TD4 - Optimisation sous contraintes
Multiplicateurs de Lagrange

Alexandra Bac

Optimisation
Master IS - 1ère année

Pour réaliser ce TP, vous pourrez vous appuyer sur le matériel de TP 4 téléchargeable sur ma page internet.

Exercice 1 (Algorithme de Newton-Raphson)**.** On veut résoudre un système de n équations non linéaires à n inconnues :

$$\begin{cases} f_1(x_1, \ldots, x_n) = 0 \\ \vdots \\ f_n(x_1, \ldots, x_n) = 0 \end{cases}$$

S'il s'agissait de résoudre une équation $f(x) = 0$ (on à une seule inconnue), on pourrait utiliser l'algorithme de Newton et itérer :

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$$

à partir d'un point x_0 proche de la solution.

Mais ici, il s'agit bien de résoudre $f(\vec{x}) = \vec{0}$, mais avec :

$$\begin{array}{rcl} f : \mathbb{R}^n & \to & \mathbb{R}^n \\ \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} & \mapsto & \begin{pmatrix} f_1(x_1, \ldots, x_n) \\ \vdots \\ f_n(x_1, \ldots, x_n) \end{pmatrix} \end{array}$$

L'algorithme de Newton peut en fait se généraliser à la dimension n de la manière suivante (algorithme de Newton-Raphson) : si f est $\mathcal{C}^2$, alors, l'algorithme :

$$\begin{cases} \text{Soit } x_0 \in \mathbb{R}^n \\ \text{On itère :} \\ \qquad f'(x_k)(x_{k+1} - x_k) = -f(x_k) \\ \text{jusqu'à } \|x_{k+1} - x_k\| < \varepsilon \end{cases}$$

où $f'(x)$ est le jacobien de f, c'est-à-dire la matrice :

$$\begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \vdots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{pmatrix}$$

1. Récupérer dans le matériel de TP la fonction :
   `function [res] = NewtonRaphson(f, fprime, x0, eps, itermax)`

2. Tester cet algorithme sur le système du cours (coca-cola avec volume 1) :

$$\begin{cases} f_1(r,h,\lambda) = 4\pi r + 2\pi h + 2\lambda\pi r h & = 0 \\ f_2(r,h,\lambda) = 2\pi r + \lambda\pi r^2 & = 0 \\ f_3(r,h,\lambda) = \pi r^2 h - 1 & = 0 \end{cases} \tag{1}$$

   On a donc :

$$f'(r,h,\lambda) = \begin{pmatrix} 4\pi + 2\lambda\pi h & 2\pi + 2\lambda\pi r & 2\pi r h \\ 2\pi + 2\lambda\pi r & 0 & \pi r^2 \\ 2\pi r h & \pi r^2 & 0 \end{pmatrix}$$

   Essayez de partir de $[0.5; 0.5; 0.5]$ (est-ce logique par rapport à la forme vue en cours ?) puis de $[1; 1; 1]$.

Exercice 2 (Multiplicateurs de Lagrange)**.** On revient à notre problème précédent ; on considère une surface implicite :

$$\mathcal{S} = \left\{(x,y,z) \in \mathbb{R}^3 \,;\, f(x,y,z) = 0\right\}$$

et un point $u = (a,b,c) \in \mathbb{R}^3$. On veut calculer le projeté de u sur la surface (donc le point le plus proche de u appartenant à la surface).

1. Rappeler la modélisation du problème comme un problème d'optimisation sous contraintes.

2. Formuler le lagrangien du problème et le système à résoudre pour trouver ce projeté.

3. Cas d'un plan :

   (a) La surface est alors donnée par $f(x,y,z) = ax + by + cz - d = 0$, que devient le système précédent dans ce cas ?

   (b) Représenter graphiquement la surface et le point en utilisant la fonction fournie en matériel de TP :
   `plotImplicit3D(f, min, max, pas)`

   (c) Résoudre et représenter le résultat.

4. Cas d'une quadrique :

(a) La surface a pour équation $(f, x, y, z) = 0$ où f est un polynôme de degré 2 (on pourra par exemple tester $x^2 + 2y^2 + 3z^2 = 1$).

(b) Représenter graphiquement la surface et le point en utilisant la fonction fournie en matériel de TP :
`plotImplicit3D(f, min, max, pas)`

(c) Ecrire le système précédent dans ce cas.

(d) Résoudre en utilisant l'algorithme de Newton Raphson de l'exercice précédent.