Représentations compactes des graphes et contraintes pseudo booléennes

Said Jabbour *, Nizar Mhadhbi*
Badran Raddaoui**, Lakhdar Sais *

*CRIL - CNRS UMR 8188, University of Artois
F-62307 Lens Cedex, France
{jabbour, mhadhbi, sais}@cril.fr
**SAMOVAR, Télécom SudParis, CNRS, Univ. Paris-Saclay
F-91011 Evry Cedex, France
badran.raddaoui@telecom-sudparis.eu

Résumé. Les graphes représentent un outil efficace pour la modélisation des relations structurelles entre les objets. Cependant, l'exploitation de ces graphes de données est très coÃ»teuse en raison de la taille. En effet, dans la plupart des applications réelles, la taille des graphes est largement grande, ce qui rend difficile à comprendre l'information et la structure codée dans ces graphes par une simple visualisation. La représentation compacte des grands graphes, appelée aussi *compression des graphes*, est une opération qui permet la diminution du nombre d'arêtes ou de nœuds du graphe pour faciliter leurs traitements. Dans cet article, nous proposons une nouvelle approche basée sur l'utilisation des contraintes pseudo booléennes pour une représentation condensée de larges graphes. L'avantage d'une telle représentation est qu'au lieu de représenter un graphe (e.g., clique) par un nombre quadratique d'arêtes, on peut l'exprimer sous forme d'une inéquation linéaire dont les modèles correspondent exactement aux arêtes du graphe initial. Notre approche permet le passage à l'échelle tout en garantissant la décompression de graphes par une simple résolution des inéquations linéaires correspondantes. Les expérimentation sur plusieurs graphes réels montrent que notre approche offre de meilleures performances comparée à plusieurs approches de l'état de l'art.

1 Introduction

Les réseaux complexes sont au cœur des sciences humaines et naturelles car ils permettent de représenter les interactions entre entités. Ces interactions sont souvent modélisés par des graphes, un outil efficace pour la modélisation des relations structurelles entre entités. Aider à comprendre le contenu en information des grands graphes est un challenge important. En effet, dans la plupart des applications réels, les graphes ont des grandes tailles, ce qui rend difficile la compréhension de leurs structures.

La taille des grands graphes présente souvent un obstacle pour comprendre les informations essentielles qu'ils contiennent. La compression de graphe a pour objectif de changer la représentation du graphe afin de diminuer la taille occupée dans la mémoire.

Plusieurs approches ont été proposées pour compacter un graphe. Ces approches dépendent fortement du type de graphe. Les principaux types de graphes étudiés dans le domaine de compression sont les graphes orientés, graphes non orientés, graphes attribués. Les principales approches de compression des graphes peuvent être classées en trois classes comme suit :

Les méthodes de la première classe sont à base de regroupement de liens ou de sommets, où le problème de compression est vu comme un problème de clustering ou de détection de communautés Liu et al. (2018). La deuxième classe est constituée des méthodes à base d'extraction de motifs fréquents, où l'idée consiste à compresser les listes d'adjacences des sommets en recherchant des motifs fréquents Maccioni et Abadi (2016). La dernière classe est constituée des méthodes à base de structures de graphes, où l'idée est de chercher des structures fréquentes dans le graphe et de trouver un meilleur résumé utilisant le principe de la longueur de description minimale (MDL) Koutra et al. (2014).

L'approche proposée dans cet article Jabbour et al. (2016) est à base de structure de graphe et repose sur une description du graphe initial par des contraintes pseudo booléennes représentant des classes particulières de graphes (incluant les cliques et plusieurs extensions de classes de graphes bipartis) et dont les solutions correspondent aux arêtes du graphe initial.

2 Préliminaires

Une contrainte pseudo booléenne est une inéquation linéaire de la forme :

$$C_{pb} = \sum_{i=1}^{n} a_i.x_i \ \# \ k$$

avec $a_i \in \mathbb{Z},\ k \in \mathbb{N},\ \# \in \{=, <, \leq, >, \geq\}$;

Un modèle de C_{pb} est une valuation des variables x_i ($1 \leq i \leq n$) satisfaisant l'inéquation linéaire. Un modèle I de C_{pb} est appelé k-modèle si le nombre de variables affectées à $vrai$ (ou 1) est égal à k. Nous notons $mod(C_{pb})$ l'ensemble de modèles de C_{pb} et $mod_k(C_{pb})$ l'ensemble de ses modèles de taille k.

Les contraintes pseudo booléennes ont fait l'objet de beaucoup de travaux pour les transformer en Forme Normale Conjonctive (CNF). Par exemple une contrainte de la forme $\sum_{i=1}^{n} x_i \leq 1$ peut s'écrire sous la forme $\bigwedge_{1 \leq i < j \leq n} (\neg x_i \vee \neg x_j)$.

3 Graphes et contraintes pseudo-booléennes

Nous considérons préalablement quelques classes de graphes bien connues, et nous montrons comment elles peuvent être formulées par des contraintes pseudo-booléennes.

Les modèles qui nous intéressent pour représenter un graphe sont ceux de l'ensemble $mod_2(C_{pb})$

Par conséquent, une contrainte pseudo-booléenne est une représentation d'un graphe $G = (V, E)$ si est seulement si ses 2-modèles (ou $mod_2(C_{pb})$) représente les arêtes du graphe :

la restriction de ces deux 2-modèles aux variables positives sont les couples représentant le graphe c'est à dire :

$$E = \bigcup_{I \in mod_2(C_{pb})} (I \cap V)$$

Dans la suite, les sommets d'un graphe sont représentés par les variables propositionnelles $x_1, x_2, \ldots, , x_n$.

Nous commenÃ§ons par la structure la plus simple à savoir la clique. Cette dernière exige que chaque deux sommets différents soient adjacents. Une clique peut être représentée par la contrainte pseudo booléenne suivante $\sum_{i=1}^{n} x_i = 2$. En effet, toute solution de cette contrainte, contient exactement deux variables affectées à 1 et toute interprétation complète avec deux littéraux positifs, est un modèle. Rappelons que le nombre d'arêtes d'une clique est égale $\frac{n(n-1)}{2}$ alors que la contrainte précédente est exprimée avec uniquement n variables, 1 entier positif et 1 opérateur. Un graphe contenant plusieurs cliques peut être codé par une disjonction de contraintes pseudo-booléennes.

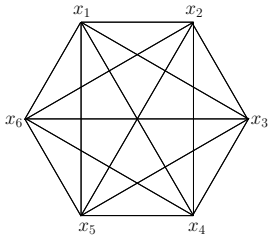

FIG. 1 – *Exemple de clique*

Considérons maintenant une clique en omettant une arête. Cela donne lieu à la structure dite pseudo-clique de la figure 2. Dans ce cas, il existe aussi une correspondance entre les arêtes du graphe et la contrainte $C = x_1 + x_2 + 2x_3 + \ldots + 2x_n \geq 3$. En effet, l'interprétation $\{x_1, x_2, \neg x_3, \ldots \neg x_n\}$ n'est pas un $2-$modèle de C. Cette arête n'est donc pas décrite par un modèle de $mod_2(C)$).

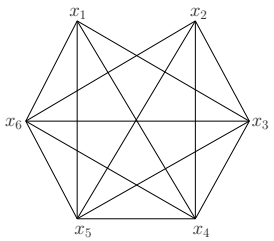

FIG. 2 – *Exemple de pseudo-clique*

Nous terminons la description de ces structures particulières en considérant le cas d'une biclique $G = (X \cup Y, E)$ (voir figure 3). Ici aussi il est possible de représenter les arêtes de ce graphe à partir des 2-modèles de la contrainte $2x_1 + \ldots + 2x_n + 3y_1 + \ldots + 3y_m = 5$. En

effet, l'unique manière de satisfaire la contrainte consiste à affecter un seul x_i et un seul y_j à 1.

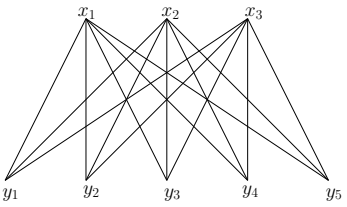

FIG. 3 – *Exemple de biclique*

Ces différents exemples illustrent le pouvoir d'expression des contraintes pseudo booléennes et leurs capacités à réduire la taille de la représentation en la faisant passer de quadratique à linéaire. Dans la suite nous proposons de généraliser ce concept en cherchant les structures pouvant être exprimées par une contrainte pseudo-booléenne.

CommenÃ§ons tout d'abord par définir la notion d'imbrication entre sommets.

Définition 1 (Séquence de sommets imbriqués) *Soit $G = (X, E)$ un graphe non orienté et $u, v \in X$. Les sommets u et v sont appelés sommets imbriqués, notés $u \subseteq_\Gamma v$, si et seulement si $N_u \subseteq N_v$. $\langle X \rangle = \langle x_1 \dots x_n \rangle$ est dite séquence de sommets imbriqués si $x_n \subseteq_\Gamma x_{n-1} \subseteq_\Gamma \dots \subseteq_\Gamma x_1$. Avec N_u l'ensemble des adjacents de u.*

Définition 2 (Graphe biparti imbriqué (NB)) *Soit $G = (X, Y, E)$ un graphe biparti. G est appelé un graphe biparti imbriqué noté NB (Nested Bipartite) si $\langle X \rangle$ est une séquence de sommets imbriqués.*

La définition 2 introduit une classe de graphe appelé *graphe biparti imbriqué*. Un graphe biparti complet est un cas particulier d'un graphe NB où $N_{x_1} = N_{x_2} = \dots = N_{x_n}$.

Définition 3 (Graphe biparti imbriqué ordonné (NOB)) *Soit $G = (X, Y, E)$ un graphe biparti. G est appelé un Graphe biparti imbriqué ordonné noté NOB (Nested-Ordered-Bipartite), s'il existe deux séquences imbriqués de sommets $\langle X \rangle = \langle x_1 \dots x_n \rangle$ et $\langle Y \rangle = \langle y_1 \dots y_m \rangle$ tel que $\forall i \in \{1 \dots n\}, \langle N_{x_i} \rangle = \langle y_1 \dots y_{m_i} \rangle$ où $m_i = |N_{x_i}|$.*

La définition 3 indique qu'un graphe biparti imbriqué ordonné (NOB) $G = (X, Y, E)$ est un graphe biparti imbriqué (NB) où Y peut être réordonné en une séquence $\langle Y \rangle$ tel que chaque N_{x_i} est représenté comme une sous-séquence de $\langle Y \rangle$ commenÃ§ant par y_1 (toutes les séquences de voisinage de x_i commencent avec le même sommet y_1). Notons que $m = m_1 \geq m_2 \geq \dots \geq m_n$, puisque $\langle x_1 \dots x_n \rangle$ est une séquence de sommets imbriqués.

Proposition 1 *Si $G = (X, Y, E)$ est un graphe NB alors G est aussi un graphe NOB.*

Proposition 2 *Soient $G = (X, Y, E)$ un graphe de type NB et $G_{nob} = (\langle X \rangle, \langle Y \rangle, E)$ une représentation NOB de G où $\langle X \rangle = \langle x_1 \ \dots \ x_n \rangle$ et $\langle Y \rangle = \langle y_1 \ \dots \ y_m \rangle$. $G = (X, Y, E)$ peut être exprimé avec la contrainte pseudo-booléenne suivante :*

$$-m \leq A \times X^T + B \times Y^T \leq 0 \quad (1)$$

avec $A = ((m_1 - m) \dots (m_n - m))$ et $B = ((m+1)(m+2) \dots 2m)$
Nous définissons une autre classe de graphe comme suit :

Définition 4 *Un graphe biparti $G = (X, Y, E)$ est dit graphe biparti séquence (SB Sequence-Bipartite graph en anglais) si X et Y peuvent être écrits comme des séquences $\langle X \rangle = \langle x_1 \dots x_n \rangle$ et $\langle Y \rangle = \langle y_1 \dots y_m \rangle$ tel qu'il existe un entier $k > 0$ où $\forall i \in \{1 \dots n\}$, $\exists k \in \{1 \dots m\}$ tel que $\langle N_{x_i} \rangle = \langle y_{1+k_i} \dots y_{k+k_i} \rangle$.*

En d'autres termes, un graphe de type SB consiste en un ensemble de sommets X où leurs voisins sont des sous-séquences de taille k translatées successivement avec k_i sur $\langle Y \rangle$.

Un graphe de type SB peut être exprimé par la contrainte pseudo-booléenne suivante :

$$1 \leq \sum_{j=1}^{m} (\alpha + j) y_j - \sum_{i=1}^{n} (\alpha + \alpha_i) x_i \leq \alpha \quad (2)$$

4 Évaluation expérimentale

Dans notre étude expérimentale nous nous restreignons aux graphes de type bipartis imbriqués pour compresser les graphes en entrée. Pour étudier la faisabilité de notre approche SuLI (Summarization using Linear Inequalities), nous l'avons testé sur plusieurs classes de graphes réels. La table 1 résume les résultats de compression obtenus par notre algorithme. Ces résultats montrent clairement que les graphes bipartis imbriqués dits graphes NB sont fréquents dans la plupart des réseaux considérés comme *Facebook*, *Twitter*, *Yahoo*, *LiveJournal*, *Youtube*, *Flickr*. Les colonnes 2, 3 et 4 de la table 1 reportent le nombre de sous-graphes bipartis trouvés, le taux de couverture qui est égal au nombre des arêtes couvertes dans les graphes NB divisé par le nombre des arêtes du graphe original et la taille minimale et maximale des graphes NB. Le nombre de sous graphes bipartis imbriqués de type NB générés par notre algorithme SuLI varie en fonction de la taille du graphe original comme indiqué dans la colonne 2 de la table 1. À titre d'exemple, pour le graphe *chocolat* le nombre de sous-graphes bipartis NB détectés est 57 couvrants 81.03% du graphe original. Pour le graphe Facebook, nous trouvons 12800 sous-graphes NB couvrants 81.20% du graphe original. Nous avons comparé notre approche par rapport à l'approche VOG Koutra et al. (2014). Les résultats montrent que notre approche obtient les meilleurs taux de compression sur 9 graphes parmi 15 (*Chocolate, CaAstroPh, Twitter, Enron, Cit-hep-th, cnr-2000, Youtube, Yahoo, Toto*).

Références

Jabbour, S., N. Mhadhbi, A. Mhadhbi, B. Raddaoui, et L. Sais (2016). Summarizing big graphs by means of pseudo-boolean constraints. In *2016 IEEE International Conference on Big Data, BigData 2016, Washington DC, USA, December 5-8, 2016*, pp. 889–894.

Network	nodes/edges	File Size	#NB graphs	coverage	min/max size	avg size	time (s)	Compression Rate	
								VOG (%)	SuLI (%)
Chocolate	4 039/87 885	940.3KB	57	81.03%	2/2 876	99	9654	39.14	**64.14**
Facebook	473 315/3 505 519	47MB	12 800	81.23%	2/2 876	98	501.94	**68.08**	62.97
Ca-AstroPh	18 772/198 110	207.7KB	3 119	81.60%	2/2 180	51	340	25	**27.78**
Twitter	18 772/198 050	4MB	3119	81.60%	2/2180	51	309.6	65	**75.14**
Enron	36 691/186 936	4MB	718	53.74%	2/2206	120	8754	32.5	**47.5**
epinions	75 877/405 739	380.4KB	924	26.20%	2/2 334	115	1 387	**60.63**	47
Gag	1 732 999/5 236 270	76.8MB	2 635	65.12%	24/5 200	1 294	303	**86.97**	84.24
Cit-hep-th	27 400/352 021	658.6KB	9 388	91.52%	2/4 203	34	1 765	67.07	**82.02**
cnr-2000	325 557/3 216 152	41.5MB	487	37.81%	8/194 103	2 126	417	39.03	**40.24**
DBLP	317 080/1 049 866	13.4MB	8 281	30.88%	2/690	39	5 785	**19.40**	14.92
LiveJournal	3 997 962/34 681 189	50.4MB	4 365	73.15%	43/7 948	1476	3 643	**80**	67.46
Youtube	1 134 890/2 987 625	38.2MB	8 000	25.89%	4/10 078	353	2 111.4	13.08	**30.36**
Flickr	105 938/2 316 948	48.7MB	8 084	75.65%	2/52 071	216	4 837	**59.54**	39.01
Yahoo	105 938/2 316 948	24.9MB	4 800	39.99%	5/52 039	709	6 511	48.99	**54.61**
Toto	19 887/367 663	3.7MB	985	22.87%	2/4 697	795	560	32.43	**56.75**

TAB. 1 – *Résultats de compression sur plusieurs graphes réels (VOG vs SuLI)*

Koutra, D., U. Kang, J. Vreeken, et C. Faloutsos (2014). VOG : summarizing and understanding large graphs. In *Proceedings of the 2014 SIAM International Conference on Data Mining, Philadelphia, Pennsylvania, USA, April 24-26, 2014*, pp. 91–99.

Liu, Y., T. Safavi, A. Dighe, et D. Koutra (2018). Graph summarization methods and applications : A survey. *ACM Comput. Surv. 51*(3), 62 :1–62 :34.

Maccioni, A. et D. J. Abadi (2016). Scalable pattern matching over compressed graphs via dedensification. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*, pp. 1755–1764.

Summary

How to succinctly represent the truly relevant information in big data graphs? The approach presented in this paper aims to discover hidden graph structures and exploit them to compactly summarize large graphs. First, we show that some special graph classes such as cliques and bicliques can be represented efficiently as *Pseudo-Boolean (PB) constraints*. Then, we propose three new graph classes representable as PB constraints, called *nested*, *sequence* and *clique-nested bi-partite* graphs. Finally, we derive a general approach for partial or complete summarization of an arbitrary graph as a disjunction of PB constraints. Our representation can be seen as an original way to represent the edges of the graph, as they correspond to particular solutions of the PB constraints. An extensive experimental evaluation on several real-world networks shows that our framework is competitive with the state-of-the-art compression technique.