Abstract

A spectral sparsifier of a graph G is a sparser graph H that approximately preserves the quadratic form of G, i.e., for all vectors x, $x^T L_G x \approx x^T L_H x$, where L_G, L_H denote the respective graph Laplacians. Spectral sparsifiers generalize cut sparsifiers, and have found several applications in designing graph algorithms. In recent years, there has been interest in computing spectral sparsifiers in the semi-streaming and dynamic settings. Natural algorithms in these settings involve repeated sparsification of a dynamic graph. We present a framework for analyzing algorithms for graph sparsification that perform repeated sparsifications. The framework yields analyses that avoid a worst-case error accumulation across various resparsification steps, and only incur the error corresponding to a single sparsification step, leading to better results. As an application, we show how to maintain a spectral sparsifier of a graph, with $O(n \log n)$ edges in a semi-streaming setting: We present a simple algorithm that, for a graph G on n vertices and m edges, computes a spectral sparsifier of G with $O(n \log n)$ edges in a single pass over G, using only $O(n \log n)$ space, and $O(m \log^2 n)$ total time. This improves on previous best constructions in the semi-streaming setting for both spectral and cut sparsifiers. The algorithm also extends to semi-streaming row sampling for general PSD matrices. As another example, we use this framework to give a parallel algorithm that achieves the best combinatorial construction of spectral graph sparsifiers, combining an algorithm due to Koutis with improved spanner construction.