Année Universitaire 2020-2021
$4^{\text{ème}}$ année GM
Projet d'optimisation en grande dimension (Code et Rapport)
A rendre pour le 06/04/2021

L'ensemble du projet devra être rendu sous forme de fichier .zip ou .rar en main propre. L'ensemble des codes devra être placé dans un dossier où il est possible de tous les exécuter sans créer de conflits entre différents scripts/définitions de variables/noms de fonctions, et qui comportera un fichier texte résumant le rôle de chaque fichier. Il est possible de hiérarchiser ces codes via des sous-dossiers, en prévoyant par exemple un sous-dossier pour les graphiques. Si aucun graphique n'est demandé explicitement pour certaines questions mais que votre argumentation peut être illustrée, n'hésitez pas à en fournir. Un dossier de code est attendu par groupe de 4 ou 5. Les réponses aux questions devront être fournies sous forme d'un rapport détaillé (compte-rendu complet et précis).
La notation globale tiendra compte de la clarté, de l'organisation, des commentaires et de l'exactitude des codes, ainsi que la manière dont vous optimiserez le nombre d'opérations et le stockage des informations. La manière de mettre en valeur les résultats graphiquement sera également un point important de la note de programmation. L'autre partie de la notation portera sur l'étude des algorithmes et l'analyse critique de vos résultats.

Exercice 1 (Comparaison des algorithmes.) On veut minimiser la fonctionnelle

$$J(u) = \frac{1}{2}\langle Au, u\rangle - \langle b, u\rangle,$$

avec $A = \mathbb{I}_N$, $u = (u_1, \ldots, u_N)^T \in \mathbb{R}^N$ et $b = (1, -1, 1, -1 \ldots)^T \in \mathbb{R}^N$, en respectant N contraintes :

$$\begin{aligned} u_i + 2u_{i+1} &\leq 0,\ i = 1, \ldots, N-1, \\ u_N &\leq 0. \end{aligned} \tag{1}$$

1. Présenter graphiquement l'ensemble des solutions admissibles pour $N = 2$, commenter brièvement.
2. Programmer trois fonctions permettant de résoudre ce problème via des méthodes de décomposition par prix, par quantités, et par prédiction en dimension N. Discuter des avantages et inconvénients algorithmiques de ces méthodes. Après l'avoir écrit pour chaque cas, étudier la stratégie de minimisation du Lagrangien.
3. Programmer une fonction générale KKT permettant à n'importe quel moment de vérifier les différentes conditions KKT pour chaque méthode. Observer les résultats obtenus à des itérations diverses pour chaque méthode, et commenter.
4. Programmer une fonction permettant de comparer les résultats obtenus via ces trois algorithmes, et comparer le temps d'exécution, le nombre d'itérations et la complexité.
5. Pour $N = 200$ et $N = 251$ avec une précision de l'ordre de $\varepsilon = 10^{-4}$, étudier le temps d'exécution et le nombre d'itérations nécessaire à chaque méthode. Étudier ces deux critères en faisant varier ε et les valeurs du pas.
6. Qu'implique l'ajout d'une sur-diagonale telle que $A_{ij} = -\alpha$, $i = 1, \ldots, N-1$, $j = i+1$, $\alpha \in \mathbb{R}$, selon les valeurs de α ?

Exercice 2 (Réseaux de distribution d'eau connectés) On a vu en cours que le problème de l'optimisation de N sous-réseaux de distribution d'eau connectés par une usine de refoulement peut se mettre sous la forme :

$$\begin{array}{l} \displaystyle\min_{\{\mathbf{u}_i=(u_i^{(1)},u_i^{(2)})\}_{i=1,N+1}} \sum_{i=1}^{N+1} J_i(\mathbf{u}_i) \\ \text{sous les contraintes couplantes :} \\ \displaystyle\sum_{i=1}^{N} \mathbf{u}_i - \mathbf{u}_{N+1} = 0, \end{array} \tag{2}$$

avec, pour $i = 1, N$, $\mathbf{u}_i \in [0, a_i] \times [0, b_i - a_i]$ $(b_i > a_i > 0)$, la matrice $A_i = diag(\alpha_i, \beta_i) \in \mathbb{R}^{2,2}$ (avec $\alpha_i > 0, \beta_i > 0$)

$$J_{N+1}(\mathbf{u}_{N+1}) = \frac{1}{2}\langle A_{N+1}\mathbf{u}_{N+1}, \mathbf{u}_{N+1}\rangle$$

et, pour $i = 1, N$, $J_i(\mathbf{u}_i)$ la fonction valeur définie par la minimisation :

$$\begin{array}{l} \displaystyle\min_{\mathbf{v}_i=(v_i^{(1)},v_i^{(2)})} J_i(\mathbf{v}_i) \\ \text{sous les contraintes :} \\ -v_i^{(1)} - u_i^{(1)} + a_i \leq 0, \\ v_i^{(1)} + v_i^{(2)} + u_i^{(1)} + u_i^{(2)} - b_i = 0. \end{array} \tag{3}$$

1. Programmer trois fonctions permettant de résoudre ce problème via des méthodes de décomposition par prix, par quantités, et par prédiction en dimension $N + 1$ (pour la décomposition par prédiction, affecter les contraintes couplantes au $(N + 1)$-ème sous-problème). Discuter des avantages et inconvénients algorithmiques de ces méthodes.
2. Comparer une programmation séquentielle et parallèle pour résoudre un tel problème.
3. Pour N entre 20 et 200 et avec un choix des valeurs des paramètres $a_i, b_i, \alpha_i, \beta_i$ et une précision ε, étudier le temps d'exécution et le nombre d'itérations nécessaire à chaque méthode ainsi que l'influence des pas et des precisions.

Exercice 3 (Optimisation d'un portefeuille d'actions et le modèle Markowitz) Supposons que l'on posséde N actions, que l'on représente par des variable aléatoires X_i, pour $i = 1, N$. Chaque action rapporte en moyenne à l'actionnaire $e_i = \mathbb{E}(X_i)$ (espérance de X_i) au bout d'un an. On suppose que l'on investit une somme S donnée, et l'on note $u_i \in \mathbb{R}$, la proportion de la somme investie dans l'action i. Ainsi, on a : $\displaystyle\sum_{i=1}^{N} u_i = 1$. Le portefeuille total est représenté par la variable aléatoire : $X = \displaystyle\sum_{i=1}^{N} u_i X_i$ et rapporte ainsi en moyenne $\mathbb{E}(X) = \displaystyle\sum_{i=1}^{N} u_i e_i$.

Dans son modèle Markowitz utilise les 2 premiers moments de la variable à étudier, qui sont l'espérance et la variance, correspondant respectivement au rendement global espéré et au risque encouru : $\theta = \mathbb{E}(X)$ et $\sigma^2 = var(X) =< U, QU >$, où $U = (u_i)_{i=1,N}$ et Q la matrice des covariances.

On peut soit imposer un rendement donné $R_e > 0$,i.e., $\sum_{i=1}^{N} u_i e_i \geq R_e$ ou fixer une dose de risque donnée $D_e > 0$ i.e., $var(X) = D_e$.

Ainsi on peut modéliser le risque du portefeuille par les deux problèmes quadratiques sous contraintes suivant :

Problème 1 :

$$\begin{aligned}
&\min_U \frac{1}{2} < U, QU > \\
&\text{sous les contraintes :} \\
&\textstyle\sum_{i=1}^N u_i e_i \geq R_e, \\
&\sum_{i=1}^N u_i = 1.
\end{aligned} \tag{4}$$

Problème 2 :

$$\begin{aligned}
&\min_U \sum_{i=1}^N u_i e_i \\
&\text{sous les contraintes :} \\
&< U, QU >= D_e, \\
&\sum_{i=1}^N u_i = 1.
\end{aligned} \tag{5}$$

Nous pouvons y ajouter la contrainte d'interdire la vente à découvert, et dans ce cas, nous nous forcerons à conserver les parts u_i à investir, positives.

Exercice 4 (Optimisation de la gestion de smart grids) Lire le projet d'étudiant ci-joint. Ensuite, formuler et résoudre le problème d'optimisation lié au modèle de la partie III (via des méthodes de décomposition par prix, et par quantités).