OPTIMIZATION METHODS: CLASS 2

IP,LP formulations

EXERCISE ONE Decide which of the following statements are true and which are false:

- Every LP instance (of polynomial size) is solvable in polynomial time.
- Every IP instance (of polynomial size) is NP-hard.
- Every LP instance can be transformed into an equivalent form which uses equality in all its constraints.
- If an LP instance has no maximization clause, we can transform it to a form with equations and then use Gaussian elimination to solve it.

EXERCISE TWO Design a linear or interer program for each problem listed below. (If you figure out an LP, it is always better, but it may not be always possible.)

- (Transportation problem with a twist.) There are n bakeries and m stores in Arbitraryville. Every day the i-th bakery bakes p_i baguettes and the j-th store sells o_j baguettes. Transporting a baguette from the i-th bakery to the j-th store costs the store c_{ij} crowns.
  However! After a few ideal months in Arbitraryville, the stores realized that they need to pay an additional fee l_{ij} for each route from the i-th bakery to the j-th store.
  The stores need to pay the fee l_{ij} only when at least a part of a baguette gets transported from the i-th bakery to the j-th store – but the cost is constant and independent of the number of baguettes transported.
  Your program should find the optimum flow of baguettes so that each bakery gets rid of all baguettes, each store sells all they can sell and the total fees are minimal.
- (Approximating with a line.) We have n points in $\mathbb{R}^2$. Find a line (and its coordinates) which minimizes the sum of vertical distances between the points and the line. By "vertical distance" we mean the distance between the line and the point, measured only in the y axis.
  To make things easier, you can assume that no vertical line is a valid solution.
- (Shortest path.) We have an undirected, unweighted graph G and two special vertices s, t. Suggest an LP which finds a shortest path from s to t.

EXERCISE THREE Suppose we have a real matrix A and appropriate vectors b, c. From those, we can build the following integer program C:

$$\begin{aligned} \max c^T x \\ Ax \leq b \\ x \in \{0,1\}^n \end{aligned}$$

Using the same givens, we could also build a linear program L:

$$\begin{aligned} \max c^T x \\ Ax \leq b \\ x \in [0,1]^n \end{aligned}$$

Assume that both programs have a solution. Suppose that we pick one optimal solution of the integer program and call it x^*_C, and we also pick one optimal solution of the linear program, denoting it x^*_L. Prove the following inequality:

$$c^T x^*_C \leq c^T x^*_L.$$

EXERCISE FOUR Let us consider the following NP-hard problem, called WEIGHTED VERTEX COVER:

Input: Undirected graph G with non-negative real weights on the vertices, given by a weight function $w : V(G) \to \mathbb{R}_0^+$.

Goal: To find a subset S of vertices such that each edge $e \in E(G)$ has at least one endpoint in S. (We say that a vertex *covers* an edge, so if we cover all edges, we get a vertex cover.)

From all such subsets S we look for the one with *minimum weight* , i.e. minimum $\sum_{s \in S} w(s)$.

Suggest an integer program with variables $x \in \{0, 1\}$ that finds the optimal solution of WEIGHTED VERTEX COVER.

EXERCISE FIVE

During the last practical we have found a 2-approximation of unweighted vertex cover. Let us now make use of the previous two exercises and figure out a polynomial-time algorithm that finds a 2-approximation of the problem WEIGHTED VERTEX COVER.

EXERCISE SIX

Josef K. got an exercise at his Optimization methods class:

Design an integer program for the travelling salesman problem: For a given graph with distances $G = (V, E, f)$, where $f : E \to \mathbb{R}_0^+$, find a Hamiltonian cycle with the shortest length.

He suggests the following:

“For every edge uv we have a variable $x_{uv} \in \{0, 1\}$, the target function is $\min \sum_{uv \in E} f(uv)x_{uv}$ and for every vertex u we create a condition of the form $\sum_{i|ui \in E} x_{ui} = 2$.”

Prove that Josef K. got the right solution – or prove him wrong and suggest a better one.

EXERCISE SEVEN

1. Find an integral matrix $A \in \mathbb{Z}^{n \times n}$ which has all numbers between -10 and 10, but Gaussian elimination (transformation to a triangular matrix) in some step or at the end creates an exponential number, i.e. one that is in $\Omega(2^n)$.
2. Consider an integral matrix $B \in \mathbb{Z}^{n \times n}$ with all numbers within range $[-K, K]$.
   Give an upper bound on the value of $\det(B)$. If we encode this upper bound in binary, will the memory size be polynomial, or exponential in n and K?
3. Suggest a way how to do Gaussian elimination on a matrix $C \in \mathbb{Z}^{n \times n}$ such that the time complexity stays polynomial at all times.