq i \leq k} f_i(x).$$

Considérons $(\mathbf{P}_i)_{1 \leq i \leq k}$ une partition de sous-pavés du pavé $\mathbf{P}$ et notons par c_i les centres des sous-pavés respectifs $\mathbf{P}_i$. Donc le meilleur sous-estimateur constant par morceaux de f sur cette partition est noté par $F_{2k}(x)$ et définie par

$$\forall i = 1, \dots, k, \forall x \in \mathbf{P}_i, \quad F_{2k}(x) = \underbrace{f(c_i) - h\left(\frac{\|b_i - a_i\|}{2}\right)^{1/\beta}}_{m_i}.$$

Le sous estimateur $F_{1k}(x)$ de f est donc relaxé par un sous-estimateur constant par morceaux sur le pavé $\mathbf{P}$. Donnons maintenant l'algorithme Branch-and-Bound :

Algorithme 11 BB-Piyavskii multidimensionnel

- 1: Etape de sélection.
- 2: Après la partition successive du pavé, une suite notée $\$$ de sous problèmes sera stockée.
On tire de $\$$ un sous problème $\bar{P} = ([\bar{a}, \bar{b}], \bar{m})$ tel que $\bar{m}$ est la plus petite borne inférieure.
Soit $[\bar{a}, \bar{b}] = \prod_{i=1}^n [\bar{a}_i, \bar{b}_i]$ le sous pavé correspondant à $\bar{P}$.
- 3: Etape de séparation (branching)
- 4: Soit $j \in \{1, 2, \dots, n\}$ tel que $\bar{l}_j = \max_{1 \leq i \leq n} (\bar{b}_i - \bar{a}_i)$ ($\bar{l}_j$ est la longueur du côté le plus long de $[\bar{a}, \bar{b}]$).
- 5: Alors $[\bar{a}, \bar{b}]$ sera subdivisé en p sous pavés $[\bar{a}, \bar{b}]_1, [\bar{a}, \bar{b}]_2, \dots, [\bar{a}, \bar{b}]_p$, ayant la même dimension, par la subdivision du j -ième côté de $[\bar{a}, \bar{b}]$ en p parties équidistantes.
- 6: Etape initiale et courante de la borne supérieure
- 7: La fonction sous estimateur constant de f sur $[\bar{a}, \bar{b}]_i$ est définie par :
- 8:

$$D_0 = c_0 = \frac{a+b}{2}$$
$$\text{et } \bar{x}_i = \bar{c}_i = \frac{\bar{a}_i + \bar{b}_i}{2}.$$

- 9: Etape initiale et courante de la borne inférieure.
- 10: La fonction sous estimateur constant de f sur $[\bar{a}, \bar{b}]_i$ est définie par :
- 11:

$$m_0 = f(c_0) - h \left(\frac{\|b-a\|}{2} \right)^{1/\beta}$$
$$\text{et } \bar{m}_i = \max \left\{ \bar{m}, f(\bar{c}_i) - h \left(\frac{\|\bar{b}_i - \bar{a}_i\|}{2} \right)^{1/\beta} \right\}.$$

- 12: Comme le sous problème actuel $\bar{P}^i$ résulte de l'étape de séparation effectuée sur le sous problème $\bar{P} = ([\bar{a}, \bar{b}], \bar{m})$, $\bar{m}$ est aussi une borne inférieure pour $\bar{P}^i$. Notons encore que la définition de la fonction sous estimateur donnée auparavant garantit que les bornes inférieures $\bar{m}$ est une suite croissante.
-

3.2.2 La méthode de la transformation réductrice Alienor

Dans le cas multidimensionnel, certains chercheurs ont pensé à transformer les problèmes d'optimisation globale aux problèmes d'optimisation plus simples en réduisant le nombre des variables. De nombreux auteurs ont exploré une procédure basée sur le remplissage de la région faisable par une courbe paramétrique. Sergeyev et al.[53], Lera [37][38], Butz [8] et Strongin [57], ont prennent en compte les approximations de courbe de type Peano. Ces courbes, ont d'abord, été présentées pour la première fois par Peano (1890), puis par Hilbert (1891), et ont la propriété de passer par tous les points d'un hyper-rectangle de $\mathbb{R}^n$. D'autre part, Cherruault et al. [1],[11],[12],[13], Ziadi et al. [22] [47],[65],[66],[67] ont élaboré, à la fin des années 1970, une méthode dite transformation réductrice Alienor dont l'idée consiste à approcher une fonction de plusieurs variables par une fonction d'une seule variable, il devenait alors assez simple de déterminer les minima globaux car il suffisait de les chercher en suivant l'évolution d'une certaine courbe avec une précision que l'on souhaite. Ainsi, la transformation est basée sur la construction d'une courbe paramétrée dite α -dense $\zeta(t) = (\zeta_1(t), \zeta_2(t), \dots, \zeta_n(t))$, dans l'ensemble faisable $\mathbf{P} = \prod_{i=1}^n [a_i, b_i] \subset \mathbb{R}^n$.

Définition 3.1 Soit S un sous-ensemble de l'espace métrique E et $\alpha > 0$. On dit qu'une courbe continue ζ de E est α -dense dans S , si $\zeta \subset S$ et $d(x, \zeta) \leq \alpha$, pour tout $x \in S$, où :

$$d(x, \zeta) = \inf_{y \in \zeta} d(x, y).$$

Définition 3.2 On dit qu'un sous-ensemble S de l'espace métrique E est α -densifiable s'il existe une courbe continue de E , rectifiable (la longueur de la courbe est finie) et dense dans S

Considérons le problème d'optimisation globale multidimensionnel :

$$\left\{ \begin{array}{l} \min_{x \in \mathbf{P} = \prod_{i=1}^n [a_i, b_i]} f(x) \end{array} \right. \quad (3.3)$$

où f est une fonction höldérienne de constante $h > 0$ et d'exposant $(1/\beta, \beta > 1)$ sur le domaine $\mathbf{P} = \prod_{i=1}^n [a_i, b_i] \subset \mathbb{R}^n$. La transformation Alienor permet de résoudre le problème (3.3) en exprimant la fonction objectif de n variables par une fonction d'une seule variable. La transformation Alienor

$$x_i = \zeta_i(t), \quad i = 1, \dots, n$$

sert à approcher la fonction f par la fonction $\mathbf{f}$ définie par :

$$\mathbf{f}(t) = f(\zeta(t)) = f(\zeta_1(t), \zeta_2(t), \dots, \zeta_n(t))$$

avec $t \in [0, T]$ et T est la borne supérieure du domaine de définition de la fonction ζ qui permet de densifier $\mathbf{P}$.

Le problème de minimisation (3.3) est alors approximé par le problème de minimisation

$$\min_{t \in [0, T]} \mathbf{f}(t), \quad (3.4)$$

Théorème 3.1 *Les minimiseurs globaux de $f(x_1, x_2, \dots, x_n)$, vérifiant les hypothèses précédentes, peuvent être approximés par les minimiseurs globaux de $\mathbf{f}(t)$. L'approximation sera d'autant meilleure que α sera choisi petit.*

Preuve. Soient $(x_1^*, x_2^*, \dots, x_n^*)$ un point de $\mathbf{P} = \prod_{i=1}^n [a_i, b_i]$ réalisant le minimum global de f et t^* un point de $[0, T]$ réalisant le minimum global de $\mathbf{f}$.

Puisque la courbe $\zeta(t)$ est α -dense dans $\mathbf{P}$ alors il existe $t_0 \in [0, T]$ tel que

$$\|(x_1^*, x_2^*, \dots, x_n^*) - \zeta(t_0)\| \leq \alpha.$$

Comme f est höldérienne on a

$$|f(x_1^*, x_2^*, \dots, x_n^*) - f(\zeta(t_0))| \leq h \|(x_1^*, x_2^*, \dots, x_n^*) - \zeta(t_0)\|^{1/\beta} \leq h\alpha^{1/\beta}.$$

Comme on a

$$f(x_1^*, x_2^*, \dots, x_n^*) \leq \mathbf{f}(t^*) \leq f(\zeta(t_0)),$$

on déduit que

$$0 \leq \mathbf{f}(t^*) - f(x_1^*, x_2^*, \dots, x_n^*) \leq h\alpha^{1/\beta}.$$

Donc, pour obtenir la précision désirée ε , il suffit de choisir $\alpha = (\frac{\varepsilon}{h})^\beta$. ■

3.2.3 Une nouvelle transformation réductrice

La caractérisation et la génération des courbes α -dense dans un pavé de $\mathbb{R}^n$ ($n \geq 2$) est un sujet fondamental. D'autre part, l'un des objectifs les plus importants est l'application des

courbes α -dense à l'optimisation globale. Mais, tout cela dépend essentiellement de la longueur de la courbe qui densifie le domaine de la fonction objectif. Des résultats intéressants, sur l'existence des courbes avec une densité fixée et une longueur minimale sur des compacts densifiables dans un espace métrique général ont été donnés par A. Ziadi et Y. Cherruault [65]. En se basant sur un résultat donné en [66]. On définit d'une manière constructive une courbe α -dense dans un pavé quelconque de $\mathbb{R}^n$.

Théorème 3.2 *Considérons dans $\mathbb{R}^n$ l'hyper-rectangle $\mathbf{P}$ et soit α un nombre strictement positif et soient $\zeta_1, \zeta_2, \dots, \zeta_n$ des fonctions, définies respectivement de $[0, T]$ dans $[a_i, b_i]$, tel que :*

- 1) $(\zeta_i)_{1 \leq i \leq n}$, sont continues et surjectives.
- 2) $(\zeta_i)_{2 \leq i \leq n}$ périodiques, respectivement, de périodes $(t_i)_{2 \leq i \leq n}$.
- 3) pour tout intervalle I de $[0, T]$ et pour tout $i \in \{2, \dots, n\}$.

Nous avons :

$$\mu(I) \leq t_i \Rightarrow \mu(\zeta_{i-1}(I)) < \alpha,$$

avec $\mu(\cdot)$ est la mesure de Lebesgue.

Alors la fonction définie par $\zeta(t) = (\zeta_1(t), \zeta_2(t), \dots, \zeta_n(t))$, pour $t \in [0, T]$, représente une courbe $\sqrt{n-1}\alpha$ -dense dans $\mathbf{P}$.

Corollaire 3.1 Soit $\zeta(t) = (\zeta_1(t), \zeta_2(t), \dots, \zeta_n(t)) : [0, \frac{\pi}{\alpha_1}] \rightarrow \prod_{i=1}^n [a_i, b_i]$ une fonction définie par :

$$\zeta_i(t) = \frac{a_i - b_i}{2} \cos(\alpha_i t) + \frac{a_i + b_i}{2} \quad i = 1, 2, \dots, n,$$

où $\alpha_1, \alpha_2, \dots, \alpha_n$ et α des constantes strictement positives satisfaisant les relations suivantes

$$\alpha_i \geq \frac{\pi}{\alpha} (b_{i-1} - a_{i-1}) \alpha_{i-1}, \forall i = 2, \dots, n.$$

Alors la courbe définie par la représentation paramétrique $\zeta(t) = (\zeta_1(t), \zeta_2(t), \dots, \zeta_n(t))$ pour $t \in [0, \frac{\pi}{\alpha_1}]$, est $\sqrt{n-1}\alpha$ -dense dans $\mathbf{P}$.

Lors de l'utilisation de l'approche de transformation réductrice, on fournit d'abord explicitement une représentation paramétrique : $x_i = \zeta_i(t)$, où $i = 1, \dots, n$, de la courbe α -dense dans l'hyper-rectangle $\mathbf{P}$, pour $t \in [0, T]$, où T est la borne supérieure du domaine de définition de la fonction ζ .

1) Etape initiale. On définit la fonction

$$\zeta(t) = (\zeta_1(t), \zeta_2(t), \dots, \zeta_n(t)) : \left[0, \frac{\pi}{\alpha_1}\right] \rightarrow \mathbf{P}.$$

Par

$$\zeta_i(t) = \frac{a_i - b_i}{2} \cos(\alpha_i t) + \frac{a_i + b_i}{2}, \quad i = 1, \dots, n,$$

où $(\alpha_i)_{1 \leq i \leq n}$ sont fournis par :

$$\begin{cases} \alpha_1 = 1 \\ \alpha_i = \frac{\pi}{\alpha} (b_{i-1} - a_{i-1}) \alpha_{i-1}, \quad i = 2, \dots, n, \end{cases}$$

avec

$$\alpha = \left(\frac{\varepsilon}{2h}\right)^\beta \cdot \frac{1}{\sqrt{n-1}}, \quad h \text{ la constante de Hölder de } f.$$

D'après le corollaire 3.1, la courbe paramétrée $\zeta(t) = (\zeta_1(t), \zeta_2(t), \dots, \zeta_n(t))$ est α -dense dans l'hyper-rectangle $\prod_{i=1}^n [a_i, b_i]$. De plus, il est facile de prouver que la fonction ζ est lipschitzienne de constante :

$$\bar{L} = 1/2 \left(\sum_{i=1}^n (b_i - a_i)^2 \alpha_i^2 \right)^{\frac{1}{2}}.$$

Alors la fonction f est approchée par la fonction d'une seule variable définie par $\mathbf{f}(t) = f(\zeta(t)) = f(\zeta_1(t), \zeta_2(t), \dots, \zeta_n(t))$. Le problème de minimisation (3.3) est alors approché par le problème de minimisation unidimensionnel suivant.

$$\left\{ \min_{t \in [0, T]} \mathbf{f}(t) \right\}$$

Exemple 3.1 Considérons le problème d'optimisation global hölderien suivant :

$$\min_{x \in \mathbf{P}} f(x) = \sin(x_1) \sin(x_1 x_2),$$

avec $\mathbf{P}$ l'ensemble des points $x = (x_1, x_2) \in [0, 4]^2$, la constante de Hölder $h = 10.2248$ et $\beta = 2$. On construit une courbe $\zeta(t)$ α -dense dans le compact $\mathbf{P}$, comme suit :

$$\begin{aligned} \zeta_1(t) &= -2 \cos(t) + 2 \\ \zeta_2(t) &= -2 \cos\left(\frac{4\pi}{\alpha} t\right) + 2, \end{aligned}$$

on prend $\alpha = 0.1$.

Le problème initial est approximé par le problème unidimensionnel suivant :

$$\min_{t \in [0, \pi]} \mathbf{f}(t) = f(\zeta(t)).$$

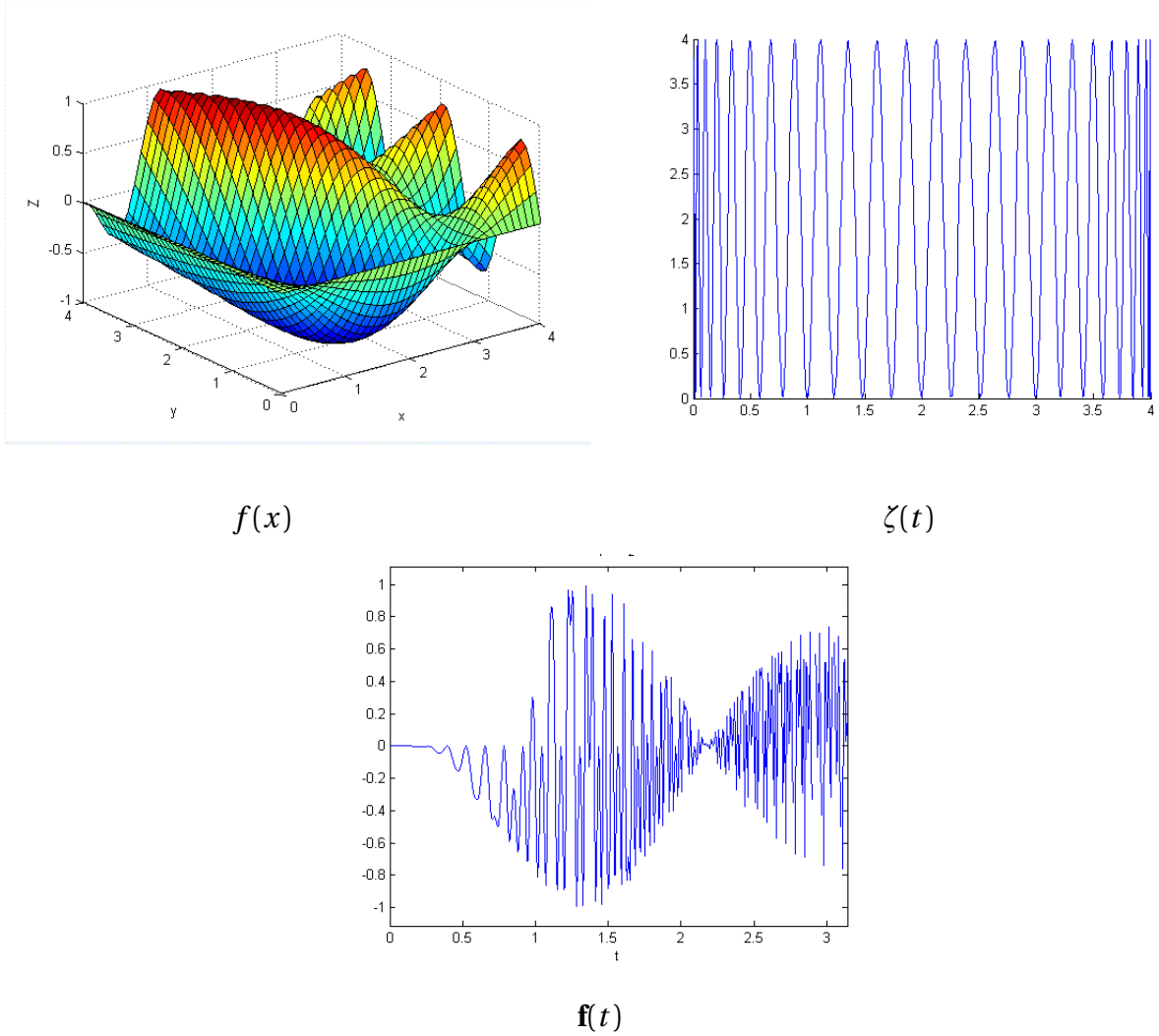


FIGURE 3.1 – Illustration graphique de la transformation d'un problème multidimensionnel en un problème unidimensionnel à l'aide d'une courbe α -dense

Théorème 3.3 La fonction $\mathbf{f}(t) = f(\zeta(t))$ pour $t \in [0, T]$ est une fonction de Hölder de constante $\mathbf{h}$ et d'exposant $(1/\beta, \beta > 1)$ où $\mathbf{h}$ est donné par $\mathbf{h} = h \bar{L}^{1/\beta}$

Preuve. Soit t_1 et t_2 de $[0, T]$, on a

$$|\mathbf{f}(t_1) - \mathbf{f}(t_2)| = |f(\zeta(t_1)) - f(\zeta(t_2))| \leq h \|\zeta(t_1) - \zeta(t_2)\|^{1/\beta}.$$

De plus, les fonctions ζ_i sont lipschitziennes de constantes $\bar{L}$, $\forall t_1, t_2 \in [0, T]$, en effet,

$$|\zeta(t_1) - \zeta(t_2)| \leq \bar{L} |t_1 - t_2|,$$

alors

$$|\mathbf{f}(t_1) - \mathbf{f}(t_2)| \leq h (\bar{L} |t_1 - t_2|)^{1/\beta},$$

d'où

$$|\mathbf{f}(t_1) - \mathbf{f}(t_2)| \leq h \bar{L}^{1/\beta} |t_1 - t_2|^{1/\beta}.$$

■

3.2.4 La méthode mixte Alienor-TPA

Afin de déterminer le minimum global de $f(x)$, la méthode mixte modifiée Alienor-TPA est composée en deux étapes : l'étape de transformation réductrice et l'application de l'algorithme TPA [10] à la fonction $\mathbf{f}(t) = f(\zeta(t))$, qui est une fonction hölderienne de constante $\mathbf{h} = h \bar{L}^{1/\beta}$.

Algorithme 12 La méthode mixte Alienor-TPA

- 1: **Entrée :** $\mathbf{P} = \prod_{i=1}^n [a_i, b_i]$ Domaine de recherche, la fonction objectif f (fonction höldérienne à plusieurs variables), la constante de Hölder h , l'exposant $1/\beta$ et la dimension de l'espace : n .
 - 2: **Sortie :** *partie 1 :* $\zeta(t)$ la courbe paramétrique, $\mathbf{f}$ la fonction höldérienne unidimensionnelle.
 - 3: *partie 2 :* f_{opt} le meilleur minimum.
 - 4: *partie 1 :*
 - 5: Initialisation : ε précision donnée, $\alpha = (\frac{\varepsilon}{2h})^\beta$, $\alpha_1 = 1$
 - 6: **for** $i = 2$ to k **do**
 - 7: $\alpha_i = \frac{\pi}{\alpha}(b_i - a_i)\alpha_{i-1}$
 - 8: **end for**
 - 9: **for** $i = 1$ to k **do**
 - 10: $\zeta_i(t) = \frac{a_i - b_i}{2} \cos(\alpha_i t) + \frac{a_i + b_i}{2}$
 - 11: **end for**
 - 12: $\zeta(t) = (\zeta_1(t), \zeta_2(t), \dots, \zeta_n(t))$, $\mathbf{f}(t) = f(\zeta(t))$.
 - 13: *partie 2 :*
 - 14: Initialisation : $k \leftarrow 2$, $t_1 \leftarrow 0$, $t_2 \leftarrow \frac{\pi}{\alpha_1}$.
 - 15: Etape k : $t_1, t_2, \dots, t_k$ sont ordonné tel que $0 = t_1 < t_2 < \dots < t_k = \frac{\pi}{\alpha_1}$.
 - 16: **for** $i = 2$ to k **do**
 - 17: $\omega_i = \varsigma_i + \frac{\beta(\mathbf{f}(t_{i-1}) - \mathbf{f}(t_i))}{2he_i^{(1/\beta)-1}}$, $\mathbf{T}_i = \min \{ \mathbf{f}(t_{i-1}) - \mathbf{h}(\omega_i - t_{i-1})^{1/\beta}, \mathbf{f}(t_i) - \mathbf{h}(t_i - \omega_i)^{1/\beta} \}$
avec $\varsigma_i = \frac{t_{i-1} + t_i}{2}$ et $e_i = \frac{t_i - t_{i-1}}{2}$
 - 18: **end for**
 - 19: $\mathbf{T}_\theta = \min \{ \mathbf{T}_i, 2 \leq i \leq k \}$, $t_\theta = \omega_\theta$
 - 20: **if** $|t_\theta - t_{\theta-1}| > \varepsilon$ **then**
 - 21: $t_{k+1} = \omega_\theta$, $k \leftarrow k + 1$ et aller à l'Etape k
 - 22: **else**
 - 23: $f_{opt} = \min \{ \mathbf{f}(t_i) : 1 \leq i \leq k \}$, Stop.
 - 24: **end if**
-

3.3 Résultats de convergence

3.3.1 Convergence de la méthode mixte Alienor-TPA

Théorème 3.1 [23] Soient f une fonction $(h, 1/\beta)$ -höldérienne sur $\mathbf{P}$ et M le minimum global de f sur $\mathbf{P}$. Alors l'algorithme mixte Alienor-TPA converge vers le minimum global avec une précision inférieure ou égale à ε .

Preuve. Notons $\overline{M}$ le minimum global de $\mathbf{f}$ sur $[0, \frac{\pi}{\alpha_1}]$ où $\mathbf{f} = f(\zeta(t))$. D'autre part, désignons par $\mathbf{f}_\varepsilon$ le minimum global du problème (3.3) obtenu par la méthode d'Alienor-TPA.

Montrons que :

$$\mathbf{f}_\varepsilon - M \leq \varepsilon.$$

a) Comme f est continue sur $\mathbf{P}$, il existe un point $\mathbf{z} \in \mathbf{P}$ tel que $M = f(\mathbf{z})$. De plus, il existe $t_0 \in [0, \frac{\pi}{\alpha_1}]$ tel que $\|\mathbf{z} - \zeta(t_0)\| \leq \left(\frac{\varepsilon}{2h}\right)^\beta$, de sorte que $|f(\mathbf{z}) - f(\zeta(t_0))| \leq \frac{\varepsilon}{2}$.

Et donc :

$$f(\zeta(t_0)) - M \leq \frac{\varepsilon}{2}.$$

Mais le fait que $M \leq \overline{M} \leq f(\zeta(t_0))$, on déduit :

$$\overline{M} - M \leq \frac{\varepsilon}{2}. \quad (3.5)$$

b) Comme $\mathbf{f}$ est continue sur $[0, \frac{\pi}{\alpha_1}]$, il existe un point $t^* \in [0, \frac{\pi}{\alpha_1}]$ tel que $\overline{M} = \mathbf{f}(t^*)$, impliquant t^* est un minimiseur global de $\mathbf{f}$. Alors t^* est un point limite de la suite $(t_k)_{k \geq 1}$ obtenue par l'algorithme mixte.

Par conséquent : $t^* \in [t_{\rho(k)-1}, t_{\rho(k)}]$ et $\lim_{k \rightarrow +\infty} (t_{\rho(k)} - t_{\rho(k)-1}) = 0$.

C'est-à-dire

$$\exists t_\varepsilon \in [t_{s-1}, t_s] : |t_s - t_{s-1}| \leq \left(\frac{\varepsilon}{2h}\right)^\beta \text{ et } \mathbf{f}_\varepsilon = \mathbf{f}(t_\varepsilon).$$

Pour que

$$\mathbf{f}(t^*) \leq \mathbf{f}(t_\varepsilon)$$

Par conséquent

$$\mathbf{f}_\varepsilon - \overline{M} = \mathbf{f}(t_\varepsilon) - \mathbf{f}(t^*) \leq \mathbf{h} |t_\varepsilon - t^*|^{1/\beta} \leq \frac{\varepsilon}{2}. \quad (3.6)$$

Enfin, d'après (3.5) et (3.6) le résultat du théorème est prouvé. ■

3.4 Optimisation globale höldérienne avec contraintes

Considérons le problème d'optimisation global avec contraintes suivant :

$$\left\{ \begin{array}{l} \min f(x) \\ \text{s.c } g_i(x) \leq 0, i \in \mathfrak{I} \\ x \in \Omega, \end{array} \right. \quad (3.7)$$

où $x = (x_1, \dots, x_n)$ est le vecteur réel de $\mathbb{R}^n$ représentant les n variables, $\mathfrak{I}$ ensemble finie d'indices et Ω est un hyper-rectangle de $\mathbb{R}^n$ donné par

$$\Omega = [a_1, b_1] \times [a_2, b_2] \times \dots \times [a_n, b_n],$$

et $a_i, b_i, i = 1, \dots, n$ sont des constantes spécifiant l'hyper-rectangle Ω . Les fonctions $f, g_i (i \in \mathfrak{I}) : \Omega \rightarrow \mathbb{R}$ sont supposés être höldériennes et non-différentiables sur Ω .

Désignant la région faisable des solutions pour le problème (3.7) associée à des contraintes d'inégalité par

$$D = \{x \in \Omega, g_i(x) \leq 0, \forall i \in \mathfrak{I}\},$$

avec l'hypothèse que la région faisable D est borné et robuste

Définition 3.1 *Un sous-ensemble fermé $X \subset \mathbb{R}^n$ est dit robuste s'il est la fermeture d'un ensemble ouvert non vide.*

Le but de cette section est d'étudier l'extension de l'algorithme *TPA* que nous avons introduit dans le chapitre 2 pour minimiser des fonctions höldériennes à une seule variable sur un intervalle de $\mathbb{R}$ aux problèmes de minimisation de fonctions höldériennes à plusieurs variables soumises à un ensemble fini de contraintes qui sont aussi höldériennes. Dans de nombreux cas, la région faisable est un ensemble compact non convexe dont la frontière de D sont des

fonctions non différentiables. On traite le problème (3.7) dans le cas où l'ensemble faisable D peut être formulé, avec un nombre fini d'inégalités, comme suit :

$$D = \left\{ \begin{array}{c} x \in \Omega, \\ a \leq x_1 \leq b \\ : \\ \varphi_i(x_1, \dots, x_{i-1}) \leq x_i \leq \psi_i(x_1, \dots, x_{i-1}), \quad 2 \leq i \leq n. \end{array} \right\} \quad (3.8)$$

où $a, b \in \mathbb{R}$ tel que $a \leq a_1$ et $b \leq b_1$ et les fonctions φ_i et ψ_i sont toutes höldériennes respectivement avec les constantes : ($h_i > 0$, $\beta_i < 1$) et ($H_i > 0$, $\gamma_i < 1$) et non différentiable sur un hyper-rectangle Ω_i qui donné par :

$$\Omega_i = [a_1, b_1] \times [a_2, b_2] \times \dots \times [a_{i-1}, b_{i-1}],$$

Le problème (3.7) étant généralement difficile à résoudre, il n'existe que peu de travaux dans la littérature [15], [42]. Ici, on fait la réduction de la dimension du problème (3.7) où l'ensemble faisable D est défini par (3.8), au moyen d'une transformation permettant d'écrire les n variables x_i , $1 \leq i \leq n$ en fonction d'une seule variable t , donné par $x_i = \ell_i(t)$, $1 \leq i \leq n$. La fonction objectif $f(x)$ du problème initial (3.7) est approchée par une fonction d'une seule variable définie par $\mathbf{f}^*(t) = f(\ell_1(t), \dots, \ell_n(t))$. Une courbe paramétrique α -dense calculable $\ell(t) = (\ell_1(t), \dots, \ell_n(t))$ dans l'ensemble faisable (3.8) devrait être utilisée et assurer le passage du problème multidimensionnel avec contraintes (3.7) et (3.8) en un problème d'optimisation globale sans contraintes plus simple

$$\min_{t \in I} \mathbf{f}^*(t) \quad (3.9)$$

où la fonction objectif unidimensionnelle $\mathbf{f}^*(t)$ du problème (3.9) est höldérienne et non-différentiable sur l'intervalle $I = [0, T]$. La constante T est la borne supérieure de la fonction $\ell(t)$ sur laquelle se situera le minimum global de $\mathbf{f}^*(t)$ dans I . On peut montrer que le minimum global de la fonction $f(x)$ sur le compact D est approché par le minimum global de la fonction $\mathbf{f}^*(t)$ sur l'intervalle I . Ceci est dû au fait que $\mathbf{f}^*(t)$ est la restriction de $f(x)$ à la courbe α -dense dans D . Ainsi, on peut résoudre le problème (3.9) en appliquant un algorithme plus efficace et plus performant proposé pour minimiser les fonctions dépendant d'une seule variable et satisfaisant la condition de Hölder (3.2).

3.4.1 Préliminaires et notation

Borne inférieure et supérieure

Commençons par rappeler que le diamètre d'un hyper-rectangle Ω_i pour chaque $i = 1, \dots, n$ est définie par

$$d_i = \max \{ \|x - \tilde{x}\|, x, \tilde{x} \in \Omega_i \} = \sqrt{(a_1 - b_1)^2 + \dots + (a_{i-1} - b_{i-1})^2},$$

et le centre de Ω_i est

$$c_i = \frac{1}{2}((a_1 + b_1), \dots, (a_{i-1} + b_{i-1})).$$

Soit les nombres suivants :

$$\begin{cases} L_i = \varphi_i(c_i) - h_i \left(\frac{d_i}{2}\right)^{\beta_i} \\ U_i = \psi_i(c_i) + H_i \left(\frac{d_i}{2}\right)^{\gamma_i}. \end{cases}$$

Puisque les fonctions φ_i et ψ_i ($i \geq 2$) sont höldériennes sur Ω_i alors de la condition de Hölder (3.2) on a

$$\begin{aligned} \varphi_i(y) - h_i |x - y|^{\beta_i} &\leq \varphi_i(x), \forall x, y \in \Omega_i \\ \psi_i(x) &\leq \psi_i(y) + H_i |x - y|^{\gamma_i}, \forall x, y \in \Omega_i. \end{aligned}$$

Dans les deux dernières inégalités, si une valeur de y est fixée par le centre c_i de Ω_i , alors les constantes L_i et U_i données ci-dessus, sont respectivement bornes inférieure et supérieure des fonctions φ_i et ψ_i sur l'hyper-rectangle Ω_i , pour $i \geq 2$, avec $L_1 = a$ et $U_1 = b$.

Notation et suite des paramètres

Dans tout ce travail, les constantes $h_i, H_i, \beta_i, \gamma_i$, sont supposées connues a priori et choisies telles que $(h_i)_{2 \leq i \leq n}, (H_i)_{2 \leq i \leq n}$ forment deux suites croissantes et $(\beta_i)_{2 \leq i \leq n}, (\gamma_i)_{2 \leq i \leq n}$ deux suites décroissantes. Soit $\alpha > 0$ et très petit par rapport à la région faisable D et soit, pour $i \geq 2$:

$$\begin{aligned} \delta_1 &= 1, \delta_i = 1 + \max(h_i \alpha^{\beta_{i-1}} \delta_{i-1}^{\beta_i}, H_i \alpha^{\gamma_{i-1}} \delta_{i-1}^{\gamma_i}) \\ \lambda_i &= \max(1, h_i (\alpha \delta_{i-1})^{\beta_{i-1}}, H_i (\alpha \delta_{i-1})^{\gamma_{i-1}}). \end{aligned} \tag{3.10}$$

La suite des nombres $(\delta_i)_{1 \leq i \leq n}$ est strictement croissante. En effet, pour $i = 2$, puisque $h_2 > 0$, $\beta_2 < 1$ et $H_2 > 0$, $\gamma_2 < 1$, alors $\delta_2 > \delta_1$. On suppose que $\delta_i < \delta_{i+1}$ pour tout i , on a $h_{i+1} < h_{i+2}$ et $\beta_{i+2} < \beta_{i+1}$ alors pour $\alpha \ll 1$ on a $\alpha^{\beta_{i+1}} < \alpha^{\beta_{i+2}}$. Analogiquement,

$$H_{i+1} < H_{i+2} \text{ et } \gamma_{i+2} < \gamma_{i+1} \text{ alors } \alpha^{\gamma_{i+1}} < \alpha^{\gamma_{i+2}}.$$

En choisissant convenablement la suite δ_i telle que $\delta_i < \delta_{i+1}$ on a

$$\delta_i^{\beta_{i+1}} < \delta_{i+1}^{\beta_{i+2}} \text{ et } \delta_i^{\gamma_{i+1}} \leq \delta_{i+1}^{\gamma_{i+2}},$$

cependant,

$$\begin{aligned} h_{i+1} \alpha^{\beta_{i+1}-1} \delta_i^{\beta_{i+1}} &< h_{i+2} \alpha^{\beta_{i+2}-1} \delta_{i+1}^{\beta_{i+2}} \\ H_{i+1} \alpha^{\gamma_{i+1}-1} \delta_i^{\gamma_{i+1}} &< H_{i+2} \alpha^{\gamma_{i+2}-1} \delta_{i+1}^{\gamma_{i+2}}. \end{aligned}$$

Alors

$$\delta_{i+1} < \delta_{i+2}.$$

3.4.2 Génération de courbes α -dense dans un ensemble compact

non convexe

Dans cette sous-section nous, donnerons la relation existant entre les composantes ℓ_i , $i = 1, \dots, n$ de la paramétrisation afin de générer une nouvelle classe de courbes α -dense dans un compact $D \subset \mathbb{R}^n$ dont le bord est défini par des fonctions non différentiable. Notons par μ la mesure de Lebesgue, le nombre $\alpha > 0$ est supposé inférieur à la longueur des intervalles $[\ell_i(t'_i), \ell_i(t''_i)]$, $t'_i, t''_i \in I$, $\forall i = 2, \dots, n$. Dans la suite, la sup-norm $\|x\| = \max_{1 \leq i \leq n} |x_i|$ est utilisée afin de simplifier les calculs, mais d'autres normes peuvent également être envisagées. Ensuite, en gardant les mêmes notations introduites ci-dessus, nous donnons un résultat général pour générer des courbes α -dense dans D .

Théorème 3.1 [50] Soient $\ell = (\ell_1, \ell_2, \dots, \ell_n) : I \rightarrow D$ une fonction continue et $t_2, t_3, \dots, t_n, \alpha$, des nombres strictement positifs tels que :

1) ℓ_1 est surjective.

2) Pour tout $i = \{2, 3, \dots, n\}$ et pour tout intervalle I_{i-1} de longueur t_i , il existe $t'_i, t''_i \in I_{i-1}$ tels que

$$\begin{aligned}\ell_i(t'_i) &= \varphi_i(\ell_1(t'_i), \dots, \ell_{i-1}(t'_i)) \\ \ell_i(t''_i) &= \psi_i(\ell_1(t''_i), \dots, \ell_{i-1}(t''_i)).\end{aligned}$$

3) Pour tout $i = \{2, 3, \dots, n\}$ et pour tout intervalle fermé J on a :

$$\mu(J) < t_i \implies \mu(\ell_{i-1}(J)) < \alpha.$$

Alors, la courbe paramétrée définie par $\ell(t) = (\ell_1(t), \ell_2(t), \dots, \ell_n(t))$ pour $t \in I$, est $\delta_{n-1}\lambda_n\alpha$ -dense dans D , avec δ_{n-1} et λ_n sont des paramètres données dans (3.10)

Corollaire 3.1 Soit $\ell = (\ell_1, \ell_2, \dots, \ell_n)$ une fonction de I dans D , avec ℓ_i , $(1 \leq i \leq n)$ des fonctions höldériennes de constantes v_i et d'exposants ρ_i respectivement. Et soient $t_2, t_3, \dots, t_n$ des nombres strictement positifs tels que :

a) ℓ_1 est surjective,

b) Pour tout $i = 2, \dots, n$ et pour tout intervalle I_{i-1} de longueur t_i , il existe t'_i et $t''_i \in I_{i-1}$ tels que :

$$\begin{aligned}\ell_i(t'_i) &= \varphi_i(\ell_1(t'_i), \dots, \ell_{i-1}(t'_i)), \\ \ell_i(t''_i) &= \psi_i(\ell_1(t''_i), \dots, \ell_{i-1}(t''_i)).\end{aligned}$$

c) Pour tout $i = \{2, 3, \dots, n\}$,

$$v_{i-1} \leq \frac{\alpha}{t_i^{\rho_{i-1}}}.$$

Alors, la courbe paramétrée définie par $\ell(t) = (\ell_1(t), \ell_2(t), \dots, \ell_n(t))$ pour $t \in I$, est $\delta_{n-1}\lambda_n\alpha$ -dense dans D .

3.4.3 Construction des courbes α -dense

Nous utilisons les notations suivantes :

$$\Phi_1(t) = a \text{ et } \Phi_i(t) = \varphi_i(\ell_1(t), \dots, \ell_{i-1}(t)), \text{ pour } i \geq 2$$

$$\Psi_1(t) = b \text{ et } \Psi_i(t) = \psi_i(\ell_1(t), \dots, \ell_{i-1}(t)), \text{ pour } i \geq 2$$

tel que φ_i, ψ_i sont les fonctions définies précédemment.

Théorème 3.2 Considérons le compact D donné dans le problème (3.7) et soit la courbe paramétrée $\ell = (\ell_1, \ell_2, \dots, \ell_n) : I \rightarrow D$ définie par

$$\ell_i(t) = \Phi_i(t) \cos^2\left(\frac{\alpha_i t}{2}\right) + \Psi_i(t) \sin^2\left(\frac{\alpha_i t}{2}\right), \quad \text{pour } i = 1, \dots, n$$

où $\alpha_1, \dots, \alpha_n$, et α sont des nombres strictement positifs satisfaisant la relation

$$\left(\frac{\alpha_i}{\pi}\right)^{\beta \dots \beta_{i-1}} \geq \left(\frac{1}{2\alpha}\right) ((U_{i-1} - L_{i-1})\alpha_{i-1} + 2(h_{i-1}z_{i-2}^{\beta_{i-1}} + H_{i-1}z_{i-2}^{\gamma_{i-1}})),$$

avec

$$z_i = \max_{1 \leq k \leq i} \left(\frac{1}{2} (U_k - L_k) \alpha_k + (h_k z_{k-1}^{\beta_k} + H_k z_{k-1}^{\gamma_k}) \right), \quad z_0 = 0$$

et L_i, U_i sont respectivement les bornes inférieure et supérieure des fonctions φ_i et ψ_i sur les hyper-rectangles Ω_i . Alors, la courbe paramétrée $\ell(t) = (\ell_1(t), \ell_2(t), \dots, \ell_n(t))$ pour $t \in [0, \frac{\pi}{\alpha_1}]$ est $\delta_{n-1} \lambda_n \alpha$ -dense dans D .

Preuve. Ce résultat est une conséquence du théorème 3.1. En effet, la preuve s'obtient de manière récurrente :

i) Pour $i = 2$.

- ℓ_1 est surjective par définition.

- Si on prend $t_2 = \frac{\pi}{\alpha_2}$, alors pour tout intervalle I_1 de longueur t_2 , il existe $t', t'' \in I_1$ tel que :

$$\begin{cases} \ell_2(t') = \varphi_2(\ell_1(t')) \\ \ell_2(t'') = \psi_2(\ell_1(t'')) \end{cases} \implies \begin{cases} \cos^2\left(\frac{\alpha_2 t'}{2}\right) = 1 \\ \sin^2\left(\frac{\alpha_2 t''}{2}\right) = 1 \end{cases} \implies \begin{cases} t' = \frac{2\pi k}{\alpha_2} \\ t'' = \frac{(2k+1)\pi}{\alpha_2} \end{cases}, k \in \mathbb{N}.$$

- Pour tout intervalle I_1 vérifiant $\mu(I_1) \leq \frac{\pi}{\alpha_2}$, on a $\mu(\ell_1(I_1)) < \alpha$.

$$\begin{aligned} \forall t', t'' \in I_1 : \\ |\ell_1(t') - \ell_1(t'')| &= \left| a \left(\cos^2\left(\frac{\alpha_1 t'}{2}\right) - \cos^2\left(\frac{\alpha_1 t''}{2}\right) \right) + b \left(\sin^2\left(\frac{\alpha_1 t'}{2}\right) - \sin^2\left(\frac{\alpha_1 t''}{2}\right) \right) \right| \\ &\leq (b-a) \left| \sin \frac{\alpha_1 t' - \alpha_1 t''}{2} \right| \leq \frac{1}{2} (b-a) \alpha_1 |t' - t''| \\ &\leq \frac{1}{2} (b-a) \alpha_1 \frac{\pi}{\alpha_2} \leq \alpha. \end{aligned}$$

D'après le théorème 3.1, la courbe paramétrée $\ell(t) = (\ell_1(t), \ell_2(t))$ définie sur $[0, \frac{\pi}{\alpha_1}]$ est $\lambda_2 \alpha$ -dense dans D . avec $\frac{\alpha_2}{\pi} \geq \frac{(b-a)\alpha_1}{2\alpha}$.

(ii) On suppose que le théorème est vrai pour i :

Pour tout $i = 2, \dots, n$ et pour tout intervalle I_{i-1} de longueur $t_i = \frac{\pi}{\alpha_i}$ il existe $t'_i = \frac{2\pi k}{\alpha_i}$ et $t''_i = \frac{\pi}{\alpha_i} + \frac{2\pi k}{\alpha_i} \in I_{i-1}$, $k \in \mathbb{N}$ tel que

$$\ell_i(t') = \varphi_i(\ell_1(t'), \ell_2(t'), \dots, \ell_{i-1}(t'))$$

$$\ell_i(t'') = \psi_i(\ell_1(t''), \ell_2(t''), \dots, \ell_{i-1}(t'')).$$

On suppose que l'inégalité est vraie pour $i = 2, \dots, n$, c'est-à-dire pour tout intervalle fermé J de longueur inférieure à $\frac{\pi}{\alpha_i}$, on a

$$\mu(J) \leq \frac{\pi}{\alpha_i} \implies \mu(\ell_{i-1}(J)) < \alpha,$$

par conséquent

$$\forall t', t'' \in J : |\ell_{i-1}(t') - \ell_{i-1}(t'')| \leq \left[\frac{1}{2}(U_{i-1} - L_{i-1})\alpha_{i-1} + (h_{i-1}z_{i-1}^{\beta_{i-1}} + H_{i-1}z_{i-1}^{\gamma_{i-1}}) \right] |t' - t''|^{\beta_1\beta_2\dots\beta_{i-1}} \leq \alpha.$$

Considérons l'intervalle fermé J tel que : $\mu(J) < \frac{\pi}{\alpha_{i+1}}$, pour $t', t'' \in J$, on a :

$$\begin{aligned} |\ell_i(t') - \ell_i(t'')| &= \left| \Phi_i(t') \cos^2\left(\frac{\alpha_i t'}{2}\right) + \Psi_i(t') \sin^2\left(\frac{\alpha_i t'}{2}\right) - \Phi_i(t'') \cos^2\left(\frac{\alpha_i t''}{2}\right) - \Psi_i(t'') \sin^2\left(\frac{\alpha_i t''}{2}\right) \right. \\ &\quad \left. + (\Phi_i(t') - \Psi_i(t')) \cos^2\left(\frac{\alpha_i t'}{2}\right) - (\Phi_i(t') - \Psi_i(t')) \cos^2\left(\frac{\alpha_i t''}{2}\right) \right| \\ &= \left| (\Phi_i(t') - \Psi_i(t')) (\cos^2\left(\frac{\alpha_i t'}{2}\right) - \cos^2\left(\frac{\alpha_i t''}{2}\right)) + (\Phi_i(t') - \Phi_i(t'')) \cos^2\left(\frac{\alpha_i t''}{2}\right) \right. \\ &\quad \left. + (\Psi_i(t') - \Psi_i(t'')) (\sin^2\left(\frac{\alpha_i t'}{2}\right)) \right| \\ &\leq \left| \Phi_i(t') - \Psi_i(t') \right| \left| \cos^2\left(\frac{\alpha_i t'}{2}\right) - \cos^2\left(\frac{\alpha_i t''}{2}\right) \right| + \left| \Phi_i(t') - \Phi_i(t'') \right| + \left| \Psi_i(t') - \Psi_i(t'') \right|. \end{aligned}$$

Tenons compte du fait que les constantes L_i, U_i sont respectivement les bornes inférieures et supérieures des fonctions φ_i, ψ_i sur les hyper-rectangles Ω_i . Si nous supposons que $\gamma_i \geq \beta_i$

(ou $\gamma_i < \beta_i$) pour chaque i , ensuite nous avons

$$\begin{aligned}
\forall t', t'' \in J: & \left| \ell_i(t') - \ell_i(t'') \right| \leq \frac{1}{2}(U_i - L_i)\alpha_i |t' - t''| \\
& + h_i \left(\max(|\ell_1(t') - \ell_1(t'')|, \dots, |\ell_{i-1}(t') - \ell_{i-1}(t'')|) \right)^{\beta_i} \\
& + H_i \left(\max(|\ell_1(t') - \ell_1(t'')|, \dots, |\ell_{i-1}(t') - \ell_{i-1}(t'')|) \right)^{\gamma_i} \\
\leq & \frac{1}{2}(U_{i-1} - L_{i-1})\alpha_i |t' - t''| \\
& + h_i \left(\max \left\{ \frac{1}{2}(b-a)\alpha_1, \dots, \frac{1}{2}(U_i - L_i)\alpha_{i-1} + (h_{i-1}z_{i-2}^{\beta_{i-1}} + H_{i-1}z_{i-2}^{\gamma_{i-1}}) \right\} |t' - t''|^{\beta_1\beta_2\dots\beta_{i-1}} \right)^{\beta_i} \\
& + H_i \left(\max \left\{ \frac{1}{2}(b-a)\alpha_1, \dots, \frac{1}{2}(U_{i-1} - L_{i-1})\alpha_{i-1} + (h_{i-1}z_{i-2}^{\beta_{i-1}} + H_{i-1}z_{i-2}^{\gamma_{i-1}}) \right\} |t' - t''|^{\beta_1\beta_2\dots\beta_{i-1}} \right)^{\gamma_i} \\
\leq & \left[\frac{1}{2}(U_i - L_i)\alpha_i + (h_i z_{i-1}^{\beta_i} + H_i z_{i-1}^{\gamma_i}) \right] |t' - t''|^{\beta_1\beta_2\dots\beta_i} \\
\leq & \left[\frac{1}{2}(U_i - L_i)\alpha_i + (h_i z_{i-1}^{\beta_i} + H_i z_{i-1}^{\gamma_i}) \right] \left(\frac{\pi}{\alpha_{i+1}} \right)^{\beta_1\beta_2\dots\beta_i} < \alpha.
\end{aligned}$$

Donc l'inégalité est vraie pour $i + 1$. ■

Remarque 3.1 Toutes les fonctions $\ell_i(t)$ définies au théorème 3.2 sont höldériennes de constantes $v_i > 0$ et d'exposants $\rho_i < 1$, pour chaque $i = 1, \dots, n$, donné par

$$\begin{aligned}
v_i &= \frac{1}{2}(U_i - L_i)\alpha_i + (h_i z_{i-1}^{\beta_i} + H_i z_{i-1}^{\gamma_i}) \\
\rho_i &= \beta_1 \dots \beta_i
\end{aligned}$$

Il est bien connu que si une fonction à plusieurs variables $f(x)$ est höldérienne de constante H et d'exposant β' sur le compact Ω , alors la fonction d'une seule variable $\mathbf{f}^*(t) = f(\ell(t))$ est höldérienne de constante $h = H(\max_{1 \leq i \leq n} v_i)^{\beta'}$ et d'exposant $\beta = \beta' \min(\rho_i)$ sur l'intervalle $[0, \frac{\pi}{\alpha_1}]$.

3.4.4 Exemples d'ensembles compacts α -densifiables

D'après le théorème 3.2, les illustrations des figures données ci-dessous, représentent une densification de certains ensembles hyper-rectangles non convexes (pour $n = 2$), par les supports des différents courbes α -dense $\ell(t)$ sur $[0, \frac{\pi}{\alpha_1}]$ (avec $\alpha_1 = 1$). A noter que les courbes paramétrées obtenues dans l'exemple D_1 est régulière de classe C^∞ et D_2 est continue.

Exemple 3.1 soit $D_1 = \{(x_1, x_2) \in \mathbb{R}^2 / -1 \leq x_1 \leq 1, -\frac{1}{4}x_1^2 + \frac{1}{2}x_1 - 1 \leq x_2 \leq \frac{1}{4}x_1^2 + \frac{1}{2}x_1 + 1\}$, ici, on a $\beta_2 = \gamma_2 = 1$

$$D_2 = \{(x_1, x_2) \in \mathbb{R}^2 / x_1^2 + x_2^2 \leq 1, |x_1| - (x_2 + 1)^2 \leq 0\}.$$

$$\varphi_2(x_1) = \sqrt{|x_1|} - 1 \text{ et } \psi_2(x_1) = \sqrt{1 - x_1^2} \text{ et } \beta_2 = \gamma_2 = \frac{1}{2}.$$

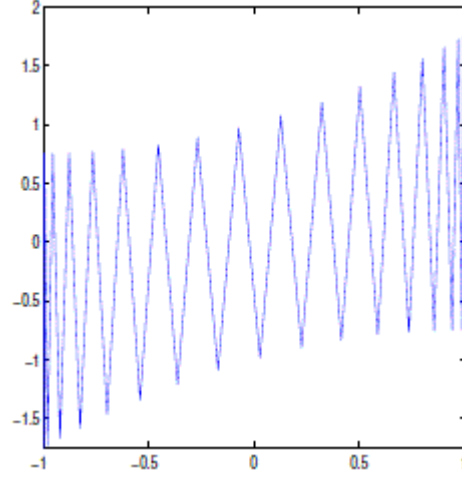


FIGURE 3.2 – La courbe α -dense $\ell(t)$ dans D_1 avec $\alpha = 0.2$

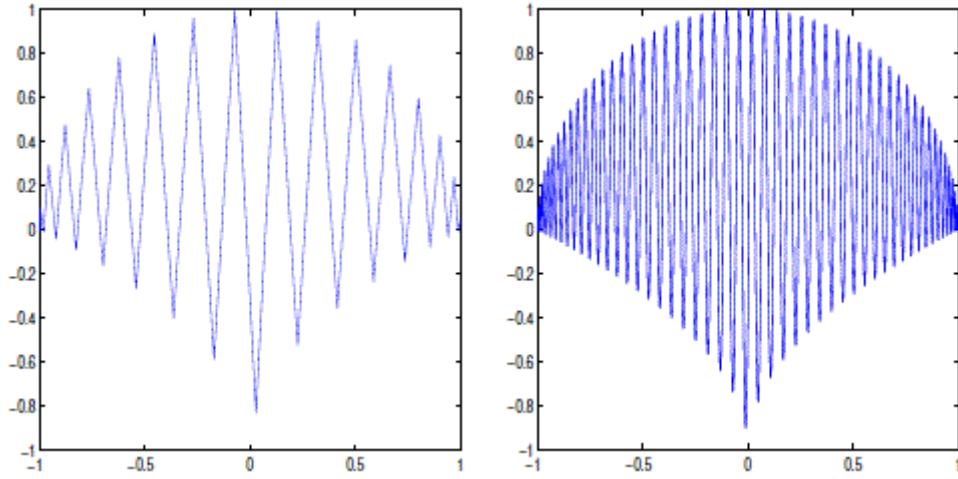


FIGURE 3.3 – La courbe α -dense $\ell(t)$ dans D_2 avec $\alpha = 0.2$ et $\alpha = 0.01$

Remarque 3.2 En particulier, si les fonctions φ_i et ψ_i ($i \geq 2$) dans la définition de l'ensemble faisable D , sont toutes des constantes et si $D = \Omega$, c'est-à-dire,

$$a = a_1 \text{ et } \varphi_i(x_1, x_2, \dots, x_{i-1}) = a_i, \quad 2 \leq i \leq n.$$

$$b = b_1 \text{ et } \psi_i(x_1, x_2, \dots, x_{i-1}) = b_i, \quad 2 \leq i \leq n.$$

Alors l'ensemble faisable D devient un hyper-rectangle de $\mathbb{R}^n$. D'après le théorème 3.2, l'hyper-rectangle D est α -densifié par la courbe paramétrée $\ell(t) = (\ell_1(t), \dots, \ell_n(t))$ définie sur $[0, \frac{\pi}{\alpha_1}]$, par l'expression suivante

$$\ell_i(t) = \left(\frac{a_i - b_i}{2}\right) \cos^2\left(\frac{\alpha_i t}{2}\right) + b_i, \quad 1 \leq i \leq n,$$

où les constantes $\alpha_1, \dots, \alpha_n$ satisfont la relation :

$$\left(\frac{\alpha_i}{\pi}\right) \geq \left(\frac{\alpha_{i-1}}{2\alpha}\right)(b_{i-1} - a_{i-1}), \quad 2 \leq i \leq n.$$

Mais dans l'implémentation de la méthode de transformation réductrice Alienor, on peut choisir ces dernières constantes comme :

$$\alpha_1 = 1, \quad \alpha_i = \left(\frac{\pi}{\alpha}\right)^{i-1} \prod_{k=1}^{i-1} (b_{i-1} - a_{i-1}), \quad 2 \leq i \leq n.$$

Remarque 3.3 On doit mentionner que, dans la méthode de la transformation réductrice Alienor, le nombre de points d'évaluation nécessaires pour atteindre le minimum global, avec la précision ε , de la fonction objectif d'une seule variable $\mathbf{f}^*(t) = f(\ell(t))$, dépend de la longueur $L_{\alpha, \ell}$ de la courbe α -dense $\ell(t)$ dans l'ensemble faisable, qui permet de convertir le problème multidimensionnel avec contraintes (3.7) en un problème unidimensionnel sans contrainte. Comme pour tout $\alpha > 0$, l'ensemble des courbes α -dense dans un compact D de $\mathbb{R}^n$, contient une courbe de longueur minimale, voir Ziadi et al.[66], il est donc très intéressant de trouver d'autres courbes de petite longueur afin d'améliorer la méthode de transformation réductrice Alienor.

3.4.5 La méthode de la transformation réductrice combinée avec la méthode TPA

Considérons la courbe $\ell(t) = (\ell_1(t), \ell_2(t), \dots, \ell_n(t))$, $\delta_{n-1} \lambda_n \alpha$ -dense dans D , pour $t \in [0, \frac{\pi}{\alpha_1}]$ qui est obtenu à partir du théorème 3.2. En utilisant la transformation $x_i = \ell_i(t)$, pour $i = 1, \dots, n$, le problème initial (3.7) est approché par le problème unidimensionnel sans contrainte,

$$\left\{ \min_{t \in [0, \frac{\pi}{\alpha_1}]} \mathbf{f}^*(t) \right. \quad (3.11)$$

où $\mathbf{f}^*(t) = f(\ell(t))$ est une fonction höldérienne non différentiable sur l'intervalle $[0, \frac{\pi}{\alpha_1}]$ de constante h et d'exposant $1/\beta$, $\beta > 1$. L'algorithme *TPA* procède alors à l'évaluation du minimum global du sous-estimateur tangentes par morceaux aux fonctions paraboliques de $\mathbf{f}^*(t) = f(\ell(t))$ sur le domaine de recherche $[0, \frac{\pi}{\alpha_1}]$ par la construction d'une suite croissante de fonctions minorantes tangentes par morceaux. L'algorithme de la méthode combinée transformation réductrice Alienor et la méthode *TPA*, notée *Alienor_TPA_CTs*, est donné comme suit :

Algorithme 13 *Alienor_TPA_CTs*

Initialisation : $k \leftarrow 2$, $t_1 \leftarrow 0$, $t_2 \leftarrow \frac{\pi}{\alpha_1}$.

Etape k : $t_1, t_2, \dots, t_k$ sont ordonné tel que $0 = t_1 < t_2 < \dots < t_k = \frac{\pi}{\alpha_1}$.

for $i = 2$ to k **do**

$$\omega_i = \varsigma_i + \frac{\beta(\mathbf{f}^*(t_{i-1}) - \mathbf{f}^*(t_i))}{2he_i^{(1/\beta)-1}}, \mathbf{T}_i = \min \{ \mathbf{f}^*(t_{i-1}) - h(\omega_i - t_{i-1})^{1/\beta}, \mathbf{f}^*(t_i) - h(t_i - \omega_i)^{1/\beta} \}$$

$$\text{avec } \varsigma_i = \frac{t_{i-1} + t_i}{2} \text{ et } e_i = \frac{t_i - t_{i-1}}{2}$$

end for

$$\mathbf{T}_\theta = \min \{ \mathbf{T}_i, 2 \leq i \leq k \}, t_\theta = \omega_\theta$$

if $|t_\theta - t_{\theta-1}| > \varepsilon$ **then**

$$t_{k+1} = \omega_\theta, k \leftarrow k + 1 \text{ et aller à l'Etape } k$$

else

$$f_{opt} = \min \{ \mathbf{f}^*(t_i) : 1 \leq i \leq k \}, \text{ Stop.}$$

end if

3.4.6 Relation entre la densité α et la précision ε

Théorème 3.3 Soit ℓ une courbe α -dense dans D et ε une précision avec laquelle on cherche le minimum global du problème 3.7 sur le compact D , alors on a

$$\begin{cases} \varepsilon(\alpha) = 2H(\delta_{n-1}\lambda_n\alpha)^{1/\beta} \\ \varepsilon(\alpha) \rightarrow 0. \\ \alpha \rightarrow 0 \end{cases}$$

Preuve. Supposons que x^* est un minimiseur global exact de la fonction objectif $f(x)$ sur le compact D et t^* un minimiseur global exact de la fonction $\mathbf{f}^*(t) = f(\ell(t))$ sur l'intervalle

$I = [0, \frac{\pi}{\alpha_1}]$. Puisque la courbe paramétrée $\ell(t)$ est $\delta_{n-1}\lambda_n\alpha$ -dense dans D , alors il existe $t' \in I$ tel que

$$\|x^* - \ell(t')\| \leq \delta_{n-1}\lambda_n\alpha.$$

L'algorithme de *TPA* nous donne un minimiseur t'' de la fonction $\mathbf{f}^*(t)$ sur la courbe $\ell(t)$ pour $t \in I$, satisfaisant

$$|\mathbf{f}^*(t^*) - \mathbf{f}^*(t'')| \leq \frac{\varepsilon}{2}.$$

Alors nous avons

$$|f(x^*) - \mathbf{f}^*(t'')| \leq \underbrace{|f(x^*) - \mathbf{f}^*(t')|}_{\leq \frac{\varepsilon}{2}} + \underbrace{|\mathbf{f}^*(t') - \mathbf{f}^*(t'')|}_{\leq \frac{\varepsilon}{2}} \leq \varepsilon,$$

et comme la fonction $f(x)$ est höldérienne sur Ω , il s'ensuit que

$$|f(x^*) - \mathbf{f}^*(t')| = |f(x^*) - f(\ell(t'))| \leq H \|x^* - \ell(t')\|^{1/\beta}.$$

Alors pour avoir la précision ε , il suffit de choisir :

$$\alpha = \left(\frac{\varepsilon}{2H}\right)^\beta \frac{1}{\delta_{n-1}\lambda_n}.$$

■

3.5 Sur l'estimation de la constante de Hölder h

Les algorithmes donnés dans ce travail sont appliqués à une classe de fonctions de Hölder où les constantes h et α sont connues a priori. Cependant, dans la plupart des situations et en pratique, ces constantes ne sont pas faciles à déterminer. De plus, l'algorithme qui n'utilise que les valeurs h et α sur tout le domaine faisable convergera lentement, surtout lorsque la fonction objectif est plutôt plate au voisinage du minimum global car, l'algorithme génère une grille très dense dans ce voisinage. Pour aller plus vite, plusieurs dispositifs peuvent être mis en place. En effet, il est nécessaire de déterminer l'approximation locale de la constante h lors de l'évolution de l'algorithme. Cette technique qui estime de manière adaptative la constante de Hölder locale sur différents sous-intervalles $[x_{i-1}, x_i]$ de domaine est utilisée

pour accélérer la recherche du minimum global. La constante h peut être naturellement estimée par

$$h_k = \eta_k \max_{2 \leq i \leq k} \left\{ \frac{|f(x_{i-1}) - f(x_i)|}{|x_{i-1} - x_i|^a} \right\},$$

où $(x_k)_{k \geq 1}$ sont les points d'évaluation générés par l'algorithme à l'itération k et $(\eta_k)_{k \geq 1}$ est une suite décroissante de nombres réels convergeant vers 1. Pour plus de détails voir [4], [36], [37], [52], [60].

3.6 Conclusion

La généralisation de la méthode de Piyavskii-Shubert aux cas des fonctions höldériennes à plusieurs variables est difficile à réaliser, car le calcul des minima locaux nécessite la détermination de l'intersection de plusieurs hyper surfaces paraboliques $f(x_i) - h \|x - x_i\|^{1/\beta}$ [21]. Quelques procédures de généralisation directe des méthodes de Piyavskii-Shubert ont été élaborées. Dans ce chapitre, On a proposé un nouvel algorithme appelé *Alienor_TPA* basé sur la technique de réduction du problème multidimensionnel initial à un problème unidimensionnel en utilisant la transformation réductrice Alienor qui fait intervenir les courbes α -dense balayant le domaine faisable. La méthode couplée, ainsi obtenue, est simple et efficace et la convergence a été prouvées. Une autre méthode mixte appelée *Alienor_TPA_CTs*, a été proposée dans ce chapitre pour résoudre une classe de problèmes avec contraintes où la fonction objectif et les contraintes sont höldériennes à plusieurs variables [50]. Avant de commencer la procédure de minimisation, nous approchons d'abord l'ensemble faisable non convexe par une courbe paramétrique α -dense. Pour minimiser la fonction transformée d'une seule variable, nous avons proposé la méthode *TPA* introduite dans le chapitre 2 qui nécessite la connaissance des constantes h et β .

Concernant la méthode de la transformation réductrice, nous avons apporté de nouvelles idées pour pouvoir appliquer cette approche aux problèmes d'optimisation des fonctions höldériennes. Cette méthode est simple et sa mise en œuvre numérique est très facile et ne nécessite pas beaucoup d'espace de stockage. Le nombre de points d'évaluation de la fonction objectif pour atteindre le minimum global, dépend de la longueur de la courbe α -dense. De ce fait un intérêt particulier est accordé au choix de la courbe qui α -densifie le domaine

d'optimisation. En général, pour α fixé, plus la longueur est petite, plus le temps de calcul sera court. Il est donc naturel d'étudier d'autres aspects des courbes α -dense qui nous permettraient, peut être, de construire des transformations réductrices plus simples pour améliorer le temps de calcul.

4.1 Les testes numériques pour les problèmes unidimensionnelles

Pour illustrer l'efficacité des deux algorithmes proposés TPA et $KL_{\alpha}TPA$, trois séries de fonctions tests standard et leurs données correspondantes sont répertoriées dans les tableaux (4.1-4.3). Ces fonctions de test ont des propriétés différentes comme non convexes, non différentiables, de Lipschitz et de Hölder, beaucoup d'entre elles ont au moins deux minima locaux. La plupart de ces fonctions sont tirées de la littérature d'optimisation de Lipschitz et Hölder [21], [26], [36] et [52]. Les expériences ont été réalisées sur PC avec intel (R) Core(TM)i3-3217U CPU@1.80GHz(CPUs), ~1.8GHz et 4096MB RAM. Les codes sont implémentés dans MATLAB R2009b. Dans les expériences données dans les tableaux, 4.4, 4.5 et 4.7, chaque calcul s'est terminé comme un succès lorsqu'un point faisable x_k satisfaisant

$$|x_{k-1} - x_k| \leq 10^{-4}(b - a).$$

Dans la phase 1 de l'algorithme $KL_{\alpha}TPA$ on utilise la tolérance $\delta = 10^{-2}$. Les techniques proposées ont été comparées avec des méthodes existantes listées dans le Tab.4.4. Dans tous les tableaux, la forme en gras indique les meilleurs résultats en termes de temps $T(s)$ et de nombre de points d'évaluations Ev . Tous les minima ε -globaux ont été trouvés par tous les algorithmes dans toutes les expériences présentées dans les tableaux, 4.4, 4.5 et 4.7. L'étude comparative indique que les résultats de TPA et $KL_{\alpha}TPA$ sont très encourageants.

Remarque 4.1 Si f est une fonction lipschitzienne de constante L alors f est une fonction hölderienne avec la constante de Hölder $h = L \|b - a\|^{1-\alpha}$. Puisque, comme f est lipschitzienne alors,

$$|f(x) - f(y)| \leq L \|x - y\|, \forall x, y \in [a, b]$$

et comme $L \|x - y\| = L \|x - y\|^{1-\alpha} \|x - y\|^\alpha$ donc,

$$|f(x) - f(y)| \leq \underbrace{L \|x - y\|^{1-\alpha}}_h \|x - y\|^\alpha.$$

TABLE 4.1 – Fonctions tests lipschitziennes différentiables.

N°	Fonction	Intervalle	Constante de Lipschitz L	Ref.
1	$\sin x + \sin \frac{10}{3}x$	[2.7,7.5]	4.29	[21]
2	$(-16x^2 + 24x - 5)\exp(-x)$	[1.9,3.9]	3	[21]
3	$(3x - 1.4)\sin 18x$	[0,1.2]	36	[21]
4	$(-x - \sin x)\exp(-x^2)$	[-10,10]	2.5	[21]
5	$\ln x - 0.84x + 3$	[2.7,7.5]	6	[26]
6	$\sin x + \sin \frac{2}{3}x$	[3.1,20.4]	1.7	[26]
7	$-x \sin x$	[0,10]	11	[26]
8	$2 \cos x + \cos 2x$	[-1.57,6.28]	3	[26]
9	$\sin^3 x + \cos^3 x$	[0,6.28]	2.2	[26]
10	$-x^{\frac{2}{3}} + (x^2 - 1)^{1/3}$	[0.001,0.99]	8.5	[26]
11	$\exp(-x)\sin 2\pi x$	[0.4]	6.5	[26]
12	$x^6 - 15x^4 + 27x^2 + 250$	[-4,4]	2520	[36]
13	$-(x - \sin x)\exp(-x^2)$	[-10,10]	1.3	[26]
14	$-x + \sin 3x - 1$	[0,6.5]	4	[26]
15	$\frac{1}{6}x^6 - \frac{52}{25}x^5 + \frac{39}{80}x^4 + \frac{71}{10}x^3 - \frac{79}{20}x^2 - x + \frac{1}{10}$	[-1.5,11]	13870	[26]
16	$-\sum_{k=1}^5 k \sin[(k+1)x + k]$	[-10,10]	67	[36]
17	$-\sum_{k=1}^5 k \cos[(k+1)x + k]$	[-10,10]	67	[36]
18	$2(x-3)^2 + \exp\left(\frac{x^2}{2}\right)$	[-3,3]	85	[36]
19	$\frac{x^2 - 5x + 6}{(x^2 + 1)}$	[-5.5]	6.5	[36]
20	$-\cos x + \sin 5x - 1$	[0,7]	5.951	[52]
21	$\sum_{k=1}^5 \cos[(k+1)x]$	[-10,10]	18.119	[52]

TABLE 4.2 – Fonctions tests lipschitziennes non différentiables.

N°	Fonction	Intervalle	Constante de Lipschitz L	Ref.
22	$\begin{cases} -\cos 5x, & x \leq \frac{3}{2}\pi \\ -\cos x, & \text{sinon} \end{cases}$	[0,18]	4.999	[52]
23	$\begin{cases} -\sin x, & x \leq \pi \\ -\sin 5x, & \text{sinon} \end{cases}$	[-10,10]	4.999	[52]
24	$-x \sin x - 6$	[-10,10]	9.632	[52]
25	$- x \sin x + 1.5$	[-10,10]	9.632	[52]
26	$\begin{cases} -\sin x, & \sin x > \cos x \\ -\cos x, & \text{sinon} \end{cases}$	[-10,10]	1	[52]
27	$\begin{cases} (x-2)^2, & \text{si } x \leq 3 \\ 2\ln(x-2)+1 & \text{sinon} \end{cases}$	[0,6]	4	[36]

TABLE 4.3 – Fonctions tests höldériennes non différentiables.

N°	Fonction	Intervalle	constante de Hölder h	Ref.
28	$\begin{cases} 2\sqrt{ x-\frac{1}{4} } & \text{if } x \leq \frac{1}{2} \\ 1 & \text{sinon} \end{cases}$	[0,1]	2	[21]
29	$\min\{\sqrt{ x+4 }-1, \sqrt{ x+1 } + 1.005, \sqrt{ x-3 }+0.5\}$	[-5,5]	1	[64]
30	$ 0.5 + 3.5\sqrt{1-x^2} $	[-2,2]	7	[47]
31	$\sum_{k=1}^3 \frac{1}{k} \left \sin\left(\left(\frac{3}{k} + k\right)x + \frac{1}{k}\right) \right x-k ^{\frac{1}{4}}$	[-5,0]	6.83	[47]
32	$\sum_{k=1}^5 k \sin((3k+1)x + k) x-k ^{\frac{1}{a}}$	[0,4]	$15 + \sum_{k=1}^5 k 2^{1-a} (3k+1)^a (10-k)^a$	[36]
33	$\begin{cases} -\sqrt{2x-x^2} & \text{if } x \leq 2 \\ -\sqrt{-x^2+8x-12} & \text{sinon} \end{cases}$	[0,6]	6.94	[36]
34	$\frac{2\sin x - 1}{\sqrt{ x+1 }+2}$	[-5,5]	7.8	New

TABLE 4.4 – Comparaison des résultats entre TPA , $KL_{\alpha-}TPA$ et les algorithmes existants.

Numéro du problème		GPA		SPA		TPA		$KL_{\alpha-}TPA$ $r = 1$	
N°	$1/\alpha$	Ev	$T(s)$	Ev	$T(s)$	Ev	$T(s)$	Ev	$T(s)$
1	2	779	1.743	1050	1.522	780	3.141	78	0.112
	3	1610	3.772	1977	3.816	1621	1.813	56	0.073
	4	2456	3.832	2804	6.363	2477	5.432	62	0.052
	4/3	/	/	315	1.919	327	0.177	258	0.167
2	2	2613	4.489	2375	4.244	2615	6.331	174	1.882
	3	4681	16.173	3810	9.295	4747	11.983	4748	14.093
	4	6306	23.942	4693	13.092	4943	15.722	4944	15.031
	4/3	/	/	1075	1.644	1126	1.172	560	0.449
3	2	1080	1.139	1155	1.911	1077	1.892	64	0.172
	3	3157	9.763	4335	11.479	3159	6.271	3160	6.713
	4	4786	12.148	6985	24.355	4787	13.641	4788	13.235
	4/3	/	/	332	0.509	333	0.229	82	0.119
4	2	2485	5.385	2457	5.151	2483	6.381	198	0.273
	3	8337	36.876	8287	34.209	8335	33.223	212	0.324
	4	9217	39.213	8951	38.372	9215	39.922	222	0.275
	4/3	/	/	318	0.217	324	0.597	128	0.222
5	2	881	0.772	827	1.309	879	1.791	916	0.868
	3	2255	4.545	3417	7.388	2257	3.643	2288	3.878
	4	3585	7.283	5279	16.146	3583	8.234	3612	8.309
	4/3	/	/	199	0.162	202	0.159	239	0.219

TABLE 4.4 – (Suite).

Numéro du problème		<i>GPA</i>		<i>SPA</i>		<i>TPA</i>		<i>KL_a-TPA</i> <i>r</i> = 1	
<i>N</i> ^o	1/ <i>α</i>	<i>E v</i>	<i>T</i> (s)	<i>E v</i>	<i>T</i> (s)	<i>E v</i>	<i>T</i> (s)	<i>E v</i>	<i>T</i> (s)
6	2	899	0.666	833	1.521	900	1.319	124	0.210
	3	1991	4.316	2127	4.362	1998	4.421	124	0.131
	4	3114	6.298	4497	11.383	3122	8.353	120	0.245
	4/3	/	/	345	0.199	359	0.482	90	0.103
7	2	1128	0.952	1045	1.773	1131	2.025	212	0.444
	3	2234	5.277	1913	3.989	2238	4.812	204	0.282
	4	2365	7.133	4349	12.552	3349	8.311	186	0.266
	4/3	/	/	458	0.341	469	0.339	174	0.323
8	2	2137	4.562	2019	4.425	2147	4.982	168	0.832
	3	3695	10.252	4159	11.393	3662	9.213	156	0.701
	4	5450	13.841	5051	15.281	5831	17.938	154	0.454
	4/3	/	/	780	0.757	891	1.024	146	0.606
9	2	2131	2.772	2075	7.918	1853	4.315	112	0.534
	3	3773	10.251	4083	10.681	3249	7.824	3336	8.312
	4	4797	11.262	5611	20.223	4503	12.017	4580	13.725
	4/3	/	/	686	0.627	585	0.415	166	0.308
10	2	5221	27.724	5163	17.071	5219	16.014	5220	17.122
	3	8895	72.712	8835	48.134	8893	40.422	8894	40.839
	4	11236	102.251	10987	75.166	11234	59.614	11234	65.487
	4/3	/	/	2001	3.075	547	0.446	642	0.591
11	2	1035	1.962	1173	2.595	1033	2.021	68	0.542
	3	3092	8.817	4494	12.422	3099	7.329	296	0.544
	4	4955	14.717	7751	28.728	4956	14.981	66	0.163
	4/3	/	/	314	0.236	322	0.535	122	0.085

TABLE 4.4 – (Suite).

Numéro du problème		<i>GPA</i>		<i>SPA</i>		<i>TPA</i>		<i>KL_a-TPA</i> <i>r</i> = 1	
<i>N</i> ^o	$1/\alpha$	<i>Ev</i>	<i>T(s)</i>	<i>Ev</i>	<i>T(s)</i>	<i>Ev</i>	<i>T(s)</i>	<i>Ev</i>	<i>T(s)</i>
12	2	5569	17.289	5475	18.914	5567	18.222	8040	35.152
	3	11325	67.468	11073	61.882	11325	63.818	7504	32.505
	4	12673	72.981	15839	119.571	12671	78.033	6172	25.093
	4/3	/	/	1921	3.019	1911	3.406	6852	31.217
13	2	9265	43.351	9179	40.152	9263	41.872	652	1.209
	3	16385	150.033	16383	131.725	16383	133.411	672	1.121
	4	16385	130.125	16383	136.215	16383	134.218	662	1.175
	4/3	/	/	2481	4.079	2483	4.079	574	0.513
14	2	890	1.741	827	1.423	894	1.672	314	0.415
	3	1609	3.613	1373	2.932	1608	1.924	62	0.023
	4	2122	4.684	1771	3.711	2135	4.542	58	0.014
	4/3	/	/	369	0.357	383	0.265	72	0.072
15	2	1051	2.081	936	2.022	1052	2.793	8384	44.612
	3	1875	5.874	1580	3.434	1901	5.036	8528	45.015
	4	2671	5.402	1975	3.952	2687	7.059	7706	56.237
	4/3	/	/	430	0.485	449	1.099	6394	22.353
16	2	4132	10.943	4365	11.625	4131	12.163	982	2.229
	3	8624	37.672	11966	69.448	8622	45.224	1050	2.192
	4	10020	46.824	15997	119.147	10028	50.061	1018	2.165
	4/3	/	/	724	1.615	728	1.717	729	0.777

TABLE 4.4 – (Suite).

Numéro du problème		<i>GPA</i>		<i>SPA</i>		<i>TPA</i>		<i>KL_αTPA</i> <i>r</i> = 1	
<i>N</i> ^o	$1/\alpha$	<i>E</i> <i>ν</i>	<i>T</i> (<i>s</i>)	<i>E</i> <i>ν</i>	<i>T</i> (<i>s</i>)	<i>E</i> <i>ν</i>	<i>T</i> (<i>s</i>)	<i>E</i> <i>ν</i>	<i>T</i> (<i>s</i>)
17	2	4132	10.944	4365	11.629	4131	12.165	982	2.222
	3	8741	41.965	9496	45.224	8741	39.267	944	1.864
	4	9349	40.746	15532	116.338	9352	47.225	936	1.785
	4/3	/	/	543	3.409	549	0.816	632	1.046
18	2	2512	4.752	3739	10.327	2553	5.871	832	1.609
	3	4560	14.489	5192	15.252	4662	13.395	744	1.255
	4	5663	16.908	6123	19.064	6063	20.738	606	1.022
	4/3	/	/	990	1.879	1018	1.797	1036	1.567
19	2	4477	12.332	5543	16.921	4481	12.083	614	0.481
	3	5905	20.458	7795	30.625	5910	19.164	712	1.235
	4	6997	24.084	8860	37.593	7023	26.761	1004	0.812
	4/3	/	/	2317	5.158	2313	1.405	488	0.474
20	2	817	0.68	944	1.78	815	1.82	74	0.15
	3	2229	4.01	3210	8.29	2232	5.16	78	0.06
	4	3440	6.96	5455	17.80	3430	9.98	64	0.15
	4/3	/	/	264	0.35	267	0.37	78	0.26
21	2	4265	13.71	2415	6.22	4267	2.65	1108	2.15
	3	8881	46.27	5793	21.18	8879	2.81	1212	2.52
	4	9705	50.81	13953	105.14	9709	3.05	1180	2.53
	4/3	/	/	555	0.84	1183	1.48	784	1.54
22	2	4447	11.20	4641	14.97	4445	13.72	5394	13.33
	3	8320	40.62	8153	37.10	8318	37.69	9347	36.18
	4	10403	55.84	13294	91.29	10404	56.23	11385	52.81
	4/3	/	/	1822	3.40	1708	3.42	2367	4.28

TABLE 4.4 – (Suite).

Numéro du problème		<i>GPA</i>		<i>SPA</i>		<i>TPA</i>		KL_{a_TPA} $r = 1$	
N°	$1/\alpha$	Ev	$T(s)$	Ev	$T(s)$	Ev	$T(s)$	Ev	$T(s)$
23	2	4935	13.67	6144	21.51	4933	15.52	1104	2.86
	3	12135	85.55	11531	69.55	12133	77.42	1200	3.24
	4	12550	84.77	13633	96.70	12548	84.90	1158	2.91
	4/3	/	/	1806	4.01	1840	3.99	738	1.82
24	2	913	1.70	1309	2.98	911	0.44	274	0.36
	3	2382	6.47	2880	7.36	2380	0.46	306	0.38
	4	3777	9.67	5757	25.09	3775	0.48	292	0.31
	4/3	/	/	298	0.39	310	0.24	192	0.49
25	2	1615	3.39	2403	6.36	1613	0.82	516	0.72
	3	3821	10.59	4665	20.51	3819	0.69	580	0.84
	4	5633	18.58	8519	39.78	5631	1.02	552	0.80
	4/3	/	/	565	0.75	589	0.58	368	0.61
26	2	870	0.44	793	1.36	869	0.10	88	0.21
	3	2106	6.54	1858	3.96	2109	0.13	84	0.19
	4	3281	8.39	2907	6.76	3291	0.42	82	0.15
	4/3	/	/	181	0.21	192	0.07	59	0.22
27	2	1382	2.204	1808	3.624	1450	3.271	148	0.342
	3	2966	7.862	2982	6.802	2905	7.512	172	0.432
	4	3533	6.763	3758	9.291	3200	7.891	110	0.307
	4/3	/	/	781	1.232	568	0.527	172	0.359

TABLE 4.5 – Comparaison des résultats pour les fonctions tests höldériennes.

Numéro du problème		<i>GPA</i>		<i>SPA</i>		<i>TPA</i>		<i>KL_α-TPA</i> <i>r</i> = 1	
<i>N</i> °	$1/\alpha$	<i>Ev</i>	<i>T(s)</i>	<i>Ev</i>	<i>T(s)</i>	<i>Ev</i>	<i>T(s)</i>	<i>Ev</i>	<i>T(s)</i>
28	2	143	1.283	104	0.237	104	0.127	104	0.173
29	2	51	0.124	15	0.278	14	0.272	10	0.088
30	2	73	0.069	76	0.134	55	0.091	225	0.362
31	4	1038	2.748	1061	2.306	955	2.118	411	0.762
32	2	857	2.355	811	2.232	856	1.519	206	0.497
	3	1370	4.131	1358	3.648	1385	2.641	302	0.701
	4	1763	5.636	1566	4.337	2050	4.803	326	0.723
33	2	6286	19.30	9451	53.606	2445	23.78	188	0.355
34	2	795	1.159	732	1.222	795	0.731	61	0.092

4.1.1 Discussion et remarques finales.

Dans cette thèse, nous développons l'algorithme de Piyavskii pour résoudre un problème d'optimisation globale unidimensionnel sous la condition de Hölder. Le critère utilisé pour comparer l'efficacité des algorithmes est en termes de temps et de nombre d'évaluations de fonctions. Dans le tableau 4.4, nous utilisons les entiers $1/\alpha = 2, 3$ et 4 car il est possible d'utiliser des expressions explicites pour les coordonnées du point d'intersection $(\mathbf{x}_i, \mathbf{y}_i)$ et le deuxième choix est de considérer le cas non entier comme $1/\alpha = 4/3$. Dans ce dernier cas, nous ne pouvons pas calculer l'expression analytique $(\mathbf{x}_i, \mathbf{y}_i)$, mais nous pouvons trouver l'approximation par n'importe quel algorithme de recherche linéaire, comme la méthode de Fibonacci et le nombre d'or [21], [39]. Mais l'inconvénient est d'utiliser ces techniques de recherche linéaire à chaque itération dans l'algorithme principal de Piyavskii, donc cela prend énormément de temps de calcul. Les algorithmes *TPA* et *KL_α-TPA* que nous avons proposés sont valables pour toute valeur donnée $0 < \alpha < 1$. Pour cette raison, nous avons comparé

tous les algorithmes qui existent tels GPA , SPA avec les algorithmes TPA et $KL_{\alpha}TPA$ sauf pour le cas où $\alpha = 1/m$ et m n'est pas un nombre entier (comme on peut le voir dans le Tableau 4.4).

Les tableaux 4.4 et 4.5 présentent le résumé des résultats numériques obtenus par les méthodes existantes et par les algorithmes qui ont été proposés. Les résultats numériques donnés par l'algorithme $KL_{\alpha}TPA$ sont résumés dans le tableau 4.4, nous avons pris la valeur du paramètre $r = 1$ qu'à titre d'exemple, les résultats sont très satisfaisants pour la majorité des fonctions tests, que ce soit en terme temps de calcul $T(s)$ ou nombre de points d'évaluation Ev , par rapport aux algorithmes GPA , SPA et TPA . On observe que le nombre de points d'évaluation Ev augmente avec l'augmentation de la valeur de $1/\alpha$.

TABLE 4.6 – Comparaison avec des pourcentages.

		GPA	SPA	TPA
Ev	TPA et GPA	44%	/	54%
	TPA et SPA	/	39%	58%
$T(s)$	TPA et GPA	51%	/	49%
	TPA et SPA	/	30%	70%

Les résultats numériques donnés dans les tableaux ci-dessus prouvent que la méthode *TPA* améliore les résultats des algorithmes proposés dans la plupart des exemples. Une comparaison en pourcentages dans le tableau 4.6 entre les algorithmes *GPA*, *SPA* et l'algorithme *TPA* confirme l'amélioration des résultats pour un nombre considérable d'exemples. D'après les résultats donnés dans les tableaux précédents, on remarque que le nombre de points d'évaluation effectués par les algorithmes *TPA* et *GPA* est quasiment le même dans plusieurs exemples, cela signifie que le point d'intersection z_i utilisé dans l'algorithme *TPA* est généralement plus proche du point d'intersection exact utilisé dans *GPA*, en comparaison avec le point utilisé par *SPA*.

Une autre technique notée *TPA_SPA* peut être utilisée dans laquelle le choix des valeurs $(z_i, \mathbf{T}_i)$ données par (3.2) et (3.3) est différent. En effet, l'idée consiste, après avoir déterminé les valeurs $(z_i, \mathbf{T}_i)$ et $(y_i, \mathbf{m}_i)$ sur chaque sous-intervalle $[x_{i-1}, x_i]$, on prend la valeur $\mathbf{T} = \max(\mathbf{T}_i, \mathbf{m}_i)$. Si $\mathbf{m}_i \geq \mathbf{T}_i$, alors on fixe $z_i = y_i$, sinon on laisse z_i et on continue avec l'algorithme *TPA*.

Aussi, une autre technique notée $\overline{KL}$, dans laquelle la valeur de z_i est remplacée par la valeur $\tilde{y}_i = \operatorname{argmin}_{[x_{i-1}, x_i]} B_i(x)$ et $\mathbf{T}_i = \min\{p_{i-1}(\tilde{y}_i), p_i(\tilde{y}_i)\}$, (voir l'expression (2.36)). Nous avons également présenté quelques résultats numériques obtenus en comparant les algorithmes $KL_\alpha\text{-}TPA$, *TPA_SPA* et $\overline{KL}$ pour la différentes valeurs du paramètre r et de l'exposant de Hölder α . Dans la méthode $KL_\alpha\text{-}TPA$ nous avons choisi $r = 0.25, 0.5$ et 0.8 et $\alpha = 2, 3$ et 4 et nous observons que lorsque r est proche de 1 les résultats sont relativement meilleurs (voir Tableau 4.7).

TABLE 4.7 – Le nombre de points d'évaluation pour différentes valeurs de r et TPA_SPA et $\overline{KL}$.

Numéro du problème	$1/\alpha$	KL_{α_TPA}			TPA_SPA	$\overline{KL}$
N°		$r = 0.25$	$r = 0.5$	$r = 0.8$		$r = 1$
22	2	6320	5752	5504	4450	4956
	3	10863	9845	9485	8318	8313
	4	13083	11935	11543	10404	9425
23	2	2304	1486	1224	4933	5003
	3	2896	1890	1374	12133	8691
	4	3056	1922	1356	12548	10048
24	2	556	386	310	911	1075
	3	734	460	352	2380	2845
	4	790	482	348	3775	4476
25	2	1052	732	588	1613	1925
	3	1388	860	672	3819	4619
	4	1492	904	664	5631	6723
26	2	4295	4099	3957	3518	3868
	3	8236	7790	7588	7079	6739
	4	9914	9412	9188	8687	8365
27	2	248	166	124	1450	1755
	3	273	170	120	2905	315
	4	274	162	112	3200	178

L'algorithme qui nous a donné des résultats très satisfaisants est $KL_{\alpha}TPA$ (comme on peut le voir dans les tableaux 4.4, 4.5) et qui est composé de deux phases. Dans la phase 1, on utilise la technique KL_{α} et elle vise à réduire considérablement l'intervalle faisable à un intervalle D' qui contient souvent l'optimum global avec quelques itérations. Dans la phase 1, nous avons utilisé le critère d'arrêt suivant

$$|F_{\varepsilon} - f_{\varepsilon}| \leq \delta + L_{\alpha},$$

où l'algorithme a été testé pour deux valeurs différentes de tolérance $\delta = 10^{-2}$ et $\delta = 10^{-4}$, puis on applique la phase 2 de minimisation globale, en utilisant l'algorithme TPA sur l'intervalle D' dont la convergence est prouvée, avec le critère d'arrêt

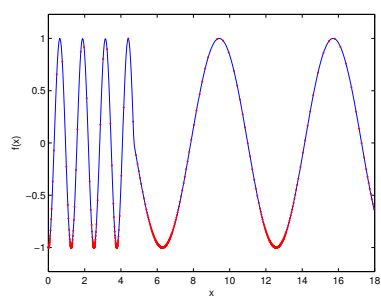
$$|x_k - x_{k-1}| \leq 10^{-6}(b - a).$$

Pour chaque δ et pour tous les problèmes tests représentés dans le Tab.4.1, la phase 1 de $KL_{\alpha}TPA$ localise le minimiseur global. La longueur du sous-intervalle D' obtenu à partir de la phase 1 avec $\delta = 10^{-4}$ est plus petite qu'avec $\delta = 10^{-2}$ et il y a quelques exemples où dans la phase 1 on peut trouver la valeur optimale de $\min_{x \in [a, b]} f(x)$ sans utiliser la phase 2 de la méthode et le nombre de points d'évaluation en phase 1 est plus petit pour $\delta = 10^{-2}$ que pour $\delta = 10^{-4}$ et inversement pour la phase 2 (voir Tab.4.8).

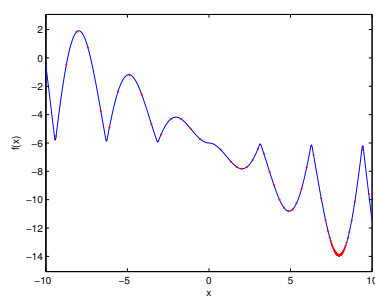
TABLE 4.8 – Le nombre de points d'évaluation des deux phases *Ph 1* et *Ph 2* dans l'algorithme KL_{α_TPA} avec $\alpha = 2, r = 1$ et l'intervalle qui localise le minimiseur global dans la phase 1 avec différentes valeurs de tolérance δ

Numro duproblme	$\delta = 10^{-2}$					$\delta = 10^{-4}$				
	<i>Ph 1</i>		<i>Ph 2</i>		<i>longueur de l'intervalle optimal</i>	<i>Ph 1</i>		<i>Ph 2</i>		<i>longueur de l'intervalle optimal</i>
	<i>N°</i>						<i>optimal D'</i>			
1	47	2047			0.00954	427	[5.1457, 5.1458]	15		7.3875.10 ⁻⁵
2	47	16383			0.02249	447	[2.8679, 2.8681]	127		1.7451.10 ⁻⁴
3	33	2047			0.00200	291	[0.9660, 0.9661]	31		3.0751.10 ⁻⁵
4	197	255			0.00362	1049	[0.6774, 0.6810]	15		2.267.10 ⁻⁴
5	37	10482			4.8	69	[2.7, 7.5]	10482		4.8
6	109	2047			0.02282	969	[17.0281, 17.0509]	63		7.1055.10 ⁻⁴
7	209	511			0.00302	1151	[7.9770, 7.9801]	31		1.8873.10 ⁻⁴
8	153	2047			0.00951	1579	[4.1838, 4.1933]	15		1.0542.10 ⁻⁴
9	35	8191			0.001554	349	[4.7030, 4.7185]	127		2.4081.10 ⁻⁴
10	1	41815			0.4945	1	[0.001, 0.99]	8108		9.890.10 ⁻¹
11	53	1023			0.0037	371	[0.2224, 0.2261]	15		5.694.10 ⁻⁵
12	8039	0			0.530.10 ⁻⁶	20000	[-3.0000, -2.9999]	0		8.14.10 ⁻⁷
13	621	32			0.0372	2285	[1.1761, 1.2133]	0		7.252.10 ⁻⁵
14	59	15647			0.0910	571	[5.8261, 5.9171]	7		4.159.10 ⁻⁵
15	8383	0			9.421.10 ⁻⁷	20000	[9.9999, 10.000]	0		2.3.10 ⁻⁷
16	981	511			8.159.10 ⁻⁴	8649	[-4.4918, -0.4909]	0		1.592.10 ⁻⁶
17	883	15			2.1623.10 ⁻⁴	8151	[5.4827, 5.4829]	0		1.641.10 ⁻⁶
18	844	127			1.562.10 ⁻⁵	8042	[1.5904, 1.5909]	3		1.5624.10 ⁻⁵
19	607	8191			0.0042	5849	[2.4117, 2.4159]	127		6.5707.10 ⁻⁵
20	67	1023			0.0037	533	[5.9797, 5.9834]	63		2.3128.10 ⁻⁴
21	1107	255			0.0035	10275	[-0.6841, -0.6806]	0		1.1027.10 ⁻⁵

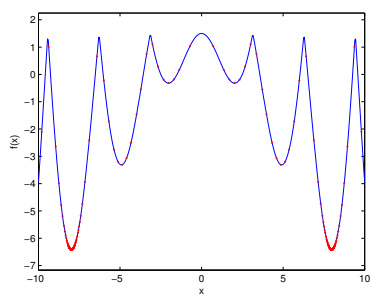
Les graphes ci-dessous montrent, pour quelques exemples, comment la technique KL_{α} en phase 1 obtient un intervalle qui contient la solution optimale (Les essais sont indiqués par une couleur rouge voir Fig.4.1).



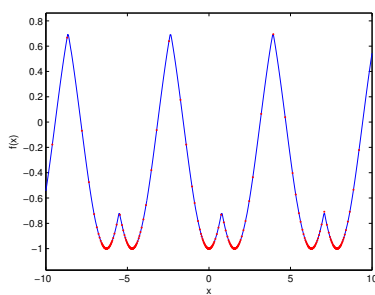
Fonction test 22, phase 1



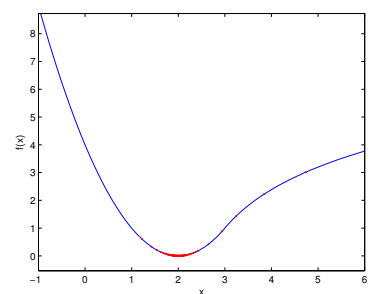
Fonction test 24, phase 1



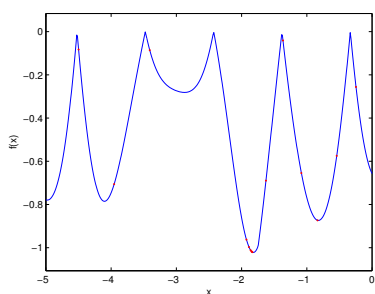
Fonction test 25, phase 1



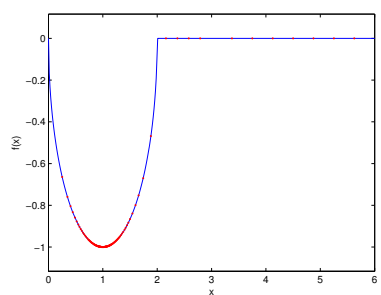
Fonction test 26, phase 1



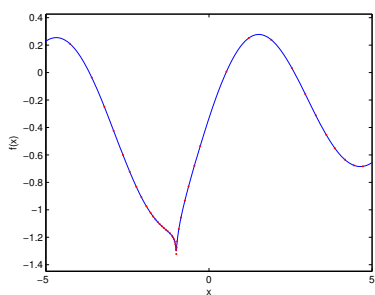
Fonction test 27, phase 1



Fonction test 31, phase 1



Fonction test 33, phase 1



Fonction test 34, phase 1

FIGURE 4.1 – Représentation graphique de quelques fonctions tests et des points de la suite obtenus en phase 1

4.2 Les testes numériques pour les problèmes multidimensionnelles sans contraintes

Dans cette section nous présentons une série de résultats numériques concernant la méthode de Piyavskii couplée avec la méthode d'Alienor appliquée à un ensemble de fonctions tests höldériennes. Les expressions analytiques des fonctions objectifs sont reportées dans le tableau ci-dessous. Les expériences ont été réalisées sur PC avec un processeur Intel (R) Core(TM)i5-7200U 2,50 GHz et 8,00 RAM. Les codes sont implémentés dans MATLAB R2017a, avec le paramètre $\alpha = 0.1$. Nous donnons, dans ce tableau, les résultats numériques obtenus par chaque méthode pour résoudre le problème (3.3) et la comparaison est faite par rapport au nombre de points d'évaluation $E v$ et au temps de calcul.

TABLE 4.9 – Problèmes tests, cas hölder sans contraintes.

$P b$	Fonction	Domaine	Constante de Hölder h	Exposant β	Ref.
1	$\sqrt{ x + y }$	$[-1, 1]^2$	$(\sqrt{2})^{1/2}$	2	[3]
2	$\max\{\sqrt{ x }, \sqrt{ y }\}$	$[-1, 1]^2$	1	2	[3]
3	$\sqrt{ x } + \sqrt{ y }$	$[-1, 1]^2$	2	2	[3]
4	$ x + y - 0.25 ^{2/3} - 3\cos(\frac{x}{2})$	$[-\frac{1}{2}, \frac{1}{2}]^2$	2.42	3	[47]
5	$\sum_{k=1}^3 \frac{1}{2k} \cos((\frac{3}{3k} + 1)x + \frac{1}{2k}) x - y ^3$	$[-5, 5]^2$	14.77	3	[47]
6	$-\cos(x)\cos(y)\exp(1 - \frac{\sqrt{x^2 + y^2}}{\pi})$	$[-6, 6]^2$	45.265	2	[2]
7	$-10\exp(-\sqrt{0.5(x + y)})$	$[-2, 12]^2$	$\frac{10}{\sqrt{2}}$	2	[3]

TABLE 4.10 – Comparaison des résultats entre Alienor- TPA et Alienor- SPA en utilisant les fonctions tests höldériennes à deux variables.

Numéro du problème	Alienor- SPA		Alienor- TPA	
	Ev	$T(s)$	Ev	$T(s)$
1	192	0.0901	196	0.1731
2	207	0.0655	212	0.1506
3	283	0.1738	248	0.0783
4	214	0.1063	206	0.0899
5	4905	1.6163	4865	1.6039
6	65549	308.5771	65546	307.6463
7	4792	13.5927	4862	12.9657

Dans le domaine de l'optimisation globale multidimensionnelle, il existe un ensemble très riche de fonctions tests. Pour nos expériences numériques on a choisi des fonctions höldériennes ni différentiables ni lipschitziennes. On peut dire à travers les résultats obtenus, que la méthode proposée TPA couplée avec la méthode d'Alienor donne des résultats compétitive avec la méthode Alienor- SPA en particulier par rapport le temps de calcul. Cependant, comme la plupart des algorithmes d'optimisation globale, les algorithmes proposées génère un nombre de points d'évaluation de f assez élevé dans le cas où la fonction objectif est assez plate dans une partie importante du domaine faisable, et particulièrement lorsque la région d'attraction du minimiseur global est très étroite. Naturellement, le nombre de points d'évaluation de f augmente aussi avec la dimension de l'espace et la taille du domaine faisable.

4.3 Les testes numériques pour les problèmes multidimensionnelles avec contraintes

Dans cette section, nous rapporterons quelques résultats numériques de 5 problèmes utilisant l'algorithme *TPA* proposé pour le problème unidimensionnel obtenu après avoir utilisé la méthode transformation reductrice Alienor au problème 3.7. Les résultats numériques sont réalisés avec les courbes concrètes α -dense du théorème 3.2. Dans la définition des courbes $\ell(t)$, nous avons substitué α_1 par 1, mais nous pouvons prendre n'importe quelle valeur strictement positive, nous avons aussi choisi ces courbes pour leur simplicité de construction. Tous les résultats numériques sont implémentés sur un PC avec intel (R) Core(TM)i3-3217U CPU@1.80GHz(CPU), ~1.8GHz et 4096MB RAM. Les codes sont implémentés dans MATLAB R2009b. Le tableau 4.11 présente les paramètres des fonctions objectif et de contraintes. Les résultats de calculs des 5 problèmes sont résumés dans le tableau 4.12. Notez que les fonctions f_i , φ_i et ψ_i , $1 \leq i \leq 5$, sont toutes höldériennes mais ni différentiables ni lipschitziennes et les domaines de faisabilité de ces problèmes, sauf les problèmes 3 et 4, sont non convexes.

Problème 1

$$(P_1) \left\{ \begin{array}{l} \min f_1(x_1, x_2) = - \left| \cos(x_1) \cos(x_2) \exp \left| 1 - \frac{(x_1^2 - x_2^2)}{\pi} \right| \right| \\ s.c. \quad \frac{1}{4} |x_2|^{1/3} - 1 \leq x_1 \leq \frac{1}{4} |x_1|^{1/3} + 1 \\ (x_1, x_2) \in [-1, 1] \times [-1, 2] \end{array} \right.$$

Dans ce problème on prend $\varphi_2(x_1) = \frac{1}{4} |x_1|^{1/3} - 1$ et $\psi_2(x_1) = \frac{1}{4} |x_1|^{1/3} + 1$.

Alors, la courbe paramétrée $\ell(t) = (\ell_1(t), \ell_2(t))$ définie par

$$\left\{ \begin{array}{l} \ell_1(t) = -\cos(\alpha_1 t) \\ \ell_2(t) = \frac{1}{4} |\cos(\alpha_1 t)|^{1/3} - \cos(\alpha_2 t) \end{array} \right.$$

avec

$$\left\{ \begin{array}{l} \alpha_2 \geq \frac{2\pi\alpha_1}{\alpha}, \quad \alpha_1 = 1 \\ \lambda_2 = \max\{1, 1/(4\alpha^{2/3})\} \end{array} \right.$$

est $\lambda_2\alpha$ —dense dans le domaine faisable.

Problème 2

$$(P_2) \left\{ \begin{array}{l} \min f_2(x_1, x_2) = \frac{1}{2} \sin \sqrt{|x_1 - x_2|} - \frac{1}{2} \cos \sqrt{|x_1 + x_2|} \\ s.c : \frac{1}{2} x_1^{1/3} \leq x_2 \leq \exp(\sqrt{x_1}) \\ (x_1, x_2) \in [0, 1] \times [0, 2.72]. \end{array} \right.$$

Dans ce problème on prend $\varphi_2(x_1) = \frac{1}{2} x_1^{1/3}$ et $\psi_2(x_1) = \exp(\sqrt{x_1})$.

Alors, la courbe paramétrée $\ell(t) = (\ell_1(t), \ell_2(t))$ d éfinie par

$$\left\{ \begin{array}{l} \ell_1(t) = \frac{1}{2}(1 - \cos(\alpha_1 t)), \text{ pour } t \in [0, \frac{\pi}{\alpha_1}] \\ \ell_2(t) = \frac{1}{2}\varphi_2(\ell_1(t))(1 + \cos(\alpha_2 t)) + \frac{1}{2}\psi_2(\ell_1(t))(1 - \cos(\alpha_2 t)), \end{array} \right.$$

avec

$$\left\{ \begin{array}{l} \alpha_2 \geq \frac{2\pi\alpha_1}{a}, \text{ et } \alpha_1 = 1 \\ \lambda_2 = \max\{1, 1/2\alpha^{2/3}, 2.72/\sqrt{\alpha}\} \end{array} \right.$$

est $\lambda_2\alpha$ —dense dans le domaine faisable.

Problème 3

$$(P_3) \left\{ \begin{array}{l} \min f_3(x_1, x_2) = -20 \exp(-0.2 \sqrt{\frac{|x_1| + |x_2|}{2}}) \\ s.c : x_1^2 + x_2^2 \leq 16 \\ (x_1, x_2) \in [-4, 4] \times [-4, 4]. \end{array} \right.$$

Dans ce problème on prend $\varphi_2(x_1) = -\sqrt{16 - x_1^2}$ et $\psi_2(x_1) = \sqrt{16 - x_1^2}$.

Alors, la courbe paramétrée $\ell(t) = (\ell_1(t), \ell_2(t))$ d éfinie par

$$\left\{ \begin{array}{l} \ell_1(t) = -4 \cos(\alpha_1 t), \text{ pour } t \in [0, \frac{\pi}{\alpha_1}] \\ \ell_2(t) = -4 \sin(\alpha_1 t) \cos(\alpha_2 t), \end{array} \right.$$

avec

$$\begin{cases} \alpha_2 \geq \frac{2\pi\alpha_1}{\alpha}, \text{ et } \alpha_1 = 1 \\ \lambda_2 = \max\{1, 4/\sqrt{\alpha}\} \end{cases}$$

est $\lambda_2\alpha$ -dense dans le domaine faisable.

Problème 4

$$(P_4) \begin{cases} \min f_4(x_1, x_2) = \sum_{k=1}^3 \frac{1}{2k} \left| \cos\left(\left(\frac{3}{2k} + 1\right)x_1 + \frac{1}{2k}\right) \right| |x_1 - x_2|^{1/3} \\ s.c : (2x_1 - 3)^2 + (2x_2 - 3)^2 - 9 \leq 0 \\ (x_1, x_2) \in [0, 3] \times [0, 3]. \end{cases}$$

Dans ce problème on prend $\varphi_2(x_1) = \frac{3}{2} - \sqrt{\frac{9}{4} - (x_1 - \frac{3}{2})^2}$ et $\psi_2(x_1) = \frac{3}{2} + \sqrt{\frac{9}{4} - (x_1 - \frac{3}{2})^2}$.

Alors, la courbe paramétrée $\ell(t) = (\ell_1(t), \ell_2(t))$ d'éfinie par

$$\begin{cases} \ell_1(t) = \frac{3}{2}(1 - \cos(\alpha_1 t)), \text{ pour } t \in [0, \frac{\pi}{\alpha_1}] \\ \ell_2(t) = \frac{1}{2}\varphi_2(\ell_1(t))(1 + \cos(\alpha_2 t)) + \frac{1}{2}\psi_2(\ell_1(t))(1 - \cos(\alpha_2 t)), \end{cases}$$

avec

$$\begin{cases} \alpha_2 \geq \frac{2\pi\alpha_1}{\alpha}, \text{ et } \alpha_1 = 1 \\ \lambda_2 = \max\{1, 3/\sqrt{\alpha}\} \end{cases}$$

est $\lambda_2\alpha$ -dense dans le domaine faisable.

Problème 5

$$(P_5) \begin{cases} \min f_5(x_1, x_2) = \left| \cos(0.5 + 9.5\sqrt{x_1^2 + x_2^2}) \right| \\ s.c : x_1^2 + x_2^2 \leq 1, \\ |x_1| - (x_2 + 1)^2 \leq 0 \\ (x_1, x_2) \in [-1, 1] \times [-1, 1]. \end{cases}$$

Dans ce problème on prend $\varphi_2(x_1) = \sqrt{|x_1|} - 1$ et $\psi_2(x_1) = \sqrt{1 - x_1^2}$.

Alors, la courbe paramétrée $\ell(t) = (\ell_1(t), \ell_2(t))$ définie par

$$\begin{cases} \ell_1(t) = -\cos(\alpha_1 t), & \text{pour } t \in [0, \frac{\pi}{\alpha_1}] \\ \ell_2(t) = (-1 + \sin(\alpha_1 t) + \sqrt{|-\cos(\alpha_1 t)|}) \cos^2(\frac{\alpha_2 t}{2}) + \sin(\alpha_1 t), \end{cases}$$

avec

$$\begin{cases} \alpha_2 \geq \frac{2\pi\alpha_1}{\alpha}, & \text{et } \alpha_1 = 1 \\ \lambda_2 = \max\{1, \sqrt{2/\alpha}\} \end{cases}$$

est $\lambda_2\alpha$ -dense dans le domaine faisable.

TABLE 4.11 – Les paramètres de Hölder des fonctions objectifs et des fonctions contraintes.

Numéro du problème		Constante de Hölder	Exposant de Hölder	Ω_i	Ω
1	f_1	5.0679	0.5		$[-1, 1] \times [-1, 2]$
	$\varphi_2(x_1)$	0.25	1/3	$[-1, 1]$	
	$\psi_2(x_1)$	0.25	1/3	$[-1, 1]$	
2	f_3	1	0.5		$[0, 1] \times [0, 2.72]$
	$\varphi_2(x_1)$	0.5	1/3	$[0, 1]$	
	$\psi_2(x_1)$	2.72	0.5	$[0, 1]$	
3	f_4	154.43	0.5		$[-4, 4]^2$
	$\varphi_2(x_1)$	4	0.5	$[-4, 4]$	
	$\psi_2(x_1)$	4	0.5	$[-4, 4]$	
4	f_5	15.8	1/3		$[0, 3]^2$
	$\varphi_2(x_1)$	$\sqrt{3}$	0.5	$[0, 3]$	
	$\psi_2(x_1)$	$\sqrt{3}$	0.5	$[0, 3]$	
5	f_6	13.43	0.5		$[-1, 1]^2$
	$\varphi_2(x_1)$	1	0.5	$[-1, 1]$	
	$\psi_2(x_1)$	$\sqrt{2}$	0.5	$[-1, 1]$	

TABLE 4.12 – Résultats de calcul des problèmes 1 à 5 avec *Alienor_TPA_CTs*.

Numéro du problème	α	$E\nu$	$T(s)$	f_{opt}
1	0.1	7022	36.8914	-2.71823
2	0.1	4314	14.6659	-0.33528
3	0.1	16383	166.155	-19.2617
4	0.1	15132	141.757	0.014413
5	0.1	14833	138.625	0.87632

Les symboles utilisés dans le tableau 4.12 sont donnés comme suit : α le paramètre strictement positif de la densification de la courbe $\ell(t)$ définie sur $[0, \pi/\alpha_1]$ avec $\alpha_1 = 1$, f_{opt} est la valeur minimale globale de $f(x)$ sur D donnée par la méthode *Alienor_TPA_CTs*.

4.4 Conclusion générale et perspectives

L'importance de l'optimisation globale a considérablement augmenté ces dernières années en raison de son applicabilité dans différents domaines de recherche et de son potentiel à résoudre des problèmes complexes difficiles à résoudre avec les méthodes d'optimisation traditionnelles.

Le défi de trouver l'optimum global d'une fonction tout en minimisant le nombre d'évaluations est un problème en théorie de l'optimisation. Cependant, dans un tel système, la fonction objectif peut ne pas présenter de propriétés qui facilitent l'optimisation, comme la convexité. Par conséquent, plusieurs termes, y compris optimisation sans dérivée, opti-

misation boîte noire et l'optimisation globale, ont été utilisée pour désigner l'optimisation séquentielle des inconnues et fonction potentiellement non convexes. Il existe plusieurs méthodes d'optimisation globale pour la caractérisation d'un minimiseur global, leurs réalisations numériques restent une chose très compliquée dans la majorité des cas particulièrement si la fonction objectif est moins régulière. Dans cette thèse, on s'est intéressé aux méthodes de recouvrement d'optimisation globale déterministes. L'idée principale dans la plupart des approches d'optimisation globales déterministes, lors de la minimisation d'une fonction non convexe f , est de construire une fonction sous estimateur de la fonction objectif f sur le domaine faisable. Cette construction est généralement utilisée pour relaxer le problème initial non convexe de sorte que le problème relaxé soit convexe. Nous avons introduit quelques techniques efficaces pour accélérer et améliorer l'algorithme de Piyavskii-Shubert afin de minimiser des fonctions de Hölder d'une seule variable définies sur un intervalle de $\mathbb{R}$ puis pour le cas à plusieurs variables où f est définie sur des hyper-rectangles de $\mathbb{R}^n$.

Pour le cas unidimensionnel, nous avons proposé deux algorithmes comme version étendue de la méthode de Piyavskii-Shubert. Le premier algorithme *TPA*, est basée sur l'utilisation des tangentes au milieu de chaque sous-intervalle de $[a, b]$ des fonctions sous-estimateur de la fonction f sur $[a, b]$. Le second est une combinaison de deux phases successives. Avec la phase 1, on obtient en quelques itérations, un sous-intervalle qui contient souvent le minimiseur global de f en utilisant une nouvelle fonction minorante K_α -lipschitzienne non tangente de f avec une tolérance δ bien supérieure à la précision donnée ε . Dans la phase 2, nous appliquons l'algorithme *TPA* sur l'intervalle obtenu en phase 1. Un résultat de convergence est prouvé et les résultats numériques sont comparés avec celles des méthodes existantes. Pour le cas d'une fonction höldérienne à plusieurs variables définies sur un hyper-rectangle de $\mathbb{R}^n$, nous avons proposé un algorithme *Alienor_TPA* qui est composé de deux étapes, la première, étape est de transformer la fonction f en fonction d'une seule variable $\mathbf{f}$ en utilisant des courbes α -dense et la deuxième étape est l'application de l'algorithme *TPA* à $\mathbf{f}$. Les propriétés théoriques de la convergence du nouvel algorithme sont données. Pour montrer l'efficacité des algorithmes *TPA* et KL_α -*TPA* et les méthodes *Alienor_TPA*, *Alienor_TPA_CTs* on a utilise des séries de fonctions tests tirées de la littérature. Ces fonctions tests ont des propriétés différentes comme non convexe, non différentiable, de Lipschitz

et de Hölder, beaucoup d'entre elles ont au moins deux minima locaux. L'étude comparative a indiqué que les résultats montrent que les techniques utilisées sont très recommandées et fiables.

Comme perspectives, il y a au moins deux types de préoccupations actuelles à travers les méthodes utilisées récemment :

i) Il est préférable de traiter des problèmes d'optimisation globale des fonctions höldériennes où la constante de Hölder h est inconnue à priori et voir l'influence des estimations de h sur les résultats numériques en termes de points d'évaluations, notamment en termes de temps de calcul.

ii) Faire une étude détaillée des algorithmes vus dans le cas unidimensionnel pour le cas multidimensionnel.

Bibliographie

- [1] H. Ammar, and Y. Cherruault, Approximation of a Several Variables function by a one Variable function and Applications to Global Optimization, Math. Comput. Modelling, Vol. 18, No. 2, pp. 17-21. (1993)
- [2] N.K. Arutynova, A.M. Dulliev and V.I. Zabotin, Algorithms for projecting a point into a level surface of a continuous function on a compact set. Computational Mathematics and Mathematical Physics 54(9), 1395-1401. (2014)
- [3] N.K. Arutynova, A.M. Dulliev and V.I. Zabotin, Global optimization of multi-variable functions satisfying the vanderbei condition, Journal of Applied Mathematics and Computing, 68, 1135-1161. (2022)
- [4] W.P. Baritomba, Accelerations for a variety of global optimization methods, J. of Global optimization, 4(1), pp. 37-45. (1993)
- [5] P. Basso, Iterative Method for the Localization of the Global Maximum, SIAM Journal on Numerical Analysis 19, 781-792, (1982)
- [6] P. Basso, Optimal Search for the Global Maximum of Function with Bounded Semi-norm, SIAM Journal on Numerical Analysis 22, 888-903, (1985).
- [7] S. BOUKAROURA, Contribution à l'étude de certaines méthodes d'optimisation globale, Thèse de Doctorat en sciences, Université Ferhat Abbas, Sétif. (2007)
- [8] A. R. Butz, Space filling curves and mathematical programming. Information and control, 12(4), 313-330. (1968)
- [9] R. P. Brent, Algorithm for Minimization without Derivatives, Prentice-Hall, Englewood Cliffs, New Jersey.195p, (1973)

- [10] C. Chenouf, M. Rahal, On Hölder global optimization method using piecewise affine bounding functions, Numerical Algorithms, [//doi.org/10.1007/s11075-023-01524-x](https://doi.org/10.1007/s11075-023-01524-x). (2023)
- [11] Y. Cherruault and A. Guillez, Une méthode pour la recherche du minimum global d'une fonctionnelle, C.R.A.S, Paris, Série I, Vol. 296, pp. 175-178. (1983)
- [12] Y. Cherruault, Optimisation, Méthodes Locales Et Globales, Presses Universitaires de France, (1999).
- [13] Y. Cherruault, A new reducing transformation for global optimization (with Alienor method), Kybernetes 34, pp. 1084-1089. (2005)
- [14] Y. M. Danilin and S.A. Piyavskii, An Algorithm for Finding an Absolute Minimum, Theory of Optimal Solutions. IK Akad.Nank, USSR, Kiev, 25- 37, 1967. (Russian)
- [15] P. M. Delgado, and E. A. Galperin, Global optimization over general compact sets by the beta algorithm : A MAPLE Code. Computers and Mathematics with Applications, 52(1-2), 33-54. (2006)
- [16] Y. G. Evtushenko, Numerical Methods for Finding the Global Extremum of a Function. USSR Computational Mathematical Physics 11, 38-54. (1971)
- [17] Y. G. Evtushenko, Numerical optimization techniques. Springer-Verlag New York, Inc. (1985)
- [18] R. Fiorenza, Hölder and locally Hölder Continuous Functions, and Open Sets of Class $C^k, C^{k,\lambda}$. Springer International Publishing ag (2016)
- [19] E. A. Galperin. The Cubic Algorithm, Journal of Mathematical Analysis and Applications 112(2), 635-640, 1985.
- [20] V. P. Gergel, A Global Optimization Algorithm for Multivariate functions with Lipschitzian First Derivatives, J. of Global Optimization, 10, No. 3, pp. 256-281. (1997)
- [21] E. Gourdin, B. Jaumard, and R. Ellaia, Global Optimization of Hölder function. Journal of Global Optimization, 8(4), pp 323-348. (1996)
- [22] D. Guettal, Efficacité et fiabilité des méthodes utilisant l'approche Branch-and-Bound pour l'optimisation globale non convexe, Thèse de doctorat en sciences, Université Ferhat Abbas, Sétif. (2014)

- [23] D. Guettal, C. Chenouf and M. Rahal, Global optimization method of multivariate non-Lipschitz functions using tangent minorants, *Nonlinear Dynamics and Systems Theory*, 23 (2) 183–194. (2023)
- [24] P. Hanjoul, P. Hansen, D. Peeters, J.F. Thisse, Uncapacitated Plant Location under Alternative Space Price Policies, *Management Science*. Vol. 36, pp 41-47. (1990)
- [25] P. Hansen, B. Jaumard and S. H. Lu. Global Optimization of Univariate Lipschitz Functions : I.Survey and Properties, *Mathematical programming* 55, 251-272, 1992
- [26] P. Hansen, B. Jaumard, Sh-H. Lu, Global optimization of univariate Lipschitz functions : II. New algorithms and computational comparison, *Mathematical Programming*. 55, pp. 273-292. (1992)
- [27] P. Hansen and B. Jaumard. Lipschitz Optimization, *Les Cahiers du GERARD*, May 1994.
- [28] R. Horst and H. Tuy. On the Convergence of Global Methods in Multi extremal Optimization, *Journal of Optimization Theory and Applications* 54, 253-271, (1987).
- [29] R. Horst, H. Tuy, *Global Optimization : Deterministic Approach*, Springer-Verlag, Berlin, (1993)
- [30] R. Horst , P.M. Pardalos, *Handbook of Global Optimization*, Kluwer Academic Publishers, Dordrecht, (1995).
- [31] Z. E. Hou, F. J. Duan, The approximation algorithm for solving a sort of non-smooth programming. *Applied Mathematics and Computation*, 186(2), pp. 1511-1519. (2007)
- [32] B. Jaumard, T. Hermann and H. Ribault, An On-line Intersection Algorithm for Global Optimization of Multivariate Lipschitz Functions. (1996)
- [33] D. R. Jones, C. D. Perttunen, and B. E. Stuckmanm. Lipschitzian optimization without the lipschitz constant, *Journal of Optimization Theory and Applications*, 79 :157, October. (1993)
- [34] A. Ketfi-cherif, Sur certaines méthodes d’optimisation globale basées sur l’introduction de fonctions auxiliaires. Thèse de Doctorat en sciences, Université Ferhat Abbas, Sétif. (2016)

- [35] S. Kiatsupaibul, R. L. Smith, Z. B. Zabinsky, Solving infinite horizon optimization problems through analysis of a one-dimensional global optimization problem, *J. of Global Optimization*. 66, pp. 711-727. (2016)
- [36] D. Lera and Y. D. Sergeyev, Global Minimization Algorithms for Hölder Functions *BIT Numerical Mathematics*, 42(1), pp. 119-133. (2002)
- [37] D. Lera, Y.D. Sergeyev, Deterministic global optimization using space-filling curves and multiple estimates of Lipschitz and Hölder constants, *Commun. Non. Sci. Num. Sim.* 23 (1-3), pp. 328-342. (2014)
- [38] D. Lera, Y.D. Sergeyev, GOSH : derivative-free global optimization using multi-dimensional space-filling curves, *J. Glob. Opt.* 71 (1), pp. 193-211. (2018)
- [39] D.G. Luenberger, *Linear and nonlinear programming* 2nd Edition, addition Wesley, Reading massachusetts, (1984)
- [40] D. Q. Mayne and E. Polak, Outer Approximation Algorithm for Non différentiable Optimi- zation Problem, *J. of Optimiz. Theory and applic.* 42(1), pp. 19-30. (1985)
- [41] C. C. Meewella and D. Q. Mayne. An Algorithm for Global Optimization of Lipschitz Functions, *Journal of Optimization Theory and Applications* 57, 307-323. (1988).
- [42] S. H. Mishra, Some new test functions for global optimization and performance of repulsive particle swarm method. Available at SSRN, 1-12. (2006)
- [43] R. H. Mladineo, An algorithm for finding the Global Maximum of a Multimodal, Multivariate function, *Mathematical Programming* 34, pp. 188-200. (1986)
- [44] J. D. Pinter, *Global Optimization in Action, Continuous and Lipschitz Optimization : Algorithm, Implementations and Applications*. Book Series of the Nonconvex Optimization and Its Application. Kluwer Academic Publisher, Dordrecht. Springer : New York, (1996)
- [45] S. A. Piyavskii, An algorithm for finding the absolute minimum for a function. *Theory of Optimal Solutions*, No.2, Kiev, IK AN USSR, pp. 13-24. (1967)
- [46] S. A. Piyavskii, An algorithm for finding the absolute extremum of a function, *USSR Comp. Mathem. and Mathem. phys.*, 12, No. 4, pp. 888-896, (1972).

- [47] M. Rahal, A. Ziadi, A new extension of Piyavskii's method to Hölder functions of several variables, *Applied Mathematics and Computation*. 197, pp. 478-488. (2008)
- [48] M. Rahal, Extention de certaines méthodes de recouvrement en optimisation globale. Thèse de Doctorat en sciences, Université Ferhat Abbas, Sétif. (2009)
- [49] M. Rahal, D. Guettal, Modified Sequential Covering Algorithm for Finding a Global Minimizer of Non différentiable Functions and Applications, *Gen. Math. Notes*, Vol. 22, No. 1, pp.100-115. (2014)
- [50] M. Rahal, A. Ziadi, R. Ellaia, Generating α -dense curves in non-convex sets to solve a class of non-smooth constrained global optimization, *Croatian Operational Research Review* 289 CRORR. 10, pp. 289-314.(2019)
- [51] F. Schoen, On a Sequential Search Strategy in Global Optimization Problems, *Calcolo* 19, 321-334. (1982)
- [52] Ya.D. Sergeyev, A. Candelieri, D E. Kvasov, R. Perego, Safe global optimization of expensive noisy black-box functions in the δ –Lipschitz framework, Springer-Verlag GmbH Germany, part of Springer Nature,2020. doi.org/10.1007/s00500-020-05030-3
- [53] Y.D. Sergeyev, R.G. Strongin, D. Lera, Introduction To Global Optimization Exploiting Space-Filling Curves, Springer Science & Business Media, (2013)
- [54] Z. Shen and Y. Zhu. An Interval Version of Shubert's Iterative Method for the Localization of the Global Maximum, *Computing* 38, 275-280. (1986).
- [55] B. O. Shubert, A sequential method seeking the global maximum of a function, *SIAM, Journal of Numerical Analysis*, Vol. 9, No. 3, pp. 379-388. (1972)
- [56] R.G. Strongin, Numerical methods in multiextremal problems, Nauka Moskow (1978)
- [57] R. G. Strongin, Algorithms for multi-extremal programming problems employing the set of joint space-filling curves. *Journal of Global Optimization* (2) 357–378. (1992)
- [58] R.G. Strongin, Ya.D.Sergeyev, Global optimization with non convex constraints : Sequential and parallel algorithms, Kluwer Academic Publishers Dordrecht, (2000)
- [59] L.N. Timonov, An algorithm for search of a global extremum, *Engineering Cybernetics*, 15, pp. 38-44. (1977)

- [60] A. Törn, A. Zilinska, Global optimization. Lecture Notes in Computer Sciences, 350. Springer-Verlag. (1989) <https://doi.org/10.1007/3-540-50871-6>
- [61] R. J. Vanderbei, Extension of Piyavskii's Algorithm to Continuous Global Optimization, J. of Global Optimization. Vol. 14 pp 205-216. (1999)
- [62] X. Wang, T-SH. CHANG, An improved Univariate Global optimization Algorithm with Improved Linear Lower Bounding Functions, J. of global optimization. 8, pp. 393-411. (1996)
- [63] A. Yahyaoui, H. Ammar, Global optimization of multivariate Hölderian functions using overestimators, Open Access Library Journal. 4, No.5. (2017)
- [64] V. I. Zabotin, P. A. Chernyshevsky, Extension of Strongin's global optimization algorithm to a function continuous on a compact interval, Computer Research and Modeling. 11(6), pp. 1111-1119. (2019)
- [65] A. Ziadi, Y. Cherruault, Generation of α -dense curves and application to global optimization, Kybernetes 29(1), pp. 71-82. (2000)
- [66] A. Ziadi, Y. Cherruault, G. Mora, Global optimization : A new variant of the Alienor method, Comput. Math. Appl. 41, pp. 63-71. (2001)
- [67] A. Ziadi, D. Guettal and Y. Cherruault, Global Optimization : Alienor mixed method with Piyavskii-Shubert technique, Kybernetes. Vol. 34 No.7/8, pp. 1049-1058. (2005)