DS 320 Algorithms for Data Science	**Lecture #19 Worksheet**
Spring 2026	Prof. Kira Goldner and Jeffrey Considine

Linear Programming II: Algorithms, Problems, and Approximations

NP-Completeness Recap

a. Decision problems return yes or no answers.

b. P is the class of decision problems solvable in polynomial time.

c. NP is the class of decision problems where yes answers can be confirmed in polynomial time with a *certificate*.

d. NP-hard problems are decision problems to which any NP problem is polynomial reducible.

e. NP-complete problems are both NP-hard and in NP.

f. The first NP-complete problem was SAT.

g. Most later NP-complete problems are proven with reductions from SAT or previous NP-complete problems.

Integer Linear Programs are NP-Complete

Via polynomial time reduction from 3CNF to integer linear programming (ILP),

a. We will map 3CNF assignments and formulas to ILP assignments and formulas.

   (a) 3CNF variables x_i maps to ILP variable z_i.

   (b) False variable assignments map to zero ILP variable assignments.

   (c) True variable assignments map to one ILP variable assignments.

b. We will then show that there is a similar correspondance between the 3CNF formulas values and the ILP formula values.

   (a) If the 3CNF formula value is zero, then the ILP formula value is also zero.

   (b) If the 3CNF formula value is one, then the ILP formula value is at least one.

c. The mapping works as follows.

   (a) Each original 3SAT variable x_i maps a new ILP variable z_i. Add two constraints $0 \leq z_i$ and $z_i \leq 1$.

   (b) Each negated variables $\bar{x_i}$ maps to an ILP expression $(1 - z_i)$.

   (c) The jth clause becomes a linear inequality where the sum of the mapped literals must be at least one. For example, $(x_1 \vee x_2 \vee \bar{x_3})$ maps to $z_1 + z_2 + (1 - z_3) \geq 1$.

d. The objective function does not matter; any feasible solution to the ILP corresponds to a satisfying assignment to the 3CNF formula.

What Does (Non-Integer) Linear Programming Buy Us?

a. We know efficient algorithms exist (and have a nice theory behind them).

b. We can relate problems to one another through relaxations, duality.

c. It gives us techniques for approximation.

Linear Programming Algorithms

a. Simplex

b. Ellipsoid

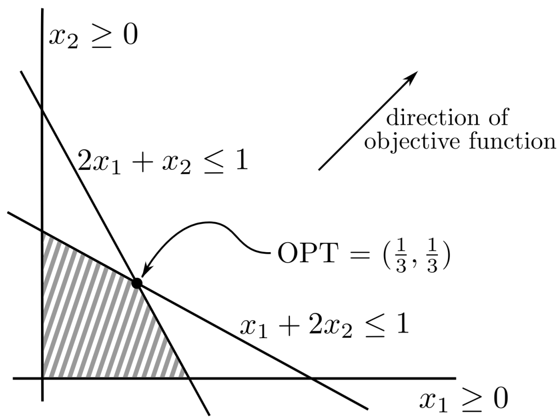


Figure 1: A toy example of a linear program.

$$
\begin{aligned}
\max \quad & x_1 + x_2 \\
\text{s.t.} \quad & x_1 \geq 0 \\
& x_2 \geq 0 \\
& 2x_1 + x_2 \leq 1 \\
& x_1 + 2x_2 \leq 1.
\end{aligned}
$$

Writing Problems We Know as Linear Programs

Independent Set

Recall from last lecture that we formulated the Independent set problem as a linear programming relaxation.

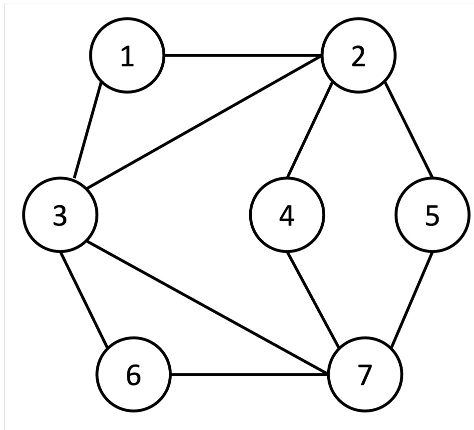
Given a graph $G = (V, E)$, each vertex i has weight w_i, find a maximum weighted *independent set*. S is an independent set if it does not contain both i and j for $(i, j) \in E$.

$$\begin{aligned}
&\max \sum_{i \in V} w_i x_i & \\
\text{s.t.} \quad & x_i + x_j \leq 1 & (i, j) \in E \\
& 0 \leq x_i \leq 1 & i \in V.
\end{aligned}$$

The Vertex Cover Problem

Given a graph $G = (V, E)$, we say that a set of nodes $S \subseteq V$ is a *vertex cover* if every edge $e = (i, j) \in E$ has at least one endpoint i or j in S. Our goal is to find a *minimum* vertex cover.

The decision version of the problem is: Given a graph G and a number k, does G contain a vertex cover of size at most k?



In this graph, the *minimum* vertex cover is

This is the same graph from last time when we discussed Independent Set. Do we notice any relationship? **Are there any implications of this?**

Vertex Cover as a Linear Program

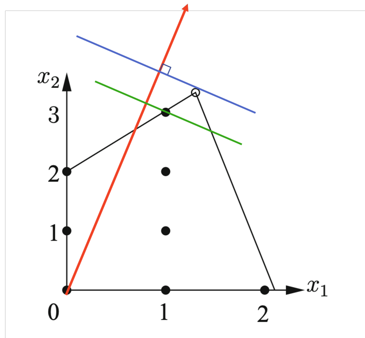
a. *Decision variables:* What are we try to solve for?

b. *Constraints:*

c. *Objective function:*

Vertex Cover as a Linear Program:

Claim 1. Let S^* denote the optimal vertex cover of minimum weight, and let x^* denote the optimal solution to the Linear Program. Then $\sum_{i \in V} w_i x_i^* \leq w(S^*)$.



$$\begin{aligned} \max \quad & 4x_1 + x_2 \\ \text{subject to} \quad & -x_1 + x_2 \leq 2 \\ & 8x_1 + 2x_2 \leq 17 \\ & x_1, x_2 \geq 0 \end{aligned}$$

Figure 2: Left: The red arrow represents the objective function, with the green line tangent to the set of feasible integer solutions, indicating the optimal integral point, and the blue line tangent to the relaxed convex feasible set, indicating the best fractional point in the relaxation, with a larger objective function. Right: The linear program for the figure on the left.

Using Linear Programming for a Vertex Cover Approximation

$$\begin{aligned} \min \sum_{i \in V} & w_i x_i & \\ \text{s.t.} \quad & x_i + x_j \geq 1 & (i,j) \in E \\ & x_i \in [0,1] & i \in V. \end{aligned}$$

Claim 2. Let S^* denote the optimal vertex cover of minimum weight, and let x^* denote the optimal solution to the Linear Program. Then $\sum_{i \in V} w_i x_i^* \leq w(S^*)$.

Proof. The vertex cover problem is equivalent to the integer program, whereas the linear program is a *relaxation*. Then there are simply more solutions allowed to the linear program, so the minimum can only be smaller. □

Claim 3. The set $S = \{i : x_i \geq 0.5\}$ is a vertex cover, and $w(S) \leq 2 \sum_{i \in V} w_i x_i^*$.