Light Spanners in Weighted Graphs with Forbidden Minors

Papa Amar Sissokho
Illinois State University

Abstract

Given an edge weighted graph G with n vertices and no K_r-minor and a small positive ϵ, we show that a simple greedy algorithm (of Althöfer et al.) finds a spanning subgraph approximating all shortest-path distances within a factor of $1+\epsilon$, and with total edge weight at most $O((r\sqrt{\log r}\cdot\log n)/\epsilon)$ times the weight of a minimum spanning tree. This result implies a quasi-polynomial time approximation scheme (QPTAS) for the traveling salesman problem (TSP) in such graphs, with running time $n^{O((r^4\sqrt{\log r}\cdot\log n\cdot\log\log n)/\epsilon^2)}$.

Our analysis uses the concept of "gap number," and it shows that a graph with *gap number* $\Omega(r\sqrt{\log r}\cdot\log n)$ has a K_r-minor. We also show that this dependence on n is nearly tight, by exhibiting graphs with no K_6-minor (apex graphs) and detour gap number $\Omega((\log n)/\log\log n)$.

As a step toward eliminating the $\log n$ factors in the first paragraph, we propose a generalized gap number, now depending on ϵ, and we show that it remains bounded for apex graphs, circular-arc graphs, and some similar graph families.

This is a joint work with M. Grigni.

Definition: Let G be a connected graph. We define
i)

$$\mathrm{gap}(e) \stackrel{\mathrm{def}}{=} d_{G-e}(u,v) - w(e) \quad \text{and} \quad \mathrm{gap}(G,T) \stackrel{\mathrm{def}}{=} \max_{w} \frac{\sum \mathrm{gap}(e) | e \in G-T\}}{w(T)},$$

where w ranges over nonnegative edge-weighting on G for which $T = \mathrm{MST}(G)$.
ii)

$$\mathrm{gap}(G) \stackrel{\mathrm{def}}{=} \max_{T} \mathrm{gap}(G,T),$$

where T ranges over all spanning trees of G.

Definition: A *detour* is a pair (P,e) where e is an edge and P is a path such that $e+P$ is a simple cycle (so, P connects the endpoints of e).

We can interpret $\mathrm{gap}(G,T)$ as the value of a Linear Program. In such case, we say *detour gap number* instead of *gap number*. However, the two notions are equivalent.