

Advanced Algorithms, Spring 2021

Homework 4. Due date: Thursday 10/5/21

The HW guidelines published in Homework 1 are valid for all the assignments in this course.

1. (30 points) Consider the linear programming problem

Maximize $x_1 + x_2$

Subject to

$ax_1 + x_2 \leq b$

$x_1, x_2 \geq 0$

- Suggest (in words) a possible real-life scenario that fits this LP.
- Find some values of a and b such that this LP has
 - a unique optimum solution
 - multiple optimal solutions
 - no feasible solution (LP is infeasible),
 - no optimal solution (LP is unbounded).

For each of the four cases:

- Illustrate each case with a figure,.
- Write down the dual LP. Clearly, all four have the same dual program when described with a and b . Write the dual program for the values selected in each case. What can you say about the dual linear program in each of the four cases (in terms of feasibility and existence/uniqueness of an optimal solution)?

2. (20 pts) In HW2 you provided a 2-approximation to the max-cut problem.

Reminder: Given an undirected graph $G=(V,E)$, a **cut** is a bipartition $(S,V-S)$ of V . The size of a cut $(S,V-S)$ is the number of edges with one endpoint on each side of the cut. The **maximum cut** problem is that of finding a cut of maximum size.

2.1 Explain why the following BIP is a formulation of the problem. For every $v \in V$, let x_v be a binary variable indicating whether $x_v \in S$. For every $e \in E$, let y_e be a binary variable indicating whether e is in the cut.

$$\max \sum_{uv \in E} y_e$$

$$\text{s.t. } y_{uv} - x_u - x_v \leq 0 \quad \text{for all } uv \in E$$

$$y_{uv} + x_u + x_v \leq 2 \quad \text{for all } uv \in E$$

$$x_v \in \{0,1\}, \quad \text{for all } v \in V$$

$$y_{uv} \in \{0,1\}, \quad \text{for all } uv \in E$$

2.2 In the LP relaxation, the integrality constraints are replaced by the constraints

$0 \leq x_v \leq 1$ for all $v \in V$, and $0 \leq y_e \leq 1$ for all $e \in E$.

Suggest a 2-approximation algorithm based on rounding, or explain why no such algorithm exists. In the first case, prove the approximation ratio and show that it is tight. In the second case, show that every graph has a useless trivial optimal fractional solution.

3. (25 pts.) Let $S = \{a_1, a_2, \dots, a_n\}$ be a finite set. Let w_i be the weight of element a_i

Let C be a collection of subsets of S . That is, $C_j \subseteq S$, for $j=1..m$.

A hitting set of C is a subset $X \subseteq S$ that contains at least one element from every subset C_j in C ($\forall D \in C \quad X \cap D \neq \emptyset$).

The minimum cost hitting set problem is the problem of finding a hitting set X of minimum weight.

1. Describe the min-cost HS problem as a BIP.
2. Describe the following instance in matrix/vector representation:
 $S = \{a_1, a_2, a_3, a_4, a_5\}$, $C_1 = \{a_2, a_3, a_4\}$, $C_2 = \{a_1, a_5\}$, $C_3 = \{a_1, a_3, a_4\}$
All elements have the same weight $w_i = 4$.
3. Describe the dual problem (of a general instance) and write its meaning.
4. Assume that for all j , $|C_j| \leq k$. Suggest a k -approximation algorithm based on rounding, or explain why no such algorithm exists. In the first case, prove the approximation ratio and show that it is tight. In the second case, show that every instance has a useless optimal fractional solution.

4. (25 pts.) Oren, our devoted HR worker would like to give special recognition awards to the workers. Every worker can select either an umbrella or a hat with the company logo. Every worker declares her preference in advance. Summing up the workers requests, it turns out that Oren needs to buy 45 umbrellas and 28 hats. The Chinese importer sells two types of packages. The first package costs 80 NIS and includes 4 umbrellas and 4 hats; the second package costs 65 NIS and includes 6 umbrellas and 3 hats. Oren must buy complete packages, and cannot use any leftovers. How many packages from each type should he buy in order to minimize his total cost and satisfy all workers' requests?

Describe the problem as an **Integer Programming** problem, and solve it using **Branch and Bound**. You can use any online LP solver in order to solve the relaxed LP problem in every stage (one possible recommended free solver is [Wolfram|Alpha](https://www.wolframalpha.com/), make sure you add 'commas' after every constraint).

Describe in a figure the B&B solution tree. For every node specify the variables' and the objective function value. Describe in words the order according to which the tree is developed.

