

# Compte-rendu - Projet d'optimisation continue

## Estimation robuste du centre d'une pièce circulaire

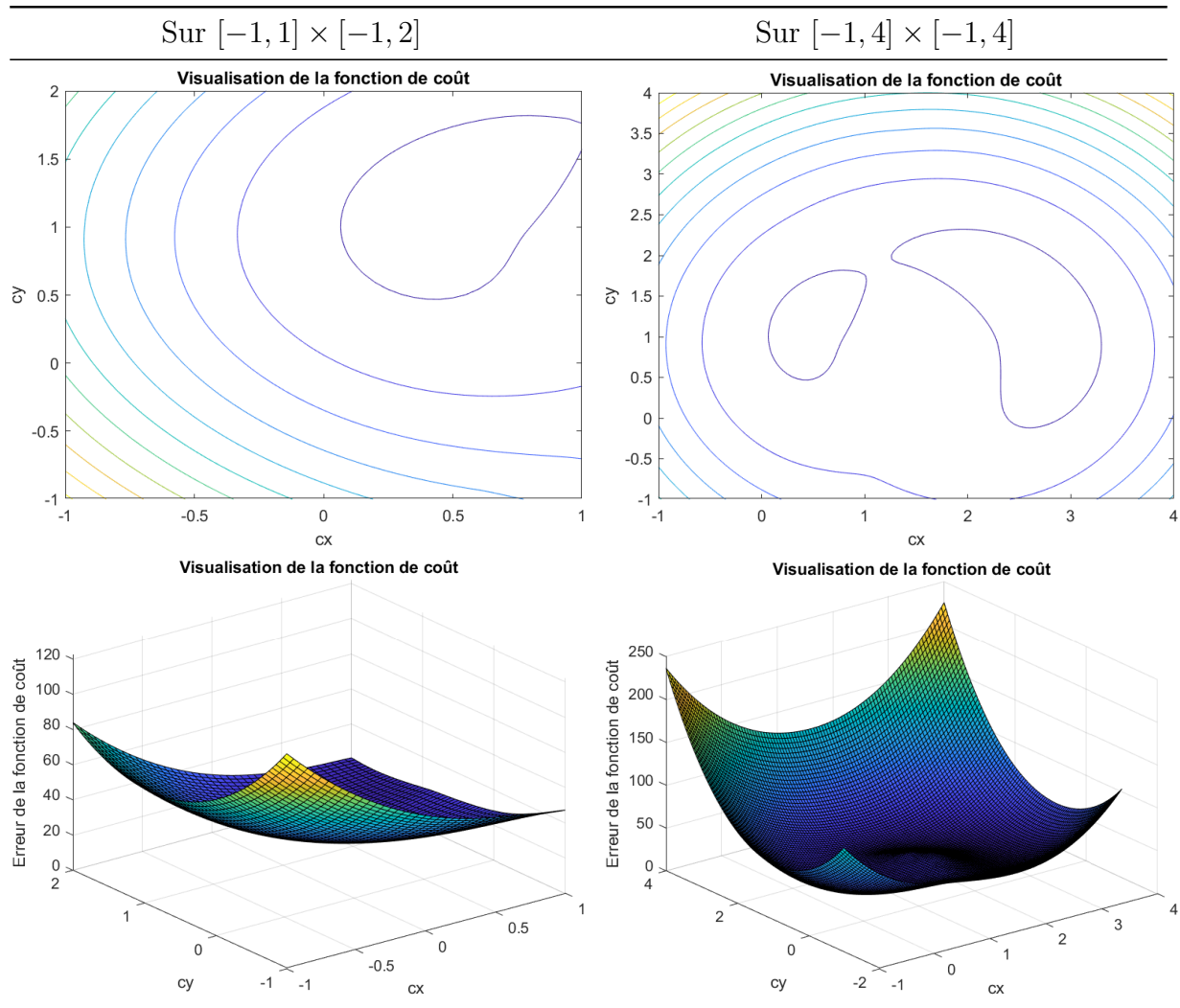
Justin Bossard

Tom Mafille

Ruben Verchere

Octobre 2024

1. On a :  $\mathcal{C}_{TLS}(c_x, c_y) = \sum_{i=1}^n (D_i - R)^2$ . On obtient les tracés suivants, avec un pas arbitraire de 0,05 :

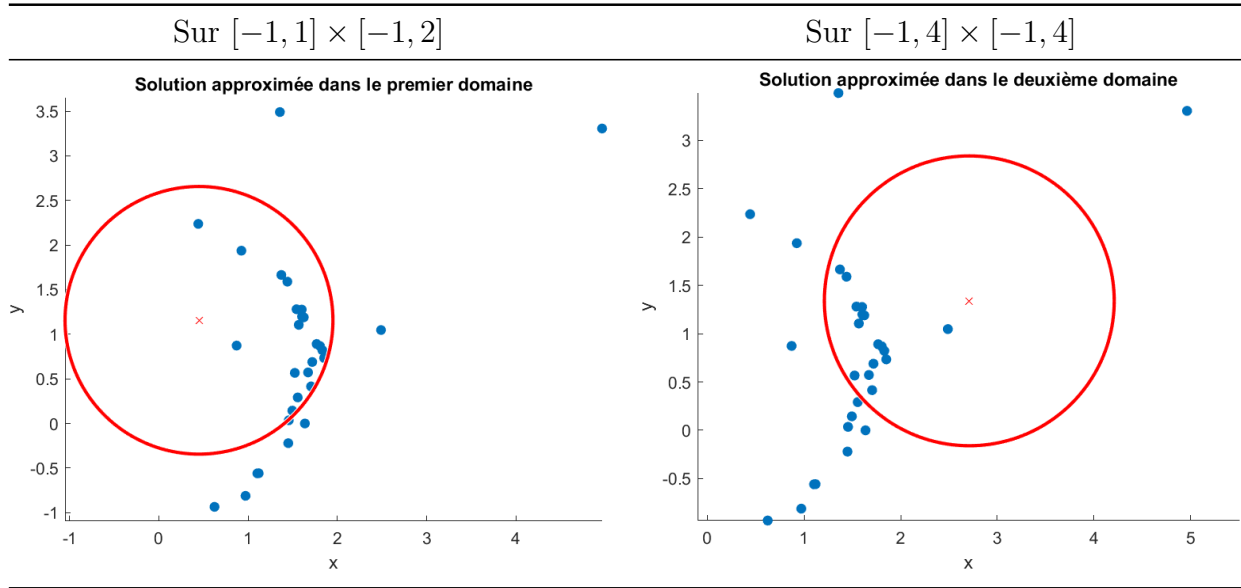


Sur la première figure, on a qu'un minimum, tandis que sur la seconde on en observe 2, qu'il faut départager. Il est préférable d'avoir une grande fenêtre, permettant de visualiser tous les minimums potentiels, plutôt qu'une petite fenêtre éliminant des solutions potentielles.

2. Pour déterminer  $c_x$  et  $c_y$  à  $10^{-4}$  près, on estime  $N = \frac{(\text{longueur intervalle } x) \times (\text{longueur intervalle } y)}{\text{pas}^2}$  fois la fonction de coût  $\mathcal{C}_{TLS}$ , dans la mesure on l'on passe dans deux boucles **for**, sur l'intervalle **départ:pas:arrivée**.

On a donc, sur  $[-1, 1] \times [-1, 2]$ ,  $N_1 = \frac{2 \times 3}{10^{-8}} = 6 \times 10^8$  estimations de la fonction de coût, et  $N_2 = 25 \times 10^8$  estimations pour l'intervalle  $[-1, 4] \times [-1, 4]$ .

On obtient les cercles et les nuages de points suivants :



On observe que les points aberrants influencent grandement le centre obtenu. La méthode n'est donc pas adaptée pour pouvoir déterminer le centre.

Pour pouvoir approximer le rayon  $R$ , il faut prendre en compte le nombre d'estimations du rayon par boucle **for**, qui est donc de  $N_R = \frac{2,5-0,5}{10^{-4}} = 2 \times 10^{-4}$ .

Le nombre d'estimations de  $\mathcal{C}_{TLS}$ , est dorénavant de  $N'_1 = N_1 \times N_R = 1,2 \times 10^{13}$  sur  $[-1, 1] \times [-1, 2]$ , et de  $N'_2 = N_2 \times N_R = 5 \times 10^{13}$  sur  $[-1, 4] \times [-1, 4]$ .

3. En calculant le gradient à la main :

$$\begin{aligned}
 \nabla \mathcal{C}_{TLS}(c_x, c_y) &= \begin{pmatrix} \frac{\partial}{\partial x_i} \sum_{i=1}^n (D_i - R)^2 \\ \frac{\partial}{\partial y_i} \sum_{i=1}^n (D_i - R)^2 \end{pmatrix} \\
 &= \begin{pmatrix} \sum_{i=1}^n \frac{\partial}{\partial x_i} (D_i - R)^2 \\ \sum_{i=1}^n \frac{\partial}{\partial y_i} (D_i - R)^2 \end{pmatrix} \\
 \nabla \mathcal{C}_{TLS}(c_x, c_y) &= \begin{pmatrix} 2 \sum_{i=1}^n (c_x - x_i) \left(1 - \frac{R}{D_i}\right) \\ 2 \sum_{i=1}^n (c_y - y_i) \left(1 - \frac{R}{D_i}\right) \end{pmatrix}
 \end{aligned}$$

#### 4. Tests à faire

5. En représentant le champ de gradient avec la fonction `quiver`, avec les lignes de contours, on a :

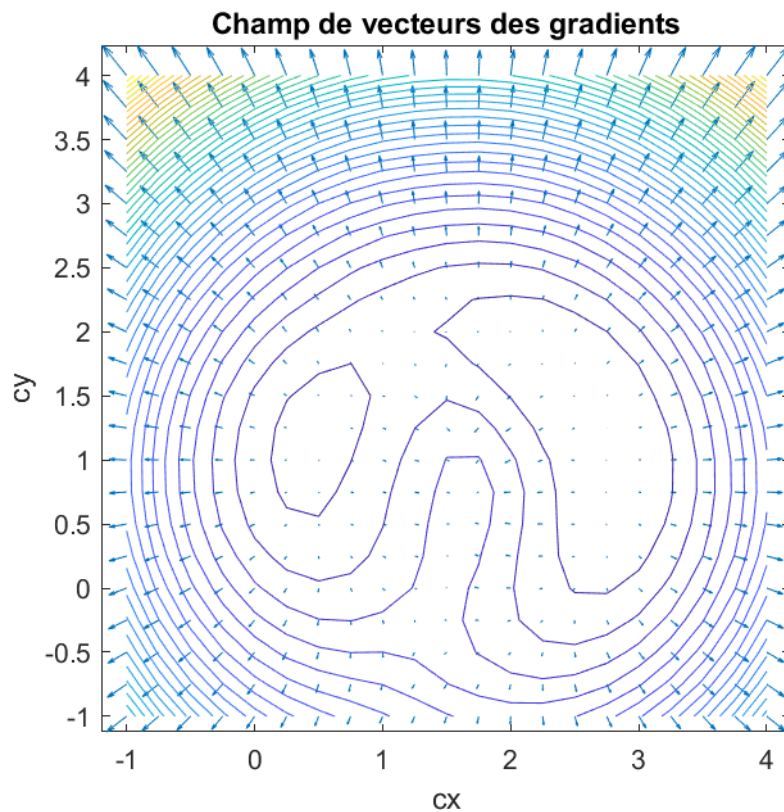
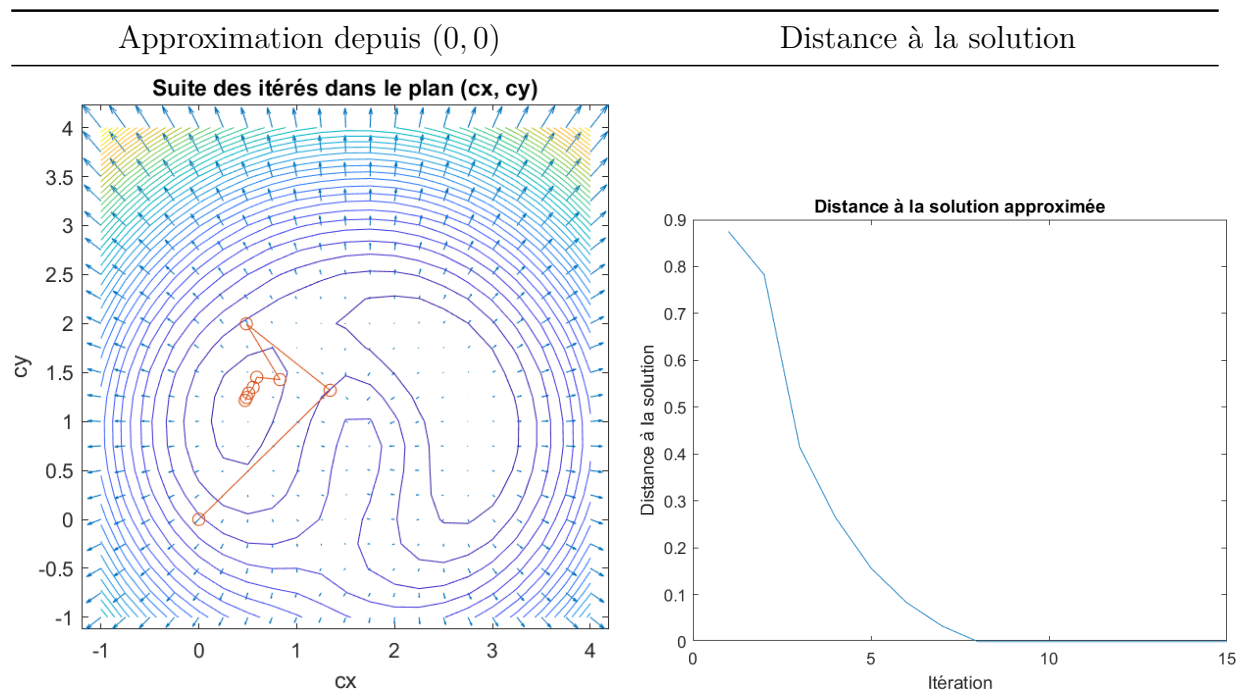


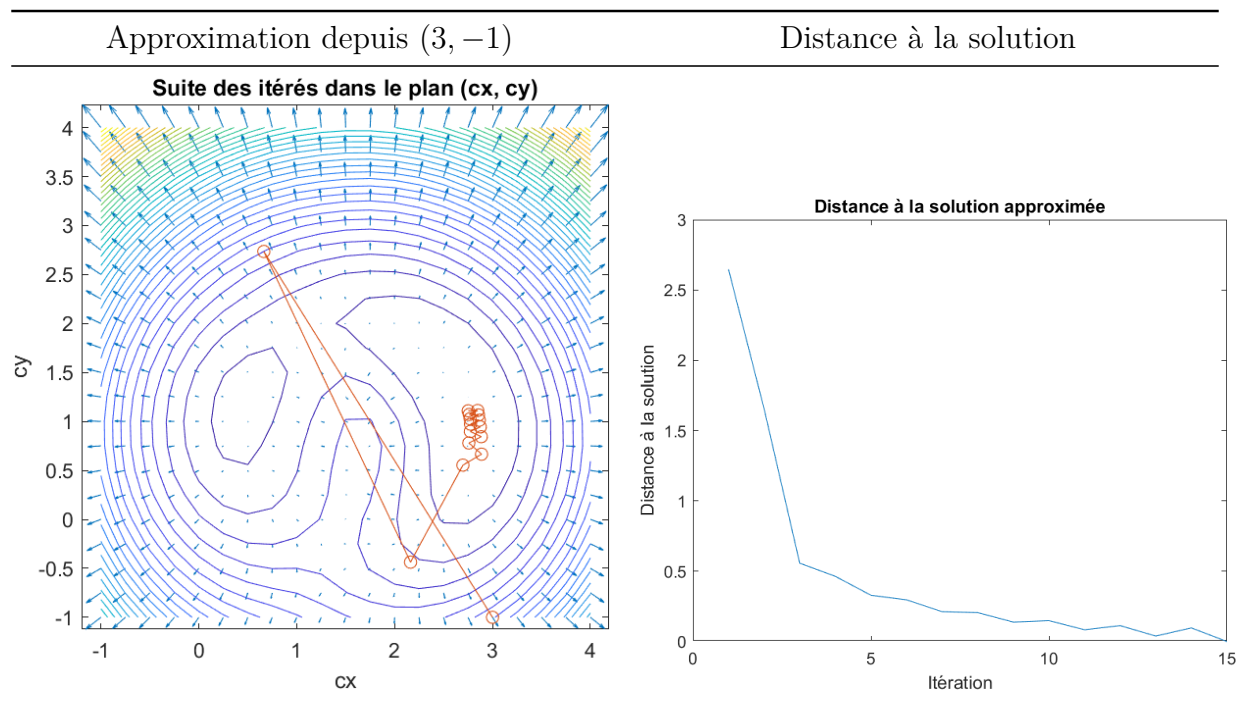
Figure 1: Champ des gradients

Le gradient est bien orthogonal aux lignes de niveaux.

6. Par la méthode des plus fortes pentes, avec l'algorithme de Fletcher et Lemaréchal, on obtient la solution suivante :



7. Selon le point de départ, on peut tomber dans le mauvais minimum, et donc avoir une solution erronée, comme l'illustre la figure suivante :



Ainsi, il faut pour utiliser cet algorithme de manière pertinente avoir au préalable une idée de la solution voulue.

8. Avec la méthode de quasi-Newton, on obtient les courbes suivantes :
9. ...
10. ...
  1. Pour la nouvelle expression du gradient, on a :
  2. Tests à faire
  3. Représentation du gradient avec la fonction `quiver` et les lignes de contours
  4. Méthode des plus fortes pentes, algo Fletcher-Lemaréchal
  5. Même étude avec d'autres points de départs