Set-Cover approximation through Primal Dual Schema

K. Subramani¹

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

April 4, 2014
Outline

1 Preliminaries
Outline

1 Preliminaries
2 Primal-Dual schema for Set Cover
Outline

1 Preliminaries

2 Primal-Dual schema for Set Cover

3 The Primal Dual Algorithm
Outline

1. Preliminaries
2. Primal-Dual schema for Set Cover
3. The Primal Dual Algorithm
4. Tightness
The primal-dual approach was used in algorithm design for problems such as matching, network flows, shortest paths in digraphs, etc., in which the LP-relaxations have integral optimal solutions.

For LP optimal solutions, we know that the primal and dual complementary slackness conditions have to be satisfied.

The approach starts with feasible primal and dual solutions and iteratively attempts to satisfy complementary slackness conditions.

When they are all satisfied, both primal and dual solutions must be optimal.

Cannot work directly for NP-hard problems, since the LP relaxations need not have integral optimal solutions.

However, a relaxation of the complementary slackness conditions helps in the derivation of approximation algorithms.
The primal-dual approach was used in algorithm design for problems such as matching, network flows, shortest paths in digraphs, etc., in which the LP-relaxations have integral optimal solutions. For LP optimal solutions, we know that the primal and dual complementary slackness conditions have to be satisfied. The approach starts with feasible primal and dual solutions and iteratively attempts to satisfy complementary slackness conditions. When they are all satisfied, both primal and dual solutions must be optimal. However, a relaxation of the complementary slackness conditions helps in the derivation of approximation algorithms. Cannot work directly for NP-hard problems, since the LP relaxations need not have integral optimal solutions.
Algorithm design for problems in P

1. The primal-dual approach was used in algorithm design for problems such as matching, network flows, shortest paths in digraphs, etc., in which the LP-relaxations have integral optimal solutions.
The primal-dual approach was used in algorithm design for problems such as matching, network flows, shortest paths in digraphs, etc., in which the LP-relaxations have integral optimal solutions.

For LP optimal solutions, we know that the primal and dual complementary slackness conditions have to be satisfied.
The primal-dual approach was used in algorithm design for problems such as matching, network flows, shortest paths in digraphs, etc., in which the LP-relaxations have integral optimal solutions.

For LP optimal solutions, we know that the primal and dual complementary slackness conditions have to be satisfied.

The approach starts with feasible primal and dual solutions and iteratively attempts to satisfy complementary slackness conditions.
The primal-dual approach was used in algorithm design for problems such as matching, network flows, shortest paths in digraphs, etc., in which the LP-relaxations have integral optimal solutions.

For LP optimal solutions, we know that the primal and dual complementary slackness conditions have to be satisfied.

The approach starts with feasible primal and dual solutions and iteratively attempts to satisfy complementary slackness conditions.

When they are all satisfied, both primal and dual solutions must be optimal.
The primal-dual approach was used in algorithm design for problems such as matching, network flows, shortest paths in digraphs, etc., in which the LP-relaxations have integral optimal solutions.

For LP optimal solutions, we know that the primal and dual complementary slackness conditions have to be satisfied.

The approach starts with feasible primal and dual solutions and iteratively attempts to satisfy complementary slackness conditions.

When they are all satisfied, both primal and dual solutions must be optimal.

Cannot work directly for NP-hard problems, since the LP relaxations need not have integral optimal solutions.
The primal-dual approach was used in algorithm design for problems such as matching, network flows, shortest paths in digraphs, etc., in which the LP-relaxations have integral optimal solutions.

For LP optimal solutions, we know that the primal and dual complementary slackness conditions have to be satisfied.

The approach starts with feasible primal and dual solutions and iteratively attempts to satisfy complementary slackness conditions.

When they are all satisfied, both primal and dual solutions must be optimal.

Cannot work directly for **NP-hard** problems, since the LP relaxations need not have integral optimal solutions.

However, a relaxation of the complementary slackness conditions helps in the derivation of approximation algorithms.
Primal and Dual forms

Primal (P):
\[ z = \min \sum_{j=1}^{n} c_j \cdot x_j \]
subject to
\[ \sum_{j=1}^{n} a_{ij} \cdot x_j \geq b_i, \quad i = 1, 2, ..., m \]
\[ x_j \geq 0, \quad j = 1, 2, ..., n \]

Dual (D):
\[ w = \max \sum_{i=1}^{m} b_i \cdot y_i \]
subject to
\[ \sum_{i=1}^{m} a_{ij} \cdot y_i \leq c_j, \quad j = 1, 2, ..., n \]
\[ y_i \geq 0, \quad i = 1, 2, ..., m \]
\[ w = \max b_i \cdot y_i \]
\[ y_i \cdot A \leq c y_i \geq 0 \]
Primal and Dual forms

Primal (P):

\[ z = \min \sum_{j=1}^{n} c_j \cdot x_j \]
Primal and Dual forms

Forms

Primal (P):

\[ z = \min \sum_{j=1}^{n} c_j \cdot x_j \]
\[ \text{s.t.} \quad \sum_{j=1}^{n} a_{ij} \cdot x_j \geq b_i, \quad i = 1, 2, \ldots m \]
Primal and Dual forms

**Forms**

**Primal (P):**

\[ z = \min \sum_{j=1}^{n} c_j \cdot x_j \]

s.t. \[ \sum_{j=1}^{n} a_{ij} \cdot x_j \geq b_i, \ i = 1, 2, \ldots m \]

\[ x_j \geq 0, \ j = 1, 2, \ldots n \]
Primal and Dual forms

Forms

Primal (P):

\[ z = \min \sum_{j=1}^{n} c_j \cdot x_j \]

s.t. \[ \sum_{j=1}^{n} a_{ij} \cdot x_j \geq b_i, \ i = 1, 2, \ldots m \]

\[ x_j \geq 0, \ j = 1, 2, \ldots n \]
Primal and Dual forms

<table>
<thead>
<tr>
<th>Forms</th>
</tr>
</thead>
<tbody>
<tr>
<td>Primal (P):</td>
</tr>
<tr>
<td>$z = \min \sum_{j=1}^{n} c_j \cdot x_j$</td>
</tr>
<tr>
<td>s.t. $\sum_{j=1}^{n} a_{ij} \cdot x_j \geq b_i, \ i = 1, 2, \ldots m$</td>
</tr>
<tr>
<td>$x_j \geq 0, \ j = 1, 2, \ldots n$</td>
</tr>
<tr>
<td>Dual (D):</td>
</tr>
<tr>
<td>$w = \max \sum_{i=1}^{m} b_i \cdot y_i$</td>
</tr>
<tr>
<td>s.t. $\sum_{i=1}^{m} a_{ij} \cdot y_i \leq c_j, \ j = 1, 2, \ldots n$</td>
</tr>
<tr>
<td>$y_i \geq 0, \ i = 1, 2, \ldots m$</td>
</tr>
<tr>
<td>$w = \max b_i \cdot y_i$</td>
</tr>
<tr>
<td>s.t. $y_i \cdot A \leq c, \ y_i \geq 0$</td>
</tr>
</tbody>
</table>
Primal and Dual forms

Forms

Primal (P):

\[ z = \min \sum_{j=1}^{n} c_j \cdot x_j \]
\[ \text{s.t.} \quad \sum_{i=1}^{m} a_{ij} \cdot x_j \geq b_i, \quad i = 1, 2, \ldots, m \]
\[ x_j \geq 0, \quad j = 1, 2, \ldots, n \]

Dual (D):

\[ z = \max \sum_{i=1}^{m} b_i \cdot y_i \]
\[ \text{s.t.} \quad \sum_{j=1}^{n} a_{ij} \cdot y_i \leq c_j, \quad j = 1, 2, \ldots, n \]
\[ y_i \geq 0, \quad i = 1, 2, \ldots, m \]
Primal and Dual forms

**Primal (P):**

\[
\begin{align*}
  z &= \min \sum_{j=1}^{n} c_j \cdot x_j \\
  \text{s.t.} \quad \sum_{j=1}^{n} a_{ij} \cdot x_j &\geq b_i, \ i = 1, 2, \ldots, m \\
  x_j &\geq 0, \ j = 1, 2, \ldots, n
\end{align*}
\]

**Dual (D):**

\[
\begin{align*}
  z &= \max \sum_{i=1}^{m} b_i \cdot y_i \\
  \text{s.t.} \quad \sum_{i=1}^{m} a_{ij} \cdot y_i &\leq c_j, \ j = 1, 2, \ldots, n \\
  y_i &\geq 0, \ i = 1, 2, \ldots, m
\end{align*}
\]
Primal and Dual forms

Forms

Primal (P):

\[
\begin{align*}
z & = \min \sum_{j=1}^{n} c_j \cdot x_j \\
s.t. & \quad \sum_{j=1}^{n} a_{ij} \cdot x_j \geq b_i, \quad i = 1, 2, \ldots m \\
& \quad x_j \geq 0, \quad j = 1, 2, \ldots n
\end{align*}
\]

Dual (D):

\[
\begin{align*}
w & = \max \sum_{i=1}^{m} b_i \cdot y_i
\end{align*}
\]
Primal and Dual forms

**Primal (P):**

\[
    z = \min \sum_{j=1}^{n} c_j \cdot x_j
\]

s.t.

\[
    \sum_{j=1}^{n} a_{ij} \cdot x_j \geq b_i, \ i = 1, 2, \ldots m
\]

\[
    x_j \geq 0, \ j = 1, 2, \ldots n
\]

**Dual (D):**

\[
    w = \max \sum_{i=1}^{m} b_i \cdot y_i
\]

subject to

\[
    \sum_{i=1}^{m} a_{ij} \cdot y_i \leq c_j, \ j = 1, 2, \ldots n
\]
Primal and Dual forms

Primal (P):

\[ z = \min \sum_{j=1}^{n} c_j \cdot x_j \]
\[ \text{s.t. } \sum_{j=1}^{n} a_{ij} \cdot x_j \geq b_i, \quad i = 1, 2, \ldots, m \]
\[ x_j \geq 0, \quad j = 1, 2, \ldots, n \]

\[ z = \min c \cdot x \]
\[ \text{s.t. } A \cdot x \geq b \]
\[ x \geq 0 \]

Dual (D):

\[ w = \max \sum_{i=1}^{m} b_i \cdot y_i \]
\[ \text{subject to } \sum_{i=1}^{m} a_{ij} \cdot y_i \leq c_j, \quad j = 1, 2, \ldots, n \]
\[ y_i \geq 0, \quad i = 1, 2, \ldots, m \]
Primal and Dual forms

**Primal (P):**

\[ z = \min \sum_{j=1}^{n} c_j \cdot x_j \]

s.t.

\[ \sum_{j=1}^{n} a_{ij} \cdot x_j \geq b_i, \quad i = 1, 2, \ldots, m \]

\[ x_j \geq 0, \quad j = 1, 2, \ldots, n \]

**Dual (D):**

\[ w = \max \sum_{i=1}^{m} b_i \cdot y_i \]

subject to

\[ \sum_{i=1}^{m} a_{ij} \cdot y_i \leq c_j, \quad j = 1, 2, \ldots, n \]

\[ y_i \geq 0, \quad i = 1, 2, \ldots, m \]

**Primal (P):**

\[ z = \min c \cdot x \]

s.t.

\[ A \cdot x \geq b \]

\[ x \geq 0 \]

**Dual (D):**

\[ w = \max b \cdot y \]
## Primal and Dual forms

### Primal (P):

$$
\begin{align*}
z &= \min \sum_{j=1}^{n} c_j \cdot x_j \\
\text{s.t.} \quad \sum_{j=1}^{n} a_{ij} \cdot x_j &\geq b_i, \quad i = 1, 2, \ldots, m \\
x_j &\geq 0, \quad j = 1, 2, \ldots, n
\end{align*}
$$

$$
\begin{align*}
z &= \min c \cdot x \\
\text{s.t.} \quad A \cdot x &\geq b \\
x &\geq 0
\end{align*}
$$

### Dual (D):

$$
\begin{align*}
w &= \max \sum_{i=1}^{m} b_i \cdot y_i \\
\text{subject to} \quad \sum_{i=1}^{m} a_{ij} \cdot y_i &\leq c_j, \quad j = 1, 2, \ldots, n \\
y_i &\geq 0, \quad i = 1, 2, \ldots, m
\end{align*}
$$

$$
\begin{align*}
w &= \max b \cdot y \\
\text{s.t.} \quad y \cdot A &\leq c
\end{align*}
$$
Primal and Dual forms

<table>
<thead>
<tr>
<th>Forms</th>
<th>Primal (P):</th>
<th>Dual (D):</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>[ z = \min \sum_{j=1}^{n} c_j \cdot x_j ]</td>
<td>[ z = \min c \cdot x ]</td>
</tr>
<tr>
<td></td>
<td>s.t. [ \sum_{j=1}^{n} a_{ij} \cdot x_j \geq b_i, i = 1,2,\ldots m ]</td>
<td>s.t. [ A \cdot x \geq b ]</td>
</tr>
<tr>
<td></td>
<td>[ x_j \geq 0, j = 1,2,\ldots n ]</td>
<td>[ x \geq 0 ]</td>
</tr>
<tr>
<td></td>
<td>[ w = \max \sum_{i=1}^{m} b_i \cdot y_i ]</td>
<td>[ w = \max b \cdot y ]</td>
</tr>
<tr>
<td></td>
<td>subject to [ \sum_{i=1}^{m} a_{ij} \cdot y_i \leq c_j, j = 1,2,\ldots n ]</td>
<td>s.t. [ y \cdot A \leq c ]</td>
</tr>
<tr>
<td></td>
<td>[ y_i \geq 0, i = 1,2,\ldots m ]</td>
<td>[ y \geq 0 ]</td>
</tr>
</tbody>
</table>
Complementary Slackness

Let $x$ and $y$ be primal and dual feasible solutions, respectively. Then, $x$ and $y$ are both optimal iff all of the following conditions are satisfied:

1. **Primal Complementary Slackness:** For each $1 \leq j \leq n$:
   
   either $x_j = 0$, or  
   
   $\sum_{i=1}^{m} a_{ij} \cdot y_i = c_j$.

2. **Dual Complementary Slackness:** For each $1 \leq i \leq m$:
   
   either $y_i = 0$, or  
   
   $\sum_{j=1}^{n} a_{ij} \cdot x_j = b_i$.
Complementary Slackness

**Theorem**

Let \( \mathbf{x} \) and \( \mathbf{y} \) be primal and dual feasible solutions, respectively.
Let $x$ and $y$ be primal and dual feasible solutions, respectively. Then, $x$ and $y$ are both optimal iff all of the following conditions are satisfied:
Complementary Slackness

Theorem

Let $\mathbf{x}$ and $\mathbf{y}$ be primal and dual feasible solutions, respectively. Then, $\mathbf{x}$ and $\mathbf{y}$ are both optimal iff all of the following conditions are satisfied:
### Theorem

Let \( \mathbf{x} \) and \( \mathbf{y} \) be primal and dual feasible solutions, respectively. Then, \( \mathbf{x} \) and \( \mathbf{y} \) are both optimal iff all of the following conditions are satisfied:

1. **Primal Complementary Slackness:**
   
   For each \( 1 \leq j \leq n \):
   
   \[
   x_j = 0, \quad \text{or} \quad \sum_{i=1}^{m} a_{ij} y_i = c_j
   \]

2. **Dual Complementary Slackness:**
   
   For each \( 1 \leq i \leq m \):
   
   \[
   y_i = 0, \quad \text{or} \quad \sum_{j=1}^{n} a_{ij} x_j = b_i
   \]
Let \( x \) and \( y \) be primal and dual feasible solutions, respectively. Then, \( x \) and \( y \) are both optimal iff all of the following conditions are satisfied:

1. **Primal Complementary Slackness:**

   \[
   \text{For each } 1 \leq j \leq n:
   \]

   \[
   x_j = 0, \quad \text{or} \quad \sum_{i=1}^{m} a_{ij} y_i = c_j
   \]
Complementary Slackness

Theorem

Let \( \mathbf{x} \) and \( \mathbf{y} \) be primal and dual feasible solutions, respectively. Then, \( \mathbf{x} \) and \( \mathbf{y} \) are both optimal iff all of the following conditions are satisfied:

1. **Primal Complementary Slackness:**
   
   For each \( 1 \leq j \leq n \) : either \( x_j = 0 \),

   For each \( 1 \leq i \leq m \): either \( y_i = 0 \),

   \[ \sum_{j=1}^{n} a_{ij} x_j = c_j \]

   \[ \sum_{j=1}^{n} a_{ij} y_i = b_i \]
Let \( x \) and \( y \) be primal and dual feasible solutions, respectively. Then, \( x \) and \( y \) are both optimal iff all of the following conditions are satisfied:

1. **Primal Complementary Slackness:**
   
   For each \( 1 \leq j \leq n \): either \( x_j = 0 \), or \( \sum_{i=1}^{m} a_{ij} \cdot y_i = c_j \)
Complementary Slackness

Theorem

Let \( x \) and \( y \) be primal and dual feasible solutions, respectively. Then, \( x \) and \( y \) are both optimal iff all of the following conditions are satisfied:

1. **Primal Complementary Slackness:**

   For each \( 1 \leq j \leq n \): either \( x_j = 0 \), or \( \sum_{i=1}^{m} a_{ij} \cdot y_i = c_j \)

2. **Dual Complementary Slackness:**

   For each \( 1 \leq i \leq m \):
Complementary Slackness

Theorem

Let \( x \) and \( y \) be primal and dual feasible solutions, respectively. Then, \( x \) and \( y \) are both optimal iff all of the following conditions are satisfied:

1. **Primal Complementary Slackness:**
   
   For each \( 1 \leq j \leq n \): either \( x_j = 0 \), or \( \sum_{i=1}^{m} a_{ij} \cdot y_i = c_j \)

2. **Dual Complementary Slackness:**
   
   For each \( 1 \leq i \leq m \): either \( y_i = 0 \),
Theorem

Let \( x \) and \( y \) be primal and dual feasible solutions, respectively. Then, \( x \) and \( y \) are both optimal iff all of the following conditions are satisfied:

1. **Primal Complementary Slackness:**
   
   For each \( 1 \leq j \leq n \): either \( x_j = 0 \), or \( \sum_{i=1}^{m} a_{ij} \cdot y_i = c_j \)

2. **Dual Complementary Slackness:**
   
   For each \( 1 \leq i \leq m \): either \( y_i = 0 \), or \( \sum_{j=1}^{n} a_{ij} \cdot x_j = b_i \).
Proof of Complementary Slackness

Proof.

(i) Recall that the primal is \( \min A \cdot x \geq b, x \geq 0 \) and the dual is \( \max y \cdot A \leq c, y \geq 0 \).

(ii) Let \((x^*, y^*)\) denote an optimal primal-dual pair.

(iii) Define \( t^* = A \cdot x^* - b \) (surplus) and \( s^* = c - y^* \cdot A \) (slack).

Clearly, \( t^* \geq 0 \) and \( s^* \geq 0 \).

(iv) The complementary slackness conditions can be rewritten as:

- **Primal**: \( \forall i, 1 \leq i \leq n \), either \( x_i = 0 \) or \( s_i = 0 \),
- **Dual**: \( \forall j, 1 \leq j \leq m \), either \( y_j = 0 \) or \( t_j = 0 \).

(v) We have, \( c \cdot x^* = (s^* + y^* \cdot A) \cdot x^* = s^* x^* + y^* \cdot A \cdot x^* = s^* x^* + y^* \cdot (t^* + b) = s^* x^* + y^* b \).
Proof of Complementary Slackness

Proof.

(i) Recall that the primal is \( \min A \cdot x \geq b, x \geq 0 \) and the dual is \( \max y \cdot A \leq c, y \geq 0 \).

(ii) Let \((x^*, y^*)\) denote an optimal primal-dual pair.

(iii) Define \( t^* = A \cdot x^* - b \) (surplus) and \( s^* = c - y^* \cdot A \) (slack).

Clearly, \( t^* \geq 0 \) and \( s^* \geq 0 \).

(iv) The complementary slackness conditions can be rewritten as:

Primal: \( \forall i, 1 \leq i \leq n, \text{ either } x_i = 0 \text{ or } s_i = 0 \),

Dual: \( \forall j, 1 \leq j \leq m, \text{ either } y_j = 0 \text{ or } t_j = 0 \).

(v) We have,

\[ c \cdot x^* = (s^* + y^* \cdot A) \cdot x^* = s^* x^* + y^* \cdot (t^* + b) = s^* x^* + y^* b \]
Proof of Complementary Slackness

Proof.

(i) Recall that the primal is $\min_{A \cdot x \geq b, x \geq 0} c \cdot x$
Proof of Complementary Slackness

Proof.

(i) Recall that the primal is $\min_{A \cdot x \geq b, x \geq 0} c \cdot x$ and the dual is $\max_{y : A \leq c, y \geq 0} b \cdot y$. 

(ii) Let $(x^*, y^*)$ denote an optimal primal-dual pair.

(iii) Define $t^* = A \cdot x^* - b$ (surplus) and $s^* = c - y^* \cdot A$ (slack).

Clearly, $t^* \geq 0$ and $s^* \geq 0$.

(iv) The complementary slackness conditions can be rewritten as:

Primal: $\forall i, 1 \leq i \leq n$, either $x_i = 0$ or $s_i = 0$, and

Dual: $\forall j, 1 \leq j \leq m$, either $y_j = 0$ or $t_j = 0$.

(v) We have, $c \cdot x^* = (s^* + y^* \cdot A) \cdot x^* = s^* x^* + y^* \cdot A \cdot t^* + y^* b$.
Proof of Complementary Slackness

Proof.

(i) Recall that the primal is \( \min_{A \cdot x \geq b, x \geq 0} c \cdot x \) and the dual is \( \max_{y \cdot A \leq c, y \geq 0} b \cdot y \).

(ii) Let \((x^*, y^*)\) denote an optimal primal-dual pair.
Proof of Complementary Slackness

Proof.

(i) Recall that the primal is \( \min_{A \cdot x \geq b, x \geq 0} c \cdot x \) and the dual is \( \max_{y \cdot A \leq c, y \geq 0} b \cdot y \).

(ii) Let \((x^*, y^*)\) denote an optimal primal-dual pair.

(iii) Define \( t^* = A \cdot x^* - b \) (surplus) and \( s^* = c - y^* \cdot A \) (slack).
Proof of Complementary Slackness

Proof.

(i) Recall that the primal is $\min_{A \cdot x \geq b, x \geq 0} c \cdot x$ and the dual is $\max_{y, A \cdot y \leq c, y \geq 0} b \cdot y$.

(ii) Let $(x^*, y^*)$ denote an optimal primal-dual pair.

(iii) Define $t^* = A \cdot x^* - b$ (surplus) and $s^* = c - y^* \cdot A$ (slack).

Clearly, $t^* \geq 0$ and $s^* \geq 0$. 
Proof of Complementary Slackness

Proof.

(i) Recall that the primal is \( \min_{A \cdot x \geq b, x \geq 0} c \cdot x \) and the dual is \( \max_{y : A \leq c, y \geq 0} b \cdot y \).

(ii) Let \((x^*, y^*)\) denote an optimal primal-dual pair.

(iii) Define \( t^* = A \cdot x^* - b \) (surplus) and \( s^* = c - y^* \cdot A \) (slack).
Clearly, \( t^* \geq 0 \) and \( s^* \geq 0 \).

(iv) The complementary slackness conditions can be rewritten as:
Proof of Complementary Slackness

Proof.

(i) Recall that the primal is $\min_{A \cdot x \geq b, \ x \geq 0} \ c \cdot x$ and the dual is $\max_{y \cdot A \leq c, \ y \geq 0} \ b \cdot y$.

(ii) Let $(x^*, y^*)$ denote an optimal primal-dual pair.

(iii) Define $t^* = A \cdot x^* - b$ (surplus) and $s^* = c - y^* \cdot A$ (slack). Clearly, $t^* \geq 0$ and $s^* \geq 0$.

(iv) The complementary slackness conditions can be rewritten as: 

Primal: $\forall i, \ 1 \leq i \leq n$, either $x_i = 0$ or $s_i = 0$, 

Dual: $\forall j, \ 1 \leq j \leq m$, either $y_j = 0$ or $t_j = 0$. 

Proof of Complementary Slackness

Proof.

(i) Recall that the primal is \( \min_{A \cdot x \geq b, x \geq 0} c \cdot x \) and the dual is \( \max_{y \cdot A \leq c, y \geq 0} b \cdot y \).

(ii) Let \( (x^*, y^*) \) denote an optimal primal-dual pair.

(iii) Define \( t^* = A \cdot x^* - b \) (surplus) and \( s^* = c - y^* \cdot A \) (slack).

Clearly, \( t^* \geq 0 \) and \( s^* \geq 0 \).

(iv) The complementary slackness conditions can be rewritten as:

**Primal:** \( \forall i, \ 1 \leq i \leq n, \) either \( x_i = 0 \) or \( s_i = 0 \), and
Proof of Complementary Slackness

Proof.

(i) Recall that the primal is $\min_{A \cdot x \geq b, x \geq 0} c \cdot x$ and the dual is $\max_{y \cdot A \leq c, y \geq 0} b \cdot y$.

(ii) Let $(x^*, y^*)$ denote an optimal primal-dual pair.

(iii) Define $t^* = A \cdot x^* - b$ (surplus) and $s^* = c - y^* \cdot A$ (slack).
Clearly, $t^* \geq 0$ and $s^* \geq 0$.

(iv) The complementary slackness conditions can be rewritten as:
Primal: $\forall i, 1 \leq i \leq n$, either $x_i = 0$ or $s_i = 0$, and
Dual: $\forall j, 1 \leq j \leq m$, either $y_j = 0$ or $t_j = 0$. 
Proof of Complementary Slackness

Proof.

(i) Recall that the primal is \( \min_{A \cdot x \geq b, x \geq 0} c \cdot x \) and the dual is \( \max_{y \cdot A \leq c, y \geq 0} b \cdot y \).

(ii) Let \((x^*, y^*)\) denote an optimal primal-dual pair.

(iii) Define \( t^* = A \cdot x^* - b \) (surplus) and \( s^* = c - y^* \cdot A \) (slack). Clearly, \( t^* \geq 0 \) and \( s^* \geq 0 \).

(iv) The complementary slackness conditions can be rewritten as:

- **Primal**: \( \forall i, 1 \leq i \leq n, \) either \( x_i = 0 \) or \( s_i = 0 \), and
- **Dual**: \( \forall j, 1 \leq j \leq m, \) either \( y_j = 0 \) or \( t_j = 0 \).

(v) We have,
Proof of Complementary Slackness

Proof.

(i) Recall that the primal is \( \min_{A \cdot x \geq b, x \geq 0} c \cdot x \) and the dual is \( \max_{y \cdot A \leq c, y \geq 0} b \cdot y \).

(ii) Let \((x^*, y^*)\) denote an optimal primal-dual pair.

(iii) Define \( t^* = A \cdot x^* - b \) (surplus) and \( s^* = c - y^* \cdot A \) (slack). Clearly, \( t^* \geq 0 \) and \( s^* \geq 0 \).

(iv) The complementary slackness conditions can be rewritten as:

Primal: \( \forall i, 1 \leq i \leq n, \) either \( x_i = 0 \) or \( s_i = 0 \), and

Dual: \( \forall j, 1 \leq j \leq m, \) either \( y_j = 0 \) or \( t_j = 0 \).

(v) We have,

\[
c \cdot x^* = \]

Proof of Complementary Slackness

Proof.

(i) Recall that the primal is \( \min_{A \cdot x \geq b, x \geq 0} c \cdot x \) and the dual is \( \max_{y \cdot A \leq c, y \geq 0} b \cdot y \).

(ii) Let \((x^*, y^*)\) denote an optimal primal-dual pair.

(iii) Define \( t^* = A \cdot x^* - b \) (surplus) and \( s^* = c - y^* \cdot A \) (slack). Clearly, \( t^* \geq 0 \) and \( s^* \geq 0 \).

(iv) The complementary slackness conditions can be rewritten as:

**Primal:** \( \forall i, 1 \leq i \leq n, \) either \( x_i = 0 \) or \( s_i = 0 \), and

**Dual:** \( \forall j, 1 \leq j \leq m, \) either \( y_j = 0 \) or \( t_j = 0 \).

(v) We have,

\[
c \cdot x^* = (s^* + y^* \cdot A) \cdot x^*
\]
Proof of Complementary Slackness

Proof.

(i) Recall that the primal is \( \min_{A \cdot x \geq b, x \geq 0} c \cdot x \) and the dual is \( \max_{y \cdot A \leq c, y \geq 0} b \cdot y \).

(ii) Let \((x^*, y^*)\) denote an optimal primal-dual pair.

(iii) Define \( t^* = A \cdot x^* - b \) (surplus) and \( s^* = c - y^* \cdot A \) (slack).

Clearly, \( t^* \geq 0 \) and \( s^* \geq 0 \).

(iv) The complementary slackness conditions can be rewritten as:

**Primal:** \( \forall i, 1 \leq i \leq n, \) either \( x_i = 0 \) or \( s_i = 0 \), and

**Dual:** \( \forall j, 1 \leq j \leq m, \) either \( y_j = 0 \) or \( t_j = 0 \).

(v) We have,

\[
c \cdot x^* = (s^* + y^* \cdot A) \cdot x^*
\]

\[
= s^* x^* + y^* \cdot A \cdot x^*
\]
Proof of Complementary Slackness

Proof.

(i) Recall that the primal is \( \min_{A \cdot x \geq b, x \geq 0} c \cdot x \) and the dual is \( \max_{y \cdot A \leq c, y \geq 0} b \cdot y \).

(ii) Let \((x^*, y^*)\) denote an optimal primal-dual pair.

(iii) Define \( t^* = A \cdot x^* - b \) (surplus) and \( s^* = c - y^* \cdot A \) (slack). Clearly, \( t^* \geq 0 \) and \( s^* \geq 0 \).

(iv) The complementary slackness conditions can be rewritten as:

- **Primal:** \( \forall i, 1 \leq i \leq n \), either \( x_i = 0 \) or \( s_i = 0 \), and
- **Dual:** \( \forall j, 1 \leq j \leq m \), either \( y_j = 0 \) or \( t_j = 0 \).

(v) We have,

\[
\begin{align*}
    c \cdot x^* &= (s^* + y^* \cdot A) \cdot x^* \\
    &= s^* x^* + y^* \cdot A \cdot x^*
    = s^* x^* + y^* \cdot (t^* + b)
\end{align*}
\]
Proof of Complementary Slackness

Proof.

(i) Recall that the primal is $\min_{A \cdot x \geq b, x \geq 0} \ c \cdot x$ and the dual is $\max_{y \cdot A \leq c, y \geq 0} \ b \cdot y$.

(ii) Let $(x^*, y^*)$ denote an optimal primal-dual pair.

(iii) Define $t^* = A \cdot x^* - b$ (surplus) and $s^* = c - y^* \cdot A$ (slack).

Clearly, $t^* \geq 0$ and $s^* \geq 0$.

(iv) The complementary slackness conditions can be rewritten as:

- **Primal**: $\forall i, 1 \leq i \leq n$, either $x_i = 0$ or $s_i = 0$, and
- **Dual**: $\forall j, 1 \leq j \leq m$, either $y_j = 0$ or $t_j = 0$.

(v) We have,

\[
\begin{align*}
    c \cdot x^* &= (s^* + y^* \cdot A) \cdot x^* \\
    &= s^* x^* + y^* \cdot A \cdot x^* \\
    &= s^* x^* + y^* \cdot (t^* + b) \\
    &= s^* x^* + y^* \cdot t^* + y^* b
\end{align*}
\]
Proof of Complementary Slackness

(i) Recall that the primal is \(\min_{A \cdot x \geq b, x \geq 0} c \cdot x\) and the dual is \(\max_{y \cdot A \leq c, y \geq 0} b \cdot y\).

(ii) Let \((x^*, y^*)\) denote an optimal primal-dual pair.

(iii) Define \(t^* = A \cdot x^* - b\) (surplus) and \(s^* = c - y^* \cdot A\) (slack). Clearly, \(t^* \geq 0\) and \(s^* \geq 0\).

(iv) The complementary slackness conditions can be rewritten as:

**Primal:** \(\forall i, 1 \leq i \leq n\), either \(x_i = 0\) or \(s_i = 0\), and

**Dual:** \(\forall j, 1 \leq j \leq m\), either \(y_j = 0\) or \(t_j = 0\).

(v) We have,

\[
c \cdot x^* = (s^* + y^* \cdot A) \cdot x^*
= s^* x^* + y^* \cdot A \cdot x^*
= s^* x^* + y^* \cdot (t^* + b)
= s^* x^* + y^* \cdot t^* + y^* b
\]
Proof of complementary slackness
Proof of complementary slackness

Proof.

1. But \( c \cdot x^* = y^* \cdot b \).
2. It follows that, \( s^* x^* + y^* t^* = 0 \).
3. Hence, \( s^* x^* = 0 \) and \( y^* t^* = 0 \), since \( x^*, y^*, s^*, t^* \geq 0 \).
4. Hence for \( 1 \leq i \leq n \), \( x_i \cdot s_i = 0 \), i.e., either \( x_i = 0 \) or \( s_i = 0 \).
5. Likewise, for \( 1 \leq j \leq n \), either \( y_j = 0 \) or \( t_j = 0 \).
Proof of complementary slackness

Proof.

\[ c \cdot x^* = y^* \cdot b. \]

It follows that,

\[ s^* x^* + y^* t^* = 0. \]

Hence,

\[ s^* x^* = 0 \quad \text{and} \quad y^* t^* = 0, \]

since \( x^*, y^*, s^*, t^* \geq 0 \).

Hence for \( 1 \leq i \leq n \),

\[ x_i \cdot s_i = 0, \text{ i.e., either } x_i = 0 \text{ or } s_i = 0. \]

Likewise, for \( 1 \leq j \leq n \), either \( y_j = 0 \) or \( t_j = 0 \).
Proof of complementary slackness

Proof.

1. But \( c \cdot x^* = y^* \cdot b \).
Proof of complementary slackness

Proof.

1. But $c \cdot x^* = y^* \cdot b$.
2. It follows that,
Proof of complementary slackness

Proof.

1. But \( c \cdot x^* = y^* \cdot b \).
2. It follows that, \( s^* x^* + y^* \cdot t^* = 0 \).
Proof of complementary slackness

Proof.

1. But $c \cdot x^* = y^* \cdot b$.
2. It follows that, $s^*x^* + y^* \cdot t^* = 0$.
3. Hence,
Proof of complementary slackness

Proof.

1. But $c \cdot x^* = y^* \cdot b$.
2. It follows that, $s^* x^* + y^* \cdot t^* = 0$.
3. Hence, $s^* \cdot x^* = 0$.
Proof.

1. But \( c \cdot x^* = y^* \cdot b \).
2. It follows that, \( s^* x^* + y^* \cdot t^* = 0 \).
3. Hence, \( s^* \cdot x^* = 0 \) and \( y^* \cdot t^* = 0 \),
Proof of complementary slackness

Proof.

1. But \( c \cdot x^* = y^* \cdot b \).
2. It follows that, \( s^* x^* + y^* \cdot t^* = 0 \).
3. Hence, \( s^* \cdot x^* = 0 \) and \( y^* \cdot t^* = 0 \), since \( x^*, y^*, s^*, t^* \geq 0 \).
Proof of complementary slackness

**Proof.**

1. But \( c \cdot x^* = y^* \cdot b \).
2. It follows that, \( s^* x^* + y^* \cdot t^* = 0 \).
3. Hence, \( s^* \cdot x^* = 0 \) and \( y^* \cdot t^* = 0 \), since \( x^*, y^*, s^*, t^* \geq 0 \).
4. Hence for \( 1 \leq i \leq n, x_i \cdot s_i = 0 \), i.e., either \( x_i = 0 \) or \( s_i = 0 \).
Proof of complementary slackness

Proof.

1. But $c \cdot x^* = y^* \cdot b$.
2. It follows that, $s^* x^* + y^* \cdot t^* = 0$.
3. Hence, $s^* \cdot x^* = 0$ and $y^* \cdot t^* = 0$, since $x^*, y^*, s^*, t^* \geq 0$.
4. Hence for $1 \leq i \leq n$, $x_i \cdot s_i = 0$, i.e., either $x_i = 0$ or $s_i = 0$.
5. Likewise, for $1 \leq j \leq n$, either $y_j = 0$ or $t_j = 0$. 
Interpretation of complementary slackness

(1) If a primal variable \( x^* \) > 0, then the corresponding dual constraint must be binding, i.e., \( s^* \) = 0.

(2) If a dual constraint is not binding, i.e., \( s^* \) > 0, then the corresponding primal variable \( x^* \) must be 0.

(3) If a dual variable \( y^* \) > 0, then the corresponding primal constraint must be binding, i.e., \( t^* \) = 0.

(4) If a primal constraint is non-binding, i.e., \( t^* \) > 0, then the corresponding dual variable \( y^* \) must be zero.
Interpretation of complementary slackness

1. If a primal variable $x^* i > 0$, then the corresponding dual constraint must be binding, i.e., $s^* i = 0$.
2. If a dual constraint is not binding, i.e., $s^* i > 0$, then the corresponding primal variable ($x^* i$) must be 0.
3. If a dual variable $y^* i > 0$, then the corresponding primal constraint must be binding, i.e., $t^* i = 0$.
4. If a primal constraint is non-binding, i.e., $t^* i > 0$, then the corresponding dual variable ($y^* i$) must be zero.
Interpretation of complementary slackness

Interpretation

(1) If a primal variable $x_i^* > 0$, then the corresponding dual constraint must be \textbf{binding}, i.e., $s_i^* = 0$. 
Interpretation of complementary slackness

<table>
<thead>
<tr>
<th>Interpretation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1) If a primal variable $x_i^* &gt; 0$, then the corresponding dual constraint must be <strong>binding</strong>, i.e., $s_i^* = 0$.</td>
<td></td>
</tr>
<tr>
<td>(2) If a dual constraint is not binding, i.e., $s_i^* &gt; 0$, then the corresponding primal variable $(x_i^*)$ must be 0.</td>
<td></td>
</tr>
</tbody>
</table>
Interpretation of complementary slackness

Interpretation

1. If a primal variable $x_i^* > 0$, then the corresponding dual constraint must be binding, i.e., $s_i^* = 0$.

2. If a dual constraint is not binding, i.e., $s_i^* > 0$, then the corresponding primal variable ($x_i^*$) must be 0.

3. If a dual variable $y_i^* > 0$, then the corresponding primal constraint must be binding, i.e., $t_i^* = 0$. 

Interpretation of complementary slackness

<table>
<thead>
<tr>
<th>Interpretation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1</strong></td>
<td>If a primal variable $x_i^* &gt; 0$, then the corresponding dual constraint must be <strong>binding</strong>, i.e., $s_i^* = 0$.</td>
</tr>
<tr>
<td><strong>2</strong></td>
<td>If a dual constraint is not binding, i.e., $s_i^* &gt; 0$, then the corresponding primal variable ($x_i^*$) must be 0.</td>
</tr>
<tr>
<td><strong>3</strong></td>
<td>If a dual variable $y_i^* &gt; 0$, then the corresponding primal constraint must be <strong>binding</strong>, i.e., $t_i^* = 0$.</td>
</tr>
<tr>
<td><strong>4</strong></td>
<td>If a primal constraint is non-binding, i.e., $t_i^* &gt; 0$, then the corresponding dual variable ($y_i^*$) must be zero.</td>
</tr>
</tbody>
</table>
Relaxed complementary slackness conditions

Let $\alpha \geq 1$. For all $j$ with $1 \leq j \leq n$:
- either $x_j = 0$ or $c_j \alpha \leq \sum_{i=1}^{m} a_{ij} \cdot y_i \leq c_j$.

Relaxed dual conditions

Let $\beta \geq 1$. For all $i$ with $1 \leq i \leq m$:
- either $y_i = 0$ or $b_i \leq \sum_{j=1}^{n} a_{ij} \cdot x_j \leq \beta \cdot b_i$. 
Relaxed complementary slackness conditions

Relaxed primal conditions

Let \( \alpha \geq 1 \).
\[ \forall j \leq j \leq n: \text{either } x_j = 0 \text{ or } c_j \alpha \leq m \sum_{i=1}^n a_{ij} \cdot y_i \leq c_j. \]

Relaxed dual conditions

Let \( \beta \geq 1 \).
\[ \forall i \leq i \leq m: \text{either } y_i = 0 \text{ or } b_i \leq n \sum_{j=1}^n a_{ij} \cdot x_j \leq \beta \cdot b_i. \]
Relaxed complementary slackness conditions

Relaxed primal conditions

Let $\alpha \geq 1$. 

Relaxed dual conditions

Let $\beta \geq 1$. 

Relaxed complementary slackness conditions

Relaxed primal conditions

Let $\alpha \geq 1$.

$$\forall j \ 1 \leq j \leq n \ : \ \text{either } x_j = 0 \ \text{or}$$
Relaxed complementary slackness conditions

Relaxed primal conditions

Let $\alpha \geq 1$.

\[ \forall j \ 1 \leq j \leq n : \text{either } x_j = 0 \text{ or } \frac{c_j}{\alpha} \leq \sum_{i=1}^{m} a_{ij} \cdot y_i \leq c_j. \]
Relaxed complementary slackness conditions

Relaxed primal conditions

Let $\alpha \geq 1$.

\[ \forall j \ 1 \leq j \leq n : \text{ either } x_j = 0 \text{ or } \frac{c_j}{\alpha} \leq \sum_{i=1}^{m} a_{ij} \cdot y_i \leq c_j. \]

Relaxed dual conditions
Relaxed complementary slackness conditions

**Relaxed primal conditions**

Let $\alpha \geq 1$.

\[
\forall j \ 1 \leq j \leq n : \text{either } x_j = 0 \text{ or } \frac{c_j}{\alpha} \leq \sum_{i=1}^{m} a_{ij} \cdot y_i \leq c_j.
\]

**Relaxed dual conditions**

Let $\beta \geq 1$.

\[
\forall i \ 1 \leq i \leq m : \text{either } y_i = 0 \text{ or } b_i \leq \sum_{j=1}^{n} a_{ij} \cdot x_j \leq \beta \cdot b_i.
\]
Relaxed complementary slackness conditions

Relaxed primal conditions
Let $\alpha \geq 1$.

$$\forall j \ 1 \leq j \leq n : \text{either } x_j = 0 \text{ or } \frac{c_j}{\alpha} \leq \sum_{i=1}^{m} a_{ij} \cdot y_i \leq c_j.$$ 

Relaxed dual conditions
Let $\beta \geq 1$.

$$\forall i \ 1 \leq i \leq m : \text{either } y_i = 0 \text{ or }$$
Relaxed complementary slackness conditions

Relaxed primal conditions

Let $\alpha \geq 1$.

\[ \forall j \ 1 \leq j \leq n : \text{ either } x_j = 0 \text{ or } \frac{c_j}{\alpha} \leq \sum_{i=1}^{m} a_{ij} \cdot y_i \leq c_j. \]

Relaxed dual conditions

Let $\beta \geq 1$.

\[ \forall i \ 1 \leq i \leq m : \text{ either } y_i = 0 \text{ or } b_i \leq \sum_{j=1}^{n} a_{ij} \cdot x_j \leq \beta \cdot b_i. \]
Application to approximation algorithms

Lemma (Main Lemma)

If $x$ and $y$ are primal and dual feasible solutions satisfying the relaxed complementary slackness conditions, then

$$\sum_{i=1}^{n} c_j \cdot x_j \leq \alpha \cdot \beta \cdot \sum_{i=1}^{m} b_i \cdot y_i.$$
Application to approximation algorithms

Lemma (Main Lemma)

If $x$ and $y$ are primal and dual feasible solutions satisfying the relaxed complementary slackness conditions, then
Application to approximation algorithms

Lemma (Main Lemma)

If $\mathbf{x}$ and $\mathbf{y}$ are primal and dual feasible solutions satisfying the relaxed complementary slackness conditions, then

$$
\sum_{i=1}^{n} c_j \cdot x_j \leq \alpha \cdot \beta \cdot \sum_{i=1}^{m} b_i \cdot y_i.
$$
Proof of Main Lemma
Proof of Main Lemma

Proof.
Proof of Main Lemma

Proof.
Proof of Main Lemma

Proof.

\[ \sum_{j=1}^{n} c_j \cdot x_j \leq \]
Proof of Main Lemma

Proof.

$$\sum_{j=1}^{n} c_j \cdot x_j \leq \sum_{j=1}^{n} (\alpha \cdot \left( \sum_{i=1}^{m} a_{ij} \cdot y_i \right)) \cdot x_j$$
Proof of Main Lemma

Proof.

\[
\sum_{j=1}^{n} c_j \cdot x_j \leq \sum_{j=1}^{n} \left( \alpha \cdot \left( \sum_{i=1}^{m} a_{ij} \cdot y_i \right) \right) \cdot x_j \\
= \alpha \cdot \left( \sum_{j=1}^{n} \left( \sum_{i=1}^{m} a_{ij} \cdot y_i \right) \right) \cdot x_j
\]
Proof of Main Lemma

Proof.

\[ \sum_{j=1}^{n} c_j \cdot x_j \leq \sum_{j=1}^{n} (\alpha \cdot (\sum_{i=1}^{m} a_{ij} \cdot y_i)) \cdot x_j \]

\[ = \alpha \cdot (\sum_{j=1}^{n} (\sum_{i=1}^{m} a_{ij} \cdot y_i)) \cdot x_j \]

\[ = \alpha \cdot (\sum_{i=1}^{m} (\sum_{j=1}^{n} a_{ij} \cdot x_j)) \cdot y_i \]
Proof.

\[
\sum_{j=1}^{n} c_j \cdot x_j \leq \sum_{j=1}^{n} \left( \alpha \cdot \left( \sum_{i=1}^{m} a_{ij} \cdot y_i \right) \right) \cdot x_j
\]

\[
= \alpha \cdot \left( \sum_{j=1}^{n} \left( \sum_{i=1}^{m} a_{ij} \cdot y_i \right) \right) \cdot x_j
\]

\[
= \alpha \cdot \left( \sum_{i=1}^{m} \left( \sum_{j=1}^{n} a_{ij} \cdot x_j \right) \right) \cdot y_i
\]

\[
\leq \alpha \cdot \left( \sum_{i=1}^{m} \left( \beta \cdot b_i \right) \right) \cdot y_i
\]
Proof.

\[
\sum_{j=1}^{n} c_j \cdot x_j \leq \sum_{j=1}^{n} (\alpha \cdot (\sum_{i=1}^{m} a_{ij} \cdot y_i)) \cdot x_j \\
= \alpha \cdot (\sum_{j=1}^{n} (\sum_{i=1}^{m} a_{ij} \cdot y_i)) \cdot x_j \\
= \alpha \cdot (\sum_{i=1}^{m} (\sum_{j=1}^{n} a_{ij} \cdot x_j)) \cdot y_i \\
\leq \alpha \cdot (\sum_{i=1}^{m} (\beta \cdot b_i)) \cdot y_i \\
= \alpha \cdot \beta \cdot \sum_{i=1}^{m} b_i \cdot y_i.
\]
The primal-dual approach

1. We start with a primal infeasible and dual feasible solution. Usually, $x = 0$ and $y = 0$.
2. Both solutions are improved iteratively. The feasibility of primal is improved and the optimality of the dual is improved.
3. The primal solution is always extended integrally, so the final primal solution is integral.
4. In the end, we want to ensure that a primal feasible solution is obtained, and all relaxed complementarity slackness conditions are met for a suitable choice of $\alpha$ and $\beta$.
5. An approximation algorithm ensures one set of complementary slackness conditions and relaxes the other.
6. If the goal is to ensure that primal conditions are ensured, then we set $\alpha = 1$. If the goal is to ensure that the dual conditions are ensured then $\beta$ is set to 1.
7. The current primal solution is used to determine the improvement to the dual and vice versa.
8. Finally, the cost of the dual solution is used as a lower bound on $\text{OPT}$ and the approximation guarantee of $\alpha \cdot \beta$ is obtained.
The primal-dual approach

Algorithmic procedure

We start with a primal infeasible and dual feasible solution. Usually, $x = 0$ and $y = 0$. Both solutions are improved iteratively. The feasibility of primal is improved and the optimality of the dual is improved. The primal solution is always extended integrally, so the final primal solution is integral. In the end, we want to ensure that a primal feasible solution is obtained, and all relaxed complementarity slackness conditions are met for a suitable choice of $\alpha$ and $\beta$. An approximation algorithm ensures one set of complementary slackness conditions and relaxes the other. If the goal is to ensure that primal conditions are ensured, then we set $\alpha = 1$. If the goal is to ensure that the dual conditions are ensured then $\beta$ is set to 1. The current primal solution is used to determine the improvement to the dual and vice versa. Finally, the cost of the dual solution is used as a lower bound on $OPT$ and the approximation guarantee of $\alpha \cdot \beta$ is obtained.
The primal-dual approach

**Algorithmic procedure**

1. We start with a primal infeasible and dual feasible solution.
The primal-dual approach

Algorithmic procedure

1. We start with a primal infeasible and dual feasible solution. Usually, $x = 0$ and $y = 0$. 
The primal-dual approach

Algorithmic procedure

1. We start with a primal infeasible and dual feasible solution. Usually, $x = 0$ and $y = 0$.
2. Both solutions are improved iteratively.
The primal-dual approach

Algorithmic procedure

1. We start with a primal infeasible and dual feasible solution. Usually, $x = 0$ and $y = 0$.
2. Both solutions are improved iteratively. The feasibility of primal is improved and the optimality of the dual is improved.
The primal-dual approach

Algorithmic procedure

1. We start with a primal infeasible and dual feasible solution. Usually, \( x = 0 \) and \( y = 0 \).
2. Both solutions are improved iteratively. The feasibility of primal is improved and the optimality of the dual is improved.
3. The primal solution is always extended integrally, so the final primal solution is integral.
The primal-dual approach

Algorithmic procedure

1. We start with a primal infeasible and dual feasible solution. Usually, \( x = 0 \) and \( y = 0 \).
2. Both solutions are improved iteratively. The feasibility of primal is improved and the optimality of the dual is improved.
3. The primal solution is **always** extended integrally, so the final primal solution is integral.
4. In the end, we want to ensure that a primal feasible solution is obtained, and all relaxed complementarity slackness conditions are met for a suitable choice of \( \alpha \) and \( \beta \).
The primal-dual approach

Algorithmic procedure

1. We start with a primal infeasible and dual feasible solution. Usually, $x = 0$ and $y = 0$.

2. Both solutions are improved iteratively. The feasibility of primal is improved and the optimality of the dual is improved.

3. The primal solution is always extended integrally, so the final primal solution is integral.

4. In the end, we want to ensure that a primal feasible solution is obtained, and all relaxed complementarity slackness conditions are met for a suitable choice of $\alpha$ and $\beta$.

5. An approximation algorithm ensures one set of complementary slackness conditions and relaxes the other.
The primal-dual approach

Algorithmic procedure

1. We start with a primal infeasible and dual feasible solution. Usually, $x = 0$ and $y = 0$.
2. Both solutions are improved iteratively. The feasibility of primal is improved and the optimality of the dual is improved.
3. The primal solution is always extended integrally, so the final primal solution is integral.
4. In the end, we want to ensure that a primal feasible solution is obtained, and all relaxed complementarity slackness conditions are met for a suitable choice of $\alpha$ and $\beta$.
5. An approximation algorithm ensures one set of complementary slackness conditions and relaxes the other.
6. If the goal is to ensure that primal conditions are ensured, then we set $\alpha = 1$. 

The primal-dual approach

Algorithmic procedure

1. We start with a primal infeasible and dual feasible solution. Usually, $x = 0$ and $y = 0$.
2. Both solutions are improved iteratively. The feasibility of primal is improved and the optimality of the dual is improved.
3. The primal solution is always extended integrally, so the final primal solution is integral.
4. In the end, we want to ensure that a primal feasible solution is obtained, and all relaxed complementarity slackness conditions are met for a suitable choice of $\alpha$ and $\beta$.
5. An approximation algorithm ensures one set of complementary slackness conditions and relaxes the other.
6. If the goal is to ensure that primal conditions are ensured, then we set $\alpha = 1$. If the goal is to ensure that the dual conditions are ensured then $\beta$ is set to 1.
The primal-dual approach

Algorithmic procedure

1. We start with a primal infeasible and dual feasible solution. Usually, $x = 0$ and $y = 0$.
2. Both solutions are improved iteratively. The feasibility of primal is improved and the optimality of the dual is improved.
3. The primal solution is always extended integrally, so the final primal solution is integral.
4. In the end, we want to ensure that a primal feasible solution is obtained, and all relaxed complementarity slackness conditions are met for a suitable choice of $\alpha$ and $\beta$.
5. An approximation algorithm ensures one set of complementary slackness conditions and relaxes the other.
6. If the goal is to ensure that primal conditions are ensured, then we set $\alpha = 1$. If the goal is to ensure that the dual conditions are ensured then $\beta$ is set to 1.
7. The current primal solution is used to determine the improvement to the dual and vice versa.
## The primal-dual approach

### Algorithmic procedure

1. We start with a primal infeasible and dual feasible solution. Usually, $x = 0$ and $y = 0$.
2. Both solutions are improved iteratively. The feasibility of primal is improved and the optimality of the dual is improved.
3. The primal solution is **always** extended integrally, so the final primal solution is integral.
4. In the end, we want to ensure that a primal feasible solution is obtained, and all relaxed complementarity slackness conditions are met for a suitable choice of $\alpha$ and $\beta$.
5. An approximation algorithm ensures one set of complementary slackness conditions and relaxes the other.
6. If the goal is to ensure that primal conditions are ensured, then we set $\alpha = 1$. If the goal is to ensure that the dual conditions are ensured then $\beta$ is set to 1.
7. The current primal solution is used to determine the improvement to the dual and vice versa.
8. Finally, the cost of the dual solution is used as a lower bound on $OPT$ and the approximation guarantee of $\alpha \cdot \beta$ is obtained.
Preliminaries

Primal-Dual Schema

Primal-Dual schema for Set Cover

The Set Cover Problem

Given,

1. A ground set \( U = \{ e_1, e_2, \ldots, e_n \} \),
2. A collection of sets \( S = \{ S_1, S_2, \ldots, S_m \} \) such that \( S_i \subseteq U \) for \( i = 1, 2, \ldots, m \),
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.
Primal-Dual Schema

Primal-Dual schema for Set Cover

Preliminaries

The Set Cover Problem

Given,
The Set Cover Problem

Given,
Preliminaries

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 \(S_P = \{S_1, S_2, \ldots S_m\}\), \(S_i \subseteq U, i = 1, 2, \ldots, m\)
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 \( c : S_i \rightarrow \mathbb{Z}_+ \),
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.
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.

Note

If all weights are unity (or the same),
Primal-Dual Schema

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 \( c : S_i \to \mathbb{Z}_+ \),

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

\[
\begin{align*}
\text{min} & \quad \sum_{S \in S} P_c(S) \cdot x_S \\
\text{subject to} & \quad \sum_{S : e \in S} x_S \geq 1, e \in U \\
& \quad x_S \in \{0, 1\}, S \in S_P
\end{align*}
\]
Formulating the Integer Program

IP formulation

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

**IP formulation**

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

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

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

Note
For our scheme, we choose \( \alpha = 1 \) and \( \beta = f \), where \( f \) is the frequency of the most frequent element.
### The Linear Program relaxation

**Linear Program**

\[
\min \sum_{S \in \mathcal{S}} \sum_{e \in S} \mathbf{c}(S) \cdot x_S
\]

subject to

\[
\sum_{S \in \mathcal{S}} x_S \geq 1, \quad e \in U
\]

\[
x_S \geq 0, \quad S \in \mathcal{S}
\]

**Note**

For our scheme, we choose \( \alpha = 1 \) and \( \beta = f \), where \( f \) is the frequency of the most frequent element.
The Linear Program relaxation

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

$$\min \sum_{S \in \mathcal{P}} c(S) \cdot x_S$$

subject to

$$\sum_{S: e \in S} x_S \geq 1, \quad e \in U$$

Note

For our scheme, we choose $\alpha = 1$ and $\beta = f$, where $f$ is the frequency of the most frequent element.
The Linear Program relaxation

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

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

Note

For our scheme, we choose \( \alpha = 1 \) and \( \beta = f \), where \( f \) is the frequency of the most frequent element.
The Linear Program relaxation

\[ \min \sum_{S \in \mathcal{S}_P} c(S) \cdot x_S \]
subject to
\[ \sum_{e \in S} x_S \geq 1, \quad e \in U \]
\[ x_S \geq 0, \quad S \in \mathcal{S}_P \]

Note

For our scheme, we choose \( \alpha = 1 \) and \( \beta = f \), where \( f \) is the frequency of the most frequent element.
Relaxed complementary slackness conditions

Primal Conditions

∀ \mathcal{S} \in \mathcal{S}^P:

x_\mathcal{S} \neq 0 \Rightarrow \sum_{e \in \mathcal{S}} y_e = c(\mathcal{S}).

Note

1. A set \mathcal{S} is tight under the current assignment to \ y, if \sum_{e \in \mathcal{S}} y_e = c(\mathcal{S}).

2. Since primal variables are incremented integrally, the primal condition can be restated as: Pick only tight sets in the cover.

3. Clearly no set can be overpacked, if dual feasibility is to be maintained.
Relaxed complementary slackness conditions

Primal Conditions

∀S ∈ S^P: x_S ≠ 0 ⇒ ∑e: e ∈ S y_e = c(S).

Note 1: A set S is tight under the current assignment to y, if ∑e ∈ S y_e = c(S).

2: Since primal variables are incremented integrally, the primal condition can be restated as: Pick only tight sets in the cover.

3: Clearly no set can be overpacked, if dual feasibility is to be maintained.
Relaxed complementary slackness conditions

Primal Conditions

\[ \forall S \in S_P : \]
Relaxed complementary slackness conditions

Primal Conditions

∀\(S \in S_P : x_S \neq 0 \Rightarrow \sum_{e : e \in S} y_e = c(S)\).
Primal-Dual Schema

Primal-Dual schema for Set Cover

Relaxed complementary slackness conditions

**Primal Conditions**

\[ \forall S \in S_P : x_S \neq 0 \Rightarrow \sum_{e : e \in S} y_e = c(S). \]

**Note**

1. A set S is tight under the current assignment to \( y \), if \[ \sum_{e : e \in S} y_e = c(S). \]
2. Since primal variables are incremented integrally, the primal condition can be restated as: Pick only tight sets in the cover.
3. Clearly no set can be overpacked, if dual feasibility is to be maintained.
Relaxed complementary slackness conditions

**Primal Conditions**

\[ \forall S \in S_P : x_S \neq 0 \Rightarrow \sum_{e \in S} y_e = c(S). \]

**Note**

- A set $S$ is tight under the current assignment to $y$, if $\sum_{e \in S} y_e = c(S)$. 
Relaxed complementary slackness conditions

**Primal Conditions**

\[ \forall S \in S_P : x_S \neq 0 \Rightarrow \sum_{e : e \in S} y_e = c(S). \]

**Note**

1. A set \( S \) is tight under the current assignment to \( y \), if \( \sum_{e \in S} y_e = c(S) \).
2. Since primal variables are incremented integrally, the primal condition can be restated as:
Relaxed complementary slackness conditions

Primal Conditions

\[ \forall S \in S_P : x_S \neq 0 \Rightarrow \sum_{e : e \in S} y_e = c(S). \]

Note

1. A set \( S \) is tight under the current assignment to \( y \), if \( \sum_{e \in S} y_e = c(S) \).
2. Since primal variables are incremented integrally, the primal condition can be restated as: Pick only tight sets in the cover.
Relaxed complementary slackness conditions

**Primal Conditions**

$$\forall S \in S_P : x_S \neq 0 \Rightarrow \sum_{e : e \in S} y_e = c(S).$$

**Note**

1. A set $S$ is tight under the current assignment to $y$, if $\sum_{e \in S} y_e = c(S)$.
2. Since primal variables are incremented integrally, the primal condition can be restated as: Pick only tight sets in the cover.
3. Clearly no set can be overpacked, if dual feasibility is to be maintained.
Relaxed complementary slackness conditions
Relaxed complementary slackness conditions

∀ e: y_e ≠ 0 ⇒ \sum_{S: e \in S} x_S ≤ f.

Note: The above conditions can be interpreted as follows: Each element having a non-zero dual can be covered at most f times. But this condition is trivially satisfied by all elements e ∈ U!
Relaxed complementary slackness conditions

∀ \( e \) : \( y_e \neq 0 \) ⇒ \( \sum_{S : e \in S} x_S \leq f \).

Note The above conditions can be interpreted as follows: Each element having a non-zero dual can be covered at most \( f \) times. But this condition is trivially satisfied by all elements \( e \in U \).
Relaxed complementary slackness conditions

<table>
<thead>
<tr>
<th>Dual Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\forall e : y_e \neq 0 \Rightarrow$</td>
</tr>
</tbody>
</table>
Relaxed complementary slackness conditions

\[ \forall e : y_e \neq 0 \Rightarrow \sum_{S : e \in S} x_S \leq \]

Note: The above conditions can be interpreted as follows: Each element having a non-zero dual can be covered at most \( f \) times. But this condition is trivially satisfied by all elements \( e \in U \).
Relaxed complementary slackness conditions

\[ \forall e : y_e \neq 0 \Rightarrow \sum_{S: e \in S} x_S \leq f. \]
Relaxed complementary slackness conditions

**Dual Conditions**

\[ \forall e : y_e \neq 0 \Rightarrow \sum_{S : e \in S} x_S \leq f. \]

**Note**

The above conditions can be interpreted as follows: Each element having a non-zero dual can be covered at most \( f \) times. But this condition is trivially satisfied by all elements \( e \in U \).
Relaxed complementary slackness conditions

Dual Conditions

\[ \forall e : y_e \neq 0 \Rightarrow \sum_{S: e \in S} x_s \leq f. \]

Note

*The above conditions can be interpreted as follows:*
Relaxed complementary slackness conditions

**Dual Conditions**

\[ \forall e : y_e \neq 0 \Rightarrow \sum_{S : e \in S} x_S \leq f. \]

**Note**

The above conditions can be interpreted as follows: Each element having a non-zero dual can be covered at most \( f \) times.
Relaxed complementary slackness conditions

Dual Conditions

\[ \forall e : y_e \neq 0 \Rightarrow \sum_{S : e \in S} x_S \leq f. \]

Note

The above conditions can be interpreted as follows: Each element having a non-zero dual can be covered at most \( f \) times. But this condition is trivially satisfied by all elements \( e \in U \)!
The Primal Dual Algorithm for Set Cover

1. Set $x \leftarrow 0$ and $y \leftarrow 0$.
2. Until (all elements are covered) do:
3. Pick an uncovered element, say $e$, and raise $y_e$, until some set goes tight.
4. Pick all tight sets in the cover and update $x$.
5. Declare all elements occurring in these sets as "covered."
6. Output the set cover $x$. 
The Primal Dual Algorithm for Set Cover

The Primal-Dual Schema

The Primal-Dual Algorithm

The Primal-Dual Algorithm for Set Cover

The Algorithm

1. Set $x \leftarrow 0$ and $y \leftarrow 0$.
2. Until (all elements are covered) do:
   3. Pick an uncovered element, say $e$, and raise $y_e$ until some set goes tight.
   4. Pick all tight sets in the cover and update $x$.
   5. Declare all elements occurring in these sets as "covered."
6. Output the set cover $x$.
The Primal Dual Algorithm for Set Cover

The Algorithm

1. Set $x \leftarrow 0$ and $y \leftarrow 0$. 
The Primal Dual Algorithm for Set Cover

The Algorithm

1. Set $x \gets 0$ and $y \gets 0$.
2. **Until** (all elements are covered) **do**:

   - Pick an uncovered element, say $e$, and raise $y_e$, until some set goes tight.
   - Pick all tight sets in the cover and update $x$.
   - Declare all elements occurring in these sets as "covered."
The Primal Dual Algorithm for Set Cover

The Algorithm

1. Set $x \leftarrow 0$ and $y \leftarrow 0$.
2. Until (all elements are covered) do:
3. Pick an uncovered element, say $e$ and raise $y_e$, until some set goes tight.
The Primal Dual Algorithm for Set Cover

The Algorithm

1. Set $x \leftarrow 0$ and $y \leftarrow 0$.
2. Until (all elements are covered) do:
   3. Pick an uncovered element, say $e$ and raise $y_e$, until some set goes tight.
   4. Pick all tight sets in the cover and update $x$.
   5. Declare all elements occurring in these sets as “covered.”
   6. Output the set cover $x$. 
The Primal Dual Algorithm for Set Cover

The Algorithm

1. Set $x \leftarrow 0$ and $y \leftarrow 0$.
2. Until (all elements are covered) do:
   3. Pick an uncovered element, say $e$ and raise $y_e$, until some set goes tight.
   4. Pick all tight sets in the cover and update $x$.
   5. Declare all elements occurring in these sets as “covered.”

Output the set cover $x$. 
The Primal Dual Algorithm for Set Cover

The Algorithm

1. Set $x \leftarrow 0$ and $y \leftarrow 0$.
2. Until (all elements are covered) do:
   3. Pick an uncovered element, say $e$ and raise $y_e$, until some set goes tight.
   4. Pick all tight sets in the cover and update $x$.
   5. Declare all elements occurring in these sets as “covered.”
6. Output the set cover $x$. 
The Primal-Dual Algorithm

Analysis

Theorem

The above algorithm achieves an approximation factor of $f$.

Proof.

1. $x$ is a primal feasible solution, i.e., are all elements covered? Yes.

2. $y$ is a dual feasible solution, i.e., is any set overpacked? No set is overpacked and hence $y$ is indeed dual feasible.

3. Note that $x$ and $y$ satisfy the relaxed complementary slackness conditions, with $\alpha = 1$ and $\beta = f$.

4. By the Main Lemma, it follows that the approximation factor is $f$. 
Theorem

The above algorithm achieves an approximation factor of $f$. 

Proof.

1. Is $x$ a primal feasible solution, i.e., are all elements covered? Yes.
2. Is $y$ a dual feasible solution, i.e., is any set overpacked? No set is overpacked and hence $y$ is indeed dual feasible.
3. Note that $x$ and $y$ satisfy the relaxed complementary slackness conditions, with $\alpha = 1$ and $\beta = f$.
4. By the Main Lemma, it follows that the approximation factor is $f$. 
## Theorem

*The above algorithm achieves an approximation factor of $f$.*

## Proof.

1. Is $x$ a primal feasible solution, i.e., are all elements covered? **Yes.**
2. Is $y$ a dual feasible solution, i.e., is any set overpacked? **No set is overpacked and hence $y$ is indeed dual feasible.**
3. Note that $x$ and $y$ satisfy the relaxed complementary slackness conditions, with $\alpha = 1$ and $\beta = f$.
4. By the Main Lemma, it follows that the approximation factor is $f$. 
Theorem

The above algorithm achieves an approximation factor of $f$. 

Proof.

1. Is $x$ a primal feasible solution, i.e., are all elements covered? Yes.
2. Is $y$ a dual feasible solution, i.e., is any set overpacked? No set is overpacked and hence $y$ is indeed dual feasible.
3. Note that $x$ and $y$ satisfy the relaxed complementary slackness conditions, with $\alpha = 1$ and $\beta = f$.
4. By the Main Lemma, it follows that the approximation factor is $f$. 
Theorem

The above algorithm achieves an approximation factor of $f$.

Proof.

1. Is $x$ a primal feasible solution, i.e., are all elements covered?
Theorem

The above algorithm achieves an approximation factor of $f$.

Proof.

1. Is $x$ a primal feasible solution, i.e., are all elements covered? Yes.
Theorem

The above algorithm achieves an approximation factor of $f$.

Proof.

1. Is $x$ a primal feasible solution, i.e., are all elements covered? Yes.
2. Is $y$ a dual feasible solution, i.e., is any set overpacked?
Theorem

The above algorithm achieves an approximation factor of $f$.

Proof.

1. Is $x$ a primal feasible solution, i.e., are all elements covered? Yes.
2. Is $y$ a dual feasible solution, i.e., is any set overpacked? No set is overpacked and hence $y$ is indeed dual feasible.
Analysis

Theorem

The above algorithm achieves an approximation factor of $f$.

Proof.

1. Is $x$ a primal feasible solution, i.e., are all elements covered? Yes.
2. Is $y$ a dual feasible solution, i.e., is any set overpacked? No set is overpacked and hence $y$ is indeed dual feasible.
3. Note that $x$ and $y$ satisfy the relaxed complementary slackness conditions, with $\alpha = 1$ and $\beta = f$. 
The Primal Dual Algorithm

Analysis

Theorem

The above algorithm achieves an approximation factor of $f$.

Proof.

1. Is $x$ a primal feasible solution, i.e., are all elements covered? Yes.
2. Is $y$ a dual feasible solution, i.e., is any set overpacked? No set is overpacked and hence $y$ is indeed dual feasible.
3. Note that $x$ and $y$ satisfy the relaxed complementary slackness conditions, with $\alpha = 1$ and $\beta = f$.
4. By the Main Lemma, it follows that the approximation factor is $f$. 
Tightness Analysis

Let $S$ consist of the following:

\[
(n-1) \text{ sets of cost } 1, \text{ viz., } \{e_1, e_n\}, \{e_2, e_n\}, \ldots, \{e_{n-1}, e_n\}
\]

and one set $\{e_1, e_2, \ldots, e_n, e_{n+1}\}$ of cost $(1+\varepsilon)$, where $\varepsilon > 0$ is a small constant.

Observe that $f = n$.

Suppose that the algorithm picks $y_{en}$ in the first iteration.

When $y_{en}$ is raised to 1, all sets $\{e_i, e_n\}, i = 1, 2, \ldots, (n-1)$, go tight.

Thus they are all picked, covering the elements in $\{e_1, e_2, \ldots, e_n\}$.

In the second iteration, $y_{en+1}$ is raised to $\varepsilon$ and the set $\{e_1, e_2, \ldots, e_n, e_{n+1}\}$ becomes tight and is picked.

The total cost of the picked cover is $(n + \varepsilon)$.

The optimal cover has cost $(1+\varepsilon)$.

This example achieves the bound of $f = n$. 
Tightness Analysis

Example

Let $S$ consist of the following:

- $(n-1)$ sets of cost 1, viz., $\{e_1, e_n\}$, $\{e_2, e_n\}$, ..., $\{e_{n-1}, e_n\}$
- One set $\{e_1, e_2, ..., e_n, e_{n+1}\}$ of cost $(1 + \varepsilon)$, where $\varepsilon > 0$ is a small constant.

Observe that $f = n$.

Suppose that the algorithm picks $y_{e_n}$ in the first iteration. When $y_{e_n}$ is raised to 1, all sets $\{e_i, e_n\}$, $i = 1, 2, ..., (n-1)$, go tight.

Thus they are all picked, covering the elements in $\{e_1, e_2, ..., e_n\}$.

In the second iteration, $y_{e_{n+1}}$ is raised to $\varepsilon$ and the set $\{e_1, e_2, ..., e_n, e_{n+1}\}$ becomes tight and is picked.

The total cost of the picked cover is $(n + \varepsilon)$.

The optimal cover has cost $(1 + \varepsilon)$.

This example achieves the bound of $f = n$. 
Let \( S \) consist of the following: \((n - 1)\) sets of cost 1, viz., \(\{e_1, e_n\}, \{e_2, e_n\}, \ldots, \{e_{n-1}, e_n\}\) and one set \(\{e_1, e_2, \ldots, e_n, e_{n+1}\}\) of cost \((1 + \varepsilon)\), where \(\varepsilon > 0\) is a small constant.
Tightness Analysis

Example

1. Let $S_P$ consist of the following: $(n - 1)$ sets of cost 1, viz., $\{e_1, e_n\}$, $\{e_2, e_n\}$, $\ldots$ $\{e_{n-1}, e_n\}$ and one set $\{e_1, e_2, \ldots, e_n, e_{n+1}\}$ of cost $(1 + \varepsilon)$, where $\varepsilon > 0$ is a small constant.

2. Observe that $f = n$. 
Let $S_P$ consist of the following: $(n - 1)$ sets of cost 1, viz., $\{e_1, e_n\}, \{e_2, e_n\}, \ldots \{e_{n-1}, e_n\}$ and one set $\{e_1, e_2, \ldots, e_n, e_{n+1}\}$ of cost $(1 + \varepsilon)$, where $\varepsilon > 0$ is a small constant.

Observe that $f = n$.

Suppose that the algorithm picks $y_{e_n}$ in the first iteration.
Primal-Dual Schema

Tightness

Tightness Analysis

Example

1. Let $S_P$ consist of the following: $(n-1)$ sets of cost 1, viz., $\{e_1, e_n\}$, $\{e_2, e_n\}$, \ldots $\{e_{n-1}, e_n\}$ and one set $\{e_1, e_2, \ldots, e_n, e_{n+1}\}$ of cost $(1 + \varepsilon)$, where $\varepsilon > 0$ is a small constant.

2. Observe that $f = n$.

3. Suppose that the algorithm picks $y_{e_n}$ in the first iteration.

4. When $y_{e_n}$ is raised to 1, all sets $\{e_i, e_n\}$, $i = 1, 2, \ldots (n-1)$, go tight.
Let $S_P$ consist of the following: $(n-1)$ sets of cost 1, viz., $\{e_1, e_n\}$, $\{e_2, e_n\}$, $\ldots$ $\{e_{n-1}, e_n\}$ and one set $\{e_1, e_2, \ldots, e_n, e_{n+1}\}$ of cost $(1 + \varepsilon)$, where $\varepsilon > 0$ is a small constant.

Observe that $f = n$.

Suppose that the algorithm picks $y_{e_n}$ in the first iteration.

When $y_{e_n}$ is raised to 1, all sets $\{e_i, e_n\}$, $i = 1, 2, \ldots (n-1)$, go tight.

Thus they are all picked, covering the elements in $\{e_1, e_2, \ldots, e_n\}$.
Example

1. Let $S_P$ consist of the following: $(n - 1)$ sets of cost 1, viz., $\{e_1, e_n\}$, $\{e_2, e_n\}$, $\ldots$, $\{e_{n-1}, e_n\}$ and one set $\{e_1, e_2, \ldots, e_n, e_{n+1}\}$ of cost $(1 + \varepsilon)$, where $\varepsilon > 0$ is a small constant.

2. Observe that $f = n$.

3. Suppose that the algorithm picks $y_{e_n}$ in the first iteration.

4. When $y_{e_n}$ is raised to 1, all sets $\{e_i, e_n\}$, $i = 1, 2, \ldots, (n - 1)$, go tight.

5. Thus they are all picked, covering the elements in $\{e_1, e_2, \ldots, e_n\}$.

6. In the second iteration, $y_{e_{n+1}}$ is raised to $\varepsilon$ and the set $\{e_1, e_2, \ldots, e_n, e_{n+1}\}$ becomes tight and is picked.

The total cost of the picked cover is $(n + \varepsilon)$.

The optimal cover has cost $(1 + \varepsilon)$.

This example achieves the bound of $f = n$. 
Tightness Analysis

Example

1. Let \( S_P \) consist of the following: \((n - 1)\) sets of cost 1, viz., \( \{e_1, e_n\}, \{e_2, e_n\}, \ldots, \{e_{n-1}, e_n\} \) and one set \( \{e_1, e_2, \ldots, e_n, e_{n+1}\} \) of cost \((1 + \varepsilon)\), where \( \varepsilon > 0 \) is a small constant.

2. Observe that \( f = n \).

3. Suppose that the algorithm picks \( y_{e_n} \) in the first iteration.

4. When \( y_{e_n} \) is raised to 1, all sets \( \{e_i, e_n\}, i = 1, 2, \ldots, (n - 1) \), go tight.

5. Thus they are all picked, covering the elements in \( \{e_1, e_2, \ldots, e_n\} \).

6. In the second iteration, \( y_{e_{n+1}} \) is raised to \( \varepsilon \) and the set \( \{e_1, e_2, \ldots, e_n, e_{n+1}\} \) becomes tight and is picked.

7. The total cost of the picked cover is \((n + \varepsilon)\).
Tightness Analysis

Example

1. Let $S_P$ consist of the following: $(n - 1)$ sets of cost 1, viz., \{e_1, e_n\}, \{e_2, e_n\}, \ldots, \{e_{n-1}, e_n\} and one set \{e_1, e_2, \ldots, e_n, e_{n+1}\} of cost $(1 + \varepsilon)$, where $\varepsilon > 0$ is a small constant.

2. Observe that $f = n$.

3. Suppose that the algorithm picks $y_{e_n}$ in the first iteration.

4. When $y_{e_n}$ is raised to 1, all sets \{e_i, e_n\}, $i = 1, 2, \ldots, (n - 1)$, go tight.

5. Thus they are all picked, covering the elements in \{e_1, e_2, \ldots, e_n\}.

6. In the second iteration, $y_{e_{n+1}}$ is raised to $\varepsilon$ and the set \{e_1, e_2, \ldots, e_n, e_{n+1}\} becomes tight and is picked.

7. The total cost of the picked cover is $(n + \varepsilon)$.

8. The optimal cover has cost $(1 + \varepsilon)$.
Tightness Analysis

Example

1. Let $S_P$ consist of the following: $(n-1)$ sets of cost 1, viz., $\{e_1, e_n\}, \{e_2, e_n\}, \ldots \{e_{n-1}, e_n\}$ and one set $\{e_1, e_2, \ldots, e_n, e_{n+1}\}$ of cost $(1 + \varepsilon)$, where $\varepsilon > 0$ is a small constant.

2. Observe that $f = n$.

3. Suppose that the algorithm picks $y_{en}$ in the first iteration.

4. When $y_{en}$ is raised to 1, all sets $\{e_i, e_n\}, i = 1, 2, \ldots (n-1)$, go tight.

5. Thus they are all picked, covering the elements in $\{e_1, e_2, \ldots, e_n\}$.

6. In the second iteration, $y_{en+1}$ is raised to $\varepsilon$ and the set $\{e_1, e_2, \ldots, e_n, e_{n+1}\}$ becomes tight and is picked.

7. The total cost of the picked cover is $(n + \varepsilon)$.

8. The optimal cover has cost $(1 + \varepsilon)$.

9. This example achieves the bound of $f = n$. 