Review

Dynamic Programming

Formulating dynamic programs – two ways

- A **dynamic program** models situations where decisions are made in a sequential process in order to optimize some objective
- **Stages** $t = 1, 2, \dots, T$
  - stage $T \leftrightarrow$ end of decision process
- **States** $n = 0, 1, \dots, N \leftarrow$ possible conditions of the system at each stage
- Two representations: **shortest/longest path** and **recursive**

Shortest/longest path		Recursive
node t_n	$\leftrightarrow$	state n at stage t
edge $(t_n, (t+1)_m)$	$\leftrightarrow$	allowable decision x_t in state n at stage t that results in being in state m at stage $t+1$
length of edge $(t_n, (t+1)_m)$	$\leftrightarrow$	cost/reward of decision x_t in state n at stage t that results in being in state m at stage $t+1$
length of shortest/longest path from node t_n to end node	$\leftrightarrow$	cost/reward-to-go function $f_t(n)$
length of edges (T_n, end)	$\leftrightarrow$	boundary conditions $f_T(n)$
shortest or longest path	$\leftrightarrow$	recursion is min or max: $f_t(n) = \min\limits_{x_t \text{ allowable}} \text{or} \max \left\{ \begin{pmatrix} \text{cost/reward of} \\ \text{decision } x_t \end{pmatrix} + f_{t+1}\begin{pmatrix} \text{new state} \\ \text{resulting} \\ \text{from } x_t \end{pmatrix} \right\}$
source node 1_n	$\leftrightarrow$	desired cost-to-go function value $f_1(n)$

- Note that the length of edge (T_n, end) is often 0, but not always!

Example 1. Simplexville Oil needs to build capacity to refine 1,000 barrels of oil and 2,000 barrels of gasoline per day. Simplexville can build a refinery at 2 locations. The cost of building a refinery is as follows:

Oil capacity per day	Gas capacity per day	Building cost ($ millions)
0	0	0 ← length of red edges
1000	0	5 ← length of blue edges
0	1000	7 ← length of orange edges
1000	1000	14 ← length of green edges

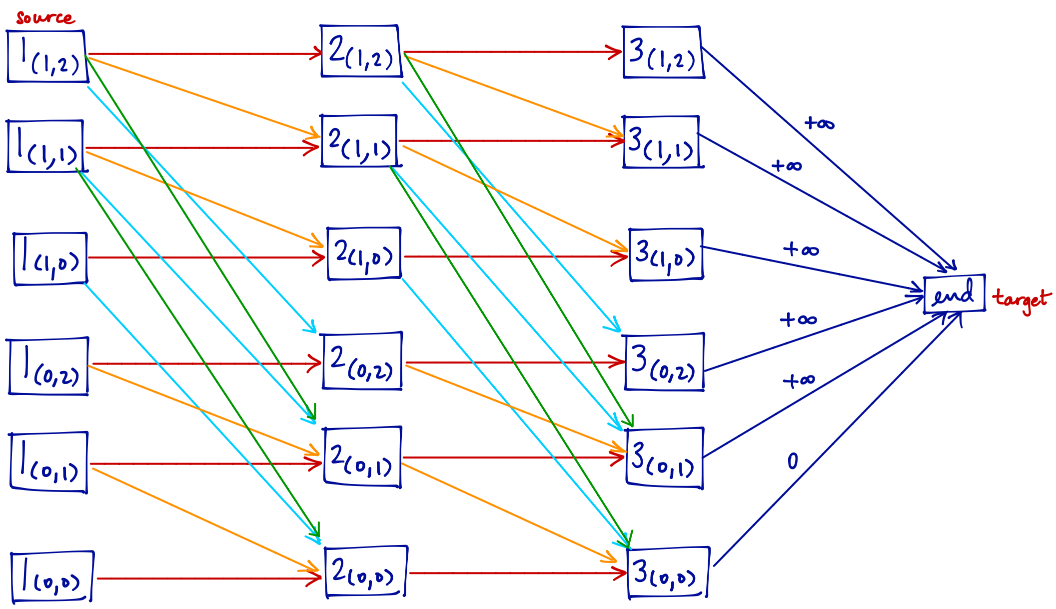
The problem is to determine how much capacity should be built at each location in order to minimize the total building cost. To make things a little simpler, assume that the capacity requirements must be met exactly.

a. Formulate this problem as a dynamic program by giving its shortest path representation.
b. Formulate this problem as a dynamic program by giving its recursive representation. Solve the dynamic program.

a. Stages: $t = 1, 2 \longleftrightarrow$ building at location t $t = 3 \longleftrightarrow$ end of process

States: $(n_1, n_2) \longleftrightarrow n_1$ oil capacity and n_2 gas capacity still needed to be built (in 1000s) for $n_1 \in \{0, 1\}$ and $n_2 \in \{0, 1, 2\}$

Find the shortest path:



b. Stages: $t = 1, 2 \longleftrightarrow$ building at location t $\qquad t = 3 \longleftrightarrow$ end of process

States: $(n_1, n_2) \longleftrightarrow n_1$ oil capacity and n_2 gas capacity still needed to be built (in 1000s) for $n_1 \in \{0,1\}$ and $n_2 \in \{0,1,2\}$

Allowable decisions x_t at stage t and state (n_1, n_2):

$t = 1, 2$: $x_t = (x_{t1}, x_{t2}) \longleftrightarrow$ build x_{t1} oil capacity and x_{t2} gas capacity at location t

x_t must satisfy: $(x_{t1}, x_{t2}) \in \{(0,0), (1,0), (0,1), (1,1)\}$

$$x_{t1} \leq n_1$$

$$x_{t2} \leq n_2$$

Cost of x_t at stage t and state (n_1, n_2): $$C(x_t) = \begin{cases} 0 & \text{if} \quad x_t = (0,0) \\ 5 & \text{if} \quad x_t = (1,0) \\ 7 & \text{if} \quad x_t = (0,1) \\ 14 & \text{if} \quad x_t = (1,1) \end{cases}$$

Cost-to-go function: $f_t(n_1, n_2) =$ minimum cost to build n_1 oil capacity and n_2 gas capacity with locations $t, t+1, \ldots$ $\quad$ for $t = 1, 2, 3$; $n_1 = 0, 1$; $n_2 = 0, 1, 2$

Boundary conditions: $$f_3(n_1, n_2) = \begin{cases} 0 & \text{if } (n_1, n_2) = (0,0) \\ +\infty & \text{o/w} \end{cases} \qquad \text{for } n_1 = 0, 1;\ n_2 = 0, 1, 2$$

Recursion: $$f_t(n_1, n_2) = \min_{\substack{(x_{t1}, x_{t2}) \\ \text{allowable}}} \left\{ C(x_{t1}, x_{t2}) + f_{t+1}(n_1 - x_{t1}, n_2 - x_{t2}) \right\} \qquad \text{for } t = 1, 2;\ n_1 = 0, 1;\ n_2 = 0, 1, 2$$

Desired cost-to-go value: $f_1(1, 2)$

Solving backwards:

Stage 3: (boundary conditions)

$$f_3(n_1, n_2) = \begin{cases} 0 & \text{if } (n_1, n_2) = (0,0) \\ +\infty & \text{o/w} \end{cases} \qquad \text{for } n_1 = 0, 1; \ n_2 = 0, 1, 2$$

Stage 2:

$$f_2(1,2) = \min\left\{0 + \overset{+\infty}{f_3(1,2)},\ 5 + \overset{+\infty}{f_3(0,2)},\ 7 + \overset{+\infty}{f_3(1,1)},\ 14 + \overset{+\infty}{f_3(0,1)}\right\} = +\infty$$

$$f_2(1,1) = \min\left\{0 + \overset{+\infty}{f_3(1,1)},\ 5 + \overset{+\infty}{f_3(0,1)},\ 7 + \overset{+\infty}{f_3(1,0)},\ 14 + \overset{0}{f_3(0,0)}\right\} = 14$$

$$f_2(1,0) = \min\left\{0 + \overset{+\infty}{f_3(1,0)},\ 5 + \overset{0}{f_3(0,0)}\right\} = 5$$

$$f_2(0,2) = \min\left\{0 + \overset{+\infty}{f_3(0,2)},\ 7 + \overset{+\infty}{f_3(0,1)}\right\} = +\infty$$

$$f_2(0,1) = \min\left\{0 + \overset{+\infty}{f_3(0,1)},\ \underline{7 + \overset{0}{f_3(0,0)}}\right\} = 7 \qquad x_2 = (0,1)$$

$$f_2(0,0) = \min\left\{0 + \overset{0}{f_3(0,0)}\right\} = 0$$

Stage 1: (desired cost-to-go)

$$f_1(1,2) = \min\left\{0 + \overset{+\infty}{f_2(1,2)},\ 5 + \overset{+\infty}{f_2(0,2)},\ 7 + \overset{14}{f_2(1,1)},\ \underline{14 + \overset{7}{f_2(0,1)}}\right\} = 21 \qquad x_1 = (1,1)$$

Note that $x_1 = (0,1)$ and $x_2 = (1,1)$ is also an optimal solution.