CC2: optimisation numérique

Exercice 1

On s'intéresse à la minimisation d'une fonction quadratique $J : \mathbb{R}^n \to \mathbb{R}$ telle que

$$J(x) = \frac{1}{2}\,{}^t xAx - {}^t bx$$

avec A une matrice symétrique définie positive de taille n et $b \in \mathbb{R}^n$, sous la contrainte

$$Cx = 0$$

avec C une matrice de taille $p \times n$ ($p < n$), de rang maximal p.

1. Montrer que J admet un unique minimum global x^* tel que $(x^*, \lambda^*) \in \mathbb{R}^n \times \mathbb{R}^p$ est solution unique de
$$\begin{cases} Ax^* + {}^tC\lambda^* = b \\ Cx^* = 0 \end{cases}$$
2. On propose d'approcher x^* par la méthode suivante pour$(x_0, \lambda_0) \in \mathbb{R}^n \times \mathbb{R}^p$ donnés :
$$\begin{cases} x_{k+1} = x_k - \rho_1(Ax_k - b +^t C\lambda_k) \\ \lambda_{k+1} = \lambda_k + \rho_1\rho_2 Cx_{k+1} \end{cases}$$
Jusitifier l'introduction de cette méthode à l'aide du Lagrangien du problème.
3. On note $\beta = |||I - \rho_1 A|||$ la norme subordonnée de $I - \rho_1 A$. On rappelle que si $x \in \mathbb{R}^n$,
$$||Ax|| \leq |||A|||.||x||$$
et que si A est symétrique définie positive, alors $|||A|||$ est égal à sa plus grande valeur propre.
4. Montrer que si ρ_1 est suffisamment petit, alors $\beta < 1$.
5. Montrer que
$$||\lambda_{k+1} - \lambda^*||^2 \leq ||\lambda_k - \lambda^*||^2 + (\rho_1\rho_2)^2|||{}^tCC|||.||x_{k+1} - x^*||^2 + 2\rho_1\rho_2 <^t C(\lambda_k - \lambda^*), x_{k+1} - x^* >$$
6. On admet alors que si ρ_2 est suffisamment petit, il existe $\gamma > 0$ tel que
$$\gamma||x_{k+1} - x^*||^2 \leq \left(\frac{||\lambda_k - \lambda^*||^2}{\rho_2} + \beta||x_k - x^*||^2\right) - \left(\frac{||\lambda_{k+1} - \lambda^*||^2}{\rho_2} + \beta||x_{k+1} - x^*||^2\right)$$
Montrer que pour un tel choix de paramètres, on a $\lim_{k \to +\infty} x_k = x^*$.
7. Que peut-on dire de la suite $(\lambda_k)_{k \in \mathbb{N}}$?

Exercice 2

Soit la fonction.

$$J(x,y) = 3x^4 + y^2 - 3xy$$

à minimiser sur l'ensemble

$$D = \{(x,y) \in \mathbb{R}^2, 1 \leq x \quad \text{et} \quad y \geq 3\}$$

1. Montrer que ce problème admet une unique solution (qu'on ne cherchera pas à déterminer).
2. Ecrire, de manière explicite une itération de l'algorithme du gradient projeté (à pas constant) sur cet exemple.

Exercice 3

Sous sa forme la plus élémentaire, une stratégie d'évolution s'écrit ainsi :

```
///////////////////////////////////////////////
n=10;
lambda=50;
mu=5;
Ngen=200;
sigma=0.01;
////////////////////////////////////////////////
function y=sphere(x)
    y=sum(x.*x);
endfunction
//////////////////////////////////////////////////
Ap=2*rand(mu,n+1)-ones(mu,n+1)// initialisation
Ae=zeros(lambda,n+1)
//
for i=1:Ngen
//
  X=1/mu*sum(Ap(:,1:n),'r')  // croisement
  for i=1:lambda
     Ae(i,1:n)=X+sigma*rand(1,n,'normal') // mutation
     Ae(i,n+1)=sphere(Ae(i,1:n))   // evaluation
  end
  [a,b]=gsort(-Ae(:,n+1))
  Ae=Ae(b,:)
  Ap=Ae(1:mu,:)  // selection virgule (enfants seulement)
end
```

1. Que représentent les paramètres `n`, `lambda` et `mu` ?
2. On souhaite enlever l'opérateur de croisement. A la place, les enfants sont issus d'un parent tiré (à chaque fois) aléatoirement. Comment proposez vous de modifier le code ?
3. Que valent `a` et `b` si on tape dans Scilab : `[a,b]=gsort([3.3,4,2,6,1])` ?
4. Quel sera le temps d'éxécution de ce code si on remplace la fonction `sphere` par une fonction nécessitant pour chaque évaluation 1 minute de calcul ?