

Correction détaillée de l'exercice sur le problème Triangle Hitting

Q1)

C'est une OUI instance car $S = \{3\}$ intersecte tous les triangles.

Q2)

Code : cf fichier java (algorithme THdec). On remarque que l'algorithme est bien un algorithme de branchement car soit il effectue un cas de base, soit il retourne $\text{THdec}(g1,k-1) \parallel \text{THdec}(g2,k-1) \parallel \text{THdec}(g3,k-1)$. D'après le théorème du cours, on branche donc avec $\Delta = 3$ fils, sur une profondeur de k , et tous les calculs faits à chaque appel sont polynomiaux (en $\mathcal{O}(n^3)$ à cause de "trouveTriangle"). La complexité de THdec est donc en $\mathcal{O}(3^k n^3)$.

Q3)

Pour prouver qu'une règle n'est pas correcte, il faut trouver un couple (G, k) sur lequel la règle produira une instance (G', k') non-équivalente.

R1) pas correcte : sens (G, k) OUI-instance $\Rightarrow (G', k')$ incorrect avec le contre exemple suivant. Soit G à 6 sommets $\{0, 1, 2, 3, 4, 5\}$ avec les 2 triangles $\{0, 1, 2\}$ et $\{3, 4, 5\}$. Si l'on applique R1 sur G avec $\Delta = \{0, 1, 2\}$ et $u = 0$, on se trouve avec

- $(G, 2)$ est une OUI-instance
- $(G - \{v, w\}, 0)$ est une NON-instance, car $G - \{v, w\} = G - \{1, 2\}$, or $G - \{1, 2\}$ contient encore le triangle $\{3, 4, 5\}$.

R2) pas correcte : sens (G, k) OUI-instance $\Rightarrow (G', k')$ incorrect avec le contre exemple suivant. Soit G à 5 sommets $\{0, 1, 2, 3, 4\}$ avec les 2 triangles $\{0, 1, 2\}$ et $\{2, 3, 4\}$. Si l'on applique R2 sur G avec $\Delta = \{0, 1, 2\}$, $u = 0$ et $v = 1$, on se trouve avec

- $(G, 1)$ est une OUI-instance (car 2 est une solution)
- $(G - \{v\}, 0)$ est une NON-instance, car $G - \{v\} = G - \{1\}$, or $G - \{1\}$ contient encore le triangle $\{2, 3, 4\}$.

R3) est correcte, écrivons donc la preuve de sa correction.

Preuve sens 1 : (G, k) OUI-instance $\Rightarrow (G', k')$ OUI-instance : version moins formelle (acceptée aussi)

Supposons que (G, k) soit une OUI-instance, et soit S une solution de G avec $|S| \leq k$. Observons d'abord qu'il existe une solution S' de taille au plus k contenant w , car on peut partir de S , et restructurer S en remplaçant dans S tous les sommets de Δ par w . Informellement, on a toujours "intérêt à prendre w " (comme le fait R3), étant donné que w est le "point d'attachement" de Δ au reste du graphe.

Montrons à présent que (G', k') est une OUI-instance, c'est à dire que $(G - \{w\}, k - 1)$ est une OUI-instance. Pour cela, on observe que $S' - \{w\}$ est bien une solution de $G - \{w\}$ de taille $k - 1$. La taille est évidente, et le fait que c'est une solution vient du fait que S était bien une solution, et touchait donc en particulier tous les triangles Δ' ne contenant pas w . Ces triangles Δ' sont toujours touchés par $S' - \{w\}$.

Preuve sens 1 : (G, k) OUI-instance $\Rightarrow (G', k')$ OUI-instance : version formelle

Supposons que (G, k) soit une OUI-instance, et soit S une solution de G avec $|S| \leq k$.

On va d'abord montrer la propriété P_1 : "il existe une solution S' de taille au plus k contenant w " (informellement, on a toujours "intérêt à prendre w ", comme le fait R3). Soit $X = S \cap \Delta$. Soit S' obtenue en remplaçant X par $\{w\}$ dans S (plus formellement, $S' = (S \setminus X) \cup \{w\}$). Notez que :

- Comme S doit toucher tous les triangles, on doit avoir en particulier que S doit intersecter Δ , et donc que $X \neq \emptyset$. Ainsi, $|S'| \leq |S|$, et donc $|S'| \leq k$.
- S' est toujours une solution (c'est à dire intersecte bien tous les triangles), car les seuls sommets potentiellement supprimés de S sont u et v , et le seul triangle dangereux utilisant u et v est Δ , qui est bien intersecté par S' (car S' contient w).

On a donc prouvé la propriété P_1 .

Montrons à présent que (G', k') est une OUI-instance, c'est à dire que $(G - \{w\}, k - 1)$ est une OUI-instance. Pour cela, prouvons que $S' - \{w\}$ est bien une solution de $G - \{w\}$ de taille $k - 1$. Tout d'abord $S' - \{w\}$ est bien de taille $k - 1$, puisque $|S'| \leq k$ et $w \in S'$.

Ensuite, prouvons que c'est bien une solution. Soit Δ' un triangle de $G - \{w\}$, et montrons que $(S' - \{w\}) \cap \Delta' \neq \emptyset$. Comme S était une solution, et que Δ' est aussi un triangle de G , on obtient que $S \cap \Delta' \neq \emptyset$. De plus, comme Δ' est un triangle de $G - \{w\}$, et comme $d(u) = d(v) = 2$, u et v ne font plus partie d'un triangle dans $G - \{w\}$, et donc Δ' ne contient ni u , ni v , ni w . Autrement dit, $\Delta' \cap \Delta = \emptyset$. Donc, si l'on enlève Δ à S , on intersecte encore Δ' , c'est à dire $(S \setminus \{\Delta\}) \cap \Delta' \neq \emptyset$. Or, par définition de S' , $S \setminus \{\Delta\} = S' \setminus \{w\}$, et on obtient donc $(S' \setminus \{w\}) \cap \Delta' \neq \emptyset$.

Preuve sens 2 : (G, k) OUI-instance $\Leftrightarrow (G', k')$ OUI-instance. Supposons que (G', k') est une OUI-instance, c'est à dire que $(G' - \{w\}, k' - 1)$ est une OUI-instance. Il existe donc un ensemble S' intersectant tous les triangles de $G' - \{w\}$, avec $|S'| \leq k' - 1$. Soit $S = S' \cup \{w\}$. On observe que

- $|S| \leq |S'| + 1 \leq k$
- S intersecte bien tous les triangles de G , car tous les triangles en plus dans G (par rapport aux triangles de $G' - \{w\}$) contiennent w , et sont donc bien intersectés par S .

Q4)

On peut définir un algorithme MTP(G) (pour Maximal Triangle Packing) qui a pour spécification de retourner la taille d'un ensemble (maximal au sens de l'inclusion) de triangles disjoints, et dont le pseudo code est :

- $res = \emptyset, G_c = G$
- tant qu'il existe un triangle Δ dans G_c , faire
 - $res = res \cup \Delta$
 - $G_c = G_c \setminus \Delta$
- retourner $|res|$

La complexité est $\mathcal{O}(nt)$ où t est le temps de détection d'un triangle (en $\mathcal{O}(n^3)$ par exemple).

Q5)

On fait définir KHdec en faisant comme THdec, mais en remplaçant :

- "trouveTriangle(g)" par "trouveClique(g, t)"
- le branchement qu'on avait sur les 3 fils par un branchement sur t fils : étant donné une t -clique K , on retourne $KHdec(g_0, k-1) || \dots || KHdec(g_{t-1}, k-1)$, où $g_i = g \setminus K.get(i)$ est le graphe obtenu en enlevant le i ème sommet de K à g .

D'après le théorème du cours, on branche donc avec $\Delta = t$ fils, sur une profondeur de k , et tous les calculs faits à chaque appel sont polynomiaux (en $\mathcal{O}(n^t)$ à cause de "trouveClique" et de la boucle de taille t pour lancer les t appels récurifs). La complexité de THdec est donc en $\mathcal{O}(t^k n^t)$. On retrouve d'ailleurs bien la complexité pour THdec quand $t = 3$.