Solving the maximum independant set problem with Tropical tensor networks

Jin-Guo Liu
Harvard University
jinguoliu@harvard.edu

Abstract

Solving the maximum independent set size problem: the maximum independent set size, the degeneracy, the optimal configuration and the equivalence between different graphs.

1 Tools

OMEinsum a package for einsum
https://github.com/under-Peter/OMEinsum.jl

TropicalGEMM a package for efficient tropical matrix multiplication (compatible with OMEinsum)
https://github.com/TensorBFS/TropicalGEMM.jl

TropicalNumbers a package providing tropical number types and tropical algebra, one o the dependency of TropicalGEMM
https://github.com/TensorBFS/TropicalNumbers.jl

2 Computing degeneracy

3 Utilizing the sparsity

Tensor network compression is an important tool to utilize sparsity.

We contract the tensors in a subregion $R \subseteq G$ of a graph G, and obtain a resulting tensor A of rank $|C|$, where C is the set of vertice tensors at the cut. The maximum independant set size in this region with boundary configuration $\sigma \in \{0, 1\} \otimes |C|$ is A_σ. We say an entry A_{σ_a} is "better" than A_{σ_b} if

$$(\sigma_a \wedge \sigma_b = \sigma_a) \wedge (A_{\sigma_a} \geq A_{\sigma_b}), \tag{1}$$

where $\wedge$ is a bitwise and operations. The first term means that whenever a bit in σ_a has boolean value 1, the corresponding bit in σ_b is also 1. While the second term means the maximum independant set size with boundary configuration fixed to σ_a is not less than that fixed to σ_b. The word "better" means the best solution with boundary configuration σ_a is never worse than that with σ_b. When Eq. 1 holds, It is easy to see that if $\sigma_b \cup \overline{\sigma_b}$ is one of the solutions for maximum independant sets in G, $\sigma_a \cup \overline{\sigma_b}$ is also a solution.

3.1 The equivalence between branching and compression

We are going the verify the Lemmas used in branching algorithm in book (Fomin & Kaski, 2013).

***Rule* 1.** *If a vertex v is in an independent set I, then none of its neighbors can be in I. On the other hand, if I is a maximum (and thus maximal) independent set, and thus if v is not in I then at least one of its neighbors is in I.*

***Rule* 2.** *Let $G = (V, E)$ be a graph, let v and w be adjacent vertices of G such that $N[v] \subseteq N[w]$. Then*

$$\alpha(G) = \alpha(G \backslash w). \tag{2}$$

***Rule* 3.** *Let $G = (V, E)$ be a graph and let v be a vertex of G. If no maximum independent set of G contains v then every maximum independent set of G contains at least two vertices of $N(v)$.*

***Rule* 4.** *Let $G = (V, E)$ be a graph and v a vertex of G. Then*

$$\alpha(G) = \max(1 + \alpha(G \backslash N[v]), \alpha(G \backslash (M(v) \cup \{v\})). \tag{3}$$

***Rule* 5.** *Let $G = (V, E)$ be a graph and v be a vertex of G such that $N[v]$ is a clique. Then*

$$\alpha(G) = 1 + \alpha(G \backslash N[v]). \tag{4}$$

***Rule* 6.** *Let G be a graph, let S be a separator of G and let $I(S)$ be the set of all subsets of S being an independent set of G. Then*

$$\alpha(G) = \max_{A \in I(S)} |A| + \alpha(G \backslash (S \cup N[A])). \tag{5}$$

***Rule* 7.** *Let $G = (V, E)$ be a disconnected graph and $C \subseteq V$ a connected com- ponent of G. Then*

$$\alpha(G) = \alpha(G[C]) + \alpha(G \backslash C)). \tag{6}$$

$$if|V| = 0 then return 0 if \exists v \in V with d(v) \leq 1 then return 1 + mis2(G \backslash N[v]) if \exists v \in V with d(v) = 2 then (let u1 and u2 be the neighbors o \tag{7}$$

References

Fedor V Fomin and Petteri Kaski. Exact exponential algorithms. *Communications of the ACM*, 56 (3):80–88, 2013.