CS 524: Introduction to Optimization
Lecture 14 : Critical Paths

Michael Ferris

Computer Sciences Department
University of Wisconsin-Madison

October 6, 2023

Widgetco

We're making widgets

Widgetco is about to introduce a new product. One unit of this product is produced by assembling subassembly 1 and subassembly 2. Before production begins on either subassembly, raw materials must be purchased and workers must be trained. Before the subassemblies can be assembled into the final product, the finished subassembly 2 must be inspected.

	Activity	**Duration**
A	Train Workers	6
B	Purchase Raw Materials	9
C	Make Subassembly 1	8
D	Make Subassembly 2	7
E	Inspect Subassembly 2	10
F	Assemble Subassemblies	12

Your Challenge

Schedule activities to complete assembly in minimum time

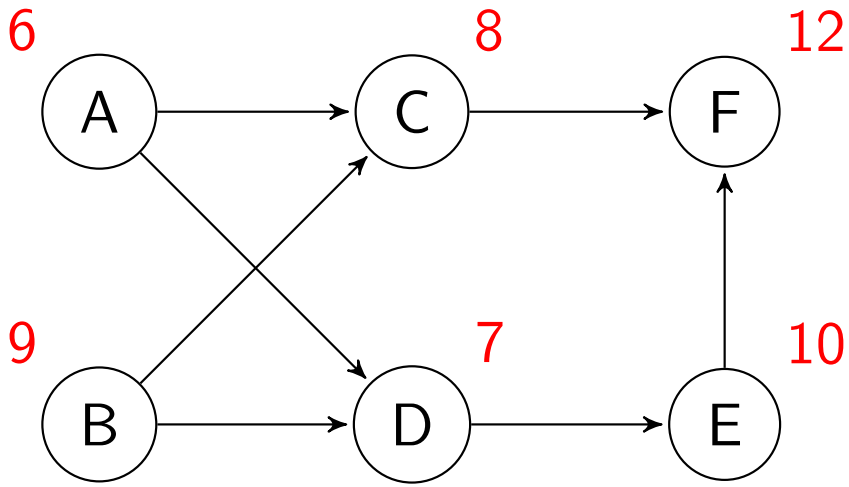
Wash and Go With

- Project Scheduling: PERT (Project Evaluation and Review Technique)
- Often used synonomously with CPM: Critical Path Method

PERT

- I: Set of projects
- $P \subset I \times I$: Precedence relationships
- a_i: Duration of activity $i \in I$

Precedence relationship (P) for Widgetco



Modeling PERT

Variables

- t_i: Time activity starts

Constraints

- j cannot begin before i finishes:

$$t_j \geq t_i + a_i \qquad \forall (i,j) \in P$$

Objective

- Minimize the latest job completion time (makespan).

$$\min \max\{t_1 + a_1, t_2 + a_2 \ldots, t_{|I|} + a_{|I|}\}.$$

Mini-Max

- Minimax will haunt you

$$
\begin{aligned}
T^* = \quad & \min_{z,t} \quad z \\
& \text{s.t.} \quad && z \geq t_i + a_i, && \forall i \in I \\
& && t_j \geq t_i + a_i, && \forall (i,j) \in P \\
& && t_i \geq 0, && \forall i \in I
\end{aligned}
$$

How can we find critical projects, or the critical path?

Critical Path

- An activity is critical if by increasing its duration, the time to complete the project increases.
- How can we find critical projects, or the critical path?
- What do the dual variables in the previous LP mean? Spoiler: positive dual variables λ_{ij} identify the critical path

---

$$
\begin{aligned}
\max \quad & \sum_{(i,j)\in P} a_i \lambda_{ij} + \sum_{i \in I} a_i \pi_i \\
\text{s.t.} \quad & \sum_{i \in I} \pi_i = 1 \\
& -\pi_i - \sum_{j \in I:(i,j)\in P} \lambda_{ij} + \sum_{j \in I:(j,i)\in P} \lambda_{ji} \leq 0, \ \forall i \in I \\
& \pi_i, \lambda_{ij} \geq 0
\end{aligned}
$$

Final Exercise

1. Fix $z = T^*$
2. Solve max $\sum_{i \in I} t_i$: Gives latest possible start times
3. Solve min $\sum_{i \in I} t_i$: Gives earliest possible start times
4. Bottleneck Projects: Which projects have early time = late time?
5. Note relationship to marginal values

Note (widgetco2.gms) that when there are multiple critical paths, then the marginal values only identify *one* critical path, not all of them. However the bottleneck code does find all.