Agenda

• Primal Network Simplex Method

• Dual Network Simplex Method

• Two-Phase Network Simplex Method

• One-Phase Primal-Dual Network Simplex Method

• Planar Graphs

• Integrality Theorem
Primal Network Simplex Method

Used when all primal flows are nonnegative (i.e., primal feasible).

Pivot Rules:

*Entering arc:* Pick a nontree arc having a negative (i.e., infeasible) dual slack.

*Leaving arc:* Add entering arc to make a cycle. Leaving arc is an arc on the cycle, pointing in the *opposite* direction to the entering arc, and of all such arcs, it is the one with the *smallest* primal flow.
Primal Method—Second Pivot

Enter arc: (c,b)
Leaving arc: (e,b)

Explanation of leaving arc rule:

- Increase flow on (c,b).
- Each unit increase produces a unit increase on arcs pointing in the same direction.
- Each unit increase produces a unit decrease on arcs pointing in the opposite direction.
- The first to reach zero will be the one pointing in the opposite direction and having the smallest flow among all such arcs.
Primal Method—Third Pivot

Entering arc: (a,d)
Leaving arc: (a,f)
Primal Method—Fourth Pivot

Entering arc: (c,a)
Leaving arc: (e,d)

Optimal!
Dual Network Simplex Method

Used when all dual slacks are nonnegative (i.e., dual feasible).

Pivot Rules:

**Leaving arc:** Pick a tree arc having a negative (i.e. infeasible) primal flow.

**Entering arc:** Remove leaving arc to split the spanning tree into two subtrees. Entering arc is an arc reconnecting the spanning tree with an arc in the **opposite** direction, and, of all such arcs, is the one with the **smallest** dual slack.
Dual Network Simplex Method—Second Pivot

Leaving arc: (a,g)
Entering arc: (g,f)

Optimal!
Recall initial tree solution:

Leaving arc: (d,c)
Entering arc: (a,d)

- Remove leaving arc. Need to find a reconnecting arc.
- Since the leaving arc has a negative flow, there is a net supply at the subtree attached to the head node and a net demand at the subtree attached to the tail node.
- So, reconnecting with an arc that spans in the same direction does not improve anything.
- Hence, only consider arcs spanning the two subtrees in the opposite direction.

- Consider a potential arc reconnecting in the opposite direction, say (a,f).
  - Its dual slack will drop to zero.
  - All other reconnecting arcs pointing in the same direction will drop by the same amount.
  - To maintain nonnegativity of all the others, must pick the one that drops the least.
Recall initial tree solution:

- Leaving arc: (d,c)
- Entering arc: (a,d)

- Remove leaving arc. Need to find a reconnecting arc.
- Since the leaving arc has a negative flow, there is a net supply at the subtree attached to the head node and a net demand at the subtree attached to the tail node.
- So, reconnecting with an arc that spans in the same direction does not improve anything.
- Hence, only consider arcs spanning the two subtrees in the opposite direction.

- Consider a potential arc reconnecting in the opposite direction, say (a,f).
  - Its dual slack will drop to zero.
  - All other reconnecting arcs pointing in the same direction will drop by the same amount.
  - To maintain nonnegativity of all the others, must pick the one that drops the least.
Example.

- Turn off display of dual slacks.
- Turn on display of artificial dual slacks.
Two-Phase Method—First Pivot

Use dual network simplex method.
Leaving arc: (b,a)  Entering arc: (a,d)

Primal Feasible!
Two-Phase Method–Phase II

- Turn off display of artificial dual slacks.
- Turn on display of dual slacks.
Two-Phase Method—Second Pivot

Entering arc: (h,c)
Leaving arc: (a,c)
Two-Phase Method—Third Pivot

Entering arc: (e, d)
Leaving arc: (d, a)

Optimal!
Click here (or on any displayed network) to try out the online network simplex pivot tool.
One-Phase Primal-Dual Method

- Artificial flows and slacks are multiplied by a parameter $\mu$.
- In the Figure, $4, 1$ represents $4 + 1\mu$.
- **Question:** For which $\mu$ values is dictionary optimal?
- **Answer:**

\[
\begin{align*}
7 + \mu &\geq 0 \quad (a, c) \\
13 + \mu &\geq 0 \quad (a, d) \\
16 + \mu &\geq 0 \quad (a, h) \\
-8 + \mu &\geq 0 \quad (b, a) \\
-10 + \mu &\geq 0 \quad (b, c) \\
8 + \mu &\geq 0 \quad (b, g) \\
-8 + \mu &\geq 0 \quad (d, b) \\
\end{align*}
\]

\[
\begin{align*}
\mu &\geq 0 \quad (d, f) \\
15 + \mu &\geq 0 \quad (e, a) \\
4 + \mu &\geq 0 \quad (e, d) \\
5 + \mu &\geq 0 \quad (e, h) \\
12 + \mu &\geq 0 \quad (f, g) \\
2 + \mu &\geq 0 \quad (g, d) \\
-13 + \mu &\geq 0 \quad (h, c) \\
\end{align*}
\]

- That is, $13 \leq \mu < \infty$.
- Lower bound on $\mu$ is generated by arc (h,c).
- Therefore, (h,c) enters.
- Arc (a,c) leaves.
Second Iteration

- Range of $\mu$ values: $8 \leq \mu \leq 13$.
- Leaving arc: (d,b)
- Entering arc: (e,d)

New tree is OPTIMAL!
Click here (or on any displayed network) to try out the online network simplex pivot tool.
**Definition.** *Network is called planar if can be drawn on a plane without intersecting arcs.*

**Theorem.** *Every planar network has a geometric dual—dual nodes are faces of primal network.*

**Note:**

- Primal spanning tree shown in red.

**Theorem.** *A dual pivot on the primal network is exactly a primal pivot on the dual network.*
**Definition.** Network is called **planar** if can be drawn on a plane without intersecting arcs.

**Theorem.** Every planar network has a geometric dual—dual nodes are faces of primal network.

Notes:
- Dual node \( A \) is “node at infinity”.
- Primal spanning tree shown in red.
- Dual spanning tree shown in blue (don’t forget node \( A \)).

**Theorem.** A dual pivot on the primal network is exactly a primal pivot on the dual network.
Planar Networks (NOTE: Not Covered)

Definition. *Network is called planar if can be drawn on a plane without intersecting arcs.*

Theorem. *Every planar network has a geometric dual—dual nodes are faces of primal network.*

Notes:
- Dual node $A$ is “node at infinity”.
- Primal spanning tree shown in red.
- Dual spanning tree shown in blue (don’t forget node $A$).

Theorem. *A dual pivot on the primal network is exactly a primal pivot on the dual network.*
Theorem. Assuming integer supplies, every basic (i.e. tree) solution assigns integer flow to every arc.

Corollary. Assuming integer supplies, every basic optimal solution assigns integer flow to every arc.