

Caleb Logemann

MATH 566 Discrete Optimization

Homework 5

1. Let S be defined as intersection of halfspaces $x_i \geq 0$ and $(1 - x_i)^k \geq 0$. Suppose $i \in \{1, 2, \dots, d\}$ and $k \geq 1$ is odd. Compute the analytic center of S . Notice that for x satisfying $(1 - x_i)^k \geq 0$, the function $(1 - x_i)^k$ is convex.
2. Compute central path for the following problem

$$(P) \begin{cases} \text{minimize} & -x_1 \\ \text{subject to} & x_1 \leq 1 \\ & x_2 \leq 1 \\ & x_1 \geq 0 \\ & x_2 \geq 0 \end{cases}$$

and find the optimal solution using the central path. Plot (sketch) the set of feasible solutions and the computed central path. Lot of calculus...

3. Let $G = (V, E)$ and $|V| = n$.

Recall that the spanning tree polytope was created by constraints *tree has $n - 1$ edges* and *tree has no cycles*. Formally,

$$STP = \left\{ \mathbf{x} \in [0, 1]^{E(G)} : \sum_{e \in E} x_e = n - 1, \sum_{uv \in E, u \in X, v \in X} x_{(u,v)} \leq |X| - 1 \text{ for } \emptyset \subset X \subset V \right\}.$$

Suppose we try to characterize the spanning tree by assuming that by constraints *tree has $n - 1$ edges* and *tree is connected*. The tree is connected can be formulated by saying that for every cut, the sum x_e of edges e in the cut is at least one. Formally,

$$P = \left\{ \mathbf{x} \in [0, 1]^{E(G)} : \sum_{e \in E} x_e = n - 1, \sum_{uv \in E, u \in X, v \notin X} x_{(u,v)} \geq 1 \text{ for } \emptyset \subset X \subset V \right\}.$$

- (a) Prove that the spanning tree polytope is a subset of P . That is, $STP \subseteq P$.
- (b) Show P does NOT have to be the same as the spanning tree polytope. To do this, show that the polytope P does NOT have to be integral (i.e., P contains a vertex that does not have all coordinates integers).

Hint: See the book Combinatorial Optimization from Korte and Vygen on the bottom of the page 150. Free PDF available from ISU library.

4. Implement any minimum spanning tree algorithm and test it on random data. You can pick any algorithm you like. You can use ANY programming language but you have to IMPLEMENT the method yourself (calling a library function `RunKruskal` is not acceptable). Obtain data by randomly generating 10 points in range $[0, 10]^2$ and the cost of every edge is the Euclidean distance in \mathbb{R}^2 . We consider all 45 edges of K_{10} . Finally, create the plot of the random points and draw edges picked to the spanning tree. You should provide: Name of the algorithm you implemented and short description of implementation, printout of the source code, pictures of two solutions. Template is provided for Sage, you do not have to use it. Time complexity DOES NOT matter.