CONTRÔLE DE RECHERCHE OPÉRATIONNELLE

- Le contrôle est noté sur 23. La note sera laissée sur 20, et éventuellement remontée en fonction des résultats globaux.
- Polycopié de cours autorisé. Tout autre document interdit. Objets electroniques interdits.
- Les questions marquées d'une * sont un peu plus difficiles.

1. Creuser une rivière artificielle à moindre coût – 3 pts

Deux bassins s et t doivent être reliés par une rivière artificielle dans une région vallonnée. Cette rivière doit permettre de faire passer l'eau de s à t. Le graphe orienté de la Figure 1 indique les possibilités : un arc est une portion de rivière possible et l'orientation de l'arc indique le sens d'écoulement de l'eau sur cette portion. Le coût attaché à l'arc est le coût d'ouverture de cette portion. Donner la rivière de coût minimum reliant s à t. Justifier la réponse de façon détaillée – une partie des points sera accordée uniquement si l'algorithme le plus efficace est utilisé.

2. Identification des tâches critiques et calcul des marges d'un projet de construction d'un pavillon – 4 pts

Dans le tableau suivant sont indiquées les tâches à effectuer pour réaliser un pavillon, leurs durées, et leurs identifiants. La dernière colonne indique pour, chaque tâche, les tâches qu'il

Date: 22 février 2018.

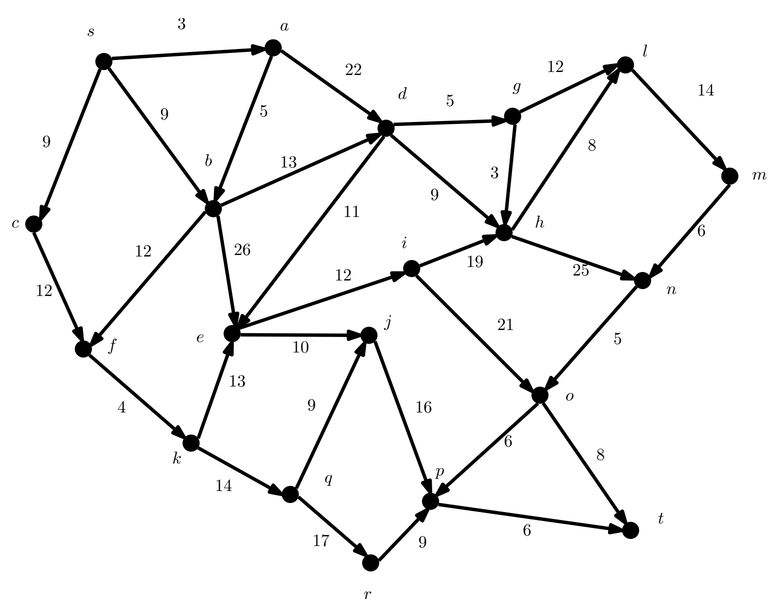

Figure 1. Région vallonée : il faut creuser une rivière de s à t au moindre coût

faut avoir complètement achevées pour pouvoir la débuter. S'il est indiqué "Début", c'est qu'aucune tâche particulière doit être effectuée au préalable.

Identifiant	Tâche	Durée (en jours)	Tâches devant être achevées
A	Déblaiement	5	Début
B	Baraquements	4	Début
C	Fondations	10	A,B
D	Murs	22	C
E	Canalisations extérieures	3	Début
F	Charpente	5	D
G	Toit	6	D,F
H	Canalisations intérieures	4	E,D
I	Electricité	3	D
J	Peinture	3	I,H,G

On considère que l'instant $t = 0$ correspond au début du chantier.

1. Pour toute tâche X, déterminer η_X l'instant au plus tôt auquel on peut commencer cette tâche. Justifier la réponse. (1 pt)

2. En déduire la durée minimale de ce projet. (1 pt)

3. Pour toute tâche X, déterminer π_X l'instant au plus tard auquel on peut commencer cette tâche. Justifier la réponse. (1 pt)

4. En déduire les tâches critiques et les marges pour chaque tâche. (1 pt)

3. Ordonnancement de voitures sur une chaîne de montage, d'après Estellon et al. – 10 pts

On s'intéresse à un problème central de l'industrie automobile : celui de la détermination de la séquence des voitures sur une chaîne de montage. Une voiture est caractérisée par une *couleur* et un sous-ensemble d'*options*. L'ensemble des couleurs est noté $\mathcal{C}$ et l'ensemble des options est noté $\mathcal{O}$. Une voiture est donc caractérisée par un couple (c, O) avec $c \in \mathcal{C}$ et $O \subseteq \mathcal{O}$. On appelle *classe* l'ensemble des voitures de même couple (c, O).

On suppose que les voitures produites par l'usine se partitionnent en un ensemble de classes $\mathcal{K}$. On note (c_k, O_k) le couple qui caractérise la classe k. Etant donné un ensemble de n voitures à produire, avec n_k voitures par classe k ($\sum_{k\in\mathcal{K}} n_k = n$), il s'agit de fixer la séquence des voitures sur la chaîne de montage, en satisfaisant la contrainte donnée au paragraphe suivant, et en minimisant le coût dont le calcul est précisé plus bas.

La contrainte que doit satisfaire la séquence de voitures est la suivante : il doit y avoir au plus r voitures de même couleur placées consécutivement sur la chaîne. En effet, si trop de voitures consécutives sont peintes de la même couleur, il est plus difficile d'évaluer la qualité de la peinture.

Le coût se décompose quant à lui en deux termes. L'un des termes est le coût de changement de couleurs. A chaque fois qu'il y a changement de couleur sur la chaîne de montage, une pénalité $\rho \geq 0$ est ajoutée. On dit qu'il y a *changement de couleur* si deux voitures consécutives doivent être peintes de couleurs différentes. Noter que si $n > r$, on est certain que la pénalité ρ sera payée au moins une fois pour toute solution réalisable. L'autre terme est le coût lié au déséquilibre des charges de travail. A toute option o est associé un couple d'entiers (p_o, q_o) qui signifie la chose suivante : Sur toute suite de q_o voitures consécutives, il serait bien qu'au plus p_o aient l'option o. Une pénalité est ajoutée en cas de dépassement. Cette pénalité est proportionnelle au dépassement : Si l'on a x voitures qui ont l'option o sur la suite de q_o voitures consécutives, avec $x > p_o$, une pénalité égale à $(x - p_o)\gamma_o$, avec $\gamma_o \geq 0$, est ajoutée. Ce calcul doit être fait pour toute suite de q_o voitures consécutives (toute "fenêtre").

Par exemple, supposons que l'on a deux options X et Y, deux couleurs `bleu` et `rouge` et 3 classes numérotées $1, 2, 3$ décrites par le tableau suivant, un 1 indiquant que l'option est présente et 0 qu'elle ne l'est pas.

Classe	1	2	3
Couleur	`bleu`	`bleu`	`rouge`
Option X	1	0	1
Option Y	0	1	1

Supposons de plus que les paramètres sont $r = 3$, $(p_{\mathrm{X}}, q_{\mathrm{X}}) = (2, 5)$, $(p_{\mathrm{Y}}, q_{\mathrm{Y}}) = (3, 4)$ et que $n_1 = 4$, $n_2 = n_3 = 3$. Le coût de la séquence (pour chaque voiture, on n'indique que sa classe)

$$1113222331$$

est égal à

$$4 \times \rho + (2+1+0+0+0+1) \times \gamma_{\mathrm{X}} + (0+0+0+1+1+1+0) \times \gamma_{\mathrm{Y}} = 4\rho + 4\gamma_{\mathrm{X}} + 3\gamma_{\mathrm{Y}}.$$

Les termes entre parenthèses correspondent aux différentes "fenêtres" de q_o voitures consécutives. Le premier 2 correspond aux 4 options X dans les 5 premières voitures de la suite, alors qu'il n'en fallait idéalement que 2. Le +1 qui le suit correspond aux 3 options X dans les 5 voitures aux positions 2 à 6 de la suite, alors qu'il n'en fallait idéalement que 2.

L'objectif est donc d'ordonner les n voitures en respectant la contrainte sur la peinture et en minimisant les coûts.

3.1. **Cas général – 6 pts.** On va construire un programme linéaire en nombres entiers modélisant ce problème. Pour cela, on va utiliser les variables suivantes.

- $x_{i,k}$ vaut 1 si la voiture en position $i \in \{1, \ldots, n\}$ est de classe $k \in \mathcal{K}$ et vaut 0 sinon.
- $y_{i,c}$ vaut 1 si la voiture en position $i \in \{1, \ldots, n\}$ est de couleur $c \in \mathcal{C}$ et vaut 0 sinon.
- $z_{i,o}$ vaut 1 si la voiture en position $i \in \{1, \ldots, n\}$ a l'option $o \in \mathcal{O}$ et vaut 0 sinon.
- u_i vaut 1 si les voitures en position i et $i+1$ sont de couleurs différentes et vaut 0 sinon.
- $v_{i,o}$ vaut 0 si le nombre de voitures en positions $i, i+1, \ldots, i+q_o-1$ avec l'option o est $\leq p_o$ et vaut ce nombre moins p_o sinon.

1. Ecrire la fonction objectif. (0.5 pt)

2. Ecrire des égalités mettant en jeu les $x_{i,k}$ et qui décrivent toutes les suites possibles – sachant qu'il y a n_k voitures de la classe k pour tout k – sans prise en compte de la contrainte qui limite le nombre de voitures consécutives de même couleur. (1 pt)

3. En introduisant le paramètre $\epsilon_{k,c}$ si la classe k est de couleur c et 0 sinon, et le paramètre $\delta_{k,o}$ qui vaut 1 si la classe k possède l'option o et 0 sinon, écrire des égalités liant d'une part les variables $x_{i,k}$ et $y_{i,c}$ et d'autre part les variables $x_{i,k}$ et $z_{i,o}$. (1 pt)

4. Ecrire les inégalités qui assurent la prise en compte de la contrainte qui limite le nombre de voitures consécutives de même couleur. (1 pt)

5. Ecrire des inégalités liant les u_i et les $y_{i,c}$ d'une part, et les $v_{i,o}$ et les $z_{i,o}$ d'autre part. (1 pt)

6. Ecrire le programme linéaire en nombres entiers complet. (0.5 pt)

7. Pour cette question, on suppose que le nombre $N_o = \sum_{k\in\mathcal{K}} \delta_{k,o} n_k$ de voitures ayant l'option o est tel que $N_o/n \leq p_o/q_o$. Déterminer alors la valeur optimale du programme linéaire obtenu par relaxation continue. La valeur de cette relaxation vous semble-t-elle adaptée à un branch-and-bound ? (1 pt)

3.2. **Cas particuliers – 4 pts.**

3.2.1. *Si $r = +\infty$ et $q_o = 2$ pour tout $o \in \mathcal{O}$.* On suppose pour les deux questions suivantes que $r = +\infty$ et $q_o = 2$ pour tout $o \in \mathcal{O}$.

8. Montrer que le problème peut se formuler comme le problème de la chaîne hamiltonienne de plus petit poids dans un graphe complet pondéré à n sommets. (1 pt)

On suppose maintenant de plus que toutes les voitures doivent être peintes de la même couleur ($|\mathcal{C}| = 1$).

9. Résoudre le cas suivant. Chaque classe k contient une seule voiture ($n_k = 1$) et l'on prendra $\gamma_o = p_o = 1$. (1 pt)

Classe	1	2	3	4	5	6	7
Option A	1	1	0	0	0	0	0
Option B	0	1	1	1	1	0	1
Option C	1	0	1	1	0	1	1
Option D	1	1	1	0	1	1	0
Option E	0	0	0	1	1	0	0
Option F	0	0	0	0	0	1	1

3.2.2. *Si $\mathcal{O} = \emptyset$.*

10*. On suppose uniquement pour cette question qu'il n'y a pas d'option sur les voitures. La chaîne de montage ne traite donc que des couleurs. On introduit pour toute couleur c la quantité $N_c = \sum_k n_k \delta_{k,c}$, qui est donc le nombre de voitures devant être peintes de la couleur c, et la quantité $m_c = \lceil N_c/r \rceil$.

Ecrire alors une condition nécessaire et suffisante qui assure l'existence d'une solution. Montrer de plus que s'il y a une solution réalisable, l'optimum vaut

$$\max\left(2m_{c^*} - 2, \sum_{c\in\mathcal{C}} m_c - 1\right),$$

où $c^* = \arg\max_{c \in \mathcal{C}} m_c$. (2 pts)

4. Voyageur de commerce et programmation dynamique – 6 pts

Considérons l'instance suivante : un graphe complet $K_n = (V, E)$ à n sommets ($n \geq 3$), des coûts positifs $c(vw)$ définis pour toute paire v, w de sommets, et un sommet particulier $s \in V$. On cherche la chaîne hamiltonienne de plus petit coût dont une des extrémités est s.

1. En prenant comme états les couples (X, v) tels que $v \in X \subseteq V \setminus \{s\}$, montrer que l'on peut écrire une équation de programmation dynamique permettant le calcul de la chaîne optimale. (2 pts)

Pour les questions suivantes, on peut se servir des identités indiquées à la fin de l'examen.

2. Quel est le nombre d'états possibles ? (1 pt)

On prend comme opération élémentaire l'addition.

3. Estimez le nombre d'additions que ferait un algorithme exploitant cette équation de programmation dynamique. Comparez ce nombre au nombre d'additions que ferait un algorithme qui énumérerait toutes les solutions. (1 pt)

4. Si votre ordinateur est capable de faire 1 million d'opérations élémentaires par seconde, pour chacune des valeurs suivantes de n, indiquez (par un calcul "à la louche") si vous serez capable de résoudre le problème avec cet algorithme en 1 seconde, 1 heure, 1 jour, 1 semaine, 1 mois, 1 an, 1 siècle (1 pt) :

$$n = 15 \quad n = 30 \quad n = 45.$$

5. Comment utiliser l'algorithme pour résoudre le problème du voyageur de commerce (cycle hamiltonien de plus petit coût) sur K_n ? (1 pt)

Quelques formules

$$\sum_{i=0}^{m} \binom{m}{i} = 2^m.$$

$$\sum_{i=0}^{m} \binom{m}{i} i = m 2^{m-1}.$$

$$\sum_{i=0}^{m} \binom{m}{i} i(i-1) = m(m-1)2^{m-2}.$$

$$m! \sim \sqrt{2\pi m} \left(\frac{m}{e}\right)^m.$$