TD6 – FPT

Exercice 1 – Édition vers des cliques

On dit qu'un graphe G est un *graphe de cluster* si G est une union disjointe de cliques, c'est-à-dire qu'on peut partitionner les sommets V de G en r ensembles $V_1, \ldots, V_r$ tels que pour tout i, $G[V_i]$ induit une clique et il n'existe aucune arête entre V_i et V_j pour $i \neq j$. Dans le problème d'édition vers des cliques, on cherche, étant donné un entier k et un graphe G, à ajouter ou supprimer au plus k arêtes pour transformer G en graphe de cluster.

1. Soit P un chemin induit à trois sommets v_1, v_2, v_3 dans G. Montrer qu'on doit supprimer une des arêtes v_1v_2, v_2v_3 ou ajouter l'arête v_1v_3.
2. En déduire un algorithme FPT de complexité $3^k \cdot Poly(n)$ pour résoudre le problème.
3. Montrer que si une arête uv apparaît dans $k+1$ copies de P_3, on doit la supprimer. Autrement dit, (G, k) est une instance positive si et seulement si $(G - uv, k-1)$ est une instance positive.
4. Montrer que si une non-arête uv apparaît dans $k+1$ copies P_3, on doit l'ajouter.
5. Montrer que si un sommet x n'apparaît dans aucun P_3 alors la suppression ou l'addition d'une arête dans un P_3 ne crée pas de P_3 contenant x.
6. Montrer que le problème admet un noyau à $O(k^2)$ sommets.

Exercice 2 – Directed Feedback Vertex Set dans les Tournois

Un graphe orienté est un graphe $G = (V, E)$ où les arêtes uv sont orientées de u vers v. On appelle u l'origine de l'arête et v la destination de uv

Un tournoi est un graphe orienté tel que pour toute paire de sommets u, v, uv ou vu est une arête. Un triangle est un cycle orienté de longueur 3. Dans le problème Directed Feedback Vertex Set dans les Tournois (DFVST), on se donne un tournoi T et un entier k et on désire déterminer si la suppression de k sommets laisse un graphe sans cycle orienté.

1. Montrer que dans un tournoi, tout cycle orienté de longueur au moins 4 contient un triangle orienté comme sous graphe.
2. En déduire un algorithme FPT pour DFVST paramétré par k. Quelle est sa complexité ?

On va essayer d'améliorer la complexité de cet algorithme en utilisant une méthode appelée *compression itérative*. On définit le problème Feedback Vertex Set dans les Tournois Quasi-Transitifs (FVSTQT) comme le problème où on se donne un tournoi T, un ensemble de sommets X tel que $T[V \setminus X]$ est acyclique et un entier k et on veut savoir s'il existe k sommets de $V \setminus X$ dont la suppression laisse un tournoi acyclique. (Autrement dit, on veut trouver un feedback vertex set de T qui est disjoint de X).

Soit T un tournoi et $u_1, \ldots, u_n$ un ordre arbitraire des sommets de T. Pour tout i, on note $V_i = \{u_1, \ldots, u_i\}$. On considère les graphes $T_1, \ldots, T_n$ où T_i est la restriction de T aux sommets $u_1, \ldots, u_i$. Soit i un entier. Dans la suite **on suppose qu'on nous donne un ensemble** X_i de sommets de taille au plus k tel que $T_i[V \setminus X_i]$ est acyclique.

3. Combien y a-t-il de sous-ensembles de X_i ?
4. On considère le tournoi T_{i+1}, c'est à dire le tournoi T_i où u_{i+1} a été ajouté. Montrer que $T_{i+1}[V_{i+1} \setminus (X_i \cup \{u_{i+1}\})]$ est transitif.
5. Montrer que si (T_{i+1}, k) est une instance positive de DFVST alors il existe $Y \subseteq X_i \cup \{u_{i+1}\}$ de taille au plus k tel que $(T_i[V_{i+1} \setminus Y], k - |Y|)$ est une instance positive de FVSTQT.
6. En déduire que si FVSTQT peut être résolu en temps polynomial n^c, alors on peut résoudre DFVST en temps $2^{k+1}n^{c+1}$.

La fin de l'exercice consiste à montrer que FVSTQT peut être décidé en temps polynomial. Soit T un tournoi et X un ensemble de sommets tels que $T[V \setminus X]$ est acyclique.

7. Montrer que l'on peut numéroter les sommets $w_1, \ldots, w_r$ de $T[V \setminus X]$ tel que, pour tout $i < j$, l'arc w_iw_j est orienté de w_i vers w_j.

8. Soit $t = abc$ un triangle de T. Montrer que si tous les sommets de t sont dans X alors l'instance est négative.
9. Montrer que si X contient a, b mais pas c, alors (T, X, k) est positive si et seulement si $(T - c, X, k - 1)$ est positive.
10. On suppose maintenant que X contient seulement le sommet a de t. Comme abc est un triangle orienté, on a $b = w_i$ et $c = w_j$ avec $i < j$. Montrer que toute solution de (T, X, k) contient tous les sommets de l'ensemble $\{w_i, w_{i+1}, \dots, w_j\}$ sauf au plus un.
11. En déduire un algorithme glouton qui résout FVSTQT en temps polynomial.

Exercice 3 – Color coding

On s'intéresse ici au problème de recherche d'un chemin de longueur (au moins) k dans un graphe G.

1. Quelle est la complexité de l'algorithme naïf permettant de résoudre ce problème ?

On cherche maintenant à améliorer cet algorithme pour qu'il soit FPT. On attribue indépendamment à chaque sommet de G une couleur aléatoire choisie uniformément entre 1 et k. Un chemin dans G est *bariolé* s'il est coloré $1, \dots, i$ où i est sa longueur.

2. Si G contient un chemin de taille k, quelle est la probabilité que ce chemin soit bariolé ?
3. On définit $S_{i,v}$ = vrai si le sommet v est l'extrémité d'un chemin bariolé de longueur i. Écrire une relation de récurrence reliant les valeurs de $S_{i,v}$.
4. En déduire un algorithme de programmation dynamique permettant de tester s'il existe un chemin bariolé dans G.
5. En déduire un algorithme FPT pour le problème initial, qui renvoie la bonne réponse avec probabilité au moins 1/2.
6. ($\star$) Améliorer l'algorithme pour obtenir une complexité $O(b^k \cdot n^c)$, où b et c sont deux constantes (indépendantes de n et k).