Formulations pour le problème de plus grand graphe partiel commun

Etienne de Gastines[1], Arnaud Knippel[1]

INSA Rouen Normandie, Laboratoire de Mathématiques, Rouen, France

{etienne.mace_de_gastines,arnaud.knippel}@insa-rouen.fr

Mots-clés : *plus grand graphe partiel commun, polyèdres, programmation linéaire 0-1*

1 Introduction

Le problème du plus grand graphe partiel commun (Maximum Common Edge Subgraph - MCES) a été introduit en premier par Bokhari dans [4], pour résoudre le problème d'affectation de tâches à des processeurs tout en maximisant les demandes de communications entre tâches. Ce problème donne également une mesure de similarité entre deux graphes et est utilisé comme outil pour comparer des molécules.

Il peut être énoncé comme suit :

Soit $G = (V_G, E_G)$ et $H = (V_H, E_H)$ deux graphes. $S = (V_S, E_S)$ est un graphe partiel de G si $V_S \subseteq V_G$ et $E_S \subseteq V_G$. Si un graphe partiel de G est isomorphe à un graphe partiel de H, alors on le qualifie de graphe partiel commun à G et H. On recherche un graphe partiel commun à G et H maximal en terme de nombre d'arêtes. Nous considérons ici les graphes G et H comme étant simples et non-orientés. Le voisinage d'un sommet u est noté $N(u)$

Le problème est $\mathcal{NP}$-difficile et généralise les problèmes d'isomorphisme de sous-graphe, de clique maximale et de chemins hamiltoniens.

La première formulation en nombres entiers a été proposée par Marenco et Loiseau ([6], [7]). Elle introduit les variables $x_{u,v}$ $\forall u \in V_G, v \in V_H$ valant 1 si le sommet u est associé au sommet v, 0 sinon, et les variables y_{u_1,u_2} $\forall u_1u_2 \in E_G$ valant 1 si l'arête u_1u_2 appartient au sous graphe commun, 0 sinon. Une deuxième formulation a été proposée par Bahiense, Manić, Piva, et de Souza ([2], [3]), qui utilise les mêmes variables x, mais introduit pour sa part les variables $c_{u_1u_2,v_1v_2}$ $\forall u_1u_2 \in E_G, v_1v_2 \in E_H$, valant 1 si l'arête u_1u_2 est associée à l'arête v_1v_2.

2 Nouvelles formulations

La nouvelle formulation que nous proposons effectue un choix encore différent des précédentes, nous introduisons des variables $z_{u,v}$ $\forall u \in V_G, v \in V_H$ valant 1 si le sommet u est associé à un voisin du sommet v dans le graphe partiel commun, et réciproquement. Cela nous permet de proposer la formulation suivante (formulation réduite) :

$$\max \quad 1/2 \sum_{u \in V_G, v \in V_H} z_{u,v} \tag{1}$$

$$\text{s.t} \quad \sum_{v \in V_H} x_{u,v} = 1, \quad \forall u \in V_G \tag{2}$$

$$\sum_{u \in V_G} x_{u,v} = 1, \quad \forall v \in V_H \tag{3}$$

$$z_{u_1,v_2} \leq \sum_{v_1 \in N(v_2)} x_{u_1,v_1}, \quad \forall u_1 \in V_G, v_2 \in V_H \tag{4}$$

$$z_{u_1,v_2} \leq \sum_{u_2 \in N(u_1)} x_{u_2,v_2}, \quad \forall u_1 \in V_G, v_2 \in V_H \tag{5}$$

$$x_{u,v} \in \{0,1\}, \quad \forall u \in V_G, v \in V_H \tag{6}$$

$$z_{u,v} \in \{0,1\}, \quad \forall u \in V_G, v \in V_H. \tag{7}$$

Nous proposons aussi une version symétrisée de la formulation proposée par [2], qui donne une meilleure relaxation.

Nous avons testé numériquement les nouvelles formulations obtenues avec les deux anciennes formulations sur deux bases de données, une base de données de graphes issus de problèmes rassemblés par Marenco et Loiseau ([7]) et provenant essentiellement d'applications pratiques, et une partie de la base de donnés ARG ([5], [1]) composées de graphes générés aléatoirement selon divers modèles. Nous observons que la formulation réduite la plus compétitive des formulations.

Références

[1] ARG Database, https ://mivia.unisa.it/datasets/graph-database/arg-database/

[2] Bahiense, L. and Piva, B. and de Souza, C., *A branch&cut algorithm for the maximum common edge subgraph problem*, Electronic Notes in Discrete Mathematics **35** (2009), 47–52.

[3] Bahiense, L. and Manić, G and Piva, B. and de Souza, C., *The maximum common edge subgraph problem : A polyhedral investigation*, Discrete Applied Mathematics **160** (2012), 2523–2541.

[4] Bokhari, *On the Mapping Problem*, IEEE Transactions on Computers **3** (1981), 207–214.

[5] De Santo, M., Foggia, P., Sansone, C. and Vento, M., *A large database of graphs and its use for benchmarking graph isomorphism algorithms*, Pattern Recognition Letters **24** (2003), 1067–1079.

[6] Marenco, Javier and Loiseau, Irene, "A branch&cut algorithm for a problem arising in parallel programming environments ", Universidade de Buenos Aires, Departamento de Computación,2000.

[7] Marenco, Javier and Loiseau, Irene, "Un algoritmo branch-and-cut para el problema de mapping ", Master thesis, Universidade de Buenos Aires, Departamento de Computación,1999.