Examen – Durée 1h45

Les notes de cours sont autorisées. L'épreuve comporte deux exercices indépendants. Il est conseillé de commencer par lire la totalité de l'énoncé.

1 Exercice 1 : programmation linéaire (12 points)

La sonde *Rosetta* s'est posée sur la comète *Tchourioumov-Guérassimenko* le 12 novembre 2014, avec à son bord des instruments de mesure et un robot permettant de prélever du sable (en réalité, c'est seulement le module *Philæ* de cette sonde qui s'est posé). Par suite d'une avarie technique, la capacité maximale de la sonde est réduite à 2500 kg et à 15 m^3. Elle ne pourra donc emporter qu'une partie des instruments de mesure et du sable déjà prélevé. On suppose pour simplifier que tous les instruments de mesure font le même poids et le même volume : 100 kg et $1,5$ m^3. Quant au sable, il pèse 250 kg par m^3. On cherche bien sûr à « rentabiliser » la mission au maximum.

1. En comparaison des PL (programmes linéaires) vus en cours, montrez que ce PL est mixte : une variable de décision est discrète, l'autre est continue.
2. Sachant que 20 instruments ont été débarqués et que 12 m^3 de sable ont été prélevés, montrez que la totalité des instruments et du sable ne peut être emportée, que ce soit à cause du poids total ou du volume total.
3. L'Agence Spatiale Européenne (ESA) doit faire face à un choix difficile : chaque instrument de mesure coûte 1 million d'euros, mais le sable de la comète *Tchourioumov-Guérassimenko* n'a pas de valeur marchande. Elle considère donc le « prix » du sable par m^3 (c'est-à-dire son intérêt à des fins scientifiques) comme un paramètre variable, mais connu, noté x. Écrivez le PL $\mathcal{P}'$, sous forme canonique, qui consiste à minimiser la perte financière de cette mission spatiale.
4. Sur la feuille quadrillée, représentez graphiquement l'ensemble admissible du PL **relaxé** $\mathcal{P}$ associé à $\mathcal{P}'$. Échelle : 1 cm par instrument de mesure en abscisse, 1 cm par m^3 de sable en ordonnée.
5. Utilisez cette représentation graphique pour déterminer la solution du PL relaxé $\mathcal{P}$, pour les « prix » suivants du m^3 de sable : $x = 0$, $x = 1$ million d'euros, $x = 100$ millions d'euros. Commentez brièvement ces résultats.
6. On souhaite désormais résoudre le PL mixte $\mathcal{P}'$, en fixant le « prix » du sable à $x = 1$ million d'euros par m^3. Tracez sur le graphique l'ensemble admissible de $\mathcal{P}'$, qui est constitué d'un ensemble de segments de droites.
7. Prédisez la solution de $\mathcal{P}'$ en vous appuyant sur le graphique précédent.
8. En vous inspirant de la méthode de séparation/évaluation vue en cours, imaginez un algorithme permettant de résoudre ce PL mixte. Détaillez chaque étape de cet algorithme, en veillant bien sûr à obtenir la même solution que celle trouvée graphiquement.
9. Construisez l'arbre des différents PL relaxés ayant permis de mener à bien cette résolution.

2 Exercice 2 : programmation dynamique (8 points)

On reprend le problème de détection de fissures dans une image de chaussée déjà abordé dans la séance de révision (cf. figure 1). Une telle image en niveaux de gris peut être considérée comme une matrice I, dont les éléments $I(i,j)$ sont des entiers compris entre 0 et 255 : i désigne le numéro de ligne, j le numéro de colonne.

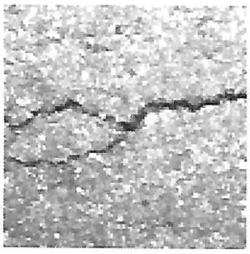

FIGURE 1 – Exemple d'image de chaussée dans laquelle on cherche à détecter les fissures.

Un chemin menant du point $D = (i_D, j_D)$ au point $A = (i_A, j_A)$ est défini comme une séquence de pixels comportant un seul pixel (i_j, j) par colonne j, de telle sorte que deux pixels consécutifs soient voisins, au sens des huit plus proches voisins. On suppose que les contraintes $j_D \leq j_A$ et $|i_D - i_A| \leq j_A - j_D$ sont satisfaites. Le programme doit déterminer le chemin entre D et A qui minimise l'objectif suivant :

$$z = \sum_{j=j_D}^{j_A} I(i_j, j)$$

On s'intéresse à l'exemple ci-dessous, où les points D et A sont indiqués en rouge :

0	2	1	3	0	2	4	3	5
2	0	1	1	3	5	1	0	1
0	2	0	0	1	1	3	1	3
2	2	1	1	3	0	2	3	3
0	3	2	0	0	5	0	2	5

1. Appliquez l'algorithme de programmation dynamique permettant de calculer, en chaque pixel atteignable, le score du chemin optimal menant de D à ce pixel.
2. Déduisez de la question précédente le chemin optimal menant de D à A, en entourant en rouge les pixels parcourus.
3. Montrez qu'en partant de D, et en se déplaçant vers le voisin de droite ayant le plus faible niveau de gris, on n'atteint pas A. De même, montrez qu'en partant de A, et en se déplaçant vers le voisin de gauche ayant le plus faible niveau de gris, on n'atteint pas D. Conclusion ?
4. Dans l'algorithme de programmation dynamique, les « poids » des voisins sont définis de telle sorte que le problème d'optimisation locale s'écrive :

$$g(i,j) = \min\{g(i-1,j-1) + p_1 * I(i,j), g(i,j-1) + p_2 * I(i,j), g(i+1,j-1) + p_3 * I(i,j)\}$$

   Donnez l'expression du nouvel objectif correspondant aux poids $(p_1, p_2, p_3) = (2, 1, 2)$. Rappel : le symbole de Kronecker $\delta(i_{j_1}, i_{j_2})$ vaut 1 si $i_{j_1} = i_{j_2}$, 0 sinon.
5. Quelle influence cette modification des poids pourrait-elle avoir sur la forme du chemin optimal ? Effectuez la résolution de ce nouveau problème, dans le cas de l'exemple précédent.