Projet d'Optimisation Continue

DAVY Valentin

MARCHET Maxime

FI2

I. Introduction du problème

Dans se projet, nous allons étudier la fonction de Rosenbrock en utilisant différentes méthodes et algorithmes afin de déterminer le minimum global de cette fonction.

Après avoir implémenter et tester ces différents algorithmes sur cette fonction, nous exposerons un problème qui concerne notre vie de tous les jours et nous essaierons de le résoudre à l'aide des fonctions mises en oeuvres.

II. Fonction de Rosenbrock

A. Un peu d'histoire

Howard Harry Rosenbrock était un ingénieur anglais, décédé en 2010 à l'âge de 90 ans, est une figure emblématique de la théorie du contrôle (étude comportementale des systèmes dynamiques). Il participa activement à la résolution des problèmes d'optimisation.
Dans ce domaine, la fonction de Rosenbrock est souvent utilisée comme test pour optimiser un problème. Dans les années 70, Rosenbrock commence à se pencher de plus en plus sur les aspects philosophiques de l'impact des nouvelles technologies sur la société, sans pour autant délaisser la recherche scientifique. Plus qu'un simple chercheur, Rosenbrock a pu appliquer ses idées à divers niveaux, que ce soit en tant qu'ingénieur ou chercheur, en mécanique et en électronique.

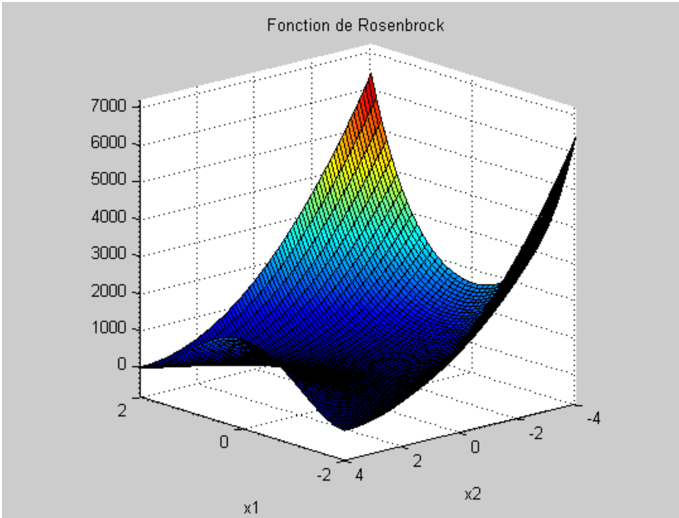
B. Définition de la fonction et caractéristiques

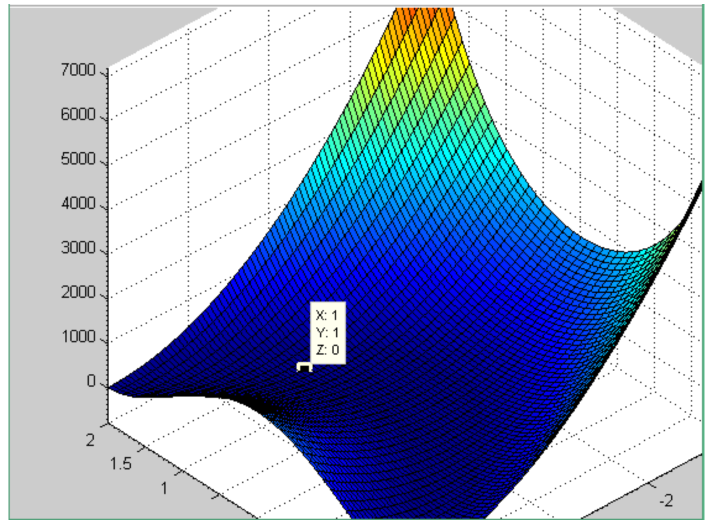
Voici la définition de cette fonction:

$$f(x_1, x_2) = 100\left(x_2 - x_1^2\right)^2 + (1 - x_1)^2$$

Le minimum global de cette fonction se situe en $x = (1\ 1)^T$. C'est ce point que l'on doit trouver via nos différentes méthodes.

télécom saint-étienne
école d'ingénieurs nouvelles technologies





C. Méthodes utilisées et paramètres employés

Nous allons utilisés 3 méthodes différentes:

- la méthode de la plus forte pente
- la méthode de Newton avec recherche linéaire
- la méthode quasi-Newton BFGS

Les différents paramètres que nous avons utilisé pour chaque algorithme sont regroupés dans ce tableau.

Grandeur	Point de départ	β_1	β_2		ε
Valeur	$(-1,5\ \ 1,5)^T$	0.001	0.99	20	0.001

Les paramètres β_1 et β_2 sont des paramètres qui entrent en jeu dans l'algorithme de Flétcher et Lemaréchal pour optimiser le pas utilisé dans chaque méthode tout comme le paramètre . (sera toujours égal à 20 dans ce projet ; $0 < \beta_1 < 1$ et $0 < \beta_2 < 1$)
Le paramètre ε est la valeur test pour laquelle on décide de sortir de la boucle qui permet de trouver le minimum de la fonction de Rosenbrock.

Dans chaque méthode, nous déterminerons:

- le temps total mis pour trouvé un résultat
- le nombre d'itérations

télécom saint-étienne
école d'ingénieurs nouvelles technologies

En plus du vecteur résultat x_f attendu, nous calculerons:

- la **fonction coût** en ce point $\left(f\left(x_f\right)\right)$
- la qualité de la solution à travers la **mesure d'optimalité du premier ordre** $\left(\left\|f\left(x_f\right)\right\|_\infty\right)$

Nous prendrons le soin d'étudier l'influence des paramètres β_1 et β_2 ainsi que du point de départ.

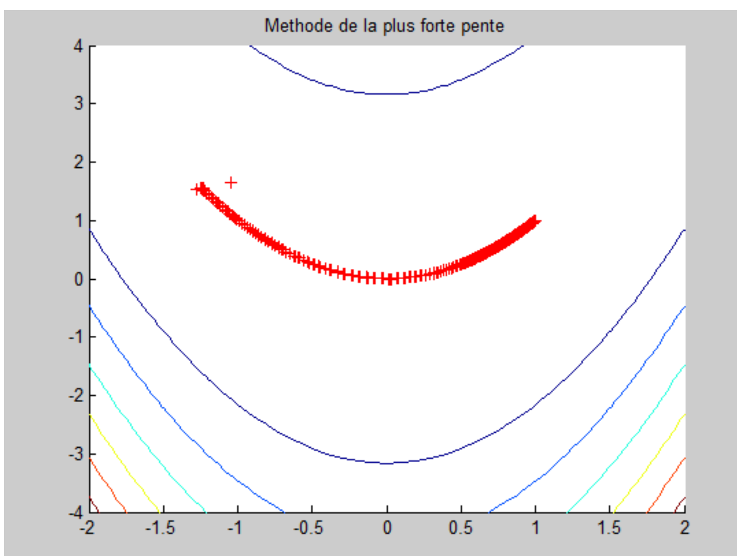
III. Méthode des plus fortes pentes avec préconditionnement

A. Description de la méthode

Cette méthode utilise l'algorithme de Fletcher et Lemaréchal afin d'optimiser le pas à chaque itération: en effet, il doit vérifier les conditions de Wolfe.
De plus, nous avons choisis comme famille de préconditionneurs celle correspondant à la matrice identité.

B. Résultats obtenus

1. Tracé du point courant



2. Résultats et performances de la méthode

Le vecteur résultat est: $x_f = (0.9989 \;\; 0.9978)^T$

Voici les différentes caractéristiques de notre méthode qui a amené à ce résultat.

Grandeur	Nombre d'itérations	Temps d'exécution	$f(x_f)$	$\|f(x_f)\|_\infty$
Valeur	1589	$3.6\ s$	1.2421×10^{-6}	-4.4535×10^{-4}

Cette méthode met un certain temps d'exécution même si nous n'avons pas encore de temps de calcul de comparaison. Par contre, on remarque que le nombre d'itérations est assez important alors que le résultat trouvé n'est pas le minimum de la fonction.

C. Influence des paramètres β_1 et β_2

Grandeur	(β_1, β_2)	Nombre d'itérations	Temps d'exécution	$f(x_f)$	$\|f(x_f)\|_\infty$
Valeur	$(0.1\ ,\ 0.9)$	1589	$2.6\ s$	1.2424×10^{-6}	-4.567×10^{-4}
	$(0.25\ ,\ 0.75)$	3416	$4.6\ s$	1.2489×10^{-6}	9.1007×10^{-4}
	$(0.4\ ,\ 0.6)$	3126	$3.9\ s$	1.2309×10^{-6}	8.8347×10^{-4}

À noter que pour les deux dernières combinaisons des paramètres beta, le point trouvé est $(1.0011\ ,\ 1.0022)^T$.

Pour cette première méthode, la combinaison idéale est $(\beta_1, \beta_2) = (0.1\ ,\ 0.9)$.

En effet, le temps d'exécution est diminué pour le même résultat.

télécom saint-étienne
école d'ingénieurs nouvelles technologies

D. Influence du point de départ

Grandeur	x_0	Nombre d'itérations	Temps d'exécution	$f(x_f)$	$\|f(x_f)\|_\infty$
Valeur	$(-5\ ,\ 5)^T$	-	-	-	-
	$(-2.5\ ,\ 2.5)^T$	4999	$7.8\ s$	1.2498×10^{-6}	$8.9369\ \times 10^{-4}$
	$(-0.5\ ,\ 0.5)^T$	1492	$2.6\ s$	1.2342×10^{-6}	-4.4388×10^{-4}

Ainsi, si l'on s'éloigne trop du minimum à trouver, la méthode ne permet pas de converger. Et plus on se rapproche du point à trouver, plus la méthode est rapide et précise.

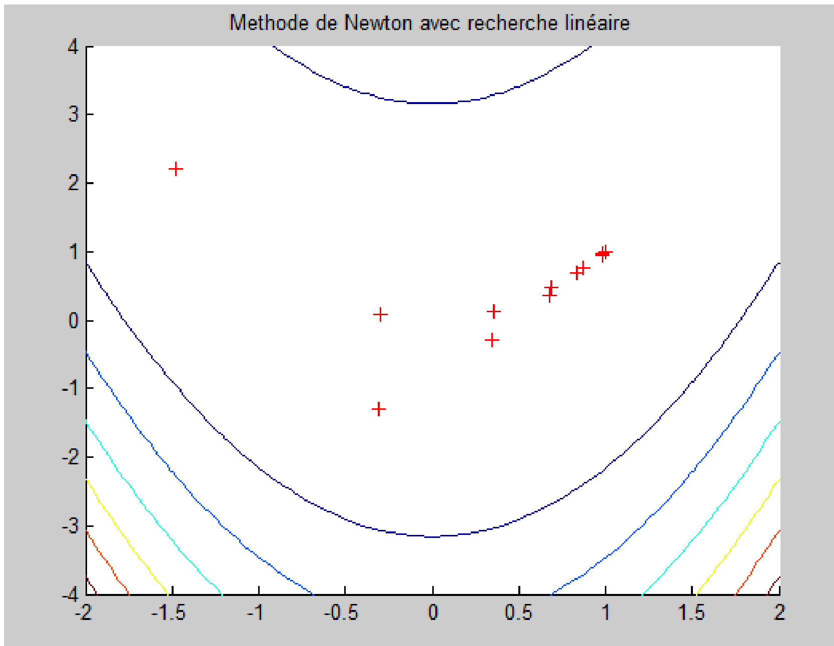
IV. Méthode de Newton avec recherche linéaire

A. Description de la méthode

Dans cette méthode, on va déterminer la direction d_k vers laquelle on s'oriente tout au long des itérations via la factorisation de Cholesky de la matrice hesienne de la fonction de Rosenbrock.
Puis, on utilise à nouveau l'algorithme de Flétcher et Lemaréchal pour déterminer le pas d'itération.
Ainsi, la différence avec la méthode précédente est la manière de déterminer la direction.

B. Résultats obtenus

1. Tracé du point courant



2. Résultats et performances de la méthode

Le vecteur résultat est: $x_f = (1 \quad 1)^T$

Voici les différentes caractéristiques de notre méthode qui a amené à ce résultat.

Grandeur	Nombre d'itérations	Temps d'exécution	$f(x_f)$	$\|f(x_f)\|_\infty$
Valeur	13	$65\ ms$	1.3191×10^{-11}	-1.5657×10^{-6}

Ainsi, on remarque que le résultat trouvé est celui attendu: le point $(1 \quad 1)^T$.

Si l'on regarde la fonction coût en ce point, elle est très proche de 0 (10^{-11}).

On peut apprécier de plus, le nombre très faible d'itérations par rapport à la méthode des plus fortes pentes. En effet, il y a environ 100 fois moins d'itérations avec cette méthode. Enfin, le temps d'exécution est aussi grandement amélioré.

télécom saint-étienne
école d'ingénieurs nouvelles technologies

C. Influence des paramètres β_1 et β_2

Grandeur	(β_1, β_2)	Nombre d'itérations	Temps d'exécution	$f(x_f)$	$\|f(x_f)\|_\infty$
Valeur	$(0.1\ ,\ 0.9)$	19	$79\ ms$	6.0767×10^{-11}	5.5286×10^{-5}
	$(0.25\ ,\ 0.75)$	20	$58\ ms$	4.1456×10^{-10}	6.8364×10^{-4}
	$(0.4\ ,\ 0.6)$	17	$49\ ms$	9.6605×10^{-13}	1.344×10^{-5}

Cette méthode varie peu en fonction des paramètres beta. Cela montre la grande flexibilité de cette méthode et prouve à nouveau son efficacité.

D. Influence du point de départ

Grandeur	x_0	Nombre d'itérations	Temps d'exécution	$f(x_f)$	$\|f(x_f)\|_\infty$
Valeur	$(-5\ ,\ 5)^T$	17	$41\ ms$	3.6453×10^{-15}	7.1690×10^{-7}
	$(-2.5\ ,\ 2.5)^T$	15	$38\ ms$	6.1733×10^{-13}	5.326×10^{-6}
	$(-0.5\ ,\ 0.5)^T$	21	$29\ ms$	7.8322×10^{-13}	6.8375×10^{-6}

Cette méthode varie peu en fonction du point de départ. Elle est toujours aussi rapide d'exécution et permet de trouver le résultat précisément.

V. Méthode quasi-Newton BFGS

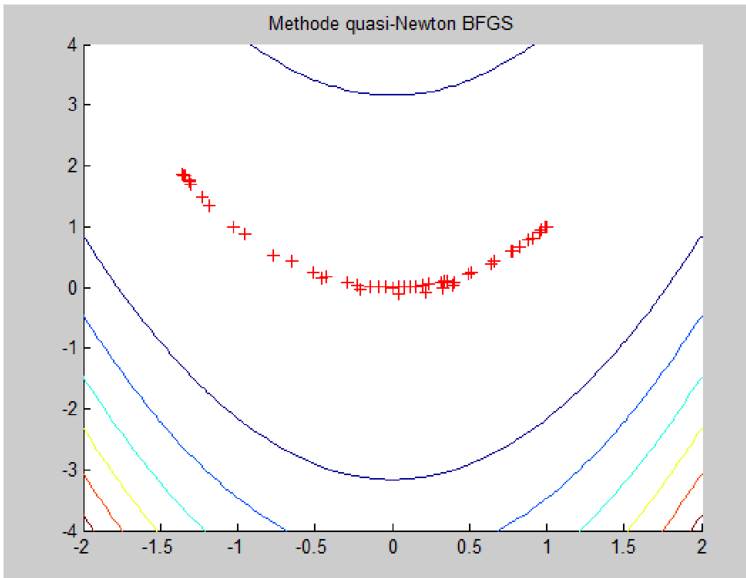
A. Description de la méthode

Dans cette méthode, on définit la direction vers laquelle on se dirige en effectuant une factorisation de Cholesky de la matrice hessienne de notre fonction. Puis, on utilise les écarts entre les gradients et les écarts entre les vecteurs trouvés pour mettre à jour l'inverse de la matrice Hessienne.

télécom saint-étienne
école d'ingénieurs nouvelles technologies

B. Résultats obtenus

1. Tracé du point courant



2. Résultats et performances de la méthode

Le vecteur résultat est: $x_f = (1 \quad 1)^T$

Voici les différentes caractéristiques de notre méthode qui a amené à ce résultat.

Grandeur	Nombre d'itérations	Temps d'exécution	$f(x_f)$	$\left\| f(x_f) \right\|_\infty$
Valeur	66	$0.13\ s$	3.5065×10^{-11}	3.3338×10^{-5}

C. Influence des paramètres β_1 et β_2

Grandeur	(β_1, β_2)	Nombre d'itérations	Temps d'exécution	$f(x_f)$	$\lVert f(x_f) \rVert_\infty$
Valeur	$(0.1\ ,\ 0.9)$	31	$0.094\ s$	$1.896\ \times 10^{-9}$	$3.7036\ \times 10^{-4}$
	$(0.25\ ,\ 0.75)$	649	$1.59\ s$	$3.2090\ \times 10^{-10}$	$5.1909\ \times 10^{-4}$
	$(0.4\ ,\ 0.6)$	-	-	-	-

On remarque que plus les paramètres béta augmentent, plus il est difficile de converger. D'ailleurs, il faut plus de 2000 itérations et donc, notre algorithme n'a pas convergé.

D. Influence du point de départ

Grandeur	x_0	Nombre d'itérations	Temps d'exécution	$f(x_f)$	$\lVert f(x_f) \rVert_\infty$
Valeur	$(-5\ ,\ 5)^T$	11	$0.20\ s$	$2.6262\ \times 10^{-11}$	$6.4005\ \times 10^{-5}$
	$(-2.5\ ,\ 2.5)^T$	87	$0.16\ s$	$1.6306\ \times 10^{-12}$	$-1.5822\ \times 10^{-7}$
	$(-0.5\ ,\ 0.5)^T$	31	$0.04\ s$	1.8960×10^{-9}	$3.7036\ \times 10^{-4}$

Cette méthode est très performante lorsque le point de départ est proche de celui qu'il faut trouver.

VI. Conclusions

De toute les méthodes utilisées, la méthode qui offre les meilleures performance est la méthode de Newton linéaire qui est très rapide en terme d'exécution. Cependant, la méthode quasi-Newton BFGS permet d'arriver au résultat avec un nombre d"opérations acceptable. Ainsi, pour le problème de Rosenbrock, la méthode qui est la plus efficace est la méthode de Newton linéaire.

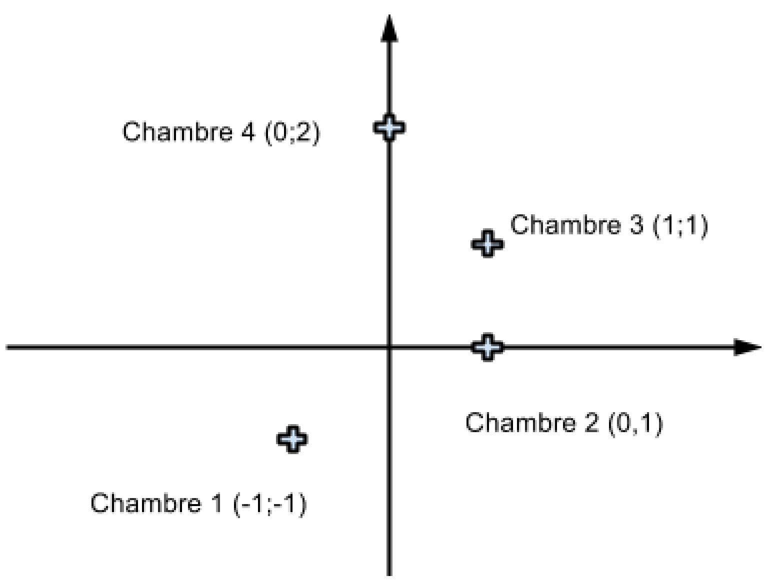
VII. Problème d'optimisation

Utilisons les algorithmes que nous avons définis précédemment afin de résoudre un problème de la vie quotidienne de manière optimale.

A. Énoncé du problème

Un groupe de 4 étudiants font une collocation ensemble dans une gigantesque maison de 2 étages au dessus du rez-de-chaussée et d'un sous-sol.
Chacun a sa propre chambre.
Voici comment l'on peut modéliser la maison ainsi que la disposition des chambres.

	Chambre 4 (0;2)	
		Chambre 3 (1;1)
	(0;0)	Chambre 2 (0;1)
Chambre 1 (-1; -1)		



Le propriétaire de la maison leur offre gracieusement l'installation d'internet. Malheureusement, la box choisie par ce dernier est un très vieux modèle et le wifi délivré par celui-ci est très faible. De plus, il ne peut y avoir qu'1 seule prise téléphonique installée dans la maison.

télécom saint-étienne
école d'ingénieurs nouvelles technologies

Les 4 étudiants doivent donc décider d'une position où placer la box sachant que certains étudiants ont plus besoin d'internet que d'autres.
Ils se réunissent donc et chacun estime sur une échelle de 1 à 10 la nécessité d'avoir le wifi dans sa chambre.

Étudiant	1	2	3	4
Nécessité	6	4	3	5

Où faut-il installer la box?

B. Traduction mathématique

1. Variables de décision

Ici, les variables de décisions sont:
- x_1 l'abscisse de la position où il faut placer la box
- x_2 l'ordonnée de la position où il faut placer la box

2. Fonction objectif

La fonction objectif ***g*** mesure la pertinence de la position de la box en prenant en compte les nécessités de chacun et en privilégiant une position proche d'un étudiant.

Ainsi, cette fonction se définit comme:

$$g(x_1, x_2) = 6\sqrt{(x_1)^2 + (x_2)^2} + 4\sqrt{(x_1-2)^2 + (x_2-1)^2} + 3\sqrt{(x_1-2)^2 + (x_2-2)^2} + 5\sqrt{(x_1-1)^2 + (x_2-3)^2}$$

3. Contraintes

Comme seule contrainte, il y a que la box doit être située dans la maison:

- $-1 \leq x_1 \leq 2$
- $-1 \leq x_2 \leq 2$

télécom saint-étienne
école d'ingénieurs nouvelles technologies

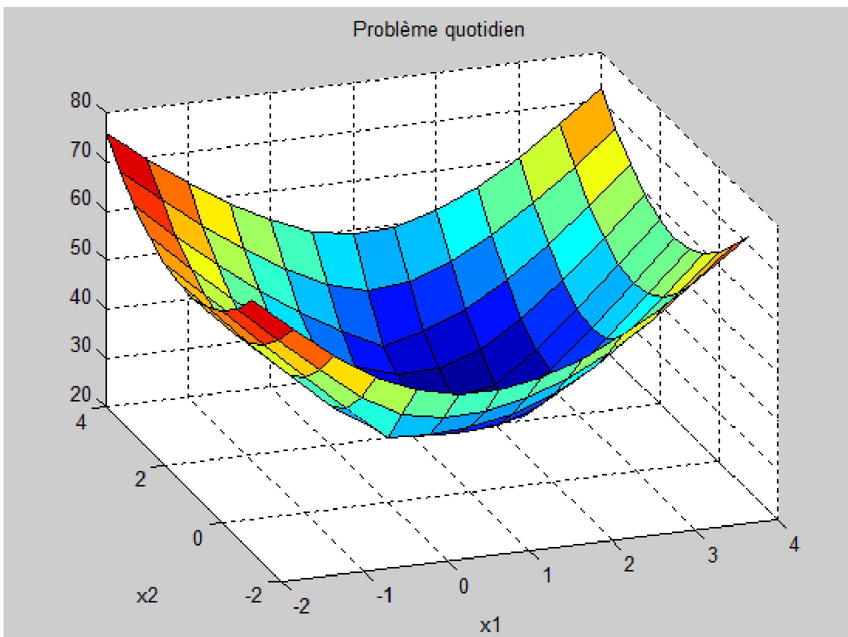
4. Reformulation du problème

Ainsi, notre problème d'optimisation est:

$$min\ (\ g(x_1, x_2)\ )_{[x_1, x_2]} \text{ sous contraintes:}$$

- $-1 \leq x_1 \leq 2$
- $-1 \leq x_2 \leq 2$

C. Représentation graphique

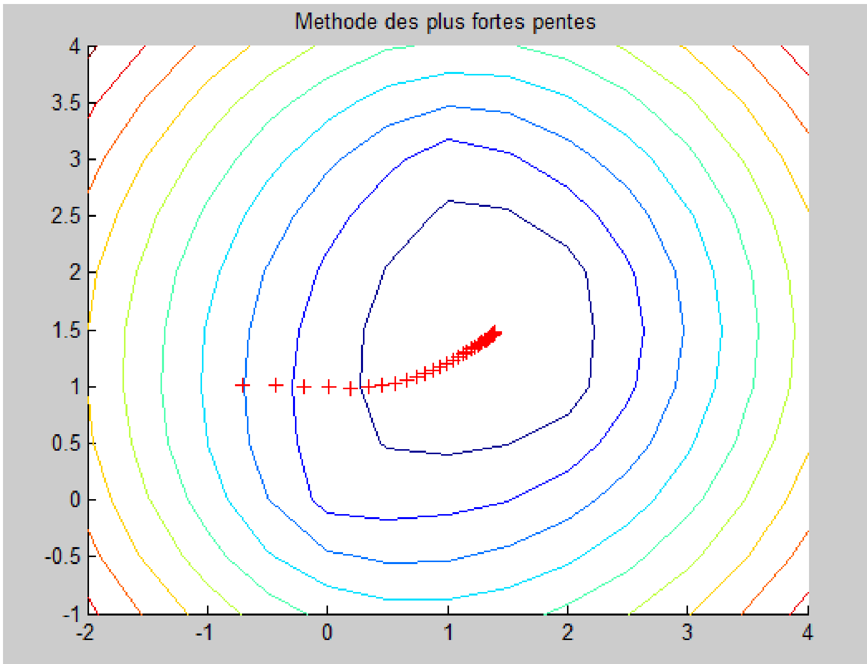


Ainsi, notre fonction présente une courbe comme pour la fonction de Rosenbrock.
Le minimum de cette fonction se trouve entre (1 , 1) et (2 , 2).
Nous allons ainsi utiliser les méthodes d'optimisation pour déterminer exactement ce minimum.

D. Utilisation des méthodes d'optimisation

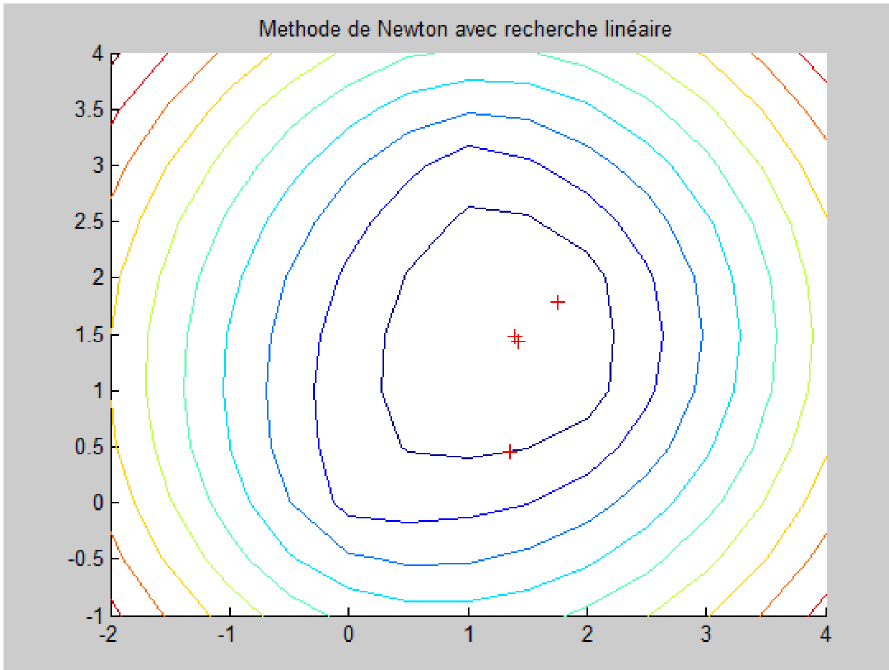
Nous avons utilisé la fonction contour de Matlab afin d'afficher le point courant au cours des itérations.

1. Méthode des fortes pentes



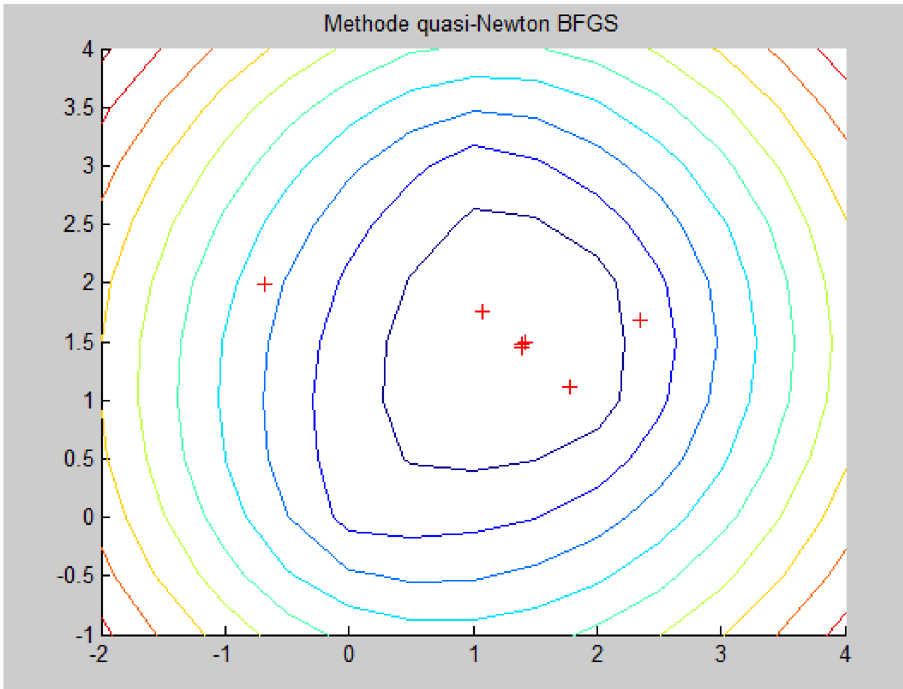
Grandeur	x_f	Nombre d'itérations	Temps d'exécution	$f\left(x_f\right)$	$\left\|f\left(x_f\right)\right\|_\infty$
Valeur	$(1.393\ ,\ 1.4716)^T$	67	$0.17\ s$	25.5375	$-4.0593\ \times 10^{-4}$

2. Méthode de Newton Linéaire



Grandeur	x_f	Nombre d'itérations	Temps d'exécution	$f(x_f)$	$\|f(x_f)\|_\infty$
Valeur	$(1.3994\ ,\ 1.4717)^T$	5	$0.016\ s$	25.5375	$-4.3121\ \times 10^{-9}$

3. Méthode Quasi-Newton BFGS

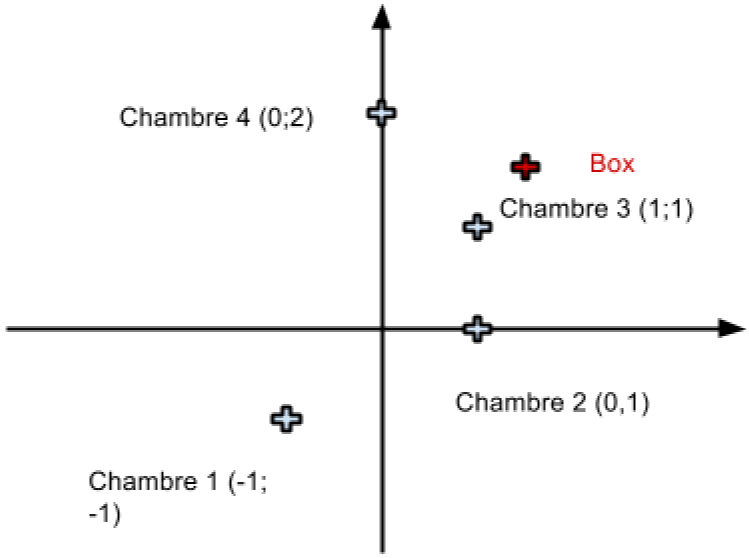


Grandeur	x_f	Nombre d'itérations	Temps d'exécution	$f(x_f)$	$\|f(x_f)\|_\infty$
Valeur	$(1.3994\ ,\ 1.4717)^T$	11	$0.032\ s$	25.5375	$2.9424\ \times 10^{-4}$

Ainsi, on observe à nouveau que la méthode de Newton linéaire est la plus performante au regard du nombre d'itération ainsi que du temps d'exécution.
On remarque que toutes les méthodes trouvent le même résultat.
Ainsi, pour répondre à notre problème, il faut placer la box dans la pièce de coordonnées:

$$(1.3994\ ,\ 1.4717)^T$$

télécom
saint-étienne
école d'ingénieurs
nouvelles technologies


Chambre 4 (0;2)
Box
Chambre 3 (1;1)
Chambre 2 (0,1)
Chambre 1 (-1;
-1)