

## Androïde - UE MAOA - Examen Janvier 2016.

Durée 2h00. Notes de cours et de TD autorisées.

### Exercice 1 (4 points)

**Rappel :** Soit un graphe non-orienté  $G = (V, E)$  avec  $V = \{1, \dots, n\}$ . Etant donné un sous-ensemble de points  $W \subset \{1, \dots, n\}$  avec  $W \neq \emptyset$  et  $W \neq \{1, \dots, n\}$ , on appelle *coupe* induite par  $W$  l'ensemble  $\delta(W)$  des arêtes ayant une extrémité dans  $W$  et une extrémité hors de  $W$ . On admet le théorème (dit de Menger) suivant :

**Theorem 1.** Soit deux points  $i$  et  $j$  pris dans  $\{1, \dots, n\}$ .

Alors il existe une chaîne reliant  $i$  et  $j$  dans  $G$  si et seulement si  $|\delta(W)| \geq 1$  pour tout sous-ensemble de sommets  $W \subset \{1, \dots, n\}$  contenant  $i$  et ne contenant pas  $j$ .

### Tracé d'une ligne de traway

On désire dessiner le tracé d'une **unique** ligne de tramway à travers une agglomération.

On considère  $n$  points dans une agglomération correspondant à des zones bien délimitées de l'agglomération (quartiers résidentiels, zones industrielles,...). Le calcul de la distance euclidienne  $d_{ij}$  entre deux points  $i, j \in \{1, \dots, n\}$  fournit une estimation du coût de construction d'une étape directe d'une ligne de tramway entre  $i$  et  $j$ . Les deux premiers points, 1 et 2 ont été décidés à l'avance parmi les  $n$  points pour être les *extrémités* de la ligne : ils sont en effet situés à l'opposé l'un de l'autre à la périphérie de l'agglomération.

On considère le graphe planaire complet  $K_n$  induit par les  $n$  points. La ligne que l'on désire tracer passera par certains des points qui seront alors appelés *stations*. La ligne de tramway est donc une chaîne dans le graphe  $K_n$ .

Une autre entrée du problème est la distances  $L$  (qui est donc fixée à l'avance). Un point  $i \in \{1, \dots, n\}$  de l'agglomération est dit *L-desservi* par la ligne de tramway si ce point est à une distance d'au plus  $L$  d'une station de la ligne (par exemple, il sera à distance nulle si ce point est une station).

Le *problème de tracé d'une ligne de tramway* (TLT) consiste à déterminer une chaîne de longueur minimale dans le graphe  $K_n$  reliant les points 1 et 2 de manière à ce que chacun des points soit L-desservi.

**Question 1** (0,5/4) — Donner et prouver la complexité du problème TLT.

On se propose de concevoir une formulation PLNE pour le problème TLT. On considère pour cela des variables binaires  $x_{ij}$  associées aux **paires non-orientées** de points  $(i, j)$  telles que  $x_{ij}$  vaut 1 si  $ij$  est une arête de la ligne et 0 sinon. Pour simplifier l'écriture, on suppose que  $x_{ii} = 0$  pour tout  $i \in V$ .

Afin de déterminer quelles sont les stations de la ligne, on ajoute les variables binaires  $y$  associées au **paires orientées** de points telles que :  $y_{ij}$  vaut 1 si le point  $i \in \{1, \dots, n\}$  est la station la plus proche du point  $j \in \{3, \dots, n\}$  et 0 sinon. On peut remarquer que  $y_{ii} = 1$  pour un point  $i \in \{3, \dots, n\}$  si et seulement si le point  $i$  est une station.

On considère à présent une valeur binaire  $(X, Y)$  pour les variables  $x$  et  $y$  qui vérifient les inégalités suivantes

$$\sum_{j=1}^n x_{ij} = 2y_{ii} \quad \forall i \in \{3, \dots, n\} \quad (1)$$

$$\sum_{j=1}^n x_{1j} = 1 \quad \text{et} \quad \sum_{j=1}^n x_{2j} = 1 \quad (2)$$

On note  $H(X)$  le graphe  $H$  composée des arêtes  $ij$  telles que  $X_{ij} = 1$ .

**Question 2** (0, 5/4) — Montrer que le graphe  $H(X)$  contient un chemin reliant 1 à 2.

**Question 3** (0, 5/4) — Décrivez une valeur  $X$  pour les variables  $x$  vérifiant les inégalités (1) et (2) qui correspond à un graphe  $H(X)$  non connexe.

**Question 4** (0, 5/4) — Montrer que les inégalité suivantes

$$\sum_{ij \in \delta(W)} x_{ij} \geq y_i \quad \forall W \subset \{1, \dots, n\} \text{ tel que } 1 \in W \text{ and } i \notin W, \quad \forall i \in \{3, \dots, n\} \quad (3)$$

impliquent que  $H(X)$  est une chaîne de 1 à 2. Expliquer comment ces inégalités peuvent être utilisées dans un PLNE.

**Question 5** (0, 5/4) — Donner une formulation PLNE pour le problème TLT. Prouver qu'elle est **équivalente** au problème TLT.

## Exercice 2 (6 points)

### Problème du sous-graphe induit sans $K_4$

On appelle  $K_p$  le graphe non-orienté complet à  $p$  sommets, c'est-à-dire un graphe où tous les sommets sont reliés à tous les sommets. Si  $K_p$  est un sous-graphe d'un graphe  $G$ , on dit que c'est une *clique* de  $G$ . Un graphe  $G$  est dit *sans  $K_4$*  si aucun sous-ensemble de 4 sommets de  $G$  ne forment un sous-graphe de  $G$  qui soit  $K_4$ .

Soit un graphe non-orienté  $G = (V, E)$  où  $V$  est l'ensemble des sommets et  $E$  l'ensemble des arêtes. Considérons  $W \subset V$  un sous-ensemble de sommets. On note  $E(W)$  l'ensemble des arêtes ayant leurs deux extrémités dans  $W$ . On dit alors que le graphe  $G[W] = (W, E(W))$  est le graphe induit par  $W$ .

Etant donné un poids  $c(v)$  associé à chaque sommet  $v$  de  $G$ , le *problème du sous-graphe induit sans  $K_4$*  (PSK4) consiste à déterminer un ensemble  $W$  induisant un sous-graphe sans  $K_4$  tel que  $\sum_{v \in W} c(v)$  soit maximum.

**Question 1** (1/6) — *Formulation*

a) Montrer que le problème PSK4 est équivalent au programme en nombres entiers ( $P$ ) suivant

$$(P) \left\{ \begin{array}{ll} \text{Max} & \sum_{u \in V} c(u)x(u) \\ & \sum_{u \in S} x(u) \leq 3 \quad \text{pour tout } S \subset V \text{ tel que } G[S] = K_4, & (4) \\ & 0 \leq x(u) \leq 1 \quad \forall u \in V & (5) \\ & x(u) \text{ entier} \quad \forall u \in V. & (6) \end{array} \right.$$

On appelle *contraintes de clique* les contraintes de type (4) et triviales celle de type (5).

b) Quels sont les caractéristiques de ce programme en terme de nombres de contraintes et de variables? Quelles sont les possibilités pour résoudre un tel programme?

**Question 2 (3/6)** — *Etude polyédrale générale*

Soit un graphe  $G = (V, E)$  avec  $n = |V|$ . On note  $P_4(G)$  le polytope des sous-graphes induits sans  $K_4$  de  $G$ , i.e.

$$P_4(G) = \text{conv}\{\chi^W \in \{0, 1\}^n \mid (W, E(W)) \text{ est un sous-graphe sans } K_4 \text{ de } G\}.$$

- a) Montrer que  $P_4(G)$  est de pleine dimension.
- b) Montrer que les contraintes triviales (5)  $x(u) \geq 0$  définissent des facettes de  $P_4(G)$  pour tout  $u \in V$ .
- c) Montrer que les contraintes triviales (5)  $x(u) \leq 1$  définissent des facettes de  $P_4(G)$  pour tout  $u \in V$ .

**Question 3 (2/6)** — *Etude faciale des contraintes de clique*

Soit  $S \subset V$  induisant une clique à  $p$  sommets (c'est-à-dire le graphe  $K_p$ ) avec  $p \geq 4$ . On dit que  $S$  induit une clique *maximale* au sens de l'inclusion s'il n'existe pas  $S'$  induisant une clique sur  $p'$  sommets tel que  $S \subset S'$  et  $p < p'$ .

- a) Montrer que l'inégalité suivante

$$\sum_{u \in S} x(u) \leq 3 \text{ pour tout } S \subset V \text{ tel que } G[S] = K_p \text{ avec } p \geq 4 \tag{7}$$

est valide pour  $P_4(G)$ . Donner un cas où l'inégalité (4) ne définit pas de facette.

- b) Soit  $S \subset V$  induisant une clique à  $p$  sommets qui est maximale au sens de l'inclusion. Montrer que, pour tout sommet  $w \in V \setminus S$ , il existe un sommet  $u_w \in S$  tel que  $wu_w \notin E$ .
- c) Soit  $u_1, u_2, u_3, u_4$  quatre sommets distincts de  $S$ . Construire  $n$  ensembles de sommets  $B_u, u \in V$ , qui appartiennent à la face définie par la contrainte (7).
- d) Rappeler les hypothèses nécessaires pour montrer que la contrainte (7) induite par  $S$  définit une facette de  $P_4(G)$ .

**Exercice 3 (5 points)**

**Ordonnement Juste à Temps**

On considère le problème noté  $P_m \mid - \mid \sum_{j=1}^n |C_j - d|$ , où  $n$  tâches indépendantes  $J_1, \dots, J_n$  doivent être exécutées sur  $m$  machines identiques de telle sorte que la déviation totale  $\sum_{j=1}^n |C_j - d|$  soit minimale avec  $d$  date d'échéance commune (non restrictive).  $C_j$  est la date de fin d'exécution de la tâche  $J_j$  et  $p_j$  sa durée opératoire pour tout  $j \in \{1, \dots, n\}$ .

**Question 1 (0.5/5)** — Les séquences optimales des  $m$  machines contiennent-elles des temps morts? Justifier.

**Question 2 (0.5/5)** — L'ordonnement optimal sur chacune des machines est-il en V (les tâches  $J_j$  avec  $C_j \leq d$  sont classées selon la règle LPT et les tâches  $J_j$  avec  $C_j > d$  sont classées selon la règle SPT)? Justifier.

**Question 3 (1/5)** — Montrer que sur chaque machine, une tâche se termine exactement à la date  $d$ .

**Question 4** (0.5/5) — Définir le nombre de tâches devant s'exécuter sur chaque machine.

**Question 5** (0.5/5) — Si  $n_i$  définit le nombre de tâches qui s'exécutent sur la machine  $M_i$ , définir l'indice de la tâche se terminant à la date  $d$ .

**Question 6** (1/5) — En utilisant les propriétés précédentes et un algorithme qui permet de calculer un ordonnancement optimal pour le même problème dans le cas où l'on ne dispose que d'une seule machine, décrire et justifier un algorithme pour résoudre ce problème. Préciser sa complexité.

**Question 7** (1/5) — Appliquer votre algorithme pour calculer la solution optimale de l'instance suivante :  $n = 14$ ,  $p = (37, 35, 32, 31, 27, 23, 21, 19, 16, 12, 11, 7, 5, 3)$ ,  $d = 100$  et  $m = 3$ .

#### Exercice 4 (5 points)

#### Lot-sizing

On considère le problème de lot-sizing LS à une seule référence sur un horizon discret de  $T$  périodes. Les paramètres  $d_t$ ,  $f_t$ ,  $p_t$  et  $h_t$  représentent respectivement la demande, le coût de setup induit lorsqu'une production a lieu à la période  $t$ , le coût unitaire de production et le coût de stockage à la période  $t$  pour  $t = 1, \dots, T$ . On considère en outre, que la quantité produite à chaque période ne peut excéder un niveau de capacité qui est défini par la somme de la capacité disponible à la période  $t$ , noté  $C_t$ , et la capacité non utilisée de la période précédente.

On note  $x_t$  la quantité produite à la période  $t$ ,  $y_t$  la variable binaire de setup associée à  $x_t$  et  $s_t$  la valeur du stock à la fin de la période  $t$ . Le paramètre  $C_{1t} = \sum_{i=1}^t C_i$  définit la capacité cumulative à la période  $t$ . On supposera que le stock initial et le stock final sont nuls et que les demandes sont positives.

On s'intéresse à la résolution du problème de lot-sizing avec contraintes de capacité cumulative LS-CC.

**Question 1** (1.5/5) — Proposer une formulation linéaire en variables mixtes du problème LS-CC en utilisant les capacités cumulatives pour exprimer les contraintes de capacité.

**Question 2** (1/5) — Proposer une formulation du problème sous forme de réseau à coût fixe en précisant le coût et la capacité de chaque arc.

Indication : On utilisera deux types de nœuds à chaque période ; des nœuds dits *de demande* et des nœuds dits *de transbordement*. A chaque période  $t$ , le flux associé à l'unique arc sortant d'un nœud de demande (respectivement de transbordement) représente le stock  $s_t$  (respectivement la production  $x_t$ ).

**Question 3** (1/5) — On définit un sous-plan  $(t, t')$  avec  $0 \leq t < t' \leq T$ , comme un ensemble de périodes consécutives entre  $t + 1$  et  $t'$  tel que :

- $s_t = 0$ ,
- $s_i > 0$ ,  $i = t + 1, \dots, t' - 1$ ,
- $s_{t'} = 0$ .

Si l'on relâche les contraintes de capacité cumulative, définir et justifier la structure d'une solution optimale en utilisant la notion de sous-plan.

**Question 4** (1.5/5) — On admet que la structure des coûts du problème LS-CC nous permet de calculer le coût d'un sous-plan indépendamment des autres sous-plans. On note  $g(t, t')$  le coût minimum du sous-plan  $(t, t')$  et  $F(t)$  le coût total minimal pour satisfaire les demandes des périodes  $t + 1$  à  $T$ .

Proposer un algorithme de programmation dynamique pour résoudre le problème LS-CC en précisant les conditions initiales et la valeur de la solution optimale. Préciser sa complexité.