Today:
– Linear Programming (con’t.)

Many of these slides are adapted from several online sources
Reading Assignments

• Today’s class:
  – Chapter 29.3, 29.5

• Reading assignment for next Thursday’s class:
  – Chapter 29.4
Recall: Formatting problems as LPs – SSSP

• Single Source Shortest Path :
  – Input: A weighted direct graph G=<V,E> with weighted function \( w: E \rightarrow \mathbb{R} \), a source \( s \) and a destination \( t \), compute \( d \) which is the weight of the shortest path from \( s \) to \( t \).
  – Formulate as a LP:
    • For each vertex \( v \), introduce a variable \( d_v \): the weight of the shortest path from \( s \) to \( v \).
    • LP:
      
      \[
      \begin{align*}
      \text{maximize} & \quad d_t \\
      \text{subject to:} & \quad d_v \leq d_u + w(u,v) \quad \text{for each edge } (u,v) \in E \\
      & \quad d_s = 0
      \end{align*}
      \]

Q: Why is this a maximization?
Q: How many variables? \(|V|\)
Q: How many constraints? \(|E|+1\)
In-Class Exercise #1

Write out explicitly the linear program corresponding to finding the shortest path from node $s$ to node $y$ in the figure below:

\[
\begin{align*}
\text{maximize} & \quad d_t \\
\text{subject to:} & \quad d_v \leq d_u + w(u,v) \text{ for each edge } (u,v) \in E \\
& \quad d_s = 0
\end{align*}
\]
In-Class Exercise #1

Write out explicitly the linear program corresponding to finding the shortest path from node $s$ to node $y$ in the figure below:

\[
\begin{align*}
\text{maximize } & \quad d_y \\
\text{subject to: } & \quad d_v \leq d_u + w(u,v) \text{ for each edge } (u,v) \in E \\
& \quad d_s = 0
\end{align*}
\]
Recall: Formatting Max-flow problem as LP

maximize $\sum_{v \in V} f_{sv} - \sum_{v \in V} f_{vs}$
subject to:

- $f_{uv} \leq c(u,v)$ for all $u, v \in V$ //capacity constraints
- $\sum_{v \in V} f_{vu} = \sum_{v \in V} f_{uv}$ for all $u \in V - \{s,t\}$ //flow conservation
- $f_{uv} \geq 0$ for all $u, v \in V$ //non-negativity constraints
In-Class Exercise #2

Write out explicitly the linear program corresponding to finding the maximum flow in the figure below:

\[
\begin{align*}
\text{maximize} & \quad \sum_{v \in V} f_{sv} - \sum_{v \in V} f_{vs} \\
\text{subject to:} & \quad f_{uv} \leq c(u,v) \quad \text{for all } u, v \in V \\
& \quad \sum_{v \in V} f_{vu} = \sum_{v \in V} f_{uv} \quad \text{for all } u \in V - \{s,t\} \\
& \quad f_{uv} \geq 0 \quad \text{for all } u, v \in V
\end{align*}
\]
maximize $\sum_{v \in V} f_{sv} - \sum_{v \in V} f_{vs}$
subject to:

$f_{uv} \leq c(u,v)$ for all $u, v \in V$

$\sum_{v \in V} f_{vu} = \sum_{v \in V} f_{uv}$ for all $u \in V - \{s, t\}$

$f_{uv} \geq 0$ for all $u, v \in V$

maximize $f_{ss} + f_{sv_1} + f_{sv_2} + f_{sv_3} + f_{sv_4} + f_{st}$
subject to:

$f_{sv_1} \leq 16$  $f_{v_1v_3} \leq 12$  $f_{v_4v_3} \leq 7$  $f_{ss} \leq 0$  $f_{st} \leq 0$

$f_{sv_2} \leq 13$  $f_{v_3v_2} \leq 9$  $f_{v_4t} \leq 4$  $f_{sv_3} \leq 0$  $f_{v_1s} \leq 0$

$f_{v_2v_1} \leq 4$  $f_{v_2v_4} \leq 14$  $f_{v_3t} \leq 20$  $f_{sv_4} \leq 0$  $f_{v_2s} \leq 0$

...
Solving LPs using SIMPLEX...

• First, another recap (via example) to remember how SIMPLEX works...
Example for Simplex algorithm

Maximize $3x_1 + x_2 + 2x_3$

Subject to:
\[
\begin{align*}
  x_1 + x_2 + 3x_3 & \leq 30 \\
  2x_1 + 2x_2 + 5x_3 & \leq 24 \\
  4x_1 + x_2 + 2x_3 & \leq 36 \\
  x_1, x_2, x_3 & \geq 0
\end{align*}
\]

Change to slack form:
\[
\begin{align*}
  z &= 3x_1 + x_2 + 2x_3 \\
  x_4 &= 30 - x_1 - x_2 - 3x_3 \\
  x_5 &= 24 - 2x_1 - 2x_2 - 5x_3 \\
  x_6 &= 36 - 4x_1 - x_2 - 2x_3 \\
  x_1, x_2, x_3, x_4, x_5, x_6 & \geq 0
\end{align*}
\]
Simplex algorithm steps

- Recall: “Feasible solutions” (infinite number of them):
  - A feasible solution is any whose values satisfy constraints
  - In previous example, solution is feasible as long as all of $x_1$, $x_2$, $x_3$, $x_4$, $x_5$, $x_6$ are nonnegative
- Basic solution:
  - set all nonbasic variables to 0 and compute all basic variable values
- Iteratively rewrite the set of equations such that:
  - There is no change to the underlying LP problem (i.e., new form is equivalent to old)
  - Feasible solutions stay the same
  - The basic solution is changed, to result in a greater objective value:
    - Select a nonbasic variable $x_e$ whose coefficient in the objective function is positive
    - Increase value of $x_e$ as much as possible without violating any of the constraints
    - Make $x_e$ a basic variable
    - Select some other variable to become nonbasic

\[
\begin{align*}
  z &= 3x_1 + x_2 + 2x_3 \\
  x_4 &= 30 - x_1 - x_2 - 3x_3 \\
  x_5 &= 24 - 2x_1 - 2x_2 - 5x_3 \\
  x_6 &= 36 - 4x_1 - x_2 - 2x_3 \\
  x_1, x_2, x_3, x_4, x_5, x_6 &\geq 0
\end{align*}
\]
Example

- **Basic solution**: \((x_1, x_2, x_3, x_4, x_5, x_6) = (0, 0, 0, 30, 24, 36)\)
  - The objective value is \(z = 3 \cdot 0 + 0 + 2 \cdot 0 = 0\) (Not a maximum)

- Try to increase the value of **nonbasic variable** \(x_1\) while maintaining constraints:
  - Increase \(x_1\) to 30: means that \(x_4\) will be OK (i.e., non-negative)
  - Increase \(x_1\) to 12 means that \(x_5\) will be OK.
  - Increase \(x_1\) to 9 means that \(x_6\) will be OK.
  - We have to choose most constraining value \(x_1\) is most constrained by \(x_6\), so we switch the roles of \(x_1\) and \(x_6\)

- Change \(x_1\) to **basic** variable by rewriting last constraint to:
  \[ x_1 = 9 - x_2/4 - x_3/2 - x_6/4 \]
  - Note: \(x_6\) becomes nonbasic.
  - Replace \(x_1\) with above formula in all equations to get...

\[
\begin{align*}
z &= 3x_1 + x_2 + 2x_3 \\
x_4 &= 30 - x_1 - x_2 - 3x_3 \\
x_5 &= 24 - 2x_1 - 2x_2 - 5x_3 \\
x_6 &= 36 - 4x_1 - x_2 - 2x_3 \\
x_1, x_2, x_3, x_4, x_5, x_6 &\geq 0
\end{align*}
\]
Example (con’t.)

\[ z = 27 + x_2/4 + x_3/2 - 3x_6/4 \]
\[ x_1 = 9 - x_2/4 - x_3/2 - x_6/4 \]
\[ x_4 = 21 - 3x_2/4 - 5x_3/2 + x_6/4 \]
\[ x_5 = 6 - 3x_2/2 - 4x_3 + x_6/2 \]

- This operation is called **pivot**
  - A pivot chooses a nonbasic variable, called **entering variable**, and a basic variable, called **leaving variable**, and changes their roles.
  - The pivot operation results in an equivalent LP.
  - Reality check: original solution (0,0,0,30,24,36) satisfies the new equations.

- In the example,
  - \( x_1 \) is entering variable, and \( x_6 \) is leaving variable.
  - \( x_2, x_3, x_6 \) are nonbasic, and \( x_1, x_4, x_5 \) becomes basic.
  - The basic solution for this new LP form is (9,0,0,21,6,0), with \( z = 27 \).
    (Yippee \( \Rightarrow \) \( z = 27 \) is better than \( z = 0! \))
• We iterate again – try to find a new variable whose value may increase.
  – $x_6$ will not work, since $z$ will decrease.
  – $x_2$ and $x_3$ are OK. Suppose we select $x_3$.
• How far can we increase $x_3$?
  – First constraint limits it to 18
  – Second constraint limits it to 42/5
  – Third constraint limits it to 3/2 – most constraining $\Rightarrow$ swap roles of $x_3$ and $x_5$
• So rewrite last constraint to:
  $$x_3 = \frac{3}{2} - \frac{3}{8}x_2 - \frac{1}{4}x_5 + \frac{1}{8}x_6$$
• Replace $x_3$ with the above in all the equations to get...
Example (con’t.)

• The new LP equations:
  \[ z = \frac{111}{4} + \frac{x_2}{16} - \frac{x_5}{8} - \frac{11x_6}{16} \]
  \[ x_1 = \frac{33}{2} - \frac{x_2}{16} + \frac{x_5}{8} - \frac{5x_6}{16} \]
  \[ x_3 = \frac{3}{2} - 3\frac{x_2}{8} - \frac{x_5}{4} + \frac{x_6}{8} \]
  \[ x_4 = \frac{69}{4} + 3\frac{x_2}{16} + 5\frac{x_5}{8} - \frac{x_6}{16} \]

• The basic solution is \((33/4,0,3/2,69/4,0,0)\) with \(z = 111/4\).

• Now we can only increase \(x_2\).
  – First constraint limits \(x_2\) to 132
  – Second to 4
  – Third to \(\infty\)

• So rewrite second constraint to:
  \[ x_2 = 4 - 8\frac{x_3}{3} - 2\frac{x_5}{3} + \frac{x_6}{3} \]

• Replace in all equations to get...
Example (con’t.)

- Rewritten LP equations:
  
  \[ z = 28 - \frac{x_3}{6} - \frac{x_5}{6} - 2\frac{x_6}{3} \]
  
  \[ x_1 = 8 + \frac{x_3}{6} + \frac{x_5}{6} - \frac{x_6}{3} \]
  
  \[ x_2 = 4 - 8\frac{x_3}{3} - 2\frac{x_5}{3} + \frac{x_6}{3} \]
  
  \[ x_4 = 18 - \frac{x_3}{2} + \frac{x_5}{2} \]

- At this point, all coefficients in objective functions are negative.
- So, no further rewrite is possible.

- Means that we’ve found the optimal solution.
- The basic solution is \((8,4,0,18,0,0)\) with objective value \(z = 28\).
- The original variables are \(x_1, x_2, x_3\), with values \((8,4,0)\)
Simplex algorithm --Pivot

\textbf{Pivot}(N, B, A, b, c, v, l, e)
1 \triangleright Compute the coefficients of the equation for new basic variable \(x_e\).
2 \(\hat{b}_e \leftarrow b_l/a_{le}\)
3 \textbf{for} each \(j \in N - \{e\}\) 
4 \hspace{1em} \textbf{do} \(\hat{a}_{ej} \leftarrow a_{ij}/a_{le}\)
5 \(\hat{a}_{el} \leftarrow 1/a_{le}\)
6 \triangleright Compute the coefficients of the remaining constraints.
7 \textbf{for} each \(i \in B - \{l\}\)
8 \hspace{1em} \textbf{do} \(\hat{b}_i \leftarrow b_i - a_{ie}\hat{b}_e\)
9 \hspace{2em} \textbf{for} each \(j \in N - \{e\}\)
10 \hspace{3em} \textbf{do} \(\hat{a}_{ij} \leftarrow a_{ij} - a_{ie}\hat{a}_{ej}\)
11 \hspace{2em} \hat{a}_{il} \leftarrow -a_{ie}\hat{a}_{el}\)
12 \triangleright Compute the objective function.
13 \(\hat{v} \leftarrow v + c_e\hat{b}_e\)
14 \textbf{for} each \(j \in N - \{e\}\)
15 \hspace{1em} \textbf{do} \(\hat{c}_j \leftarrow c_j - c_e\hat{a}_{ej}\)
16 \hspace{1em} \hat{c}_l \leftarrow -c_e\hat{a}_{el}\)
17 \triangleright Compute new sets of basic and nonbasic variables.
18 \(\hat{N} = N - \{e\} \cup \{l\}\)
19 \(\hat{B} = B - \{l\} \cup \{e\}\)
20 \textbf{return} \((\hat{N}, \hat{B}, \hat{A}, \hat{b}, \hat{c}, \hat{v})\)

\(N\): indices set of nonbasic variables
\(B\): indices set of basic variables
\(A\): \(a_{ij}\)
\(b\): \(b_i\)
\(c\): \(c_i\)
\(v\): constant coefficient.
\(e\): index of entering variable
\(l\): index of leaving variable

\[ z = v + \sum_{j \in N} c_j x_j \]
\[ x_i = b_i - \sum_{j \in N} a_{ij} x_j \quad \text{for} \ i \in B \]
Issues in Solving LP

• How to determine if LP is feasible?

• What if LP is feasible, but initial basic solution is not feasible?
  • Presume we have procedure, INITIALIZE-SIMPLEX, that takes LP in standard form and returns slack form for which initial basic solution is feasible (or states that the problem is infeasible)

• How to determine whether LP is unbounded?
  – If none of the constraints limits the amount by which the entering variable can increase, the LP is unbounded

• How to choose entering and leaving variables?
  – By selecting variable that limits entering variables the most
  – Break ties using Bland’s rule, which always chooses variable with smallest index
Formal Simplex algorithm

\textsc{Simplex}(A, b, c)

1. \((N, B, A, b, c, v) \leftarrow \text{Initialize-Simplex}(A, b, c)\)
2. \textbf{while} some index \(j \in N\) has \(c_j > 0\)
3. \hspace{1em} \textbf{do} choose an index \(e \in N\) for which \(c_e > 0\)
4. \hspace{2em} \textbf{for} each index \(i \in B\)
5. \hspace{3em} \textbf{do if} \(a_{ie} > 0\)
6. \hspace{4em} \textbf{then} \(\Delta_i \leftarrow b_i / a_{ie}\)
7. \hspace{4em} \textbf{else} \(\Delta_i \leftarrow \infty\)
8. \hspace{2em} choose an index \(l \in B\) that minimizes \(\Delta_i\)
9. \hspace{1em} \textbf{if} \(\Delta_l = \infty\)
10. \hspace{2em} \textbf{then return} “unbounded”
11. \hspace{2em} \textbf{else} \((N, B, A, b, c, v) \leftarrow \text{Pivot}(N, B, A, b, c, v, l, e)\)
12. \textbf{for} \(i \leftarrow 1\) to \(n\)
13. \hspace{2em} \textbf{do if} \(i \in B_{+m}\)
14. \hspace{3em} \textbf{then} \(\bar{x}_i \leftarrow b_i\)
15. \hspace{3em} \textbf{else} \(\bar{x}_i \leftarrow 0\)
16. \textbf{return} \((\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_n)\)
Correctness of SIMPLEX

(Presume INITIALIZE-SIMPLEX is correct, for now.)

• First:
  – Show that if solution is returned, then that solution is feasible
  – Show that if SIMPLEX says “unbounded”, then the LP is indeed unbounded

• Sketch of this part of proof:
  – 3-part invariant (at the beginning of the while loop):
    • The slack form is equivalent to that returned by INITIALIZE-SIMPLEX
    • For each $i \in B$, $b_i \geq 0$
    • The basic solution associated with slack form is feasible
  – Show that this invariant is true:
    • At the beginning (easy to show)
    • During each iteration (show via correctness of pivot)
    • At termination (look at 2 cases of when SIMPLEX terminates, and show true for each case)
Correctness of SIMPLEX (con’t.)

• Next, show that SIMPLEX does indeed terminate

• Reason why it might not terminate?
  – Cycling:
    • Would occur if SIMPLEX oscillates between solutions that leave objective value unchanged ("degeneracy")

• Helpful lemma:
  – The slack form of a LP is uniquely determined by the set of basic variables
    • Proof:
      – By contradiction. Assume there are 2 different slack forms, then work through the algebra to show that the 2 forms must be identical.
Correctness of SIMPLEX (con’t.)

• How to prevent cycling?
  • Break ties for choosing entering and leaving variables, using *Bland’s rule*:
    – Choose entering variable with smallest index (which also has positive coefficient in objective function)
    – After having chosen entering variable, if there are now ties for choosing leaving variable, chose the leaving variable with smallest index
    – Proof is tedious, so omitted here 😊
Running time of Simplex

• **Lemma:**
  – Assuming that INITIALIZE-SIMPLEX returns a slack form for which the basic solution is feasible, SIMPLEX either reports that a linear program is unbounded, or it terminates with a feasible solution in at most \( \binom{n + m}{m} \) iterations
    (where \( n = \# \text{ non-basic variables} \) and \( m = \# \text{ basic variables} \))

• **Idea:**
  – There are at most \( \binom{n + m}{m} \) ways to choose the basic variables.
  – The set of basic variables defines a unique slack from.
  – Thus, there at most \( \binom{n + m}{m} \) unique slack forms.
  – If S SIMPLEX runs for more than \( \binom{n + m}{m} \) iterations, it cycles.
    (Thus, need to ensure there isn’t cycling. Can do this using Bland’s rule, which always chooses variable with smallest index. Proof omitted)
How to find an initial basic feasible solution?

- A LP might be feasible, but the initial basic solution might not be feasible
- To address, formulate an auxiliary LP
- Given an LP in standard form, introduce new variable $x_0$ and formulate auxiliary LP as:

  Maximize: $-x_0$

  Subject to:

  \[ \sum_{j=1}^{n} a_{ij} x_j - x_0 \leq b_i \quad \text{for } i = 1, 2, \ldots, m \]

  \[ x_j \geq 0 \quad \text{for } j = 0, 1, \ldots, n \]

- Then original LP is feasible iff the optimal objective value of auxiliary LP is 0.
- Proof is based on original solution and the fact that $x_0 = 0$ must be an optimal solution to the auxiliary LP.
Design of \texttt{INITIALIZE-SIMPLEX}

- Check original slack form; if feasible, then done
- Otherwise
  - Form auxiliary LP, as defined previously
  - Perform a single pivot of auxiliary LP, selecting leaving variable as that with most negative value
    - In this form, the basic solution is feasible
  - Repeatedly call \texttt{PIVOT} (i.e., while loop of \texttt{SIMPLEX}) to solve auxiliary LP
  - If solution to auxiliary LP is 0, then original LP is feasible
    - Rewrite the auxiliary LP, to eliminate $x_0$

Proof of correctness of \texttt{INITIALIZE-SIMPLEX} is based on algebraic argument, correctness of Pivot, etc.
Optimality of SIMPLEX

• **Duality** is a way to prove that a solution is optimal

• Can you think of an example of duality we’ve already seen this semester?
  – Max Flow, Min Cut

• This is an example of duality: given a maximization problem, we define a related minimization problem s.t. the two problems have the same optimal objective value
Duality in LP

• Given an LP, we’ll show how to formulate a dual LP in which the objective is to minimize, and whose optimal value is identical to that of the original LP (now called primal LP)
Primal Dual LPs:

Primal:
maximize \( c^T x \)
subject to: \( Ax \leq b \)
\( x \geq 0 \)
(standard form)

Dual:
minimize \( y^T b \)
subject to: \( y^T A \geq c^T \)
\( y \geq 0 \)
(standard form)
Forming dual

- Change maximization to minimization
- Exchange roles of coefficients on RHSs and the objective function
- Replace each \( \leq \) with \( \geq \)

- Each of the \( m \) constraints in primal has associated variable \( y_i \) in the dual
- Each of the \( n \) constraints in the dual as associated variable \( x_i \) in the primal
Example: Primal-Dual

**PRIMAL:**

\[
\begin{align*}
\text{max} & \quad 16x_1 - 23x_2 + 43x_3 + 82x_4 \\
\text{subject to:} & \\
3x_1 + 6x_2 - 9x_3 + 4x_4 & \leq 239 \\
-9x_1 + 8x_2 + 17x_3 - 14x_4 & = 582 \\
5x_1 + 12x_2 + 21x_3 + 26x_4 & \geq -364 \\
\end{align*}
\]

\[x_1 \geq 0, \quad x_2 \leq 0, \quad x_4 \geq 0\]

**DUAL:**

\[
\begin{align*}
\text{min} & \quad 239y_1 + 582y_2 - 364y_3 \\
\end{align*}
\]

\[
\begin{align*}
\text{subject to:} & \\
3y_1 - 9y_2 + 5y_3 & \geq 16 \\
6y_1 + 8y_2 + 12y_3 & \leq -23 \\
-9y_1 + 17y_2 + 21y_3 & = 43 \\
4y_1 - 14y_2 + 26y_3 & \geq 82 \\
\end{align*}
\]

\[y_1 \geq 0, \quad y_3 \leq 0\]
Next time...

• We’ll look at how to use dual to prove optimality
Reading Assignments

• Reading assignment for Thursday’s class:
  – Chapter 29.4