Corrigé du partiel du cours d'approximation

EXERCICE 1
Question 1 Par l'absurde : soit A une $\alpha(n)$ approximation pour le k-centre non métrique, on essaye de résoudre le problème du dominant minimum.
On transforme (G,k), une instance du dominant minimum à n sommets, en G' instance du k-centre à n sommets en ajoutant toutes les arêtes pour obtenir une clique et en affectant un poids 1 aux arêtes qui étaient déjà dans G, et $\alpha(n)+1$ aux autres.
Si G admet un dominant de taille $\leq k$, alors G' admet un k-centre optimal de coût 1 et $A(G') \leq \alpha(n)$.
Si G n'admet pas de dominant de taille $\leq k$, alors le coût optimal d'un k-centre de G' vaut $\alpha(n)+1$ et $\alpha(n) < A(G')$.
Le résultat de $A(G')$ permet donc de savoir s'il existe un dominant de taille $\leq k$ dans G et donc de résoudre polynomialement ce problème, ce qui est absurde si P$\neq$ NP.

EXERCICE 2
Question 2 Soit A l'algorithme suivant : orienter chaque arête $\{u,v\} \in E$, *indépendamment*, avec probabilité 1/2 de u vers v, et avec probabilité 1/2 de v vers u.
À tout triangle (non-orienté) c du graphe, on associe $W(c)$ la variable aléatoire qui vaut 1 si c est orienté en circuit dans le graphe résultant (ce qui arrive avec probabilité 1/4 car l'orientation de chaque arête est choisie dans un sens ou dans l'autre indépendamment avec probabilité 1/2) et 0 sinon.
Par linéarité de l'espérance : $\mathbb{E}(A(G)) = \sum_{c \in G} \mathbb{E}(W(c)) = \sum_{c \in G} 1/4 \geq 1/4 \cdot \text{OPT}$ car OPT est majoré par le nombre de triangles (non-orientés) de G.

Question 3 Famille $\{I_n\}_{n \in \mathbb{N}}$ d'instances critiques : I_n= n triangles disjoints (ou partageant un même sommet, formant ainsi un moulin). $\text{OPT}(I_n) = n$, $\mathbb{E}(A(I_n)) = n/4$.

Question 4 On déterminise le choix de l'orientation d'une arête par la méthode de l'espérance conditionnelle : en maximisant l'espérance du nombre de circuits de longueur 3 si on orientait aléatoirement les arêtes qui n'ont pas encore été orientées. Les seuls triangles affectés par l'orientation d'une arête uv sont les triangles (uvw), encrés sur cette arête, et leur nombre est *polynomial* ($\leq n-2$). Il s'agit donc de choisir l'orientation $(u \rightarrow v)$ ou $(v \rightarrow u)$ qui maximise l'espérance du nombre de ces triangles orientés correctement en circuit. Si les arêtes uw et wv sont déjà orientées, en sens inverse, alors (uvw) ne sera pas un circuit quelque soit le choix de l'orientation de uv ; si ces deux arêtes sont orientées dans le même sens, alors si oriente uv dans le même sens, le triangle sera un circuit, sinon non. Si le triangle (uvw) a une seule arête déjà orientée, alors si uv est orientée dans le même sens, la probabilité que ce triangle soit correctement orienté sera 1/2, sinon 0. Si le triangle (uvw) n'a aucune arête déjà orientée, alors quelque soit le choix de l'orientation pour uv, la probabilité que le triangle soit correctement orienté sera 1/4. Nous pouvons donc calculer en temps polynomial ($O(n)$) pour chaque orientation de uv, l'espérance du nombre de circuits de longueur 3 et sélectionner l'orientation qui maximise cette espérance conditionnelle. Cette méthode calcule une solution de valeur supérieure à l'espérance de l'algorithme randomisé et est donc une 1/4-approximation en temps $O(n^2)$.

EXERCICE 3 (Couverture par sommets de poids minimum)
Question 5 □

Question 6 Soit u un sommet de V. L'égalité est vraie à l'initialisation.
Si aucune arête uv n'est traitée en phase 2, alors l'égalité est vraie car les variables ne sont pas modifiées. Si uv est choisie en phase 2, seules les variables $t(u)$ et $\text{prix}(uv)$ sont modifiées dans l'égalité : comme $t(u)$ diminue de m et $\text{prix}(uv)$ diminue de m, l'égalité reste vraie après cette phase.

Question 7 Soit C une couverture. $\text{poids}(C) = \sum_{u \in C} \text{poids}(u) \geq \sum_{u \in C} \sum_{v : uv \in E} \text{prix}(uv)$, car $t(u) \geq 0$ pour tout u. Or, C est une couverture, donc $\cup_{u \in C} \{uv : uv \in E\} = E$, ainsi : $\text{poids}(C) \geq \sum_{e \in E} \text{prix}(e)$. Ainsi, en prenant la couverture optimale : $\text{OPT} \geq \sum_{e \in E} \text{prix}(e)$.

Question 8 Soit C l'ensemble construit par l'algorithme. C est bien une couverture car l'algorithme s'arrête que lorsque toute les arêtes sont couvertes et, à chaque étape de l'itération, au moins une nouvelle arête est couverte. De plus, $\text{poids}(C) = \sum_{u \in C} \text{poids}(u) = \sum_{u \in C} \sum_{v : uv \in E} \text{prix}(uv)$, car pour tout u sélectionné $t(u) = 0$. Comme chaque arête uv apparaît au plus 2 fois dans cette somme : $\text{poids}(C) \leq 2 \sum_{e \in E} \text{prix}(e) \leq 2\,\text{OPT}$.

Question 9 Famille $\{I_n\}_{n \in \mathbb{N}}$ d'instances critiques : $I_n = n$ arêtes disjointes dont chaque extrémité a un poids 1. $\text{OPT}(I_n) = n$, $A(I_n) = 2n$.

EXERCICE 4 (Débordement minimum)
Question 10 On réduit le problème Partition au problème de décision associé au problème du débordement minimum : pour tout i, on pose $t_i = c_i = x_i$ et $W = \sum_i x_i/2$, et on pose la question "existe-t-il un débordement de coût inférieur à W ?". La réduction est bien *polynomiale*. Et, s'il existe une solution pour partition, l'ensemble S obtenu donne une solution pour débordement minimum, et inversement.

Question 11 Considérons l'instance suivante avec trois objets : $(t_1, c_1) = (W - \epsilon^2, 1)$, $(t_2, c_2) = (W - \epsilon^2, (W - \epsilon^2)/\epsilon)$ et $(t_3, c_3) = (\epsilon^2, 2\epsilon)$. L'algorithme trie les ensembles dans cet ordre et sélectionne les deux premiers pour un coût : $1 + (W - \epsilon^2)/\epsilon$, alors que $\text{OPT} = 1 + 2\epsilon$. Ainsi il peut-être arbitrairement mauvais.

Question 12 On procède par programmation dynamique. On construit le tableau $t \in \{1, \ldots, n\} \times \{\min_i c_i, \ldots, n \cdot C\}$ défini de la façon suivante :
$A[i, c] =$ volume maximal d'une combinaison d'éléments parmi $\{1, \ldots, i\}$ de coût c.
On remplit le tableau grâce à la relation de récurrence suivante :
$A[i, c] = \max(t[i-1, c], A[i-1, c - c_i] + t_i)$ pour $2 \geq i \leq n$ avec $A[1, c] = t_1$ si l'objet 1 est de coût c, 0 sinon, et $A[i, k < 0] = 0$.

On cherche ensuite $\min_c\{A[n,c] | A[n,c] \geq W\}$, on peut reconstituer l'ensemble correspondant en remontant dans le tableau. La complexité de cet algorithme est en $(n^2 \cdot C)$, ce qui est polynomial en C, donc pseudo-polynomial.

Question 13 Lorsque $C \leq OPT$, on pose $K = \frac{\varepsilon C}{n}$ et pour chaque objet i on arrondit son coût : $c'_i = \lceil \frac{c_i}{K} \rceil$.
Soit O un ensemble de coût minimum, pour tout objet i, $K \cdot c'_i$ est supérieur ou égal à c_i, mais pas de plus que K : $K \cdot \text{coût}(O') - \text{coût}(O) \leq nK$. On peut appliquer l'algorithme pseudo-polynomial sur l'instance arrondie en temps polynomial car $O(n^2C) = O(n^2\lceil \frac{n}{\varepsilon} \rceil)$.
Cet algorithme retourne un ensemble S' de coût inférieur à K fois celui de O avec les valeurs arrondies : $\text{coût}(S') \leq K \cdot \text{coût}(O') \leq nK + \text{coût}(O) = \varepsilon \cdot C + \text{OPT} \leq (1+\varepsilon)\,\text{OPT}$.
Pour traiter le cas général, on utilise la méthode de l'élagage paramétré. Pour k de 1 à $n \cdot C$, on suppose $OPT \leq k$, on restreint alors l'instance aux objets de coût inférieur à k sur laquelle on peut appliquer le FPTAS décrit ci-dessus.
Algorithme :

1. Contruire $I_1, \dots I_{n \cdot C}$ (les instances constituées des objets de coût inférieur à $1 \leq i \leq n \cdot C$).
2. Calculer un ensemble S_i de coût inférieur à $(1+\varepsilon)\,\text{OPT}_i$ pour chaque I_i à l'aide du FPTAS.
3. Calculer le plus petit indice i tel que le volume de S_i soit supérieur à W, noté j.
4. Retourner S_j.

EXERCICE 5 (Multi-couverture par ensembles)
Question 14 L'algorithme glouton est assez similaire à celui de la couvertur par ensemble : on introduit le coût efficace d'un ensemble S : $\frac{\text{coût}(S)}{\#}$, # étant le nombre d'éléments non terminés (i.e. l'élément i n'a pas encore été couvert r_i fois) qu'il couvre. Tant que tous les éléments ne sont pas couverts, on choisit l'ensemble de coût efficace minimum.
On considère les ensembles de la multi-couverture optimale. On ordonne les éléments à l'intérieur des ensembles de OPT par le nombre de fois qu'ils ont été couverts dans l'algorithme glouton. On crée ensuite une bijection avec l'univers en associant, à un élément $i \in S \subseteq \text{OPT}$, un évenement $E_{i,j}$="i est couvert pour la j fois dans le glouton".
Le prix de la $j$ème couverture de i dans le glouton est alors inférieur à $\frac{\text{coût}(S)}{|S|-k}$, où k est le nombre d'évenements placés avant $E_{i,j}$ dans S.
$\text{coût(glouton)} \leq \sum_{S \subseteq \text{OPT}} \sum_{k \leq |S|} \frac{\text{coût}(S)}{|S|-k}$
$\leq H_m \cdot \text{OPT}$.

Question 15 On peut reprendre la même instance critique que celle de la couverture par ensemble puisque c'est un sous problème et que le facteur d'approximation est le même. Sinon, on peut imaginer une belle instance pour ce problème !
