Appendix to the Paper Breaking the $O(\ln n)$ Barrier: An Enhanced Approximation Algorithm for Fault-Tolerant Minimum Weight Connected Dominating Set

Jiao Zhou, Zhao Zhang*
College of Mathematics Physics and Information Engineering, Zhejiang Normal University

Shaojie Tang
Naveen Jindal School of Management, The University of Texas at Dallas

Xiaohui Huang
College of Mathematics Physics and Information Engineering, Zhejiang Normal University

Dingzhu Du
Department of Computer Science, University of Texas at Dallas

---

▶ The Analysis of D_1

To analyze D_1 in Algorithm 1, we need the concept of polymatroid and submodular cover. Potential function f is *monotone increasing* if $f(C) \leq f(D)$ holds for any subsets $C \subseteq D \subseteq V$. It is *submodular* if and only if $\triangle_u f(C) \geq \triangle_u f(D)$ holds for any $C \subseteq D \subseteq V$ and any $u \in V \setminus D$. A monotone increasing and submodular function f with $f(\emptyset) = 0$ is called a *polymatroid*. Given an element set U with cost function $c : U \mapsto \mathbb{R}^+$ and given a polymatroid $f : 2^U \mapsto \mathbb{R}^+$, denote by $\Omega_f = \{C \subseteq U \colon \ \triangle_u f(C) = 0 \text{ for any } u \in U\}$. The *Submodular Cover* problem is:

$$\min \ \ c(C) = \sum_{u \in C} c(u)$$

* Corresponding author: hxhzz@sina.com.

$$s.t.\ \ C \in \Omega_f.$$

The following is a classic result which can be found in (Du et al. 2011) Theorem 2.29.

Theorem 0.1. *The greedy algorithm for the submodular cover problem has performance ratio $H(\gamma)$, where $\gamma = \max\{f(\{u\}): u \in U\}$ and $H(\gamma) = \sum_{i=1}^{\gamma} 1/i$ is the Harmonic number.*

Lemma 0.2. *Function q is a polymatroid.*

Proof. Obviously, $q(\emptyset) = 0$. It is easy to see that for any node $u \in V$, function $-q_C(u)$ is monotone increasing and submodular with respect to C. So, function q, being the summation of a constant function and some monotone increasing and submodular functions, is also monotone increasing and submodular. □

▶ **Proof of Theorem 4.1.**

Proof. If there exists a node $u \in V \backslash D_1$ with $|N_{D_1}(u)| < m$, then $-\triangle_u\, q_{D_1}(u) = -q_{D_1 \cup \{u\}}(u) + q_{D_1}(u) = -0 + (m - |N_{D_1}(u)|) > 0$, and $-\triangle_u\, q_{D_1}(v) \geq 0$ for any node $v \in V \setminus \{u\}$ (by the monotonicity of $-q_{D_1}(v)$ with respect to D_1). Thus $\triangle_u q(D_1) = -\sum_{v \in V} \triangle_u q_{D_1}(v) > 0$, and the algorithm does not terminate at this stage. Hence when the algorithm jumps out of the first while loop, D_1 is an m-DS.

By Lemma 0.2, the minimum weight m-fold dominating set problem is a special submodular cover problem with potential function q. So by Theorem 0.1, we have $w(D_1) \leq H(\gamma) \cdot opt'$, where opt' is the optimal value for the m-MWDS problem. Then, the result follows from the observation that $opt' \leq opt$ and $\gamma = \max_{u \in V}\{q(\{u\})\} = \max_{u \in V}\{m|V| - \sum_{v \in V \setminus \{u\}}(m - |N_{\{u\}}(v)|)\} = \max_{u \in V}\{m + d_G(u)\} = m + \delta$. □

References

Du, Ding-Zhu, Ker-I Ko, Xiaodong Hu. 2011. *Design and analysis of approximation algorithms*, vol. 62. Springer Science & Business Media.