Department of Mathematical Sciences
CARNEGIE MELLON UNIVERSITY

OPERATIONS RESEARCH II 21-393

Homework 2: Due Monday October 9.

Q1 Consider an electronic system consisting of four components each of which must work for the system to function. The reliability of the system can be improved by installing several parallel units in one or more of the components. The following table gives the probability that the respective components will function if they consist of one, two or three parallel units.

Number of units	Component 1	Component 2	Component 3	Component 4
1	0.4	0.6	0.7	0.5
2	0.6	0.7	0.8	0.7
3	0.8	0.8	0.9	0.9

The probability that the system will function is the product of the probabilities that the respective components will function. The cost (in hundred's of dollars) of installing one, two or three parallel units in the respective compoonents is given by the following table.

Number of units	Component 1	Component 2	Component 3	Component 4
1	1	2	1	1
2	2	4	3	3
3	3	5	4	4

Because of budget limitations, a maximum of \$1000 can be spent. use Dynamic Programming to determine how many parallel units should be installed in each of the four components in order to maximise the probability that the system will function.

Solution: Let $p_i(x)$ be the probability that component i functions if there are x parallel units placed in it and let $c_r(x)$ be the cost of putting x parallel units into component i. Let $\phi_r(w)$ be the maximum reliability of a system of components $r, r+1, \ldots, 4$ if one has enough money to put in w parallel components altogether. We wish to compute $\phi_1(10)$. In general

$$\begin{aligned} \phi_4(w) &= p_4(w) \\ \phi_r &= \max_x (p_r(x)\phi_{r+1}(w - c_r(x))) \end{aligned}$$

w	ϕ_1	x_1	ϕ_2	x_2	ϕ_3	x_3	ϕ_4	x_4
0			0		0		0	
1			0		0		0	
2			0		.35	1	.5	1
3			0		.35	1	.7	2
4			0		.49	1	.9	3
5			.210	1	.63	1	.9	3
6			.294	1	.63	1	.9	3
7			.378	1	.72	2	.9	3
8			.378	1	.81	3	.9	3
9			.441	2	.81	3	.9	3
10	.3024	3	.504	3	.81	3	.9	3

Solution: $x_1 = 3, x_2 = 1, x_3 = 1, x_4 = 3$. Maximum = .3024.

Q2 A system can be in 3 states 1,2,3 and the cost of moving from state i to state j in one period is $c(i,j)$, where the $c(i,j)$ are given in the matrix below. The one period discount factor α is 1/2.

The aim is to find a policy which simultaneously minimises the discounted cost of operating from any starting state. Start with the policy

$$\pi(1) = 1, \pi(2) = 1, \pi(3) = 2.$$

Find an optimal policy.

The matrix of costs is

$$\begin{bmatrix} 7 & 2 & 3 \\ 5 & 2 & 7 \\ 1 & 6 & 2 \end{bmatrix}$$

Solution: Start with an initial solution of $\pi(1) = 2, \pi(2) = 2, \pi(3) = 1$ i.e. chose the minimum cost in each row (a reasonable starting solution).

Then we solve

$$\begin{aligned} y_1 &= 2 + \frac{y_2}{2} \\ y_2 &= 2 + \frac{y_2}{2} \\ y_3 &= 1 + \frac{y_1}{2} \end{aligned}$$

Solving these equations gives $y_1 = 4, y_2 = 4, y_3 = 3$.
We now check for optimality:
Check y_1:

$$\begin{aligned} 7 &+ \frac{y_1}{2} = 9 \\ 2 &+ \frac{y_2}{2} = 4 \quad \star \\ 3 &+ \frac{y_3}{2} = \tfrac{13}{2} \end{aligned}$$

Check y_2:

$$\begin{aligned} 5 &+ \frac{y_1}{2} = 7 \\ 2 &+ \frac{y_2}{2} = 4 \quad \star \\ 7 &+ \frac{y_3}{2} = \tfrac{17}{2} \end{aligned}$$

Check y_3:

$$\begin{aligned} 1 &+ \frac{y_1}{2} = 3 \quad \star \\ 6 &+ \frac{y_2}{2} = 8 \\ 2 &+ \frac{y_3}{2} = \tfrac{7}{2} \end{aligned}$$

So our initial guess is optimal.
Q3 A mathematics student has been asked to compute the product $A_1 \times A_2 \times \cdots \times A_n$ where each A_i is an $m_i \times m_{i+1}$ matrix. It will cost the student \$$pqr$ to multiply a $p \times q$ matrix by a $q \times r$ matrix. Describe a dynamic

programming algorithm for finding the cheapest way of multiplying the n matrices together.
To see that there is a problem: let $\underline{m} = (10, 5, 8, 12)$. There are two ways to compute $A_1A_2A_3$: compute A_1A_2 first and then multiply this by A_3 – total cost 400+960=1360. The other way is to compute A_2A_3 first and then multiply this by A_1 – total cost 480+600=1080.

Solution: Let $M_{i,j}$ denote the minimum number of multiplications needed to compute $A_i \times A_{i+1} \times \cdots \times A_j$ for $1 \leq i < j \leq n$. Then, by considering the last matrix multiplication needed,

$$M_{i,j} = \min_{i \leq k < j} \{M_{i,k} + M_{k+1,j} + m_i m_{k+1} m_{j+1}\}.$$

We can use this recurrence to compute $M_{i,n}$, computing the terms $M_{i,j}$ in order of the sum $i + j$.