CS 599: Rounding Techniques in Approximation Algorithms

Homework 3: Sparsity and Iterated Rounding/Relaxation

Fall 2024

Collect at least 3 points from the following 4 problems, or do more for extra credit. Normal problems are worth 1 point. The bonus versions are worth two points. (Do not do both for any problem.)

1 Problem 1: Sparsity for Unweighted k-Edge-Connectivity

Give a $1 + O(1/k)$ approximation for the unweighted minimum k-edge-connected spanning subgraph problem using sparsity. Recall here we are given a k-edge-connected graph $G = (V, E)$ and want to return a subset of edges $F \subseteq E$ of minimum size so that $G' = (V, F)$ is k-edge-connected.

Bonus: Design a $1 + O(1/k)$ approximation for the unweighted minimum k-arc connected spanning subgraph problem instead. In other words, prove the same but for directed graphs when you want to have k edges going in each direction across every cut.

2 Problem 2: Iterative Rounding for 3-Dimensional Matching

Consider a hypergraph $H = (V, E)$ with a vertex set and a collection of hyperedges $E \subseteq 2^V$. A collection $M \subseteq E$ is a matching if all $e, f \in M$ have $e \cap f = \emptyset$. Give a $\frac{1}{2}$-approximation for the problem of finding a maximum size matching when H is 3-partite, i.e. when H is given by three vertex sets V_1, V_2, V_3 and every hyperedge contains exactly one vertex from each of V_1, V_2, and V_3.

3 Problem 3: Iterative Relaxation for Beck-Fiala

Improve the Beck-Fiala bound to $2t - 1$.

Bonus: Improve the Beck-Fiala bound to $2t - 3$ for $t \geq 3$ instead.

4 Problem 4: Integrality for Matching

Design a polynomial time algorithm for maximum weight bipartite matching using an LP. Generalize it to work in the case when all vertices v have a required degree $d_v \in \mathbb{Z}_{\geq 0}$.

Bonus: Design an LP-based polynomial time algorithm for maximum weight perfect matching on general graphs instead (which can be used to solve maximum weight matching). You may use without proof that the family of inequalities $x(\delta(S)) \geq 1$ for all $S \subset V : |S|$ odd has a polynomial time separation oracle.