Résolution approchée du problème de k-couverture maximale pour ensembles de grande taille

Lucile Mahé, Julien Darlay

Hexaly, 251 Boulevard Pereire 75017 Paris, France
{lmahe, jdarlay}@hexaly.com

Mots-clés : *max k coverage, approximation probabiliste, programmation mathématique.*

1 Introduction

Le problème de k-couverture maximale est un problème classique de la littérature. Un total de M ensembles (notés S_i) couvrant une population de taille N est donné. Chaque élément peut être présent dans un ou plusieurs ensembles. Le but est alors de sélectionner k ensembles tels que le nombre d'éléments couverts est maximal. Ce problème peut être modélisé par un modèle linéaire en nombre entiers, ou résolu de manière approchée par un algorithme glouton de complexité $O(kNM)$. Cependant, si la taille de la population devient très importante devant le nombre d'ensembles, ces deux approches peuvent devenir très coûteuses. Nous proposons une modélisation alternative permettant la résolution approchée d'instances pour N très grand.

2 Modélisation et approche choisie

Le modèle linéaire en nombres entiers classique pour le problème de k-couverture maximale est le suivant :

max.	$\sum_{j\in[\![1;N]\!]} y_j$	Maximisation du nombre d'éléments couverts
s.t	$\sum_{i\in[\![1;M]\!]} x_i \leq k$	Pas plus de k ensembles sélectionnés
	$\sum_{i\|j\in S_i} x_i \geq y_j$	L'élément j est couvert si l'un des ensembles i contenant j est pris
	$y_j \in \{0,1\}$	Prend la valeur 1 si l'élément $j \in [\![1;N]\!]$ est couvert
	$x_i \in \{0,1\}$	Prend la valeur 1 si l'ensemble $i \in [\![1;M]\!]$ est sélectionné

Ce modèle contient $N+M$ variables et $N+1$ contraintes et, d'après nos expériences, n'est plus utilisable lorsque N dépasse plusieurs centaines de milliers. De plus le nombre de non-zéros de la matrice de contraintes peut devenir très grand si les ensembles sont denses. Ce problème a été rencontré avec l'un de nos clients, dans le cadre d'une application industrielle.

La communauté des algorithmes de *streaming* propose des structures de données permettant d'obtenir une approximation du nombre distinct d'éléments dans un ensemble sans représenter l'intégralité de la population. Ces structures, basées sur des estimateurs probabilistes, permettent en outre de fournir la cardinalité de l'union de plusieurs ensembles en temps linéaire et sans augmentation de l'erreur d'approximation [1, 2, 3]. L'estimateur stocke pour chaque ensemble i une liste H_i d'entiers de longueur $n << N$. La taille n choisie ne dépendra que de la précision du résultat souhaitée, et non de la taille de la population à mesurer. L'union de deux ensembles i_1 et i_2 peut être représentée par le maximum terme à terme des listes H_{i_1} et H_{i_2} [1].

Le modèle s'écrit alors comme le programme non linéaire :

max.	$f(H)$	Maximisation du nombre d'éléments couverts
s.t.	$\sum_{i \in [\![1;M]\!]} x_i \leq k$	Pas plus de k ensembles sélectionnés
	$H[l] = \max_{i \in [\![1;M]\!]} \{H_i[l] * x_i\}$	Calcul de l'estimateur (union)
	$x_i \in \{0, 1\}$	Prend la valeur 1 si l'ensemble i est sélectionné
	$H[l] \in \mathbb{R}^+$	La valeur de la structure de l'union

La fonction f pour l'estimateur LogLog est définie par $f(H) = \alpha_n n 2^{\frac{1}{n}\sum_l H[l]}$. Dans le cadre de la maximisation, on peut donc considérer simplement la somme des coefficients de H, le reste étant des constantes positives dépendant uniquement de la précision choisie pour l'estimateur. Ainsi, le nombre de variables dépend uniquement du nombre d'ensembles, et le nombre de contraintes est indépendant de la taille de la population.

3 Résultats et conclusion

Ce modèle peut s'écrire directement avec la variable de type `set` de Hexaly Optimizer (anciennement LocalSolver), prenant la forme suivante avec une seule variable et une seule contrainte :

```
function model() {
    x <- set(M);                          // Indices des ensembles sélectionnés
    constraint count(x) <= k;             // Pas plus de k ensembles sélectionnés
    H[l in 0...n] <- max(x, i => H[i][l]); // Calcul de l'estimateur (union)
    maximize sum[l in 0...n](H[l]);       // Maximisation du nombre d'éléments couverts
}
```

FIG. 1 – Modèle LSP du problème de k-couverture maximale avec estimateur

Le modèle a été testé avec Hexaly Optimizer 12.0 sur l'instance client avec $N = 10^6$, $M = 108$, $k = 10$ et des estimateurs H_i avec 4096 éléments. On obtient alors une valeur optimale à moins de 10^{-5} de gap, en quelques secondes. Des tests plus généraux ont également été menés, en retirant les ensembles de trop grande taille qui influençaient les résultats, et pour des valeurs de k variables de 2 à 60. Le gap d'optimalité prouvé avec Hexaly Optimizer est toujours compris entre 0 et 10^{-3}, en moins de 10 secondes de calcul.

Avec cette approche probabiliste, il est possible de résoudre le problème de k-couverture maximale à l'aide d'un modèle dont la taille est indépendante de la population à couvrir tout en conservant la précision des estimateurs probabilistes. La rapidité de résolution pourra être ajustée au détriment de la précision souhaitée sur le résultat, en modifiant la taille de l'estimateur.

Références

[1] Durand, M., & Flajolet, P. (2003). Loglog counting of large cardinalities. In *Algorithms-ESA 2003 : 11th Annual European Symposium, Budapest, Hungary, September 16-19, 2003. Proceedings 11* (pp. 605-617). Springer Berlin Heidelberg.

[2] Flajolet, P., Fusy, É., Gandouet, O., & Meunier, F. (2007). Hyperloglog : the analysis of a near-optimal cardinality estimation algorithm. *Discrete mathematics & theoretical computer science*, (Proceedings).

[3] Heule, S., Nunkesser, M., & Hall, A. (2013, March). Hyperloglog in practice : Algorithmic engineering of a state of the art cardinality estimation algorithm. In *Proceedings of the 16th International Conference on Extending Database Technology* (pp. 683-692).