Approximation Algorithms

Frank

December 12, 2021

1 Vertex Cover

1.1 Approximation Algorithms

- Algorithms that
  - Run efficiently (poly time)
  - Find solution that is guaranteed to be "close enough" to optimal
- Close to optimal means: Within a constant factor $c > 1$ of optimal

  Note that $c = 1$ gives optimal solution

  c is the "approximation ratio"
- Minimization problem: solution $\leq c \times$ optimal solution
- Maximization problem: solution $\geq \frac{1}{c} \times$ optimal solution

1.2 Polytime Approximation Scheme

- Even better: arbitrarily close to optimal with parameter $\epsilon > 0$ and approximation ratio of $1 + \epsilon$
  - minimization: solution $\leq (1 + \epsilon) \times$ optimal
  - maximization: solution $\geq \frac{1}{(1+\epsilon)} \times$ optimal
- Running time depends on ϵ

  ex. $O(n^2(\frac{1}{\epsilon})^3)$

  Less desirably: $O(n^2 \cdot 2^{\frac{1}{\epsilon}})$

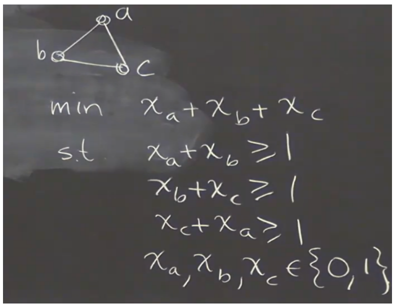
1.3 Problem

- **Input**: $G = (V, E)$
- **Output**: $\hat{V} \subseteq V$ s.t. $\forall\{u, v\} \in E$, $u \in \hat{V}$ or $v \in \hat{V}$

1.4 Algorithm 1 (LP rounding/LP relaxation)

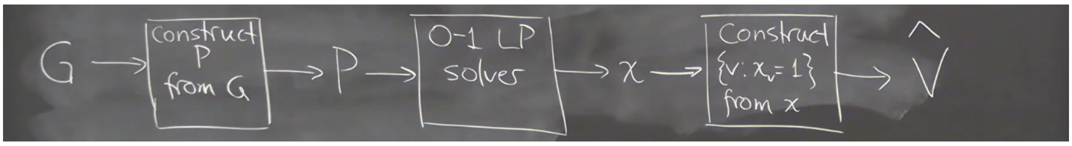
1.4.1 Introduction

- Recall: Reduction of VC problem to 0-1 LP.
- From $G = (V, E)$, define 0-1 LP $P =$
  - **variables**: $\forall v \in V$, x_v
  - **Objective Function**: minimize $\sum_{v \in V} x_v$
  - **Constraints**: $\forall \{u, v\} \in E$: $x_u + x_v \geq 1$ and $\forall v$, $x_v \in \{0, 1\}$
- Example:



- And we have that $x = (x_v : v \in V)$ is a feasible solution to $P \iff \{v : x_v = 1\}$ is a v.c.

  So x is optimal solution to $P \iff \{v : x_v = 1\}$ is <u>minimum</u> v.c.
- reduction is as follows:



  Note that constructing P from G takes poly time, and constructing the v.c. from x takes poly time as well. However, we do not know if 0-1 solver is poly time.
- So let us replace the 0-1 constraint with a linear constraint:

  $$\forall v, x_v \in \{0, 1\} \to x_v \geq 0, x_v \leq 1$$

  and let's call the new LP P', which is a normal LP, solvable efficiently by LP algorithm.
- Then it is not hard to see that the optimal solution to our original 3 vertices problem is $x^\star = (\frac{1}{2}, \frac{1}{2}, \frac{1}{2})$ under the relaxed constraints

1.4.2 Algorithm

- Solve LP P' to obtain optimal solution $x^\star = (x_v : v \in V)$
- Define

$$\hat{x}_v = \begin{cases} 1 & x_v^\star \geq \frac{1}{2} \\ 0 & x_v^\star < \frac{1}{2} \end{cases} \quad \text{(rounding of } x^\star\text{)}$$

- In the example before, $\hat{x} = (1, 1, 1)$ by rounding.
- In general, this rounding approach is at most 2 times worse than the optimal solution

1.5 Correctness/Error Margin

- Is $\hat{x}$ a feasible solution to P (i.e. is $\hat{V} = \{v : \hat{x}_v = 1\}$ a v.c. ?)
  - Yes, if $\{u, v\} \in E$ then $u \in \hat{V}$ or $v \in \hat{V}$
    Since we must have $x_u^\star + x_v^\star \geq 1$ by our relaxed constraint $\implies x_u^\star \geq \frac{1}{2}$ or $x_v^\star \geq \frac{1}{2}$ $\implies \hat{x}_u = 1$ or $\hat{x}_v = 1 \implies u \in \hat{V}$ or $v \in \hat{V}$
- How much bigger is the v.c. obtained this way compared to minimum v.c.
  - $|\hat{V}| \leq 2 \cdot$ optimal
  - Let $\bar{x} =$ optimal solution to 0-1 LP P, then optimal is $\sum_{v \in V} \bar{x}_v$.
  - Observation 1:

    $\sum_{v \in V} x_v^\star \leq \sum_{v \in V} \bar{x}_v$, that the optimal solution to P' has a more relaxed constraint, and therefore has more possibilities.

    More rigorously, feasible solutions to $P \subseteq$ feasible solutions to P'
  - Observation 2:

    $\forall v \in V\ \hat{x}_v \leq 2x_v^\star$ by definition of rounding

  - Thus, the size of v.c. $\hat{V}$ found by algo is equal to $\sum_{v \in V} \hat{x}_v$, and consider

$$\begin{aligned} \sum_{v \in V} \hat{x}_v &\leq 2 \cdot \sum_{v \in V} x_v^\star \quad \text{[by observation 2]} \\ &\leq 2 \cdot \sum_{v \in V} \bar{x}_v \quad \text{[by observation 1]} \\ &= 2 \cdot \text{optimal} \end{aligned}$$

- Thus, we have a polynomial time algorithm, because the LP solver is polynomial, and also the additional work we did including round are also polynomial.
- But can we do better? Like something with approximation ratio closer to 1 or a polytime approximation scheme. Probably not, unless P=NP. However, $c < 1.361\ldots$ is achievable.