CS 473: Algorithms

Ruta Mehta

University of Illinois, Urbana-Champaign

Spring 2021

LP & Strong Duality

Lecture 19
April 15, 2021

Some of the slides are courtesy Prof. Chekuri

Part I

Recall

Linear Program

Properties and the Simplex Algorithm

- Solution at a vertex of the polyhedron $\mathcal{P}$.
- If vertex v is not optimal then it has a neighbor where the objective value improves.
- If the $\mathcal{P}$ in d dimension, then every vertex has exactly d neighboring vertices (almost always).

Properties and the Simplex Algorithm

- Solution at a vertex of the polyhedron $\mathcal{P}$.
- If vertex v is not optimal then it has a neighbor where the objective value improves.
- If the $\mathcal{P}$ in d dimension, then every vertex has exactly d neighboring vertices (almost always).

Simplex: Moves from a vertex to its neighboring vertex

Questions + Answers

- Which neighbor to move to? One where objective value increases.

Properties and the Simplex Algorithm

- Solution at a vertex of the polyhedron $\mathcal{P}$.
- If vertex v is not optimal then it has a neighbor where the objective value improves.
- If the $\mathcal{P}$ in d dimension, then every vertex has exactly d neighboring vertices (almost always).

Simplex: Moves from a vertex to its neighboring vertex

Questions + Answers

- Which neighbor to move to? One where objective value increases.
- When to stop? When no neighbor with better objective value.

Properties and the Simplex Algorithm

- Solution at a vertex of the polyhedron $\mathcal{P}$.
- If vertex v is not optimal then it has a neighbor where the objective value improves.
- If the $\mathcal{P}$ in d dimension, then every vertex has exactly d neighboring vertices (almost always).

Simplex: Moves from a vertex to its neighboring vertex

Questions + Answers

- Which neighbor to move to? One where objective value increases.
- When to stop? When no neighbor with better objective value.
- How much time does it take? At most d neighbors to consider in each step.

Issues

1. Starting vertex
2. The linear program could be infeasible: No point satisfy the constraints.
3. The linear program could be unbounded: Polygon unbounded in the direction of the objective function.

Computing the Starting Vertex

Equivalent to solving another LP!

Find an x such that $Ax \leq b$.
If $b \geq \mathbf{0}$ then trivial!

Computing the Starting Vertex

Equivalent to solving another LP!

Find an x such that $Ax \leq b$.
If $b \geq \mathbf{0}$ then trivial! $x = \mathbf{0}$. Otherwise.

Computing the Starting Vertex

Equivalent to solving another LP!

Find an x such that $Ax \leq b$.
If $b \geq 0$ then trivial! $x = 0$. Otherwise.

$$
\begin{aligned}
\textbf{min}: \quad & s \\
s.t. \quad & \textstyle\sum_j a_{ij} x_j - s \leq b_i, \quad \forall i \\
& s \geq 0
\end{aligned}
$$

Trivial feasible solution:

Computing the Starting Vertex

Equivalent to solving another LP!

Find an x such that $Ax \leq b$.
If $b \geq \mathbf{0}$ then trivial! $x = \mathbf{0}$. Otherwise.

$$\begin{aligned} \textbf{min}: \quad & s \\ s.t. \quad & \sum_j a_{ij}x_j - s \leq b_i, \quad \forall i \\ & s \geq \mathbf{0} \end{aligned}$$

Trivial feasible solution: $x = \mathbf{0}$, $s = |\min_i b_i|$.

Computing the Starting Vertex

Equivalent to solving another LP!

Find an x such that $Ax \leq b$.
If $b \geq \mathbf{0}$ then trivial! $x = \mathbf{0}$. Otherwise.

$$\begin{aligned} \textbf{min}: \quad & s \\ s.t. \quad & \textstyle\sum_j a_{ij} x_j - s \leq b_i, \quad \forall i \\ & s \geq 0 \end{aligned}$$

Trivial feasible solution: $x = \mathbf{0}$, $s = |\min_i b_i|$.

If $Ax \leq b$ feasible then optimal value of the above LP is $s = 0$.

Computing the Starting Vertex

Equivalent to solving another LP!

Find an x such that $Ax \leq b$.
If $b \geq \mathbf{0}$ then trivial! $x = \mathbf{0}$. Otherwise.

$$\begin{aligned} \textbf{min}: \quad & s \\ s.t. \quad & \sum_j a_{ij}x_j - s \leq b_i, \quad \forall i \\ & s \geq 0 \end{aligned}$$

Trivial feasible solution: $x = \mathbf{0}$, $s = |\min_i b_i|$.

If $Ax \leq b$ feasible then optimal value of the above LP is $s = 0$.

Checks Feasibility!

Part II

Duality

Feasible Solutions and Lower Bounds

Consider the program

$$
\begin{array}{llrcl}
\text{maximize} & 4x_1 + & 2x_2 & & \\
\text{subject to} & x_1 + & 3x_2 & \leq & 5 \\
 & 2x_1 - & 4x_2 & \leq & 10 \\
 & x_1 + & x_2 & \leq & 7 \\
 & x_1 & & \leq & 5
\end{array}
$$

Feasible Solutions and Lower Bounds

Consider the program

$$\begin{array}{llrrl} \text{maximize} & 4x_1 + & 2x_2 & & \\ \text{subject to} & x_1 + & 3x_2 & \leq & 5 \\ & 2x_1 - & 4x_2 & \leq & 10 \\ & x_1 + & x_2 & \leq & 7 \\ & x_1 & & \leq & 5 \end{array}$$

1. $(0, 1)$ satisfies all the constraints and gives value 2 for the objective function.

Feasible Solutions and Lower Bounds

Consider the program

$$\begin{array}{lrrl} \text{maximize} & 4x_1 + & 2x_2 & \\ \text{subject to} & x_1 + & 3x_2 & \leq 5 \\ & 2x_1 - & 4x_2 & \leq 10 \\ & x_1 + & x_2 & \leq 7 \\ & x_1 & & \leq 5 \end{array}$$

1. $(0, 1)$ satisfies all the constraints and gives value 2 for the objective function.
2. Thus, optimal value σ^* is at least 2.

Feasible Solutions and Lower Bounds

Consider the program

$$
\begin{array}{lrrl}
\text{maximize} & 4x_1 + & 2x_2 & \\
\text{subject to} & x_1 + & 3x_2 & \leq 5 \\
 & 2x_1 - & 4x_2 & \leq 10 \\
 & x_1 + & x_2 & \leq 7 \\
 & x_1 & & \leq 5
\end{array}
$$

1. $(0, 1)$ satisfies all the constraints and gives value 2 for the objective function.
2. Thus, optimal value σ^* is at least 2.
3. $(2, 0)$ also feasible, and gives a better bound of 8.

Feasible Solutions and Lower Bounds

Consider the program

$$\begin{array}{lrrl} \text{maximize} & 4x_1 + & 2x_2 & \\ \text{subject to} & x_1 + & 3x_2 & \leq 5 \\ & 2x_1 - & 4x_2 & \leq 10 \\ & x_1 + & x_2 & \leq 7 \\ & x_1 & & \leq 5 \end{array}$$

1. $(0, 1)$ satisfies all the constraints and gives value 2 for the objective function.
2. Thus, optimal value σ^* is at least 2.
3. $(2, 0)$ also feasible, and gives a better bound of 8.
4. How good is 8 when compared with σ^*?

Obtaining Upper Bounds

$$
\begin{array}{llll}
\text{maximize} & 4x_1 + & 2x_2 & \\
\text{subject to} & x_1 + & 3x_2 & \leq 5 \\
 & 2x_1 - & 4x_2 & \leq 10 \\
 & x_1 + & x_2 & \leq 7 \\
 & x_1 & & \leq 5
\end{array}
$$

1. Let us multiply the first constraint by 2 and the and add it to second constraint

Obtaining Upper Bounds

$$\begin{array}{llrrl} \text{maximize} & 4x_1 + & 2x_2 & & \\ \text{subject to} & x_1 + & 3x_2 & \leq & 5 \\ & 2x_1 - & 4x_2 & \leq & 10 \\ & x_1 + & x_2 & \leq & 7 \\ & x_1 & & \leq & 5 \end{array}$$

1. Let us multiply the first constraint by 2 and the and add it to second constraint

$$\begin{array}{rrrl} 2(& x_1 + & 3x_2 &) \leq 2(5) \\ +1(& 2x_1 - & 4x_2 &) \leq 1(10) \\ \hline & 4x_1 + & 2x_2 & \leq 20 \end{array}$$

Obtaining Upper Bounds

$$\begin{array}{lrrl} \text{maximize} & 4x_1 + & 2x_2 & \\ \text{subject to} & x_1 + & 3x_2 & \leq 5 \\ & 2x_1 - & 4x_2 & \leq 10 \\ & x_1 + & x_2 & \leq 7 \\ & x_1 & & \leq 5 \end{array}$$

1. Let us multiply the first constraint by 2 and the and add it to second constraint

$$\begin{array}{rrrl} 2(& x_1 + & 3x_2 &) \leq 2(5) \\ +1(& 2x_1 - & 4x_2 &) \leq 1(10) \\ \hline & 4x_1 + & 2x_2 & \leq 20 \end{array}$$

2. Thus, 20 is an upper bound on the optimum value!

1. Multiply first equation by y_1, second by y_2, third by y_3 and fourth by y_4 $(y_1, y_2, y_3, y_4 \geq 0)$ and add

$$
\begin{array}{rrrl}
y_1(& x_1 + & 3x_2 &) \leq y_1(5) \\
+y_2(& 2x_1 - & 4x_2 &) \leq y_2(10) \\
+y_3(& x_1 + & x_2 &) \leq y_3(7) \\
+y_4(& x_1 & &) \leq y_4(5) \\
\hline
\multicolumn{4}{l}{(y_1 + 2y_2 + y_3 + y_4)x_1 + (3y_1 - 4y_2 + y_3)x_2 \leq \ldots}
\end{array}
$$

Generalizing . . .

1. Multiply first equation by y_1, second by y_2, third by y_3 and fourth by y_4 ($y_1, y_2, y_3, y_4 \geq 0$) and add

$$
\begin{array}{rrrl}
y_1(& x_1 + & 3x_2 &) \leq y_1(5) \\
+y_2(& 2x_1 - & 4x_2 &) \leq y_2(10) \\
+y_3(& x_1 + & x_2 &) \leq y_3(7) \\
+y_4(& x_1 & &) \leq y_4(5) \\
\hline
\multicolumn{4}{l}{(y_1 + 2y_2 + y_3 + y_4)x_1 + (3y_1 - 4y_2 + y_3)x_2 \leq \ldots}
\end{array}
$$

2. $5y_1 + 10y_2 + 7y_3 + 5y_4$ is an upper bound,

Generalizing . . .

1. Multiply first equation by y_1, second by y_2, third by y_3 and fourth by y_4 $(y_1, y_2, y_3, y_4 \geq 0)$ and add

$$
\begin{array}{rrrl}
y_1(& x_1 + & 3x_2 &) \leq y_1(5) \\
+y_2(& 2x_1 - & 4x_2 &) \leq y_2(10) \\
+y_3(& x_1 + & x_2 &) \leq y_3(7) \\
+y_4(& x_1 & &) \leq y_4(5) \\
\hline
\multicolumn{4}{c}{(y_1 + 2y_2 + y_3 + y_4)x_1 + (3y_1 - 4y_2 + y_3)x_2 \leq \ldots}
\end{array}
$$

2. $5y_1 + 10y_2 + 7y_3 + 5y_4$ is an upper bound, provided coefficients of x_i are same as in the objective function $(4x_1 + 2x_2)$,

$$y_1 + 2y_2 + y_3 + y_4 = 4 \quad 3y_1 - 4y_2 + y_3 = 2$$

Generalizing ...

1. Multiply first equation by y_1, second by y_2, third by y_3 and fourth by y_4 ($y_1, y_2, y_3, y_4 \geq 0$) and add

$$\begin{array}{rrrl} y_1(& x_1+ & 3x_2 &) \leq y_1(5) \\ +y_2(& 2x_1- & 4x_2 &) \leq y_2(10) \\ +y_3(& x_1+ & x_2 &) \leq y_3(7) \\ +y_4(& x_1 & &) \leq y_4(5) \\ \hline \multicolumn{4}{l}{(y_1 + 2y_2 + y_3 + y_4)x_1 + (3y_1 - 4y_2 + y_3)x_2 \leq \ldots} \end{array}$$

2. $5y_1 + 10y_2 + 7y_3 + 5y_4$ is an upper bound, provided coefficients of x_i are same as in the objective function $(4x_1 + 2x_2)$,

$$y_1 + 2y_2 + y_3 + y_4 = 4 \quad 3y_1 - 4y_2 + y_3 = 2$$

3. Subject to these constrains, the best upper bound is $\min : 5y_1 + 10y_2 + 7y_3 + 5y_4$!

Dual LP: Example

Thus, the optimum value of program

$$\begin{aligned} \text{maximize} \quad & 4x_1 + 2x_2 \\ \text{subject to} \quad & x_1 + 3x_2 \leq 5 \\ & 2x_1 - 4x_2 \leq 10 \\ & x_1 + x_2 \leq 7 \\ & x_1 \leq 5 \end{aligned}$$

is upper bounded by the optimal value of the program

$$\begin{aligned} \text{minimize} \quad & 5y_1 + 10y_2 + 7y_3 + 5y_4 \\ \text{subject to} \quad & y_1 + 2y_2 + y_3 + y_4 = 4 \\ & 3y_1 - 4y_2 + y_3 = 2 \\ & y_1, y_2 \geq 0 \end{aligned}$$

Dual Linear Program

Given a linear program $\mathbf{\Pi}$ in canonical form

$$\begin{array}{ll} \text{maximize} & \sum_{j=1}^{d} c_j x_j \\ \text{subject to} & \sum_{j=1}^{d} a_{ij} x_j \leq b_i \quad i = 1, 2, \ldots n \end{array}$$

the dual $\mathbf{Dual}(\mathbf{\Pi})$ is given by

$$\begin{array}{lll} \text{minimize} & \sum_{i=1}^{n} b_i y_i & \\ \text{subject to} & \sum_{i=1}^{n} y_i a_{ij} = c_j & j = 1, 2, \ldots d \\ & y_i \geq 0 & i = 1, 2, \ldots n \end{array}$$

Dual Linear Program

Given a linear program Π in canonical form

$$\begin{array}{ll} \text{maximize} & \sum_{j=1}^{d} c_j x_j \\ \text{subject to} & \sum_{j=1}^{d} a_{ij} x_j \leq b_i \quad i = 1, 2, \ldots n \end{array}$$

the dual $\mathrm{Dual}(\Pi)$ is given by

$$\begin{array}{lll} \text{minimize} & \sum_{i=1}^{n} b_i y_i & \\ \text{subject to} & \sum_{i=1}^{n} y_i a_{ij} = c_j & j = 1, 2, \ldots d \\ & y_i \geq 0 & i = 1, 2, \ldots n \end{array}$$

Proposition

$\mathrm{Dual}(\mathrm{Dual}(\Pi))$ *is equivalent to* Π

Duality Theorems

Theorem (Weak Duality)

If x' *is a feasible solution to* Π *and* y' *is a feasible solution to* $\mathrm{Dual}(\Pi)$ *then* $c \cdot x' \leq y' \cdot b$.

Duality Theorems

Theorem (Weak Duality)

If x' *is a feasible solution to* Π *and* y' *is a feasible solution to* $\mathrm{Dual}(\Pi)$ *then* $c \cdot x' \leq y' \cdot b$.

Theorem (Strong Duality)

If x^* *is an optimal solution to* Π *and* y^* *is an optimal solution to* $\mathrm{Dual}(\Pi)$ *then* $c \cdot x^* = y^* \cdot b$.

Many applications! Maxflow-Mincut theorem can be deduced from duality.

Weak Duality

Theorem (Weak Duality)

If x is a feasible solution to Π and y is a feasible solution to $\mathrm{Dual}(\Pi)$ then $c \cdot x \leq y \cdot b$.

We already saw the proof by the way we derived it but we will do it again formally.

Proof.

Since y' is feasible in $\mathrm{Dual}(\Pi)$: $y'A = c$

Weak Duality

Theorem (Weak Duality)

If x is a feasible solution to Π and y is a feasible solution to $\mathrm{Dual}(\Pi)$ then $c \cdot x \leq y \cdot b$.

We already saw the proof by the way we derived it but we will do it again formally.

Proof.

Since y' is feasible in $\mathrm{Dual}(\Pi)$: $y'A = c$

Therefore $c \cdot x' = y'Ax'$

Weak Duality

Theorem (Weak Duality)

If x is a feasible solution to Π and y is a feasible solution to $\mathrm{Dual}(\Pi)$ then $c \cdot x \leq y \cdot b$.

We already saw the proof by the way we derived it but we will do it again formally.

Proof.

Since y' is feasible in $\mathrm{Dual}(\Pi)$: $y'A = c$

Therefore $c \cdot x' = y'Ax'$

Since x' is feasible in Π, $Ax' \leq b$ and hence,

$$c \cdot x' = y'Ax' \leq y' \cdot b$$

Strong Duality and Complementary Slackness

$$\begin{array}{ll} \textbf{maximize}: & c \cdot x \\ \text{subject to} & Ax \leq b \end{array} \quad \xrightarrow{\textit{Dual}} \quad \begin{array}{ll} \textbf{minimize}: & y \cdot b \\ \text{subject to} & yA = c \\ & y \geq 0 \end{array}$$

Definition (Complementary Slackness)

x feasible in Π and y feasible in $\mathrm{Dual}(\Pi)$, s.t.,

$$\forall i = 1..n, \quad y_i > 0 \;\Rightarrow\; (Ax)_i = b_i$$

Strong Duality and Complementary Slackness

$$\begin{array}{ll} \textbf{maximize}: & c \cdot x \\ \text{subject to} & Ax \leq b \end{array} \quad \xrightarrow{Dual} \quad \begin{array}{ll} \textbf{minimize}: & y \cdot b \\ \text{subject to} & yA = c \\ & y \geq 0 \end{array}$$

Definition (Complementary Slackness)

x feasible in Π and y feasible in $\mathrm{Dual}(\Pi)$, s.t.,

$$\forall i = 1..n, \quad y_i > 0 \Rightarrow (Ax)_i = b_i$$

Geometric Interpretation: c is in the cone of the normal vectors of the tight hyperplanes at x.

Strong Duality and Complementary Slackness

Definition (Complementary Slackness)

x feasible in Π and y feasible in $\mathrm{Dual}(\Pi)$, s.t.,

$$\forall i = 1..n, \quad y_i > 0 \Rightarrow (Ax)_i = b_i$$

Theorem

(x^*, y^*) satisfies complementary Slackness if and only if strong duality holds, i.e., $c \cdot x^* = y^* \cdot b$.

Proof.

$$\begin{aligned} c \cdot x^* &= (y^* A) \cdot x^* \\ &= y^* \cdot (Ax^*) \end{aligned}$$

$(\Rightarrow)$

Strong Duality and Complementary Slackness

Definition (Complementary Slackness)

x feasible in Π and y feasible in $\mathrm{Dual}(\Pi)$, s.t.,

$$\forall i = 1..n, \quad y_i > 0 \;\Rightarrow\; (Ax)_i = b_i$$

Theorem

(x^*, y^*) satisfies complementary Slackness if and only if strong duality holds, i.e., $c \cdot x^* = y^* \cdot b$.

Proof.

$(\Rightarrow)$

$$\begin{aligned} c \cdot x^* &= (y^* A) \cdot x^* \\ &= y^* \cdot (Ax^*) \\ &= \textstyle\sum_{i=1}^{n} y_i^* (Ax^*)_i \\ &= \textstyle\sum_{i: y_i > 0} y_i^* (Ax^*)_i \end{aligned}$$

Strong Duality and Complementary Slackness

Definition (Complementary Slackness)

x feasible in Π and y feasible in $\mathrm{Dual}(\Pi)$, s.t.,

$$\forall i = 1..n, \quad y_i > 0 \;\Rightarrow\; (Ax)_i = b_i$$

Theorem

(x^*, y^*) satisfies complementary Slackness if and only if strong duality holds, i.e., $c \cdot x^* = y^* \cdot b$.

Proof.

$(\Rightarrow)$

$$\begin{aligned} c \cdot x^* &= (y^* A) \cdot x^* \\ &= y^* \cdot (Ax^*) \\ &= \textstyle\sum_{i=1}^n y_i^* (Ax^*)_i \\ &= \textstyle\sum_{i: y_i > 0} y_i^* (Ax^*)_i \\ &= \textstyle\sum_i y_i^* b_i = y^* \cdot b \end{aligned}$$

□

Strong Duality and Complementary Slackness

Definition (Complementary Slackness)

x feasible in Π and y feasible in $\mathrm{Dual}(\Pi)$, s.t.,

$$\forall i = 1..n, \quad y_i > 0 \;\Rightarrow\; (Ax)_i = b_i$$

Theorem

(x^*, y^*) satisfies complementary Slackness if and only if strong duality holds, i.e., $c \cdot x^* = y^* \cdot b$.

Proof.

$(\Leftarrow)$

Strong Duality and Complementary Slackness

Definition (Complementary Slackness)

x feasible in Π and y feasible in $\mathrm{Dual}(\Pi)$, s.t.,

$$\forall i = 1..n, \quad y_i > 0 \; \Rightarrow \; (Ax)_i = b_i$$

Theorem

(x^*, y^*) satisfies complementary Slackness if and only if strong duality holds, i.e., $c \cdot x^* = y^* \cdot b$.

Proof.

$(\Leftarrow)$ **Exercise** □

Strong Duality ≡ Complementary Slackness

We want

Theorem (Strong Duality)

If x^* *is an optimal solution to* Π *and* y^* *is an optimal solution to* $\mathrm{Dual}(\Pi)$ *then* $c \cdot x^* = y^* \cdot b$.

We showed

Theorem

(x^*, y^*) *satisfies complementary slackness* $\Leftrightarrow$ *Strong duality holds, i.e.,* $c \cdot x^* = y^* \cdot b$.

Strong Duality $\equiv$ Complementary Slackness

We want

Theorem (Strong Duality)

If x^* is an optimal solution to Π and y^* is an optimal solution to $\mathrm{Dual}(\Pi)$ then $c \cdot x^* = y^* \cdot b$.

We showed

Theorem

(x^*, y^*) satisfies complementary slackness $\Leftrightarrow$ Strong duality holds, i.e., $c \cdot x^* = y^* \cdot b$.

If (x^*, y^*) optimum $\Rightarrow$ complementary slackness, then done.

Complementary Slackness: Geometric View

$$\begin{array}{ll} \textbf{maximize}: & c \cdot x \\ \text{subject to} & Ax \le b \end{array} \quad \xrightarrow{\textit{Dual}} \quad \begin{array}{ll} \textbf{minimize}: & y \cdot b \\ \text{subject to} & yA = c \\ & y \ge 0 \end{array}$$

$y^*)$ satisfies complementary slackness: $\forall i = 1..n, \quad y_i^* > 0 \;\Rightarrow\; (Ax^*)_i$

Complementary Slackness: Geometric View

$$\begin{array}{ll} \textbf{maximize}: & c \cdot x \\ \text{subject to} & Ax \leq b \end{array} \quad \xrightarrow{\textit{Dual}} \quad \begin{array}{ll} \textbf{minimize}: & y \cdot b \\ \text{subject to} & yA = c \\ & y \geq 0 \end{array}$$

$y^*)$ satisfies complementary slackness: $\forall i = 1..n, \quad y_i^* > 0 \Rightarrow (Ax^*)_i$

Suppose first d inequalities are tight at x^*.

Complementary Slackness: Geometric View

$$\begin{array}{ll} \textbf{maximize}: & c \cdot x \\ \text{subject to} & Ax \le b \end{array} \quad \xrightarrow{Dual} \quad \begin{array}{ll} \textbf{minimize}: & y \cdot b \\ \text{subject to} & yA = c \\ & y \ge 0 \end{array}$$

$y^*)$ satisfies complementary slackness: $\forall i = 1..n, \quad y_i^* > 0 \ \Rightarrow \ (Ax^*)_i$

Suppose first d inequalities are tight at x^*.

c is in the cone of

Optimality implies Complementary Slackness

x^* **:** Optimum vertex.

Optimality implies Complementary Slackness

x^* **:** Optimum vertex. First d inequalities tight at x^*.

$$Ax^* \leq b \quad \text{splits into} \quad \hat{A}x^* = \hat{b}, \qquad \tilde{A}x^* < \tilde{b}$$

Optimality implies Complementary Slackness

$\boldsymbol{x^*}$ **:** Optimum vertex. First $\boldsymbol{d}$ inequalities tight at $\boldsymbol{x^*}$.

$$\boldsymbol{Ax^* \leq b} \quad \text{splits into} \quad \boldsymbol{\hat{A}x^* = \hat{b}}, \qquad \boldsymbol{\tilde{A}x^* < \tilde{b}}$$

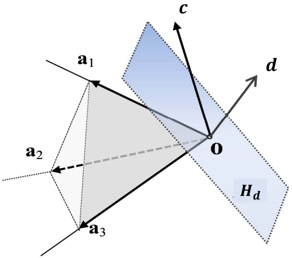
Suppose $\boldsymbol{c}$ is NOT in the cone of rows of $\boldsymbol{\hat{A}}$.

Optimality implies Complementary Slackness

x^* : Optimum vertex. First d inequalities tight at x^*.

$$Ax^* \leq b \quad \text{splits into} \quad \hat{A}x^* = \hat{b}, \qquad \tilde{A}x^* < \tilde{b}$$

Suppose c is NOT in the cone of rows of $\hat{A}$.

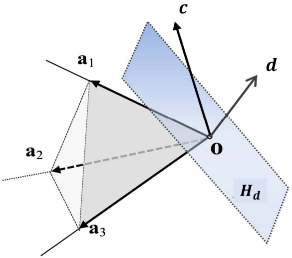


Optimality implies Complementary Slackness

$\boldsymbol{x}^*$ **:** Optimum vertex. First $\boldsymbol{d}$ inequalities tight at $\boldsymbol{x}^*$.

$$\boldsymbol{A}\boldsymbol{x}^* \leq \boldsymbol{b} \quad \text{splits into} \quad \hat{\boldsymbol{A}}\boldsymbol{x}^* = \hat{\boldsymbol{b}}, \qquad \tilde{\boldsymbol{A}}\boldsymbol{x}^* < \tilde{\boldsymbol{b}}$$

Suppose $\boldsymbol{c}$ is NOT in the cone of rows of $\hat{\boldsymbol{A}}$.



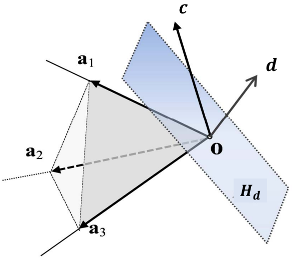
$\Rightarrow$ There exists a hyperplane separating $\boldsymbol{c}$ from the cone.

Optimality implies Complementary Slackness

x^* : Optimum vertex. First d inequalities tight at x^*.

$$Ax^* \leq b \quad \text{splits into} \quad \hat{A}x^* = \hat{b}, \qquad \tilde{A}x^* < \tilde{b}$$

Suppose c is NOT in the cone of rows of $\hat{A}$.



$\Rightarrow$ There exists a hyperplane separating c from the cone.
Suppose cone is on the negative side, and c on the positive size. If the d is the normal vector of the hyperplane, then formally,

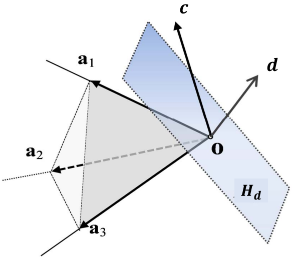
$$\hat{A}d < 0, \qquad c \cdot d > 0$$

Optimality implies Complementary Slackness

$\boldsymbol{x}^*$ **:** Optimum vertex. First $\boldsymbol{d}$ inequalities tight at $\boldsymbol{x}^*$.

$$\boldsymbol{A}\boldsymbol{x}^* \leq \boldsymbol{b} \quad \text{splits into} \quad \hat{\boldsymbol{A}}\boldsymbol{x}^* = \hat{\boldsymbol{b}}, \qquad \tilde{\boldsymbol{A}}\boldsymbol{x}^* < \tilde{\boldsymbol{b}}$$

Suppose $\boldsymbol{c}$ is NOT in the cone of rows of $\hat{\boldsymbol{A}}$.



$\Rightarrow$ There exists a hyperplane separating $\boldsymbol{c}$ from the cone.
Suppose cone is on the negative side, and $\boldsymbol{c}$ on the positive size. If the $\boldsymbol{d}$ is the normal vector of the hyperplane, then formally,

$$\hat{\boldsymbol{A}}\boldsymbol{d} < \boldsymbol{0}, \qquad \boldsymbol{c} \cdot \boldsymbol{d} > \boldsymbol{0} \qquad \text{Also known as Farkas' Lemma}$$

Optimality implies Complementary Slackness

x^* **:** Optimum vertex.

Optimality implies Complementary Slackness

$\boldsymbol{x}^*$ **:** Optimum vertex. First $\boldsymbol{d}$ inequalities tight at $\boldsymbol{x}^*$.

$$\boldsymbol{A}\boldsymbol{x}^* \leq \boldsymbol{b} \quad \text{splits into} \quad \hat{\boldsymbol{A}}\boldsymbol{x}^* = \hat{\boldsymbol{b}}, \qquad \tilde{\boldsymbol{A}}\boldsymbol{x}^* < \tilde{\boldsymbol{b}}$$

Suppose $\boldsymbol{c}$ is NOT in the cone of rows of $\hat{\boldsymbol{A}}$.

$\Rightarrow$ There exists a hyperplane separating $\boldsymbol{c}$ from the cone.

$$\hat{\boldsymbol{A}}\boldsymbol{d} < \boldsymbol{0}, \qquad \boldsymbol{c} \cdot \boldsymbol{d} > \boldsymbol{0} \qquad \text{Also known as Farkas' Lemma}$$

Optimality implies Complementary Slackness

x^* **:** Optimum vertex. First d inequalities tight at x^*.

$$Ax^* \leq b \quad \text{splits into} \quad \hat{A}x^* = \hat{b}, \qquad \tilde{A}x^* < \tilde{b}$$

Suppose c is NOT in the cone of rows of $\hat{A}$.

$\Rightarrow$ There exists a hyperplane separating c from the cone.

$$\hat{A}d < 0, \qquad c \cdot d > 0 \quad \text{Also known as Farkas' Lemma}$$

Choose v. v. tiny $\epsilon > 0$ such that $\tilde{A}(x^* + \epsilon d) \leq \tilde{b}$.

$$\hat{A}(x^* + \epsilon d) = \hat{A}x^* + \epsilon \hat{A}d < \hat{b} \ \Rightarrow$$

Optimality implies Complementary Slackness

x^* : Optimum vertex. First d inequalities tight at x^*.

$$Ax^* \leq b \quad \text{splits into} \quad \hat{A}x^* = \hat{b}, \qquad \tilde{A}x^* < \tilde{b}$$

Suppose c is NOT in the cone of rows of $\hat{A}$.

$\Rightarrow$ There exists a hyperplane separating c from the cone.

$$\hat{A}d < 0, \qquad c \cdot d > 0 \quad \text{Also known as Farkas' Lemma}$$

Choose v. v. tiny $\epsilon > 0$ such that $\tilde{A}(x^* + \epsilon d) \leq \tilde{b}$.

$$\hat{A}(x^* + \epsilon d) = \hat{A}x^* + \epsilon \hat{A}d < \hat{b} \;\Rightarrow\; (x^* + \epsilon d) \text{ is feasible in } \Pi$$

Optimality implies Complementary Slackness

x^* **:** Optimum vertex. First d inequalities tight at x^*.

$$Ax^* \leq b \quad \text{splits into} \quad \hat{A}x^* = \hat{b}, \qquad \tilde{A}x^* < \tilde{b}$$

Suppose c is NOT in the cone of rows of $\hat{A}$.

$\Rightarrow$ There exists a hyperplane separating c from the cone.

$$\hat{A}d < 0, \qquad c \cdot d > 0 \qquad \text{Also known as Farkas' Lemma}$$

Choose v. v. tiny $\epsilon > 0$ such that $\tilde{A}(x^* + \epsilon d) \leq \tilde{b}$.

$$\hat{A}(x^* + \epsilon d) = \hat{A}x^* + \epsilon \hat{A}d < \hat{b} \;\Rightarrow\; (x^* + \epsilon d) \text{ is feasible in } \Pi$$

$c \cdot (x^* + \epsilon d) = c \cdot x^* + \epsilon(c \cdot d)$

Optimality implies Complementary Slackness

x^* **:** Optimum vertex. First d inequalities tight at x^*.

$$Ax^* \leq b \quad \text{splits into} \quad \hat{A}x^* = \hat{b}, \qquad \tilde{A}x^* < \tilde{b}$$

Suppose c is NOT in the cone of rows of $\hat{A}$.

$\Rightarrow$ There exists a hyperplane separating c from the cone.

$$\hat{A}d < 0, \qquad c \cdot d > 0 \qquad \text{Also known as Farkas' Lemma}$$

Choose v. v. tiny $\epsilon > 0$ such that $\tilde{A}(x^* + \epsilon d) \leq \tilde{b}$.

$$\hat{A}(x^* + \epsilon d) = \hat{A}x^* + \epsilon \hat{A}d < \hat{b} \;\Rightarrow\; (x^* + \epsilon d) \text{ is feasible in } \Pi$$

$$c \cdot (x^* + \epsilon d) = c \cdot x^* + \epsilon(c \cdot d) > c \cdot x^* \Rightarrow$$

Optimality implies Complementary Slackness

$\boldsymbol{x}^*$ **:** Optimum vertex. First $\boldsymbol{d}$ inequalities tight at $\boldsymbol{x}^*$.

$$\boldsymbol{A}\boldsymbol{x}^* \leq \boldsymbol{b} \quad \text{splits into} \quad \hat{\boldsymbol{A}}\boldsymbol{x}^* = \hat{\boldsymbol{b}}, \qquad \tilde{\boldsymbol{A}}\boldsymbol{x}^* < \tilde{\boldsymbol{b}}$$

Suppose $\boldsymbol{c}$ is NOT in the cone of rows of $\hat{\boldsymbol{A}}$.

$\Rightarrow$ There exists a hyperplane separating $\boldsymbol{c}$ from the cone.

$$\hat{\boldsymbol{A}}\boldsymbol{d} < \boldsymbol{0}, \qquad \boldsymbol{c} \cdot \boldsymbol{d} > \boldsymbol{0} \qquad \text{Also known as Farkas' Lemma}$$

Choose v. v. tiny $\epsilon > \boldsymbol{0}$ such that $\tilde{\boldsymbol{A}}(\boldsymbol{x}^* + \epsilon \boldsymbol{d}) \leq \tilde{\boldsymbol{b}}$.

$$\hat{\boldsymbol{A}}(\boldsymbol{x}^* + \epsilon \boldsymbol{d}) = \hat{\boldsymbol{A}}\boldsymbol{x}^* + \epsilon \hat{\boldsymbol{A}}\boldsymbol{d} < \hat{\boldsymbol{b}} \;\Rightarrow\; (\boldsymbol{x}^* + \epsilon \boldsymbol{d}) \text{ is feasible in } \boldsymbol{\Pi}$$

$\boldsymbol{c} \cdot (\boldsymbol{x}^* + \epsilon \boldsymbol{d}) = \boldsymbol{c} \cdot \boldsymbol{x}^* + \epsilon(\boldsymbol{c} \cdot \boldsymbol{d}) > \boldsymbol{c} \cdot \boldsymbol{x}^* \Rightarrow \boldsymbol{x}^*$ **is NOT optimum!**

Proof of Strong Duality

x^* : Optimum vertex. First d inequalities tight at x^*.

$$Ax^* \leq b \quad \text{splits into} \quad \hat{A}x^* = \hat{b}, \qquad \tilde{A}x^* < \tilde{b}$$

Then c IS in the cone of rows of $\hat{A}$.

Proof of Strong Duality

x^* **:** Optimum vertex. First d inequalities tight at x^*.

$$Ax^* \leq b \quad \text{splits into} \quad \hat{A}x^* = \hat{b}, \qquad \tilde{A}x^* < \tilde{b}$$

Then c IS in the cone of rows of $\hat{A}$.

$\Leftrightarrow$ y^* feasible in $\mathrm{Dual}(\Pi)$ such that (x^*, y^*) satisfies complementary slackness.

Proof of Strong Duality

x^* **:** Optimum vertex. First d inequalities tight at x^*.

$$Ax^* \leq b \quad \text{splits into} \quad \hat{A}x^* = \hat{b}, \qquad \tilde{A}x^* < \tilde{b}$$

Then c IS in the cone of rows of $\hat{A}$.

$\Leftrightarrow$ y^* feasible in $\mathrm{Dual}(\Pi)$ such that (x^*, y^*) satisfies complementary slackness.

$\Leftrightarrow$ (x^*, y^*) satisfies strong duality, $c \cdot x^* = y^* \cdot b$.

Proof of Strong Duality

x^* **:** Optimum vertex. First d inequalities tight at x^*.

$$Ax^* \leq b \quad \text{splits into} \quad \hat{A}x^* = \hat{b}, \qquad \tilde{A}x^* < \tilde{b}$$

Then c IS in the cone of rows of $\hat{A}$.

$\Leftrightarrow$ y^* feasible in $\mathbf{Dual(\Pi)}$ such that (x^*, y^*) satisfies complementary slackness.

$\Leftrightarrow$ (x^*, y^*) satisfies strong duality, $c \cdot x^* = y^* \cdot b$. (implies y^* optimal in $\mathbf{Dual(\Pi)}$).

Proof of Strong Duality

$\boldsymbol{x}^*$ **:** Optimum vertex. First $\boldsymbol{d}$ inequalities tight at $\boldsymbol{x}^*$.

$$\boldsymbol{A}\boldsymbol{x}^* \leq \boldsymbol{b} \quad \text{splits into} \quad \hat{\boldsymbol{A}}\boldsymbol{x}^* = \hat{\boldsymbol{b}}, \qquad \tilde{\boldsymbol{A}}\boldsymbol{x}^* < \tilde{\boldsymbol{b}}$$

Then $\boldsymbol{c}$ IS in the cone of rows of $\hat{\boldsymbol{A}}$.

$\Leftrightarrow$ $\boldsymbol{y}^*$ feasible in $\mathbf{Dual}(\boldsymbol{\Pi})$ such that $(\boldsymbol{x}^*, \boldsymbol{y}^*)$ satisfies complementary slackness.

$\Leftrightarrow$ $(\boldsymbol{x}^*, \boldsymbol{y}^*)$ satisfies strong duality, $\boldsymbol{c} \cdot \boldsymbol{x}^* = \boldsymbol{y}^* \cdot \boldsymbol{b}$. (implies $\boldsymbol{y}^*$ optimal in $\mathbf{Dual}(\boldsymbol{\Pi})$).

Theorem (Strong Duality)

If x^* is an optimal solution to Π and y^* is an optimal solution to $\mathbf{Dual}(\Pi)$ then $c \cdot x^* = y^* \cdot b$.

Duality for another canonical form

Compactly, for the primal LP Π

$$\begin{array}{ll} \max & c \cdot x \\ \text{subject to} & Ax \leq b, \; x \geq 0 \end{array}$$

the dual LP is $\mathrm{Dual}(\Pi)$

$$\begin{array}{ll} \min & y \cdot b \\ \text{subject to} & yA \geq c, \; y \geq 0 \end{array}$$

Definition (Complementary Slackness)

x feasible in Π and y feasible in $\mathrm{Dual}(\Pi)$, s.t.,

$$\forall i = 1, \ldots, n, \quad y_i > 0 \; \Rightarrow \; (Ax)_i = b_i$$
$$\forall j = 1, \ldots, d, \quad x_j > 0 \; \Rightarrow \; (yA)_j = c_j$$

In General...

from Jeff's notes

Primal	Dual
$\max c \cdot x$	$\min y \cdot b$
$\sum_j a_{ij} x_j \leq b_i$	$y_i \geq 0$
$\sum_j a_{ij} x_j \geq b_i$	$y_i \leq 0$
$\sum_j a_{ij} x_j = b_i$	–
$x_j \geq 0$	$\sum_i y_i a_{ij} \geq c_j$
$x_j \leq 0$	$\sum_i y_i a_{ij} \leq c_j$
–	$\sum_i y_i a_{ij} = c_j$
$x_j = 0$	–

Primal	Dual
$\min c \cdot x$	$\max y \cdot b$
$\sum_j a_{ij} x_j \leq b_i$	$y_i \leq 0$
$\sum_j a_{ij} x_j \geq b_i$	$y_i \geq 0$
$\sum_j a_{ij} x_j = b_i$	–
$x_j \leq 0$	$\sum_i y_i a_{ij} \geq c_j$
$x_j \geq 0$	$\sum_i y_i a_{ij} \leq c_j$
–	$\sum_i y_i a_{ij} = c_j$
$x_j = 0$	–

Figure H.4. Constructing the dual of an arbitrary linear program.

Part III

Examples of Duality

Network flow

s-t flow in directed graph $G = (V, E)$ with capacities c. Assume for simplicity that no incoming edges into s.

$$
\begin{aligned}
\textbf{max} \quad & \sum_{(s,v)\in \mathrm{E}} x(s,v) \\
& \sum_{(u,v)\in \mathrm{E}} x(u,v) - \sum_{(v,w)\in \mathrm{E}} x(v,w) = 0 \quad \forall v \in \mathrm{V} \setminus \{s,t\} \\
& x(u,v) \leq c(u,v) \qquad \forall (u,v) \in \mathrm{E} \\
& x(u,v) \geq 0 \qquad \forall (u,v) \in \mathrm{E}.
\end{aligned}
$$

Network flow: Equivalent formulation

Directed graph $G = (V, E)$, with capacities on edges. Add a t to s edge with infinite capacity. Now maximize flow on this edge.

$$\begin{aligned}
\textbf{max} \quad & x(t, s) \\
& \sum_{(u,v)\in \mathrm{E}} x(u, v) - \sum_{(v,w)\in \mathrm{E}} x(v, w) = 0 && \forall v \in V \\
& x(u, v) \leq c(u, v) && \forall (u, v) \in \mathrm{E} \setminus (t, s) \\
& x(u, v) \geq 0 && \forall (u, v) \in \mathrm{E}.
\end{aligned}$$

Dual of Network Flow

Summary

1. Linear Programming is a useful and powerful (modeling) problem.

Summary

1. Linear Programming is a useful and powerful (modeling) problem.
2. Can be solved in polynomial time. Practical solvers available commercially as well as in open source. Whether there is a strongly polynomial time algorithm is a major open problem.

Summary

1. Linear Programming is a useful and powerful (modeling) problem.
2. Can be solved in polynomial time. Practical solvers available commercially as well as in open source. Whether there is a strongly polynomial time algorithm is a major open problem.
3. Geometry and linear algebra are important to understand the structure of LP and in algorithm design. Vertex solutions imply that LPs have poly-sized optimum solutions. This implies that LP is in **NP**.

Summary

1. Linear Programming is a useful and powerful (modeling) problem.
2. Can be solved in polynomial time. Practical solvers available commercially as well as in open source. Whether there is a strongly polynomial time algorithm is a major open problem.
3. Geometry and linear algebra are important to understand the structure of LP and in algorithm design. Vertex solutions imply that LPs have poly-sized optimum solutions. This implies that LP is in **NP**.
4. Duality is a critical tool in the theory of linear programming. Duality implies the Linear Programming is in **co-NP**. Do you see why?

Part IV

Integer Linear Programming

Integer Linear Programming

Problem

Find a vector $x \in Z^d$ (integer values) that

$$\begin{array}{ll} \text{maximize} & \sum_{j=1}^{d} c_j x_j \\ \text{subject to} & \sum_{j=1}^{d} a_{ij} x_j \leq b_i \quad \text{for } i = 1 \dots n \end{array}$$

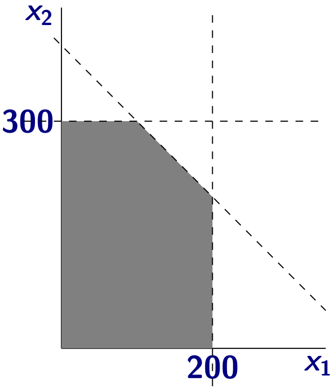
Input is matrix $A = (a_{ij}) \in \mathbb{R}^{n \times d}$, column vector $b = (b_i) \in \mathbb{R}^n$, and row vector $c = (c_j) \in \mathbb{R}^d$

Factory Example

$$\begin{array}{ll} \text{maximize} & x_1 + 6x_2 \\ \text{subject to} & x_1 \leq 200 \quad x_2 \leq 300 \quad x_1 + x_2 \leq 400 \\ & x_1, x_2 \geq 0 \end{array}$$

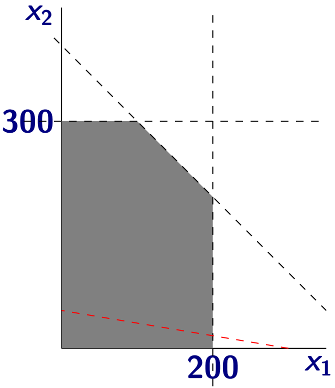
Suppose we want x_1, x_2 to be integer valued.

Factory Example Figure



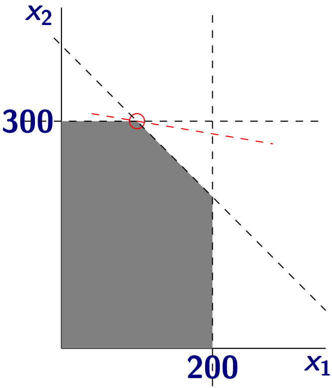
1. Feasible values of x_1 and x_2 are integer points in shaded region
2. Optimization function is a line; moving the line until it just leaves the final integer point in feasible region, gives optimal values

Factory Example Figure



1. Feasible values of x_1 and x_2 are integer points in shaded region
2. Optimization function is a line; moving the line until it just leaves the final integer point in feasible region, gives optimal values

Factory Example Figure



1. Feasible values of x_1 and x_2 are integer points in shaded region
2. Optimization function is a line; moving the line until it just leaves the final integer point in feasible region, gives optimal values

Integer Programming

Can model many difficult discrete optimization problems as integer programs!

Therefore integer programming is a hard problem. NP-hard.

Integer Programming

Can model many difficult discrete optimization problems as integer programs!

Therefore integer programming is a hard problem. NP-hard.

Can relax integer program to linear program and *approximate*.

Integer Programming

Can model many difficult discrete optimization problems as integer programs!

Therefore integer programming is a hard problem. NP-hard.

Can relax integer program to linear program and *approximate*.

Practice: integer programs are solved by a variety of methods

1. branch and bound
2. branch and cut
3. adding cutting planes
4. linear programming plays a fundamental role

Example: Maximum Independent Set

Definition

Given undirected graph $G = (V, E)$ a subset of nodes $S \subseteq V$ is an independent set (also called a stable set) if for there are no edges between nodes in S. That is, if $u, v \in S$ then $(u, v) \notin E$.

Input Graph $G = (V, E)$

Goal Find maximum sized independent set in G

Example: Dominating Set

Definition

Given undirected graph $G = (V, E)$ a subset of nodes $S \subseteq V$ is a dominating set if for all $v \in V$, either $v \in S$ or a neighbor of v is in S.

Input: Graph $G = (V, E)$, weights $w(v) \geq 0$ for $v \in V$

Goal: Find minimum weight dominating set in G

Example: s-t minimum cut and implicit constraints

Input Graph $G = (V, E)$, edge capacities $c(e), e \in E$.
$s, t \in V$

Goal Find minimum capacity s-t cut in G.

Linear Programs with Integer Vertices

Suppose we know that for a linear program *all* vertices have integer coordinates.

Linear Programs with Integer Vertices

Suppose we know that for a linear program *all* vertices have integer coordinates.
Then solving linear program is same as solving integer program. We know how to solve linear programs efficiently (polynomial time) and hence we get an integer solution for free!

Linear Programs with Integer Vertices

Suppose we know that for a linear program *all* vertices have integer coordinates.
Then solving linear program is same as solving integer program. We know how to solve linear programs efficiently (polynomial time) and hence we get an integer solution for free!

Luck or *Structure*:

1. Linear program for flows with integer capacities have integer vertices
2. Linear program for matchings in bipartite graphs have integer vertices
3. A complicated linear program for matchings in general graphs have integer vertices.

All of above problems can hence be solved efficiently.

Linear Programs with Integer Vertices

Meta Theorem: A combinatorial optimization problem can be solved efficiently if and only if there is a linear program for problem with integer vertices.

Consequence of the Ellipsoid method for solving linear programming.

In a sense linear programming and other geometric generalizations such as convex programming are the most general problems that we can solve efficiently.

Summary

1. Linear Programming is a useful and powerful (modeling) problem.
2. Can be solved in polynomial time. Practical solvers available commercially as well as in open source. Whether there is a strongly polynomial time algorithm is a major open problem.
3. Geometry and linear algebra are important to understand the structure of LP and in algorithm design. Vertex solutions imply that LPs have poly-sized optimum solutions. This implies that LP is in **NP**.
4. Duality is a critical tool in the theory of linear programming. Duality implies the Linear Programming is in **co-NP**. Do you see why?
5. **Integer Programming in NP-Complete. LP-based techniques critical in heuristically solving integer programs.**