CS 5150/6150 Graduate (Advanced) Algorithms
Fall 2022
Assignment 3: Linear Programming
Total Points: 40

Submission notes

- Due at 11:59 pm on **Thursday, September 29, 2022**.
- Solutions must be typeset (not hand-written or scanned).
- You should strive to write clear, concise solutions to each problem, ideally fitting on a page or less. The easier your work is to read & follow, the easier it is for the TAs to award you points.
- Upload a PDF version of your completed problem set to Gradescope.
- Teaching staff reserve the right to request original source/tex files during the grading process, so please retain these until an assignment has been returned.
- Please remember that for problem sets, *collaboration with other students must be limited to a high-level discussion of solution strategies.* If you do collaborate with other students in this way, you must identify the students and describe the nature of the collaboration at the top of your homework submission. **You are not allowed to create a group solution, and all work that you hand in must be written in your own words. Do not base your solution on any other written solution, regardless of the source.**

Instructions on How to Answer Algorithm Design Questions

Most homework questions in this semester will be algorithm design questions. Please read the following instructions on how to answer them properly (you may lose points if you do not follow the instructions).

1. **Algorithm Description** You are required to clearly describe the main idea of your algorithm.

2. **Pseudocode** Providing pseudocode for your algorithm is optional. However, pseudocode is very helpful for explaining algorithms clearly, and you are *strongly encouraged* to provide pseudocode for your algorithms. Additionally, having pseudocode available often leads to more partial credit for incorrect solutions, as the TAs can more easily identify correct ideas in your solution.

   Note: copying code from an implementation (e.g. python/java/c++) as "pseudocode" does not satisfy this requirement, as grading it is difficult, and key steps may be difficult to identify. In principle, pseudocode should be relatively easy to read while keeping all relevant operations of the algorithms. *You are encouraged to do your best to write good pseudocode, following examples given in class.*

3. **Correctness** You also need to explain/argue why your algorithm is correct, and handles all necessary cases – especially when the correctness of your algorithm is not obvious.

4. **Time Analysis** You are required to analyze the time complexity of your algorithm (i.e., provide a written justification for the claimed asymptotic runtime in terms of the algorithm's operations, not just state the time complexity). You may refer to your pseudocode in this analysis, as it often simplifies the descriptors (e.g. "the `for` loop on line 2").

1. **(20 points)** My son just got a new robotics kit, and is excited to start seeing what it can do. Unfortunately, as is often the case, there's a bit of adult assembly required before he can get started. The robot's controller board has n terminal blocks, each of which is colored red (positive) or blue (negative). As they program the robot, children must connect various wiring patterns, each of which pairs a different subset of the red and blue blocks. Even though the kit was expensive, it's up to an adult to attach a wire to each terminal and fit it with a special red (or blue) wire-to-wire connector so they can easily plug and un-plug things repeatedly.

   Because of the myriad of things this robot supposedly does, it must be the case that for every pair of red/blue terminals, the two wires must be long enough so that they can be plugged together (which is true if the ends can touch before adding the connectors). Each configuration will ask the child to connect some set of pairs $(r_1, b_1), \ldots, (r_k, b_k)$ for distinct r_i and b_j (and $k \leq n/2$).

   For example: if there's a red block at point $a = (0, 0)$, and blue blocks at $b = (0, 10)$ and $c = (10, 0)$ (coordinates measured in mm from the lower left corner), then one solution is to put a wire of length 5 at all three terminals. Another solution is to put a wire of length 10 at a and length zero at the others (assume we can just attach the connector directly to the block if the wire has length 0). More generally, for any $0 \leq x \leq 10$, we can put wires of length at least $10 - x$ at a, and at least x at both b and c.

   Given the locations and colors (polarities) of the blocks, your challenge is to use as little wire as possible to satisfy the stated requirements. Specifically, find the minimum total length of wire needed so that every red block could be connected to any blue one.

   (a) [15 points] Formulate this problem as a linear program (LP). Write your LP in canonical form. (b) [5 points] Find the dual of the LP from part (a). You do not need to use canonical form (e.g. you may leave this as a minimization problem).

2. **(20 points)** Some of the algorithms for linear programming (e.g. simplex) start off with one of the corner points of the feasible set. This turns out to be tricky in general. In this problem, we will see that **in general**, finding one feasible point is as difficult as actually performing the optimization!

   Consider the following linear program (in n variables $x_1, \dots, x_n$, represented by the vector x):

$$\begin{aligned} &\text{minimize } c^T x \ \text{ subject to} \\ &a_1^T x \geq b_1 \\ &a_2^T x \geq b_2 \\ &\dots \\ &a_m^T x \geq b_m. \end{aligned}$$

   Suppose you know that the optimum value (i.e. the minimum of $c^T x$ over the feasible set) lies in the interval $[-M, M]$ for some real number M (this is typically possible in practice). Suppose also that you have an **oracle** that can take any linear program and say whether it is feasible or not. Prove that using $O(\log(M/\epsilon))$ calls to the oracle, one can determine the optimum value of the LP above up to an error of $\pm\epsilon$, for any given accuracy $\epsilon > 0$. [*Hint:* can you write a new LP that is feasible only if the LP above has optimum value $\leq z$, for some z?]