Set-Cover approximation through LP-Rounding

K. Subramani

1 Lane Department of Computer Science and Electrical Engineering
West Virginia University

April 1, 2014
Outline

1 Preliminaries
Outline

1 Preliminaries

2 A Simple Rounding Algorithm
Outline

1. Preliminaries
2. A Simple Rounding Algorithm
3. A Randomized Rounding Algorithm
Outline

1 Preliminaries
2 A Simple Rounding Algorithm
3 A Randomized Rounding Algorithm
4 Half-integrality of Vertex Cover
The Set Cover Problem

Given,

1. A ground set \( U = \{ e_1, e_2, \ldots, e_n \} \),
2. A collection of sets \( S = \{ S_1, S_2, \ldots, S_m \} \), where \( S_i \subseteq U \), \( i = 1, 2, \ldots, m \),
3. A weight function \( c: S_i \rightarrow \mathbb{Z}^+ \),

find a collection of subsets \( S_i \), whose union covers the elements of \( U \) at minimum cost.

Note: If all weights are unity (or the same), the problem is called the Cardinality Set Cover problem.
The Set Cover Problem

Given,
The Set Cover Problem

Given,
The Set Cover Problem

Given,

1. A ground set $U = \{e_1, e_2, \ldots, e_n\}$,
The Set Cover Problem

Given,

1. A ground set \( U = \{e_1, e_2, \ldots, e_n\} \),
2. A collection of sets \( S_P = \{S_1, S_2, \ldots S_m\} \), \( S_i \subseteq U, i = 1, 2, \ldots, m \)
The Set Cover Problem

Given,

1. A ground set $U = \{e_1, e_2, \ldots, e_n\}$,
2. A collection of sets $S_P = \{S_1, S_2, \ldots S_m\}$, $S_i \subseteq U$, $i = 1, 2, \ldots, m$
3. A weight function $c : S_i \rightarrow \mathbb{Z}_+$,
Preliminaries

The Set Cover Problem

Given,

1. A ground set \( U = \{e_1, e_2, \ldots, e_n\} \),
2. A collection of sets \( S_P = \{S_1, S_2, \ldots, S_m\} \), \( S_i \subseteq U \), \( i = 1, 2, \ldots, m \)
3. A weight function \( c : S_i \rightarrow \mathbb{Z}_+ \),

find a collection of subsets \( S_i \), whose union covers the elements of \( U \) at minimum cost.
The Set Cover Problem

Given,

1. A ground set \( U = \{e_1, e_2, \ldots, e_n\} \),
2. A collection of sets \( S_P = \{S_1, S_2, \ldots S_m\} \), \( S_i \subseteq U \), \( i = 1, 2, \ldots, m \)
3. A weight function \( c : S_i \rightarrow \mathbb{Z}_+ \),

find a collection of subsets \( S_i \), whose union covers the elements of \( U \) at minimum cost.

Note

*If all weights are unity (or the same)*,
The Set Cover Problem

Given,

1. A ground set $U = \{e_1, e_2, \ldots, e_n\}$,
2. A collection of sets $S_P = \{S_1, S_2, \ldots S_m\}$, $S_i \subseteq U$, $i = 1, 2, \ldots, m$
3. A weight function $c : S_i \rightarrow \mathbb{Z}_+$

find a collection of subsets $S_i$, whose union covers the elements of $U$ at minimum cost.

Note

If all weights are unity (or the same), the problem is called the Cardinality Set Cover problem.
Formulating the Integer Program

**IP formulation**

\[
\min \sum_{S \in S} P \cdot x_S
\]

subject to

\[
\sum_{S: e \in S} x_S \geq 1, \quad e \in U, \quad x_S \in \{0, 1\}, \quad S \in S_P
\]
Formulating the Integer Program

IP formulation

\[ \min \sum_{S \in \mathcal{P}} c(S) \cdot x_S \]
Formulating the Integer Program

\[
\begin{align*}
\text{min} & \quad \sum_{S \in \mathcal{P}} c(S) \cdot x_S \\
\text{subject to} & \quad \sum_{S : e \in S} x_S \geq 1, \quad e \in U
\end{align*}
\]
Formulating the Integer Program

**IP formulation**

\[
\min \sum_{S \in S_P} c(S) \cdot x_S \\
\text{subject to} \quad \sum_{S: e \in S} x_S \geq 1, \quad e \in U \\
x_S \in \{0, 1\}, \quad S \in S_P
\]
The Linear Program relaxation

\[
\begin{align*}
\min & \quad \sum_{S \in \mathcal{S}} P_c(S) \cdot x_S \\
\text{subject to} & \quad \sum_{S : e \in S} x_S \geq 1, \\
& \quad x_S \geq 0, \\
& \quad S \in \mathcal{P}
\end{align*}
\]
The Linear Program relaxation

\[
\begin{align*}
\min & \quad \sum_{S \in \mathcal{S}} P_c(S) \cdot x_S \\
\text{subject to} & \quad \sum_{S \in \mathcal{S}} e \in S x_S \geq 1, \\
& \quad x_S \geq 0, \\
& \quad S \in \mathcal{P}
\end{align*}
\]
The Linear Program relaxation

\[ \min \sum_{S \in \mathcal{P}} c(S) \cdot x_S \]
The Linear Program relaxation

\[
\min \sum_{S \in \mathcal{P}} c(S) \cdot x_S \\
\text{subject to } \sum_{S : e \in S} x_S \geq 1, \quad e \in U
\]
The Linear Program relaxation

\[
\begin{align*}
\text{min} & \quad \sum_{S \in S_P} c(S) \cdot x_S \\
\text{subject to} & \quad \sum_{S: e \in S} x_S \geq 1, \quad e \in U \\
& \quad x_S \geq 0, \quad S \in S_P
\end{align*}
\]
Simple rounding

1. Find an optimal solution to the LP relaxation.
2. Let $f$ denote the frequency of the most frequent element.
3. Pick all sets $S$ for which $x_S \geq \frac{1}{f}$ in this solution.

Lemma
The above algorithm achieves an approximation factor of $f$ for the set cover problem.
Simple rounding

Rounding Algorithm

1. Find an optimal solution to the LP relaxation.
2. Let $f$ denote the frequency of the most frequent element.
3. Pick all sets $S$ for which $x_S \geq \frac{1}{f}$ in this solution.

Lemma
The above algorithm achieves an approximation factor of $f$ for the set cover problem.
Simple rounding

Rounding Algorithm

1. Find an optimal solution to the LP relaxation.
A Simple Rounding Algorithm

1. Find an optimal solution to the LP relaxation.
2. Let $f$ denote the frequency of the most frequent element.
Simple rounding

**Rounding Algorithm**

1. Find an optimal solution to the LP relaxation.
2. Let $f$ denote the frequency of the most frequent element.
3. Pick all sets $S$ for which $\chi_S \geq \frac{1}{f}$ in this solution.
Simple rounding

Rounding Algorithm

1. Find an optimal solution to the LP relaxation.
2. Let $f$ denote the frequency of the most frequent element.
3. Pick all sets $S$ for which $\chi_S \geq \frac{1}{f}$ in this solution.

Lemma

The above algorithm achieves an approximation factor of $f$ for the set cover problem.
Proof.

Let $C$ denote the collection of sets picked by the algorithm.

Focus on an arbitrary element $e \in U$. Assume it belongs to the sets $S_1, S_2, \ldots, S_r$, where $r \leq f$.

Since $\sum_r x_j = 1 \cdot x_j \geq 1$, at least one of the $x_j \geq 1 \cdot r \geq 1 \cdot f$.

Thus, the corresponding set will be picked and $e$ will be covered, i.e., $C$ is a valid cover.

The rounding process increases $x_S$ for each $S$ by at most a factor of $f$.

Thus, the cost of $C$ is at most $f$ times the cost of the optimal fractional cover and hence at most $f$ times the cost of the optimal integer cover!
Proof.

Let $C$ denote the collection of sets picked by the algorithm. Focus an arbitrary element $e \in U$. Assume it belongs to the sets $S_1, S_2, ..., S_r$, where $r \leq f$. Since $\sum r_j = 1 \cdot x_j \geq 1$, at least one of the $x_j \geq 1$ and $r \geq 1$. Thus, the corresponding set will be picked and $e$ will be covered, i.e., $C$ is a valid cover.

The rounding process increases $x_S$ for each $S$ by at most a factor of $f$. Thus, the cost of $C$ is at most $f$ times the cost of the optimal fractional cover and hence at most $f$ times the cost of the optimal integer cover!
Proof.

Let $C$ denote the collection of sets picked by the algorithm.
Proof.

1. Let $C$ denote the collection of sets picked by the algorithm.
2. Focus an arbitrary element $e \in U$. Assume it belong to the sets $S_1, S_2, \ldots S_r$, where $r \leq f$. 

Since $\sum_{j=1}^{r} x_j \geq 1$, at least one of the $x_j \geq \frac{1}{r} \geq \frac{1}{f}$. 

Thus, the corresponding set will be picked and $e$ will be covered, i.e., $C$ is a valid cover.

The rounding process increases $x_S$ for each $S$ by at most a factor of $f$. 

Thus, the cost of $C$ is at most $f$ times the cost of the optimal fractional cover and hence at most $f$ times the cost of the optimal integer cover!
Analysis

Proof.

1. Let $C$ denote the collection of sets picked by the algorithm.
2. Focus an arbitrary element $e \in U$. Assume it belong to the sets $S_1, S_2, \ldots S_r$, where $r \leq f$.
3. Since $\sum_{j=1}^{r} x_j \geq 1$, 

Proof.

1. Let $C$ denote the collection of sets picked by the algorithm.
2. Focus an arbitrary element $e \in U$. Assume it belong to the sets $S_1, S_2, \ldots S_r$, where $r \leq f$.
3. Since $\sum_{j=1}^{r} x_j \geq 1$, at least one of the $x_j \geq \frac{1}{r}$.
Proof.

1. Let $C$ denote the collection of sets picked by the algorithm.
2. Focus an arbitrary element $e \in U$. Assume it belong to the sets $S_1, S_2, \ldots S_r$, where $r \leq f$.
3. Since $\sum_{j=1}^{r} x_j \geq 1$, at least one of the $x_j \geq \frac{1}{r} \geq \frac{1}{f}$.
Proof.

1. Let $C$ denote the collection of sets picked by the algorithm.
2. Focus an arbitrary element $e \in U$. Assume it belong to the sets $S_1, S_2, \ldots S_r$, where $r \leq f$.
3. Since $\sum_{j=1}^{r} x_j \geq 1$, at least one of the $x_j \geq \frac{1}{r} \geq \frac{1}{f}$.
4. Thus, the corresponding set will be picked and $e$ will be covered, i.e., $C$ is a valid cover.
Analysis

Proof.

1. Let $C$ denote the collection of sets picked by the algorithm.
2. Focus an arbitrary element $e \in U$. Assume it belong to the sets $S_1, S_2, \ldots S_r$, where $r \leq f$.
3. Since $\sum_{j=1}^{r} x_j \geq 1$, at least one of the $x_j \geq \frac{1}{r} \geq \frac{1}{f}$.
4. Thus, the corresponding set will be picked and $e$ will be covered, i.e., $C$ is a valid cover.
5. The rounding process increases $x_S$ for each $S$ by at most a factor of $f$. 
Proof.

1. Let $C$ denote the collection of sets picked by the algorithm.
2. Focus an arbitrary element $e \in U$. Assume it belong to the sets $S_1, S_2, \ldots, S_r$, where $r \leq f$.
3. Since $\sum_{j=1}^{r} x_j \geq 1$, at least one of the $x_j \geq \frac{1}{r} \geq \frac{1}{f}$.
4. Thus, the corresponding set will be picked and $e$ will be covered, i.e., $C$ is a valid cover.
5. The rounding process increases $x_S$ for each $S$ by at most a factor of $f$.
6. Thus, the cost of $C$ is at most $f$ times the cost of the optimal fractional cover.
Proof.

1. Let $C$ denote the collection of sets picked by the algorithm.

2. Focus an arbitrary element $e \in U$. Assume it belong to the sets $S_1, S_2, \ldots S_r$, where $r \leq f$.

3. Since $\sum_{j=1}^{r} x_j \geq 1$, at least one of the $x_j \geq \frac{1}{r} \geq \frac{1}{f}$.

4. Thus, the corresponding set will be picked and $e$ will be covered, i.e., $C$ is a valid cover.

5. The rounding process increases $x_S$ for each $S$ by at most a factor of $f$.

6. Thus, the cost of $C$ is at most $f$ times the cost of the optimal fractional cover and hence at most $f$ times the cost of the optimal integer cover!
A Randomized Rounding Algorithm

Solve the LP relaxation optimally. Let $x$ denote the optimal fractional solution.

Set probability vector $p = x$.

Round each $x_S$ to 1 by flipping a coin with "head" bias $p_S$. If the coin turns up heads, set $x_S$ to 1. Otherwise, set $x_S$ to 0.

Output all sets $S$, such that $x_S = 1$. 
Solve the LP relaxation optimally. Let $x$ denote the optimal fractional solution.

Set probability vector $p = x$.

Round each $x_S$ to 1 by flipping a coin with "head" bias $p_S$. If the coin turns up heads, set $x_S$ to 1. Otherwise, set $x_S$ to 0.

Output all sets $S$, such that $x_S = 1$. 

A Randomized Rounding Algorithm
Randomized Approach

1. Solve the LP relaxation optimally. Let $x$ denote the optimal fractional solution.
A Randomized Rounding Algorithm

Randomized Approach

1. Solve the LP relaxation optimally. Let $x$ denote the optimal fractional solution.
2. Set probability vector $p = x$. 
Randomized Approach

1. Solve the LP relaxation optimally. Let $\mathbf{x}$ denote the optimal fractional solution.
2. Set probability vector $\mathbf{p} = \mathbf{x}$.
3. Round each $x_S$ to 1 by flipping a coin with "head" bias $p_S$. 
Randomized Approach

1. Solve the LP relaxation optimally. Let $\mathbf{x}$ denote the optimal fractional solution.
2. Set probability vector $\mathbf{p} = \mathbf{x}$.
3. Round each $x_S$ to 1 by flipping a coin with "head" bias $p_S$. If the coin turns up heads, set $x_S$ to 1.
A Randomized Rounding Algorithm

Randomized Approach

1. Solve the LP relaxation optimally. Let $x$ denote the optimal fractional solution.
2. Set probability vector $p = x$.
3. Round each $x_S$ to 1 by flipping a coin with "head" bias $p_S$. If the coin turns up heads, set $x_S$ to 1. Otherwise, set $x_S$ to 0.
A Randomized Rounding Algorithm

Randomized Approach

1. Solve the LP relaxation optimally. Let $\mathbf{x}$ denote the optimal fractional solution.
2. Set probability vector $\mathbf{p} = \mathbf{x}$.
3. Round each $x_S$ to 1 by flipping a coin with "head" bias $p_S$. If the coin turns up heads, set $x_S$ to 1. Otherwise, set $x_S$ to 0.
4. Output all sets $S$, such that $x_S = 1$. 
Analysis
Approximation guarantee

\[ \text{cost}(C) = \sum_{S \in S} \Pr[\text{S is picked}] \cdot c_S = \sum_{S \in S} \Pr[p_S] \cdot c_S = \text{OPT} \]
Analysis

Approximation guarantee

\[ \mathbf{E}[\text{cost}(C)] = \]
### Approximation guarantee

\[
\mathbb{E}[\text{cost}(C)] = \sum_{S \in S_P} \Pr[\text{S is picked}] \cdot c_S
\]
Analysis

Approximation guarantee

\[ E[\text{cost}(C)] = \sum_{S \in S_P} \Pr[S \text{ is picked}] \cdot c_S \]

\[ = \sum_{S \in S_P} p_S \cdot c_S \]
Approximation guarantee

$$E[\text{cost}(C)] = \sum_{S \in S_P} \Pr[ S \text{ is picked }] \cdot c_S$$

$$= \sum_{S \in S_P} p_S \cdot c_S$$

$$= OPT_f$$
Facts from calculus and probability

1. \((1 - \frac{1}{k})^k \leq 1/e\), for all \(k = 1, 2, \ldots, \infty\).

2. The function \(\Pi_k = \prod_{i=1}^{k} \left(1 - p_i\right)\), subject to \(\sum_{i=1}^{k} p_i = 1\) and \(0 \leq p_i \leq 1\), is maximized at \(p_i = \frac{1}{k}\) for all \(i = 1, 2, \ldots, k\).

3. \(\Pr(E_1 \cup E_2) \leq \Pr(E_1) + \Pr(E_2)\).

4. If \(X\) is a non-negative random variable and \(a > 0\) is a positive constant, then \(\Pr\left[X \geq a \cdot \mathbb{E}[X]\right] \leq \frac{1}{a}\). (Markov's inequality!)
Facts from calculus and probability

1. \((1 - \frac{1}{k})^k \leq \frac{1}{e}\), for all \(k = 1, 2, \ldots\).

2. The function \(\Pi_k \prod_{i=1}^{k} (1 - p_i)\), subject to \(\sum_{i=1}^{k} p_i = 1\), \(0 \leq p_i \leq 1\), \(i = 1, 2, \ldots, k\), is maximized at \(p_i = \frac{1}{k}\) for all \(i = 1, 2, \ldots, k\).

3. \(\Pr(E_1 \cup E_2) \leq \Pr(E_1) + \Pr(E_2)\).

4. If \(X\) is a non-negative random variable and \(a > 0\) is a positive constant, then \(\Pr[X \geq a \cdot \mathbb{E}[X]] \leq \frac{1}{a}\). (Markov's inequality!)
Facts from calculus and probability

Elementary facts

1. $(1 - \frac{1}{k})^k \leq \frac{1}{e}$, for all $k = 1, 2, \ldots\infty$. 
Facts from calculus and probability

Elementary facts

1. \((1 - \frac{1}{k})^k \leq \frac{1}{e}\), for all \(k = 1, 2, \ldots \infty\).

2. The function \(\prod_{i=1}^{k} (1 - p_i)\), subject to \(\sum_{i=1}^{k} p_i \geq 1, 0 \leq p_i \leq 1, i = 1, 2, \ldots k\), is maximized at \(p_i = \frac{1}{k}\) for all \(i = 1, 2, \ldots k\).
Facts from calculus and probability

Elementary facts

1. \((1 - \frac{1}{k})^k \leq \frac{1}{e}\), for all \(k = 1, 2, \ldots \infty\).

2. The function \(\Pi_{i=1}^{k} (1 - p_i)\), subject to \(\sum_{i=1}^{k} p_i \geq 1, 0 \leq p_i \leq 1, i = 1, 2, \ldots k\), is maximized at \(p_i = \frac{1}{k}\) for all \(i = 1, 2, \ldots k\).

3. \(\Pr(E_1 \cup E_2) \leq \Pr(E_1) + \Pr(E_2)\).
Facts from calculus and probability

**Elementary facts**

1. \((1 - \frac{1}{k})^k \leq \frac{1}{e}\), for all \(k = 1, 2, \ldots \infty\).

2. The function \(\prod_{i=1}^{k} (1 - p_i)\), subject to \(\sum_{i=1}^{k} p_i \geq 1, 0 \leq p_i \leq 1, i = 1, 2, \ldots k\), is maximized at \(p_i = \frac{1}{k}\) for all \(i = 1, 2, \ldots k\).

3. \(\Pr(E_1 \cup E_2) \leq \Pr(E_1) + \Pr(E_2)\).

4. If \(X\) is a non-negative random variable and \(a > 0\) is a positive constant, then
Facts from calculus and probability

**Elementary facts**

1. \((1 - \frac{1}{k})^k \leq \frac{1}{e}\), for all \(k = 1, 2, \ldots, \infty\).

2. The function \(\prod_{i=1}^{k} (1 - p_i)\), subject to \(\sum_{i=1}^{k} p_i \geq 1, 0 \leq p_i \leq 1, i = 1, 2, \ldots, k\), is maximized at \(p_i = \frac{1}{k}\) for all \(i = 1, 2, \ldots, k\).

3. \(\Pr(E_1 \cup E_2) \leq \Pr(E_1) + \Pr(E_2)\).

4. If \(X\) is a non-negative random variable and \(a > 0\) is a positive constant, then \(\Pr[X \geq a \cdot \mathbb{E}[X]] \leq \frac{1}{a}\).
Facts from calculus and probability

**Elementary facts**

1. \((1 - \frac{1}{k})^k \leq \frac{1}{e}\), for all \(k = 1, 2, \ldots \infty\).

2. The function \(\Pi_{i=1}^k (1 - p_i)\), subject to \(\sum_{i=1}^k p_i \geq 1\), \(0 \leq p_i \leq 1\), \(i = 1, 2, \ldots k\), is maximized at \(p_i = \frac{1}{k}\) for all \(i = 1, 2, \ldots k\).

3. \(\Pr(E_1 \cup E_2) \leq \Pr(E_1) + \Pr(E_2)\).

4. If \(X\) is a non-negative random variable and \(a > 0\) is a positive constant, then \(\Pr[X \geq a \cdot E[X]] \leq \frac{1}{a}\). (Markov’s inequality!)
Feasibility Analysis

1. Pick an arbitrary element $a \in U$. We will study the probability that it is covered in the set cover that is output by the randomized algorithm discussed above.

2. W.l.o.g. assume that $a \in S_1, S_2, ..., S_k$.

3. Let $x_1 = p_1, x_2 = p_2, ..., x_k = p_k$.

4. Since $a$ is fractionally covered, $\sum_{i=1}^{k} x_i = \sum_{i=1}^{k} p_i = 1 \geq 1$.

5. The probability that $a$ is not covered by set $S_i$ is $(1 - p_i)$.

6. The probability that $a$ is not covered by any of the $S_i, i = 1, 2, ..., k$ is $\prod_{i=1}^{k} (1 - p_i)$.

7. Thus, the probability that $a$ is not covered by any of the sets is at most $(1 - \frac{1}{e})^k \leq \frac{1}{e}$.

8. Thus, the probability that $a$ is covered by some set in the cover is at least $1 - \frac{1}{e}$. 

Pick an arbitrary element \( a \in U \).
Feasibility Analysis

**Feasibility**

Pick an arbitrary element $a \in U$. We will study the probability that it is covered in the set cover that is output by the randomized algorithm discussed above.
Feasibility Analysis

Feasibility

1. Pick an arbitrary element $a \in U$. We will study the probability that it is covered in the set cover that is output by the randomized algorithm discussed above.

2. W.l.o.g. assume that $a \in S_1, S_2, \ldots S_k$. 


Feasibility Analysis

### Feasibility

1. Pick an arbitrary element $a \in U$. We will study the probability that it is covered in the set cover that is output by the randomized algorithm discussed above.

2. W.l.o.g. assume that $a \in S_1, S_2, \ldots S_k$.

3. Let $x_1 = p_1$, $x_2 = p_2$, $\ldots$, $x_k = p_k$. 

Since $a$ is fractionally covered, $\sum_{i=1}^{k} p_i = 1$. The probability that $a$ is not covered by set $S_i$ is $(1 - p_i)$. The probability that $a$ is not covered by any of the $S_i$, $i = 1, 2, \ldots, k$ is $\prod_{i=1}^{k} (1 - p_i)$. Thus, the probability that $a$ is not covered by any of the sets is at most $(1 - \frac{1}{e^k}) \leq 1 - e$. Therefore, the probability that $a$ is covered by some set in the cover is at least $(1 - (1 - \frac{1}{e})) = 1 - e$. 

---

**Note:** The approach in this section provides a feasibility analysis for the randomized rounding algorithm, focusing on the probability of covering an arbitrary element in the set cover.
Feasibility Analysis

**Feasibility**

1. Pick an arbitrary element $a \in U$. We will study the probability that it is covered in the set cover that is output by the randomized algorithm discussed above.

2. W.l.o.g. assume that $a \in S_1, S_2, \ldots, S_k$.

3. Let $x_1 = p_1, x_2 = p_2, \ldots, x_k = p_k$.

4. Since $a$ is fractionally covered, $\sum_{i=1}^{k} p_i$
Feasibility Analysis

Feasibility

1. Pick an arbitrary element $a \in U$. We will study the probability that it is covered in the set cover that is output by the randomized algorithm discussed above.

2. W.l.o.g. assume that $a \in S_1, S_2, \ldots S_k$.

3. Let $x_1 = p_1, x_2 = p_2, \ldots, x_k = p_k$.

4. Since $a$ is fractionally covered, $\sum_{i=1}^{k} p_i \geq 1$. 
Feasibility Analysis

Feasibility

1. Pick an arbitrary element $a \in U$. We will study the probability that it is covered in the set cover that is output by the randomized algorithm discussed above.
2. W.l.o.g. assume that $a \in S_1, S_2, \ldots S_k$.
3. Let $x_1 = p_1, x_2 = p_2, \ldots, x_k = p_k$.
4. Since $a$ is fractionally covered, $\sum_{i=1}^{k} p_i \geq 1$.
5. The probability that $a$ is not covered by set $S_i$ is...
Feasibility Analysis

Feasibility

1. Pick an arbitrary element \( a \in U \). We will study the probability that it is covered in the set cover that is output by the randomized algorithm discussed above.

2. W.l.o.g. assume that \( a \in S_1, S_2, \ldots S_k \).

3. Let \( x_1 = p_1, x_2 = p_2, \ldots, x_k = p_k \).

4. Since \( a \) is fractionally covered, \( \sum_{i=1}^{k} p_i \geq 1 \).

5. The probability that \( a \) is not covered by set \( S_i \) is \( (1 - p_i) \).
Feasibility Analysis

Feasibility

1. Pick an arbitrary element \( a \in U \). We will study the probability that it is covered in the set cover that is output by the randomized algorithm discussed above.

2. W.l.o.g. assume that \( a \in S_1, S_2, \ldots S_k \).

3. Let \( x_1 = p_1, x_2 = p_2, \ldots, x_k = p_k \).

4. Since \( a \) is fractionally covered, \( \sum_{i=1}^{k} p_i \geq 1 \).

5. The probability that \( a \) is not covered by set \( S_i \) is \((1 - p_i)\).

6. The probability that \( a \) is not covered by any of the \( S_i, i = 1, 2, \ldots, k \) is \( \prod_{i=1}^{k} (1 - p_i) \).
Feasibility Analysis

Feasibility

1. Pick an arbitrary element \( a \in U \). We will study the probability that it is covered in the set cover that is output by the randomized algorithm discussed above.

2. W.l.o.g. assume that \( a \in S_1, S_2, \ldots S_k \).

3. Let \( x_1 = p_1, x_2 = p_2, \ldots, x_k = p_k \).

4. Since \( a \) is fractionally covered, \( \sum_{i=1}^{k} p_i \geq 1 \).

5. The probability that \( a \) is not covered by set \( S_i \) is \( (1 - p_i) \).

6. The probability that \( a \) is not covered by any of the \( S_i, i = 1, 2, \ldots, k \) is \( \prod_{i=1}^{k} (1 - p_i) \).
Feasibility Analysis

1. Pick an arbitrary element $a \in U$. We will study the probability that it is covered in the set cover that is output by the randomized algorithm discussed above.

2. W.l.o.g. assume that $a \in S_1, S_2, \ldots S_k$.

3. Let $x_1 = p_1, x_2 = p_2, \ldots, x_k = p_k$.

4. Since $a$ is fractionally covered, $\sum_{i=1}^k p_i \geq 1$.

5. The probability that $a$ is not covered by set $S_i$ is $(1 - p_i)$.

6. The probability that $a$ is not covered by any of the $S_i, i = 1, 2, \ldots, k$ is $\prod_{i=1}^k (1 - p_i)$.

7. Thus, the probability that $a$ is not covered by any of the sets is at most $(1 - \frac{1}{k})^k$. 

8. Thus, the probability that $a$ is covered by some set in the cover is at least $(1 - \frac{1}{k})^k$. 

Feasibility Analysis

Feasibility

1. Pick an arbitrary element $a \in U$. We will study the probability that it is covered in the set cover that is output by the randomized algorithm discussed above.
2. W.l.o.g. assume that $a \in S_1, S_2, \ldots S_k$.
3. Let $x_1 = p_1, x_2 = p_2, \ldots, x_k = p_k$.
4. Since $a$ is fractionally covered, $\sum_{i=1}^{k} p_i \geq 1$.
5. The probability that $a$ is not covered by set $S_i$ is $(1 - p_i)$.
6. The probability that $a$ is not covered by any of the $S_i, i = 1, 2, \ldots, k$ is $\prod_{i=1}^{k} (1 - p_i)$.
7. Thus, the probability that $a$ is not covered by any of the sets is at most $(1 - \frac{1}{k})^k \leq \frac{1}{e}$.
Feasibility Analysis

Feasibility

1. Pick an arbitrary element \( a \in U \). We will study the probability that it is covered in the set cover that is output by the randomized algorithm discussed above.

2. W.l.o.g. assume that \( a \in S_1, S_2, \ldots S_k \).

3. Let \( x_1 = p_1, x_2 = p_2, \ldots, x_k = p_k \).

4. Since \( a \) is fractionally covered, \( \sum_{i=1}^{k} p_i \geq 1 \).

5. The probability that \( a \) is not covered by set \( S_i \) is \( (1 - p_i) \).

6. The probability that \( a \) is not covered by any of the \( S_i, i = 1, 2, \ldots, k \) is \( \prod_{i=1}^{k} (1 - p_i) \).

7. Thus, the probability that \( a \) is not covered by any of the sets is at most \( (1 - \frac{1}{k})^k \leq \frac{1}{e} \).

8. Thus, the probability that \( a \) is covered by some set in the cover is at least
Feasibility Analysis

**Feasibility**

1. Pick an arbitrary element $a \in U$. We will study the probability that it is covered in the set cover that is output by the randomized algorithm discussed above.

2. W.l.o.g. assume that $a \in S_1, S_2, \ldots S_k$.

3. Let $x_1 = p_1, x_2 = p_2, \ldots, x_k = p_k$.

4. Since $a$ is fractionally covered, $\sum_{i=1}^{k} p_i \geq 1$.

5. The probability that $a$ is not covered by set $S_i$ is $(1 - p_i)$.

6. The probability that $a$ is not covered by any of the $S_i, i = 1, 2, \ldots, k$ is $\prod_{i=1}^{k} (1 - p_i)$.

7. Thus, the probability that $a$ is not covered by any of the sets is at most $(1 - \frac{1}{k})^k \leq \frac{1}{e}$.

8. Thus, the probability that $a$ is covered by some set in the cover is at least $(1 - \frac{1}{e})$. 
Improving the bound

1. Run the randomized algorithm $c \cdot \ln n$ times independently and merge all the sets obtained into a set $C'$, where $(1/e)^{c \cdot \ln n} \leq 1/4 \cdot n$.

2. Observe that $\Pr[a \text{ is not covered by } C']$ is at most: $(1/e)^{c \cdot \ln n} \leq 1/4 \cdot n$.

3. Summing over all elements, $\Pr[C' \text{ is not a valid cover}]$ is at most $n \cdot 1/4 \cdot n = 1/4$.

4. Clearly, $E[\text{cost}(C')] \leq \text{OPT} f \cdot c \cdot \ln n$.

5. Applying Markov's inequality, $\Pr[\text{cost}(C') \geq 4 \cdot \text{OPT} f \cdot c \cdot \ln n] \leq 1/4$.

6. The probability of these two undesirable events is at most $1/2$.

7. Hence, the probability that $C'$ is a valid set cover and has cost at most $4 \cdot c \cdot \text{OPT} f \cdot \ln n$ is at least $1/2$.

8. If either condition is violated, repeat the experiment. Since the number of trials is a geometric random variable, the expected number of repetitions is at most 2.
Improving the bound

1. Run the randomized algorithm $c \cdot \ln n$ times independently and merge all the sets obtained into a set $C'$, where $(1/e) c \cdot \ln n \leq 1/4 \cdot n$.

2. Observe that $\Pr[a \text{ is not covered by } C']$ is at most: $(1/e) c \cdot \ln n \leq 1/4 \cdot n$.

3. Summing over all elements, $\Pr[C' \text{ is not a valid cover}]$ is at most $n \cdot 1/4 \cdot n = 1/4$.

4. Clearly, $E[\text{cost}(C')] \leq \text{OPT} f \cdot c \cdot \ln n$.

5. Applying Markov's inequality, $\Pr[\text{cost}(C') \geq 4 \cdot \text{OPT} f \cdot c \cdot \ln n] \leq 1/4$.

6. The probability of these two undesirable events is at most $1/2$.

7. Hence, the probability that $C'$ is a valid set cover and has cost at most $4 \cdot c \cdot \text{OPT} f \cdot \ln n$ is at least $1/2$.

8. If either condition is violated, repeat the experiment. Since the number of trials is a geometric random variable, the expected number of repetitions is at most 2.
Improving the bound

**Boosting**

1. Run the randomized algorithm $c \cdot \ln n$ times independently and merge all the sets obtained into a set $C'$, where $\left(\frac{1}{e}\right)^{c \cdot \ln n} \leq \frac{1}{4 \cdot n}$. 

[$\sqrt{2}$]
Boosting

1. Run the randomized algorithm $c \cdot \ln n$ times independently and merge all the sets obtained into a set $C'$, where $(\frac{1}{e})^c \cdot \ln n \leq \frac{1}{4 \cdot n}$.

2. Observe that $\Pr[a \text{ is not covered by } C']$ is at most:
### Improving the bound

#### Boosting

1. Run the randomized algorithm $c \cdot \ln n$ times independently and merge all the sets obtained into a set $C'$, where $(\frac{1}{e})^{c \cdot \ln n} \leq \frac{1}{4n}$.

2. Observe that $\Pr[a \text{ is not covered by } C']$ is at most:

$$
(\frac{1}{e})^{c \cdot \ln n}
$$
Improving the bound

Boosting

1. Run the randomized algorithm $c \cdot \ln n$ times independently and merge all the sets obtained into a set $C'$, where $(\frac{1}{e})^{c \cdot \ln n} \leq \frac{1}{4n}$.

2. Observe that $\Pr[a \text{ is not covered by } C']$ is at most:

$$\left(\frac{1}{e}\right)^{c \cdot \ln n} \leq \frac{1}{4 \cdot n}.$$
Improving the bound

### Boosting

1. Run the randomized algorithm $c \cdot \ln n$ times independently and merge all the sets obtained into a set $C'$, where \((\frac{1}{e})^{c \cdot \ln n} \leq \frac{1}{4 \cdot n}\).

2. Observe that $\Pr[a \text{ is not covered by } C']$ is at most:

   \[
   \left(\frac{1}{e}\right)^{c \cdot \ln n} \leq \frac{1}{4 \cdot n}.
   \]

3. Summing over all elements, $\Pr[C' \text{ is not a valid cover}]$ is at most
Improving the bound

**Boosting**

1. Run the randomized algorithm $c \cdot \ln n$ times independently and merge all the sets obtained into a set $C'$, where $(\frac{1}{e})^{c \cdot \ln n} \leq \frac{1}{4 \cdot n}$.

2. Observe that $\Pr[a \text{ is not covered by } C']$ is at most:

   $$(\frac{1}{e})^{c \cdot \ln n} \leq \frac{1}{4 \cdot n}.$$  

3. Summing over all elements, $\Pr[C' \text{ is not a valid cover}]$ is at most $n \cdot \frac{1}{4 \cdot n}$. 

4. Clearly, $\mathbb{E}[\text{cost}(C')] \leq \text{OPT}$.

5. Applying Markov’s inequality, $\Pr[\text{cost}(C') \geq 4 \cdot \text{OPT} \cdot c \cdot \ln n] \leq \frac{1}{4 \cdot n}$.

6. The probability of these two undesirable events is at most $\frac{1}{2}$.

7. Hence, the probability that $C'$ is a valid set cover and has cost at most $4 \cdot c \cdot \text{OPT} \cdot \ln n$ is at least $\frac{1}{2}$.

8. If either condition is violated, repeat the experiment. Since the number of trials is a geometric random variable, the expected number of repetitions is at most 2.
Improving the bound

Boosting

1. Run the randomized algorithm \( c \cdot \ln n \) times independently and merge all the sets obtained into a set \( C' \), where \( \left( \frac{1}{e} \right)^{c \cdot \ln n} \leq \frac{1}{4 \cdot n} \).

2. Observe that \( \Pr[a \text{ is not covered by } C'] \) is at most:

\[
\left( \frac{1}{e} \right)^{c \cdot \ln n} \leq \frac{1}{4 \cdot n}.
\]

3. Summing over all elements, \( \Pr[C' \text{ is not a valid cover}] \) is at most \( n \cdot \frac{1}{4 \cdot n} = \frac{1}{4} \).
LP-Rounding
A Randomized Rounding Algorithm

Improving the bound

Boosting

1. Run the randomized algorithm \( c \cdot \ln n \) times independently and merge all the sets obtained into a set \( C' \), where \( \left( \frac{1}{e} \right)^{c \cdot \ln n} \leq \frac{1}{4 \cdot n} \).

2. Observe that \( \Pr[a \text{ is not covered by } C'] \) is at most:

\[
\left( \frac{1}{e} \right)^{c \cdot \ln n} \leq \frac{1}{4 \cdot n}.
\]

3. Summing over all elements, \( \Pr[C' \text{ is not a valid cover}] \) is at most \( n \cdot \frac{1}{4 \cdot n} = \frac{1}{4} \).

4. Clearly, \( \mathbb{E}[\text{cost}(C')] \leq OPT_f \cdot c \cdot \ln n \).
Improving the bound

**Boosting**

1. Run the randomized algorithm \( c \cdot \ln n \) times independently and merge all the sets obtained into a set \( C' \), where \( \left( \frac{1}{e} \right)^{c \cdot \ln n} \leq \frac{1}{4 \cdot n} \).

2. Observe that \( \Pr[a \text{ is not covered by } C'] \) is at most:
   \[
   \left( \frac{1}{e} \right)^{c \cdot \ln n} \leq \frac{1}{4 \cdot n}.
   \]

3. Summing over all elements, \( \Pr[C' \text{ is not a valid cover}] \) is at most \( n \cdot \frac{1}{4 \cdot n} = \frac{1}{4} \).

4. Clearly, \( \mathbb{E}[\text{cost}(C')] \leq \text{OPT}_f \cdot c \cdot \ln n \).

5. Applying Markov's inequality, \( \Pr[\text{cost}(C') \geq 4 \cdot \text{OPT}_f \cdot c \cdot \ln n] \)
### Improving the bound

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Run the randomized algorithm $c \cdot \ln n$ times independently and merge all the sets obtained into a set $C'$, where $(\frac{1}{e})^{c \cdot \ln n} \leq \frac{1}{4 \cdot n}$.</td>
</tr>
<tr>
<td>2</td>
<td>Observe that $\Pr[a$ is not covered by $C']$ is at most: $$(\frac{1}{e})^{c \cdot \ln n} \leq \frac{1}{4 \cdot n}.$$</td>
</tr>
<tr>
<td>3</td>
<td>Summing over all elements, $\Pr[C'$ is not a valid cover] is at most $n \cdot \frac{1}{4 \cdot n} = \frac{1}{4}$.</td>
</tr>
<tr>
<td>4</td>
<td>Clearly, $\mathbb{E}[\text{cost}(C')] \leq OPT_f \cdot c \cdot \ln n$.</td>
</tr>
<tr>
<td>5</td>
<td>Applying Markov's inequality, $\Pr[\text{cost}(C') \geq 4 \cdot OPT_f \cdot c \cdot \ln n] \leq \frac{1}{4}$.</td>
</tr>
</tbody>
</table>
Improving the bound

Boosting

1. Run the randomized algorithm $c \cdot \ln n$ times independently and merge all the sets obtained into a set $C'$, where $(\frac{1}{e})^{c \cdot \ln n} \leq \frac{1}{4 \cdot n}$.

2. Observe that $\Pr[a \text{ is not covered by } C']$ is at most:

   $$(\frac{1}{e})^{c \cdot \ln n} \leq \frac{1}{4 \cdot n}.$$

3. Summing over all elements, $\Pr[C' \text{ is not a valid cover}]$ is at most $n \cdot \frac{1}{4 \cdot n} = \frac{1}{4}$.

4. Clearly, $\mathbb{E}[\text{cost}(C')] \leq OPT_f \cdot c \cdot \ln n$.

5. Applying Markov's inequality, $\Pr[\text{cost}(C') \geq 4 \cdot OPT_f \cdot c \cdot \ln n] \leq \frac{1}{4}$.

6. The probability of these two undesirable events is at most $\frac{1}{2}$.
Improving the bound

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Boosting</strong></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Run the randomized algorithm $c \cdot \ln n$ times independently and merge all the sets obtained into a set $C'$, where $(\frac{1}{e})^{c \cdot \ln n} \leq \frac{1}{4n}$.</td>
</tr>
<tr>
<td>2</td>
<td>Observe that $\Pr[a \text{ is not covered by } C']$ is at most:</td>
</tr>
<tr>
<td></td>
<td>$(\frac{1}{e})^{c \cdot \ln n} \leq \frac{1}{4 \cdot n}$.</td>
</tr>
<tr>
<td>3</td>
<td>Summing over all elements, $\Pr[C' \text{ is not a valid cover}]$ is at most $n \cdot \frac{1}{4n} = \frac{1}{4}$.</td>
</tr>
<tr>
<td>4</td>
<td>Clearly, $\mathbb{E}[\text{cost}(C')] \leq \text{OPT}_f \cdot c \cdot \ln n$.</td>
</tr>
<tr>
<td>5</td>
<td>Applying Markov's inequality, $\Pr[\text{cost}(C') \geq 4 \cdot \text{OPT}_f \cdot c \cdot \ln n] \leq \frac{1}{4}$.</td>
</tr>
<tr>
<td>6</td>
<td>The probability of these two undesirable events is at most $\frac{1}{2}$.</td>
</tr>
<tr>
<td>7</td>
<td>Hence, the probability that $C'$ is a valid set cover and has cost at most $4 \cdot c \cdot \text{OPT}_f \cdot \ln n$ is at least $\frac{1}{2}$.</td>
</tr>
</tbody>
</table>
Improving the bound

Boosting

1. Run the randomized algorithm $c \cdot \ln n$ times independently and merge all the sets obtained into a set $C'$, where $(\frac{1}{e})^c \ln n \leq \frac{1}{4 \cdot n}$.

2. Observe that $\Pr[a \text{ is not covered by } C']$ is at most:

$$ (\frac{1}{e})^c \ln n \leq \frac{1}{4 \cdot n}. $$

3. Summing over all elements, $\Pr[C' \text{ is not a valid cover}]$ is at most $n \cdot \frac{1}{4 \cdot n} = \frac{1}{4}$.

4. Clearly, $\mathbb{E}[\text{cost}(C')] \leq \text{OPT}_f \cdot c \cdot \ln n$.

5. Applying Markov's inequality, $\Pr[\text{cost}(C') \geq 4 \cdot \text{OPT}_f \cdot c \cdot \ln n] \leq \frac{1}{4}$.

6. The probability of these two undesirable events is at most $\frac{1}{2}$.

7. Hence, the probability that $C'$ is a valid set cover and has cost at most $4 \cdot c \cdot \text{OPT}_f \cdot \ln n$ is at least $\frac{1}{2}$.

8. If either condition is violated, repeat the experiment.
Improving the bound

1. Run the randomized algorithm \( c \cdot \ln n \) times independently and merge all the sets obtained into a set \( C' \), where \( \left( \frac{1}{e} \right)^{c \cdot \ln n} \leq \frac{1}{4n} \).

2. Observe that \( \Pr[a \text{ is not covered by } C'] \) is at most:
   \[
   \left( \frac{1}{e} \right)^{c \cdot \ln n} \leq \frac{1}{4 \cdot n}.
   \]

3. Summing over all elements, \( \Pr[C' \text{ is not a valid cover}] \) is at most \( n \cdot \frac{1}{4n} = \frac{1}{4} \).

4. Clearly, \( \mathbb{E}[\text{cost}(C')] \leq \text{OPT}_f \cdot c \cdot \ln n \).

5. Applying Markov’s inequality, \( \Pr[\text{cost}(C') \geq 4 \cdot \text{OPT}_f \cdot c \cdot \ln n] \leq \frac{1}{4} \).

6. The probability of these two undesirable events is at most \( \frac{1}{2} \).

7. Hence, the probability that \( C' \) is a valid set cover and has cost at most \( 4 \cdot c \cdot \text{OPT}_f \cdot \ln n \) is at least \( \frac{1}{2} \).

8. If either condition is violated, repeat the experiment. Since the number of trials is a geometric random variable, the expected number of repetitions is at most 2.
The IP formulation for Vertex Cover and its LP Relaxation

Let $V$ denote the vertex set, $E$ denote the edge set and $c: V \to \mathbb{Q}^+$ denote the weight function.

The IP formulation for the vertex cover problem is:

$$\min \sum_{v \in V} c(v) \cdot x_v$$

subject to

$$x_u + x_v \geq 1, \quad \forall (u, v) \in E$$

$$x_v \in \{0, 1\} \quad \forall v \in V$$

LP relaxation

$$\min \sum_{v \in V} c(v) \cdot x_v$$

subject to

$$x_u + x_v \geq 1, \quad \forall (u, v) \in E$$

$$x_v \geq 0 \quad \forall v \in V$$
The IP formulation for Vertex Cover and its LP Relaxation

**IP formulation**

Let $V$ denote the vertex set, $E$ denote the edge set and $c : V \to Q^+$ denote the weight function.
The IP formulation for Vertex Cover and its LP Relaxation

**IP formulation**

Let $V$ denote the vertex set, $E$ denote the edge set and $c : V \rightarrow Q^+$ denote the weight function. The IP formulation for the vertex cover problem is:

$$\text{min } \sum_{v \in V} c(v) \cdot x_v$$

subject to

$$x_u + x_v \geq 1, \quad \forall (u, v) \in E$$

$$x_v \in \{0, 1\} \quad \forall v \in V$$
LP-Rounding

Half-integrality of Vertex Cover

The IP formulation for Vertex Cover and its LP Relaxation

**IP formulation**

Let $V$ denote the vertex set, $E$ denote the edge set and $c : V \rightarrow Q^+$ denote the weight function. The IP formulation for the vertex cover problem is:

$$\min \sum_{v \in V} c(v) \cdot x_v$$

$$\text{subject to } x_u + x_v \geq 1, \forall (u, v) \in E$$

$$x_v \in \{0, 1\} \forall v \in V$$
The IP formulation for Vertex Cover and its LP Relaxation

**IP formulation**

Let $V$ denote the vertex set, $E$ denote the edge set and $c : V \rightarrow Q^+$ denote the weight function. The IP formulation for the vertex cover problem is:

$$\min \sum_{v \in V} c(v) \cdot x_v$$

subject to

$$x_u + x_v \geq 1, \quad \forall (u, v) \in E$$
The IP formulation for Vertex Cover and its LP Relaxation

**IP formulation**

Let $V$ denote the vertex set, $E$ denote the edge set and $c : V \rightarrow \mathbb{Q}^+$ denote the weight function. The IP formulation for the vertex cover problem is:

$$\min \sum_{v \in V} c(v) \cdot x_v$$

subject to

$$x_u + x_v \geq 1, \quad \forall (u, v) \in E$$

$$x_v \in \{0, 1\} \quad \forall v \in V$$
The IP formulation for Vertex Cover and its LP Relaxation

**IP formulation**

Let $V$ denote the vertex set, $E$ denote the edge set and $c : V \to Q^+$ denote the weight function. The IP formulation for the vertex cover problem is:

$$\begin{align*}
\min & \sum_{v \in V} c(v) \cdot x_v \\
\text{subject to} & \quad x_u + x_v \geq 1, \quad \forall (u, v) \in E \\
& \quad x_v \in \{0, 1\}, \quad \forall v \in V
\end{align*}$$
The IP formulation for Vertex Cover and its LP Relaxation

**IP formulation**

Let $V$ denote the vertex set, $E$ denote the edge set and $c : V \rightarrow Q^+$ denote the weight function. The IP formulation for the vertex cover problem is:

$$\min \sum_{v \in V} c(v) \cdot x_v$$

subject to

$$x_u + x_v \geq 1, \quad \forall (u, v) \in E$$

$$x_v \in \{0, 1\} \quad \forall v \in V$$

**LP relaxation**
The IP formulation for Vertex Cover and its LP Relaxation

**IP formulation**

Let $V$ denote the vertex set, $E$ denote the edge set and $c : V \rightarrow Q^+$ denote the weight function. The IP formulation for the vertex cover problem is:

$$\min \sum_{v \in V} c(v) \cdot x_v$$

subject to

$$x_u + x_v \geq 1, \quad \forall (u, v) \in E$$

$$x_v \in \{0, 1\} \quad \forall v \in V$$

**LP relaxation**

$$\min \sum_{v \in V} c(v) \cdot x_v$$
The IP formulation for Vertex Cover and its LP Relaxation

**IP formulation**

Let $V$ denote the vertex set, $E$ denote the edge set and $c : V \to \mathbb{Q}^+$ denote the weight function. The IP formulation for the vertex cover problem is:

$$\min \sum_{v \in V} c(v) \cdot x_v$$

subject to

$$x_u + x_v \geq 1, \quad \forall (u, v) \in E$$

$$x_v \in \{0, 1\} \quad \forall v \in V$$

**LP relaxation**

$$\min \sum_{v \in V} c(v) \cdot x_v$$

subject to

$$x_u + x_v \geq 1, \quad \forall (u, v) \in E$$
The IP formulation for Vertex Cover and its LP Relaxation

**IP formulation**

Let $V$ denote the vertex set, $E$ denote the edge set and $c : V \rightarrow Q^+$ denote the weight function. The IP formulation for the vertex cover problem is:

$$\min \sum_{v \in V} c(v) \cdot x_v$$

subject to

$$x_u + x_v \geq 1, \quad \forall (u, v) \in E$$

$$x_v \in \{0, 1\} \quad \forall v \in V$$

**LP relaxation**

$$\min \sum_{v \in V} c(v) \cdot x_v$$

subject to

$$x_u + x_v \geq 1, \quad \forall (u, v) \in E$$

$$x_v \geq 0 \quad \forall v \in V$$
The IP formulation for Vertex Cover and its LP Relaxation

**IP formulation**

Let \( V \) denote the vertex set, \( E \) denote the edge set and \( c : V \rightarrow Q^+ \) denote the weight function. The IP formulation for the vertex cover problem is:

\[
\min \sum_{v \in V} c(v) \cdot x_v \\
\text{subject to} \quad x_u + x_v \geq 1, \quad \forall (u, v) \in E \\
x_v \in \{0, 1\}, \quad \forall v \in V
\]

**LP relaxation**

\[
\min \sum_{v \in V} c(v) \cdot x_v \\
\text{subject to} \quad x_u + x_v \geq 1, \quad \forall (u, v) \in E \\
x_v \geq 0, \quad \forall v \in V
\]
Some concepts from polyhedral theory
Some concepts from polyhedral theory

<table>
<thead>
<tr>
<th>Concepts</th>
</tr>
</thead>
<tbody>
<tr>
<td>Convex sets.</td>
</tr>
<tr>
<td>Polyhedral sets.</td>
</tr>
<tr>
<td>Convexity of polyhedral sets.</td>
</tr>
<tr>
<td>Extreme point solution.</td>
</tr>
<tr>
<td>Half-integral solution.</td>
</tr>
</tbody>
</table>
Some concepts from polyhedral theory

- Convex sets.
Some concepts from polyhedral theory

Concepts

1. Convex sets.
2. Polyhedral sets.
Some concepts from polyhedral theory

Concepts

1. Convex sets.
2. Polyhedral sets.
3. Convexity of polyhedral sets.
Some concepts from polyhedral theory

Concepts

1. Convex sets.
2. Polyhedral sets.
3. Convexity of polyhedral sets.
4. Extreme point solution.
Some concepts from polyhedral theory

<table>
<thead>
<tr>
<th>Concepts</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Convex sets.</td>
</tr>
<tr>
<td>2 Polyhedral sets.</td>
</tr>
<tr>
<td>3 Convexity of polyhedral sets.</td>
</tr>
<tr>
<td>4 Extreme point solution.</td>
</tr>
<tr>
<td>5 Half-integral solution.</td>
</tr>
</tbody>
</table>
Lemma

Let $x$ denote a feasible solution of the above LP that is not half-integral. Then $x$ is not an extreme point solution of the LP.

Proof.

1. Consider the set $S$ of vertices for which the extreme point solution $x$ does not assign half-integral values.
2. Partition the vertices in $S$ into $V_+ = \{ v : 1/2 < x_v < 1 \}$, $V_- = \{ v : 0 < x_v < 1/2 \}$.
3. Let $\epsilon > 0$ denote a constant. Define $y_v$ and $z_v$ as follows:
   
   $\begin{align*}
   y_v &= \begin{cases} 
   x_v + \epsilon, & x_v \in V_+ \\
   x_v - \epsilon, & x_v \in V_- \\
   x_v, & \text{otherwise}
   \end{cases} \\
   z_v &= \begin{cases} 
   x_v - \epsilon, & x_v \in V_+ \\
   x_v + \epsilon, & x_v \in V_- \\
   x_v, & \text{otherwise}
   \end{cases}
   \end{align*}$
Lemma

Let $x$ denote a feasible solution of the above LP that is not half-integral.
Lemma

Let $x$ denote a feasible solution of the above LP that is not half-integral. Then $x$ is not an extreme point solution of the LP.
Half-integrality of vertex cover

Lemma

Let $x$ denote a feasible solution of the above LP that is not half-integral. Then $x$ is not an extreme point solution of the LP.

Proof.
Half-integrality of vertex cover

Lemma

Let $\mathbf{x}$ denote a feasible solution of the above LP that is not half-integral. Then $\mathbf{x}$ is not an extreme point solution of the LP.

Proof.

1. Consider the set $S$ of vertices for which the extreme point solution $\mathbf{x}$ does not assign half-integral values.
Lemma

Let $x$ denote a feasible solution of the above LP that is not half-integral. Then $x$ is not an extreme point solution of the LP.

Proof.

1. Consider the set $S$ of vertices for which the extreme point solution $x$ does not assign half-integral values.

2. Partition the vertices in $S$ into
Let \( \mathbf{x} \) denote a feasible solution of the above LP that is not half-integral. Then \( \mathbf{x} \) is not an extreme point solution of the LP.

Proof.

1. Consider the set \( S \) of vertices for which the extreme point solution \( \mathbf{x} \) does not assign half-integral values.
2. Partition the vertices in \( S \) into

\[
V_+ = \{ v : \frac{1}{2} < x_v < 1 \},
\]

\[
V_- = \{ v : 0 < x_v < \frac{1}{2} \}.
\]
Half-integrality of vertex cover

**Lemma**

Let $x$ denote a feasible solution of the above LP that is not half-integral. Then $x$ is not an extreme point solution of the LP.

**Proof.**

1. Consider the set $S$ of vertices for which the extreme point solution $x$ does not assign half-integral values.
2. Partition the vertices in $S$ into

$$V_+ = \{ v : \frac{1}{2} < x_v < 1 \}, \quad V_- = \{ v : 0 < x_v < \frac{1}{2} \}$$
Lemma

Let $x$ denote a feasible solution of the above LP that is not half-integral. Then $x$ is not an extreme point solution of the LP.

Proof.

1. Consider the set $S$ of vertices for which the extreme point solution $x$ does not assign half-integral values.

2. Partition the vertices in $S$ into

   $$V_+ = \{ v : \frac{1}{2} < x_v < 1 \}, \quad V_- = \{ v : 0 < x_v < \frac{1}{2} \}$$

3. Let $\varepsilon > 0$ denote a constant.
**Half-integrality of vertex cover**

**Lemma**

Let \( \mathbf{x} \) denote a feasible solution of the above LP that is not half-integral. Then \( \mathbf{x} \) is not an extreme point solution of the LP.

**Proof.**

1. Consider the set \( S \) of vertices for which the extreme point solution \( \mathbf{x} \) does not assign half-integral values.
2. Partition the vertices in \( S \) into

\[
\begin{align*}
V_+ &= \{ v : \frac{1}{2} < x_v < 1 \}, \\
V_- &= \{ v : 0 < x_v < \frac{1}{2} \}
\end{align*}
\]

3. Let \( \varepsilon > 0 \) denote a constant. Define \( y_v \) and \( z_v \) as follows:

\[
\begin{align*}
y_v &= \begin{cases} 
  x_v + \varepsilon, & x_v \in V_+ \\
  x_v - \varepsilon, & x_v \in V_- \\
  x_v, & \text{otherwise}
\end{cases} \\
\end{align*}
\]

\[
\begin{align*}
z_v &= \begin{cases} 
  x_v - \varepsilon, & x_v \in V_+ \\
  x_v + \varepsilon, & x_v \in V_- \\
  x_v, & \text{otherwise}
\end{cases}
\end{align*}
\]
Completing the proof

Proof.

1. $x$ is distinct from $y$ and $z$, since $V^+ \cup V^- \neq \emptyset$.

2. If $y$ and $z$ are feasible, then $x$ cannot be an extreme point, since $x = \frac{1}{2} (y + z)$.

3. It is easy to choose $\varepsilon$, so that $y$ and $z$ are non-negative.

4. Focus on a specific edge $(u, v)$.
   - We consider the following cases:
     1. $x_u + x_v > 1$ - Clearly, we can choose $\varepsilon$ small enough so that $y$ and $z$ do not violate the constraint for this edge.
     2. $x_u + x_v = 1$ - In this case, there are three possibilities for $x_u$ and $x_v$, viz., $x_u = x_v = \frac{1}{2}, x_u = 0, x_v = 1$, and $u \in V^+, v \in V^-$. In all three cases, for any choice of $\varepsilon$, we must have, $x_u + x_v = y_u + y_v = z_u + z_v = 1$. 
Completing the proof

**Proof.**

1. $x$ is distinct from $y$ and $z$,
Completing the proof

Proof.

1. \( x \) is distinct from \( y \) and \( z \), since \( V_+ \cup V_- \neq \emptyset \).
Completing the proof

Proof.

1. \( x \) is distinct from \( y \) and \( z \), since \( V_+ \cup V_- \neq \emptyset \).

2. If \( y \) and \( z \) are feasible, then \( x \) cannot be an extreme point, since
Completing the proof

Proof.

1. $x$ is distinct from $y$ and $z$, since $V_+ \cup V_- \neq \emptyset$.
2. If $y$ and $z$ are feasible, then $x$ cannot be an extreme point, since $x = \frac{1}{2} \cdot (y + z)$. 
Completing the proof

Proof.

1. $x$ is distinct from $y$ and $z$, since $V_+ \cup V_- \neq \emptyset$.
2. If $y$ and $z$ are feasible, then $x$ cannot be an extreme point, since $x = \frac{1}{2} \cdot (y + z)$.
3. It is easy to choose $\varepsilon$, so that $y$ and $z$ are non-negative.
Completing the proof

Proof.

1. \(x\) is distinct from \(y\) and \(z\), since \(V_+ \cup V_- \neq \emptyset\).
2. If \(y\) and \(z\) are feasible, then \(x\) cannot be an extreme point, since \(x = \frac{1}{2} \cdot (y + z)\).
3. It is easy to choose \(\varepsilon\), so that \(y\) and \(z\) are non-negative.
4. Focus on a specific edge \((u, v)\).
Completing the proof

Proof.

1. **x** is distinct from **y** and **z**, since \( V_+ \cup V_- \neq \emptyset \).
2. If **y** and **z** are feasible, then **x** cannot be an extreme point, since \( x = \frac{1}{2} \cdot (y + z) \).
3. It is easy to choose \( \varepsilon \), so that **y** and **z** are non-negative.
4. Focus on a specific edge \((u, v)\). We consider the following cases:
Completing the proof

Proof.

1. \( \mathbf{x} \) is distinct from \( \mathbf{y} \) and \( \mathbf{z} \), since \( V_+ \cup V_- \neq \emptyset \).
2. If \( \mathbf{y} \) and \( \mathbf{z} \) are feasible, then \( \mathbf{x} \) cannot be an extreme point, since \( \mathbf{x} = \frac{1}{2} \cdot (\mathbf{y} + \mathbf{z}) \).
3. It is easy to choose \( \varepsilon \), so that \( \mathbf{y} \) and \( \mathbf{z} \) are non-negative.
4. Focus on a specific edge \((u, v)\). We consider the following cases:
   1. \( x_u + x_v > 1 \) - Clearly, we can choose \( \varepsilon \) small enough so that \( \mathbf{y} \) and \( \mathbf{z} \) do not violate the constraint for this edge.
Completing the proof

Proof.

1. $x$ is distinct from $y$ and $z$, since $V_+ \cup V_- \neq \emptyset$.
2. If $y$ and $z$ are feasible, then $x$ cannot be an extreme point, since $x = \frac{1}{2} \cdot (y + z)$.
3. It is easy to choose $\varepsilon$, so that $y$ and $z$ are non-negative.
4. Focus on a specific edge $(u, v)$. We consider the following cases:
   1. $x_u + x_v > 1$ - Clearly, we can choose $\varepsilon$ small enough so that $y$ and $z$ do not violate the constraint for this edge.
   2. $x_u + x_v = 1$ - In this case, there are three possibilities for $x_u$ and $x_v$, viz.,
Completing the proof

**Proof.**

1. **x** is distinct from **y** and **z**, since $V_+ \cup V_- \neq \emptyset$.

2. If **y** and **z** are feasible, then **x** cannot be an extreme point, since $x = \frac{1}{2} \cdot (y + z)$.

3. It is easy to choose $\epsilon$, so that **y** and **z** are non-negative.

4. Focus on a specific edge $(u, v)$. We consider the following cases:

   1. $x_u + x_v > 1$ - Clearly, we can choose $\epsilon$ small enough so that **y** and **z** do not violate the constraint for this edge.

   2. $x_u + x_v = 1$ - In this case, there are three possibilities for $x_u$ and $x_v$, viz., $x_u = x_v = \frac{1}{2}$,
Completng the proof

Proof.

1. $x$ is distinct from $y$ and $z$, since $V_+ \cup V_- \neq \emptyset$.
2. If $y$ and $z$ are feasible, then $x$ cannot be an extreme point, since $x = \frac{1}{2} \cdot (y + z)$.
3. It is easy to choose $\varepsilon$, so that $y$ and $z$ are non-negative.
4. Focus on a specific edge $(u, v)$. We consider the following cases:
   - $x_u + x_v > 1$ - Clearly, we can choose $\varepsilon$ small enough so that $y$ and $z$ do not violate the constraint for this edge.
   - $x_u + x_v = 1$ - In this case, there are three possibilities for $x_u$ and $x_v$, viz., $x_u = x_v = \frac{1}{2}$,
     $x_u = 0, x_v = 1$, $x_u = 1, x_v = 0$.
Completing the proof

Proof.

1. \( x \) is distinct from \( y \) and \( z \), since \( V_+ \cup V_- \neq \emptyset \).
2. If \( y \) and \( z \) are feasible, then \( x \) cannot be an extreme point, since \( x = \frac{1}{2} \cdot (y + z) \).
3. It is easy to choose \( \varepsilon \), so that \( y \) and \( z \) are non-negative.
4. Focus on a specific edge \((u, v)\). We consider the following cases:
   1. \( x_u + x_v > 1 \) - Clearly, we can choose \( \varepsilon \) small enough so that \( y \) and \( z \) do not violate the constraint for this edge.
   2. \( x_u + x_v = 1 \) - In this case, there are three possibilities for \( x_u \) and \( x_v \), viz., \( x_u = x_v = \frac{1}{2} \), \( x_u = 0, x_v = 1 \), and \( u \in V_+, v \in V_- \).
Completing the proof

Proof.

1. $x$ is distinct from $y$ and $z$, since $V_+ \cup V_- \neq \emptyset$.
2. If $y$ and $z$ are feasible, then $x$ cannot be an extreme point, since $x = \frac{1}{2} \cdot (y + z)$.
3. It is easy to choose $\varepsilon$, so that $y$ and $z$ are non-negative.
4. Focus on a specific edge $(u, v)$. We consider the following cases:
   1. $x_u + x_v > 1$ - Clearly, we can choose $\varepsilon$ small enough so that $y$ and $z$ do not violate the constraint for this edge.
   2. $x_u + x_v = 1$ - In this case, there are three possibilities for $x_u$ and $x_v$, viz., $x_u = x_v = \frac{1}{2}$, $x_u = 0, x_v = 1$, and $u \in V_+, v \in V_-$. In all three cases, for any choice of $\varepsilon$, we must have,
Completing the proof

Proof.

1. \( \mathbf{x} \) is distinct from \( \mathbf{y} \) and \( \mathbf{z} \), since \( V_+ \cup V_- \neq \emptyset \).
2. If \( \mathbf{y} \) and \( \mathbf{z} \) are feasible, then \( \mathbf{x} \) cannot be an extreme point, since \( \mathbf{x} = \frac{1}{2} \cdot (\mathbf{y} + \mathbf{z}) \).
3. It is easy to choose \( \varepsilon \), so that \( \mathbf{y} \) and \( \mathbf{z} \) are non-negative.
4. Focus on a specific edge \((u, v)\). We consider the following cases:
   
   - **Case 1:** \( x_u + x_v > 1 \) - Clearly, we can choose \( \varepsilon \) small enough so that \( \mathbf{y} \) and \( \mathbf{z} \) do not violate the constraint for this edge.
   
   - **Case 2:** \( x_u + x_v = 1 \) - In this case, there are three possibilities for \( x_u \) and \( x_v \), viz., \( x_u = x_v = \frac{1}{2} \), \( x_u = 0, x_v = 1 \), and \( u \in V_+, v \in V_- \). In all three cases, for any choice of \( \varepsilon \), we must have,
     \[
     x_u + x_v = y_u + y_v = z_u + z_v = 1
     \]
Approximation algorithm for vertex cover

Corollary

All extreme point solutions to the above linear programming relaxation of the vertex-cover problem are half-integral.
Approximation algorithm for vertex cover

Corollary

All extreme point solutions to the above linear programming relaxation of the vertex-cover problem are half-integral.

Note
Approximation algorithm for vertex cover

Corollary

All extreme point solutions to the above linear programming relaxation of the vertex-cover problem are half-integral.

Note

We now have a 2-approximation algorithm for weighted vertex cover.
**Approximation algorithm for vertex cover**

**Corollary**

All extreme point solutions to the above linear programming relaxation of the vertex-cover problem are half-integral.

**Note**

We now have a 2-approximation algorithm for weighted vertex cover.

1. Solve the LP to obtain an extreme point solution.
Approximation algorithm for vertex cover

Corollary

All extreme point solutions to the above linear programming relaxation of the vertex-cover problem are half-integral.

Note

We now have a 2-approximation algorithm for weighted vertex cover.

1. Solve the LP to obtain an extreme point solution.
2. Pick all the vertices that are set to $\frac{1}{2}$ or 1.