

P. Bastide et C. Groenland : Reconstruction de graphes avec un oracle de distances

Paul Bastide, LaBRI, Bordeaux, paul.bastide@ens-rennes.fr

Carla Groenland, Utrecht University, Utrecht, c.e.groenland@uu.nl

À quelle vitesse peut-on reconstruire un réseau en utilisant seulement la commande `ping`? Ou formellement, en combien de requêtes peut-on reconstruire un graphe en ayant accès seulement à un oracle connaissant la distance entre deux sommets?

On se place dans le modèle restreint où G est un graphe connecté, simple, non orienté et de **degré borné**. Le but est de trouver toutes les arêtes de G en utilisant uniquement des requêtes de la forme $\text{dist}(u, v)$ pour u et v deux sommets de G .

Sampath Kannan, Claire Mathieu et Hang Zhou [1] ont répondu à cette question en créant plusieurs algorithmes randomisés spécialisés pour différentes classes de graphes (e.g arbres, graphes cordaux, graphes planaire extérieures, etc...). Tous leurs algorithmes ont une complexité en moyenne quasi-linéaire en la taille du graphe à reconstruire. Mais la question de l'existence d'algorithmes déterministes quasi-linéaires était ouverte même pour une classe aussi simple que celle des arbres binaires.

Theorem 1 *Il existe un algorithme déterministe qui reconstruit la classe des graphes cordaux en un nombre quasi-linéaire de requêtes.*

De plus, nous avons généralisé l'approche utilisée dans [1] pour reconstruire la classe des graphes cordaux, et nous l'avons étendue à la classe des graphes de *connected treewidth* bornées.

Theorem 2 *Il existe un algorithme randomisé qui reconstruit n'importe quelle classe des graphes de *connected treewidth* bornée en un nombre moyen quasi-linéaire de requêtes.*

Ceci implique l'existence d'algorithmes randomisés quasi-linéaires pour des classes des graphes intéressantes telles que la classe des graphes de permutations et la classe des graphes *AT-free*.

Références

- [1] Sampath Kannan, Claire Mathieu and Hang Zhou, Graph Verification and Reconstruction via Distance Oracles, ACM Trans. Algorithms, 2018.