Department of Mathematical Sciences

CARNEGIE MELLON UNIVERSITY

OPERATIONS RESEARCH II 21-393

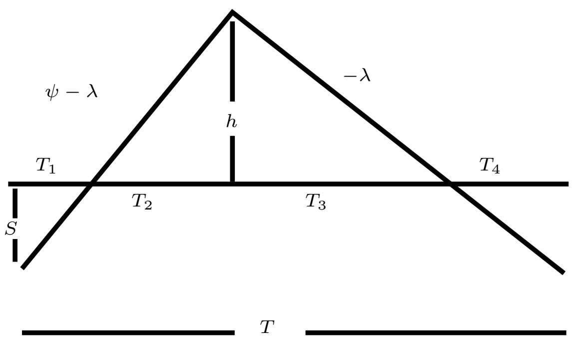
Homework 4: Due Monday October 29.

Q1
Find an optimal inventory policy for the model with the following parameters: It is a generalisation of Models 2 and 3 of notes.

A	Cost of making an order.
λ	Demand per period for items.
ψ	Arrival rate of ordered items.
I	Inventory cost per item per period.
π	Penalty cost per item out of stock per period.

1. First draw a diagram showing the inventory level over time and various parameters.
2. Then identify the various costs per period.
3. Optimize total cost.

Solution:



S and h will be our independent variables. Then

$$T_1 = \frac{S}{\psi - \lambda};\ T_2 = \frac{h}{\psi - \lambda};\ T_3 = \frac{h}{\lambda};\ T_4 = \frac{S}{\lambda}.$$

$$T = T_1 + T_2 + T_3 + T_4 = \frac{(S+h)\psi}{\lambda(\psi - \lambda)}.$$

Let K denote total cost. Then

$$\begin{aligned} K &= \frac{A}{T} + \frac{hI}{2} \cdot \frac{T_2 + T_3}{T} + \frac{\pi S}{2} \cdot \frac{T_1 + T_4}{T} \\ &= \frac{1}{S+h}\left(\frac{A\lambda(\psi - \lambda)}{\psi} + \frac{1}{2} I h^2 + \frac{1}{2}\pi S^2\right) \end{aligned}$$

Putting $\frac{\partial K}{\partial S} = \frac{\partial K}{\partial h} = 0$ we get

$$S^2 = \frac{2AI\lambda(\psi - \lambda)}{\pi\psi(I + \pi)} \ \textit{and}\ h^2 = \frac{2A\pi\lambda(\psi - \lambda)}{I\psi(I + \pi)}$$

Q2

Describe a modification of Dijkstra's algorithm that can be used to solve the following problem: Prove that your algorithm finds optimum paths.
(The standard proof for Dijkstra will work, with only very minor changes.)

You are given a digraph D with a distinguished vertex s. If $P = (s = x_0, x_1, \ldots, x_k)$ is a path and $P_i = (x_0, x_1, \ldots, x_i)$ for $i = 0, 1, \ldots, k$ then we define its *weight* $w(P)$ by

$$w(P_0) = 0 \ \textit{and}\ w(P_i) = w(P_{i-1}) + \phi_{e_i}(w(P_{i-1}))$$

where for every edge e, ϕ_e is a non-negative monotone increasing function. (What this is saying is that the time to cross edge $e = (x, y)$ is a function of the time of arrival at x and increases with this time.)

Solution: Run the usual Dijksra algorithm: We have a set S, initially $S = \{s\}$ and for each $v \in S$, $d(v)$ is the weight of the shortest path from s to v. For $v \notin S$, $d(v)$ is the minimum weight of a path that goes s, P, w, v where $w \in S$ and where s, P, w is a minimum weight path from s to w. Initially, $d(v) = \phi_{sv}(0)$. The update rule is to add u to S where $d(u) = \min_{v \notin S} d(v)$ and then to update $d(v), v \notin S$ by

$$d(v) := \min\{d(v), d(u) + \phi_{uv}(d(u))\}.$$

All we have to show is that $d(u)$ is the minimum weight of a path from s to u. Let P be any other path from s to u and let x be the last vertex of S on P. Let P_x be the sub-path of P from S to x. Let y be the vertex that follows x on P. Then, by induction on $|S|$,

$$\begin{aligned}
w(P) &\geq w(P_x, y) \\
&= w(P_x, x) + \phi_{xy}(w(P_x)) \\
&\geq d(x) + \phi_{xy}(w(P_x)) \\
&\geq d(x) + \phi_{xy}(d(x)) \\
&\geq d(y) \\
&\geq d(u).
\end{aligned}$$

Q3

How would you solve the following *bottleneck* assignment problem: You are given an $m \times m$ matrix of costs A and you are asked to find an assignment a which minimises

$$\max_{i=1,\ldots,m} A(i, a(i)).$$

Solution: We re-phrase the problem as follows: Suppose that the ordered pairs $(i, j) \in [m]^2$ are ordered by $\preceq$ where $(i, j) \preceq (i', j')$ if $A(i, j) \leq A(i', j')$ (ties can be broken arbitrarily). Let (i_s, j_s) be the sth largest pair in this order. Then we wish to find the smallest s such that there exists assignment a for which $(i, a(i)) \preceq (i_s, j_s)$ for $i = 1, 2, \ldots, m$. So for each s we define

$$B(i,j) = \begin{cases} 1 & (i,j) \preceq (i_s, j_s) \\ 2 & (i,j) \not\preceq (i_s, j_s) \end{cases}.$$

We try $s = 1, 2, \ldots, m^2$ and stop at the first s from which the minimum cost assignment using costs B is m. (We can perhaps speed things up using binary search).