# 6.1220 LECTURE 16

BRENDON JIANG

## Contents



## 1. Approximation Algorithms

In the previous lecture we studied decision problems, now we move to optimization problems.

**Definition 1.1: $\alpha$ Approximation**

$\mathcal{A}$ is an $\alpha$-approximation for $\mathcal{P}$(minimization) if and only if for all $P \in \mathcal{P}$, $\dfrac{\mathcal{A}(P)}{\mathrm{OPT}(P)} \leq \alpha$. For maximization problems, we define $\mathcal{A}$ as an $\alpha-$approximation for $\mathcal{P}$(maximization) if and only if for all $P \in \mathcal{P}$, $\dfrac{\mathrm{OPT}(P)}{\mathcal{A}(P)} \leq \alpha$.

### 1.1. Maximum Matching

We can find the maximum matching using flow for polynomial time. We studied this case for bipartite graphs, but the solution can be extended for any graph. Thus, there is a $1-$approximation algorithm if we just take the problem's algorithmic solution. However, we look at another simple approximation algorithm.

**Example 1.2: Greedy Algorithm**

Given $G = (V, E)$, we iterate through $e \in E$ and add $e$ if it does not violate our current matching $M$. This is both a greedy and polynomial time algorithm. We now want to show that this algorithm is a $2-$approximation algorithm

*Date*: November 2024

**Claim 1.3**

Suppose $|M^*|$ is the optimal matching for a problem instance. We claim that $|M^*| \leq 2 \cdot |M|$.

**Proof**

Let $M^*$ be a maximum matching. Label every edge $e \in M$ by $e^* \in M^*$ if $e$ "blocks $e^*$, or in other words $e$ and $e^*$ share an endpoint. Note here that each edge in $M$ can only block two edges in $M^*$. Since $M$ is greedy, every edge in $M^*$ must also be blocked by some edge in $M$. Thus, we have $|M^*| \leq \sum \#$ of labels $\leq 2|M|$, and we are done.

## 1.2. Vertex Cover

As a reminder, the vertex cover is a subset of the vertices such that all edges touch a vertex in the vertex cover. The hard problem here is minimizing the size of the vertex cover. Again, there is quite a simple 2-approximation algorithm.

**Example 1.4**

Given $G$, find a maximal matching $M$. Output all endpoints of the edge in $M$. Here, maximal matching means the greedy matching algorithm.

**Claim 1.5: Claim 1**

$S$ is a vertex cover.

**Proof**

Proof by contradiction. Suppose that an edge $e = (u, v)$ is not covered. This means neither $u$ or $v$ belongs to $S$. But this means $u$ and $v$ are not endpoints of any edge in $M$, so we can add $e$ greedily to $M$, which is a contradiction.

**Claim 1.6: Claim 2**

$|S| \leq 2|S^*|, S^*$ is a minimum vertex cover.

**Proof**

If we simply consider the edges in $M$, we need at least $|M|$ vertex to cover the edges, because they are disjoint. This means $|M| \leq |S^*|$. By construction, we know that $|S| = 2|M|$, so $|S| \leq 2|S^*|$.

## 1.3. Weighted Vertex Cover

Now the vertices have weights, and we want to minimize the minimum weight vertex cover. This is a more general version of the original vertex cover problem, but we can still design a 2−approximation algorithm. We will use a new technique called LP rounding.

First we construct the LP. For each vertex, construct an indicator variable $x_v \in \{0, 1\}$ for whether if the vertex is in the weighted vertex cover. The constraints are then $x_u + x_v \geq 1$ for each $(u, v) \in E$, and we want to minimize the objective function $\sum_{v \in V} x_v w(v)$. Since this is a binary integer linear program, it is NP-hard. But we can relax the binary constraint, and instead of having $x_v \in \{0, 1\}$ we use $0 \leq x_v \leq 1$. Now, we get the LP $\mathcal{P}$

$$
\begin{aligned}
\min \sum_{v \in V} & x_v w(v) \\
& x_u + x_v \geq 1 \ \forall (u, v) \in E \\
& 0 \leq x_v \leq 1 \ \forall v \in V
\end{aligned}
$$

Now suppose I solve $\mathcal{P}$ to get $x_v^*$ for $v \in V$. Now, round $x_v^*$ to $x_v = 0$ if $x_v^* < \frac{1}{2}$ and $x_v = 1$ if $x_v^* \geq \frac{1}{2}$. Lastly, output all the vertices with $x_v = 1$.

**Claim 1.7**

$S$ is a vertex cover.

**Proof**

Since $x_u + x_v \geq 1$ for all edges, at least one of $x_u, x_v$ is rounded up to 1.

**Claim 1.8**

$\sum_{v \in V} x_v w(v) \leq 2 \sum_{v \in V} x_v^* w(v)$.

**Proof**

$x_v \leq 2x_v^*$.

**Claim 1.9**

$\sum_{v \in V} w(v) \leq 2\text{OPT}(IP)$.

**Proof**

$\sum_{v \in V} w(v) \leq 2\text{OPT}(\mathcal{P})$ by claim 2. Since the problem $\mathcal{P}$ is more general than $IP$, the optimal minimum must be less. Thus, $\sum_{v \in V} w(v) \leq 2\text{OPT}(\mathcal{P}) \leq 2\text{OPT}(IP)$ and we are done.

## 1.4. MAX-3-SAT

Given a $3 - CNF$ formula F, find an assignment that satisfies most clauses. We will use randomization. Assign $x_i$ to be $T/F$ with one half probability. There is a 7/8th chance each clause is satisfied. $\Pr[\text{clause is satisfied}] = \frac{7}{8}$. The expected number of satisfied clauses is then $E[\#\text{ satisfied clauses}] = \frac{7m}{8}$. $OPT \leq m, \mathcal{A}(P) \geq \frac{7}{8}\text{OPT}$.