CC2: optimisation numérique

Exercice 1

On s'intéresse au code Python suivant :

```
import numpy as np
def scalar(x, y):
    return np.sum(x * y)
def HOptim(J, nablaJ, cE, nablacE, X0, lamb0, alpha1, alpha2, N):
    X = X0
    lamb = lamb0
    for i in range(N):
        X = X - alpha1 * (nablaJ(X) + lamb * nablacE(X))
        lamb = lamb + alpha2 * cE(X)
    return X,lamb
n = 4
A = np.array([[2, -1, 0, 0], [-1, 2, -1, 0], [0, -1, 2, -1], [0, 0, -1, 2]])
b = np.ones((n, 1))
def quad(x):
    return 0.5 * scalar(A @ x, x) - scalar(b, x)
def nablaquad(x):
    return A @ x - b
def cE(x):
    return np.sum(x) - 1
def nablacE(x):
    return np.ones((n,1))
# Parameters
alpha1 = 0.1
alpha2 = 0.1
X0 = np.zeros((n, 1))
lamb0 = 3
N = 100
X,lamb = HOptim(quad, nablaquad, cE, nablacE, X0, lamb0, alpha1, alpha2, N)
```

1. Indiquer quel est l'objectif de ce code, en particulier celui de la fonction `HOptim`.
2. Déterminier la solution exacte du problème pour l'exemple considéré dans le code.
3. Expliquer précisément le principe des lignes `X = X - alpha1 * (nablaJ(X) + lamb * nablacE(X))` et `lamb = lamb + alpha2 * cE(X)`.
4. On remplace $N = 100$ par $N = 1$ (1 seule itération). Que valent X et `lamb` après l'exécution du programme ?

Exercice 2

On s'intéresse à la minimisation d'une fonction $J : \mathbb{R}^n \to \mathbb{R}$ strictement convexe, coercive et continue sous la contrainte égalité affine $Dx - d = 0$ avec $D \in \mathcal{M}_{m,n}(\mathbb{R})$ de rang $m < n$ et $d \in \mathbb{R}^m$.

1. Montrer que le problème de minimisation possède une unique solution $x^* \in \mathbb{R}^n$.
   On considère le Lagrangien associé :
   $$\mathcal{L}(x, \lambda) = J(x) + {}^t\lambda(Dx - d)$$
   .
2. Soit $\lambda \in \mathbb{R}^m$ fixé. Montrer qu'il existe un unique x^*_λ minimum de $x \mapsto \mathcal{L}(x, \lambda)$ sur $\mathbb{R}^n$.
3. Montrer que la fonction
   $$H(\lambda) = \min(\mathcal{L}(x, \lambda), x \in \mathbb{R}^n\}$$
   est concave sur $\mathbb{R}^m$.
4. On suppose ici que J est une fonctionnelle quadratique :
   $$J(x) = \frac{1}{2}\,{}^t xAx - {}^t bx$$
   avec A matrice symétrique définie positive.
   Calculer explicitement x^*_λ et $H(\lambda)$ en fonction de A, D, b, d. Montrer que
   $$\nabla H(\lambda) = Dx^*_\lambda - d$$
   Déterminer l'unique maximum λ_0 de H et monrtrer que
   $$x^*_{\lambda_0} = x^*$$
5. Montrer qu'on a toujours dans le cas général
   $$\nabla H(\lambda) = Dx^*_\lambda - d$$
   On écrira l'expression de $H(\lambda)$ en fonction de x^*_λ et on admettra que $\lambda \mapsto x^*_\lambda$ est différentiable.