

KTH Mathematics

Suggested solutions for the exam in SF2863 Systems Engineering. January 13, 2014 14.00–19.00

Examiner: Per Enqvist, phone: 790 62 98

1. (a) Model 0: With $\rho = \lambda/\mu = 1/2$ we get

$$W = \frac{L}{\lambda} = \frac{1}{\lambda} \frac{\rho}{1-\rho} = 1/10$$

Now choose μ_1, μ_2, μ_3 so that the time to pass through the systems are 1/10 hours, that is 6 minutes.

Model 1: With infinitely many servers the time to pass through the system is only the service time, which has a mean value of $1/\mu_1$ hours, *i.e.*, let $\mu_1 = 10$. Model 2: With the feedback in the system the internal arrival rate to the queue is 20 customers per hour. The mean time to pass through the queue once is $V = 1/(\mu_2 - 20)$. The mean time to pass through the system is W = V + 1/2W, so W = 2V, *i.e.*, $10 = 2/(\mu_2 - 20)$, and $\mu_2 = 40$.

Model 3: With the feedback in the system the internal arrival rate to the queue is 20 customers per hour. The mean time to pass through the queue once is $V = 1/\mu_3$. The mean time to pass through the system is W = V + 1/2W, so W = 2V, *i.e.*, $10 = 2/\mu_3$, and $\mu_2 = 20$.

(b) Model 0: $W_q = W - 1/\mu_0 = 1/10 - 1/20 = 1/20$ and mean number of persons in the queue is $L_q = W_q/\lambda = 1/2$

If the arrival rate doubles it is the same as the service rate and the mean queue length will tend to infinity.

Model 1: Since there infinitely many servers, nobody has to stand in the queue. If the arrival rate doubles the average queue length is still zero.

Model 2: The arrival rate to the queue is 20 and service intensity is 40 so $\rho = 1/2$ and then $L_q = 1/2$ as for Model 0.

If the arrival rate doubles the average queue length will go to infinity.

Model 3: Since there infinitely many servers, nobody has to stand in the queue. If the arrival rate doubles the average queue length is still zero.

2. (a) This can be solved using the economic order quantity model Let K = 100, d = 2, h = 1 and c = 10.

Optimal order quantity D is $D = \sqrt{\frac{2dK}{h}} = \sqrt{2 \cdot 2 \cdot 100/1} = 20$. The time between orders is Q/d = 20/2 = 10 days. (b) This a deterministic periodic review model.

$$C_i = \min_j \{C_i^{(j)} | i \le j \le N\}$$

where

$$C_i^{(j)} = C_{j+1} + K + h(r_{i+1} + 2r_{i+2} + \dots + (j-i)r_j).$$

When

$$C_{5}^{(5)} = 100$$
then

$$C_{5} = \min_{j=5} \{C_{5}^{(5)}\} = 100$$
When

$$C_{4}^{(4)} = 100 + 100 = 200$$

$$C_{4}^{(5)} = 100 + 7(10) + 0 = 170$$
then

$$C_{4} = \min_{j=4,\dots,5} \{C_{4}^{(4)}, C_{4}^{(5)}\} = 170$$
When

$$C_{3}^{(3)} = 100 + 170 = 270$$

$$C_{3}^{(4)} = 100 + 7(20) + 100 = 340$$
then

$$C_{3} = \min_{j=3,\dots,5} \{C_{2}^{(2)}, C_{2}^{(3)}, C_{2}^{(4)}, C_{2}^{(5)}\} = 270$$
When

$$C_{2}^{(2)} = 0 + 270 = 270$$

$$C_{2}^{(3)} = 100 + 7(30) + 170 = 480$$

$$C_{2}^{(4)} = 100 + 7(30 + 2 \cdot 20) + 100 = 690$$
then

$$C_{2}^{(5)} = 100 + 7(30 + 2 \cdot 20 + 3 \cdot 10) + 0 = 800$$
then

$$C_{2} = \min_{j=2,\dots,5} \{C_{2}^{(2)}, C_{2}^{(3)}, C_{2}^{(4)}, C_{2}^{(5)}\} = 270$$
When

$$C_{1}^{(1)} = 100 + 270 = 370$$

$$C_{1}^{(2)} = 100 + 7(0) + 270 = 370$$

$$C_{1}^{(3)} = 100 + 7(0 + 2 \cdot 30) + 270 = 790$$

$$C_{1}^{(4)} = 100 + 7(0 + 2 \cdot 30 + 3 \cdot 20) + 100 = 1040$$

$$C_{1}^{(5)} = 100 + 7(0 + 2 \cdot 30 + 3 \cdot 20 + 4 \cdot 10) + 0 = 1220$$
then

$$C_{1} = \min_{j=1,\dots,5} \{C_{1}^{(1)}, C_{1}^{(2)}, C_{1}^{(3)}, C_{1}^{(4)}, C_{1}^{(5)}\} = 370$$
So the optimal strategy is: First week buy 10, third week buy 30, fourth week
buy 30. Total cost is then 370 SEK.

(c) Let $V_n(x)$ =optimal expected cost when at stage n with x liters of julmust in storage.

The dynamic programming recursion is given by

$$V_n(x) = \min_{u_n, u_n \ge 100 - x} \left\{ 10u_n + 100u_n + \mathbf{E}[h(x + u_n - D) + V_{n+1}(x + u - D)] \right\}$$

where u_n is 1 if $u_n > 0$ and 0 if $u_n = 0$ and the expectation is taken over the stochastic variable D. The constraint $u_n \ge 100 - x$ ensures that no shortage occurs.

We have that

$$V_3(x) = \begin{cases} 0 & \text{if } x \ge 100 & \text{and then } \hat{u} = 0\\ 10(100 - x) + 100 & \text{if } x < 100 & \text{and then } \hat{u} = 100 - x \end{cases}$$

For the particular distribution we get an upper limit of 30 instaed of 100.

$$V_3(x) = \begin{cases} 0 & \text{if } x \ge 30 \\ 10(30-x) + 100 = 400 & \text{if } x = 0 \end{cases} \text{ and then } \hat{u} = 30 - x$$

It is also clear that the inventory x will always be a multiple of 30. Consider the case x = 0, u = 30 at stage n = 2 $30 \cdot 10 + 100 + 0.6[(30 - 30)7 + V_3(0)] + 0.4[(30 - 0)7 + 0.4V_3(30)] = 724$ Consider the case x = 0, u = 60 at stage n = 2 $60 \cdot 10 + 100 + 0.6[(60 - 30)7 + V_3(30)] + 0.4[(60 - 0)7 + 0.4V_3(60)] = 994$ Then $V_2(0) = 724$. Consider the case x = 30, u = 0 at stage n = 2 $0.6[(30 - 30)7 + V_3(0)] + 0.4[(30 - 0)7 + 0.4V_3(30)] = 324$ Consider the case x = 30, u = 30 at stage n = 2 $30 \cdot 10 + 100 + 0.6[(60 - 30)7 + V_3(30)] + 0.4[(60 - 0)7 + 0.4V_3(60)] = 694$ Then $V_2(30) = 324$. Consider the case x = 0, u = 30 at stage n = 1 $30 \cdot 10 + 100 + 0.6[(30 - 30)7 + V_2(0)] + 0.4[(30 - 0)7 + 0.4V_2(30)] = 970.24$ Consider the case x = 0, u = 60 at stage n = 2 $60 \cdot 10 + 100 + 0.6[(60 - 30)7 + V_2(30)] + 0.4[(60 - 0)7 + 0.4V_2(60)] > 1000$ Then $V_1(0) = 970.24$. First week buy 30 liters, if it is consumed buy 30 liters more, otherwise buy nothing.

3. (a) Let d_i denote the number of days Frasse visits companions i, i = 1, 2, 3, 4.. Define functions f and g that we want to minimize, with the right properties. Maximizing pleasure is the same as minimizing minus the pleasure. Let $f(d_1, d_2, d_3, d_4) = -\left[(-d_1^2 + 20d_1) + (-d_2^2 + 19d_2) + (-d_3^2 + 18d_3) + (-d_4^2 + 17d_4)\right]$ and $g(d_1, d_2, d_3, d_4) = -\left[7 - \sum_{i=1}^N d_i\right]$. Clearly g is a separable function, increasing and integer-convex.

The continuous version of function f has a gradient $\nabla f = (2d_1 - 20, 2d_2 - 19, 2d_3 - 18, 2d_4 - 17)$ which has negative elements for d_i less than 8, which makes the function decreasing. Furthermore, f is separable, $f = f_1(d_1) + f_2(d_2) + f_3(d_3) + f_4(d_4)$, and the functions f_i are quadratic functions which are convex, since the Hessian is 2 times the identity matrix.

$-\Delta f$	$-\Delta f_1$	$-\Delta f_2$	$-\Delta f_3$	$-\Delta f_4$
$d_i = 0$	19	18	17	16
$d_i = 1$	17	16	15	14
$d_i = 2$	15	14	13	12
$d_i = 3$	13	12	11	10

Since the values are decreasing in each column the function f is integer-convex for the tabulated values.

Note that $g_i = 1$ for all i.

We can then apply the Marginal Allocation algorithm.

Note that the quotients $-\Delta f_i(d)/\Delta g_i(d) = \Delta f_i(d_i)$ are given in the table above. The efficient allocations are therefore,

$$\begin{split} S^{(0)} &= (s_1 = 0, s_2 = 0, s_3 = 0, s_4 = 0), \, \mathrm{f}(\mathrm{s}^{(0)}) = 0, g(s^{(0)}) = -7 \\ \mathrm{S}^{(1)} &= (s_1 = 1, s_2 = 0, s_3 = 0, s_4 = 0), \, \mathrm{f}(\mathrm{s}^{(1)}) = -19, g(s^{(1)}) = -6 \\ \mathrm{S}^{(2)} &= (s_1 = 1, s_2 = 1, s_3 = 0, s_4 = 0), \, \mathrm{f}(\mathrm{s}^{(2)}) = -37, g(s^{(2)}) = -5 \\ \mathrm{S}^{(3)} &= (s_1 = 1, s_2 = 1, s_3 = 1, s_4 = 0), \, \mathrm{f}(\mathrm{s}^{(3)}) = -54, g(s^{(3)}) = -4 \\ \mathrm{S}^{(4)} &= (s_1 = 2, s_2 = 1, s_3 = 1, s_4 = 0), \, \mathrm{f}(\mathrm{s}^{(4)}) = -71, g(s^{(4)}) = -3 \\ \mathrm{S}^{(5)} &= (s_1 = 2, s_2 = 1, s_3 = 1, s_4 = 1), \, \mathrm{f}(\mathrm{s}^{(5)}) = -87, g(s^{(5)}) = -2 \\ \mathrm{S}^{(6)} &= (s_1 = 2, s_2 = 2, s_3 = 1, s_4 = 1), \, \mathrm{f}(\mathrm{s}^{(6)}) = -103, g(s^{(6)}) = -1 \\ \mathrm{S}^{(7)} &= (s_1 = 2, s_2 = 2, s_3 = 2, s_4 = 1), \, \mathrm{f}(\mathrm{s}^{(7)}) = -118, g(s^{(7)}) = 0 \end{split}$$

For $s^{(3)}$ there is a an alternative solution with $s_1 = 2$ and $s_3 = 0$. For $s^{(5)}$ there is a an alternative solution with $s_2 = 2$ and $s_4 = 0$. For $s^{(7)}$ there is a an alternative solution with $s_3 = 1$ and $s_1 = 3$.

4. We need to keep track of if the resolution was kept the year before, define the state

$$s_k = \begin{cases} 0 & \text{if the resolution was not kept the year before year } k \\ 1 & \text{if the resolution was kept the year before year } k \end{cases}$$

Define the decisions

$$x_k = \begin{cases} 1 & \text{if Frasse decides to make an ambitious resolution year } k \\ 2 & \text{if Frasse decides to make an easy resolution year } k \end{cases}$$

The transition probabilities are

 $p_{ij}(x)$ = the probability of jumping from state *i* to *j* if we make decision *x*.

$$P(x = 1) = \begin{bmatrix} 0.8 & 0.2\\ 0.5 & 0.5 \end{bmatrix}$$
$$P(x = 2) = \begin{bmatrix} 0.4 & 0.6\\ 0.2 & 0.8 \end{bmatrix}$$

Let the costs be the happiness and maximize instead of minimize.

Let $q_{ij}(x) =$ expected cost incurred when the state is in state *i* decision *x* is made and the system evolves to state *j*.

$$Q(x=1) = \begin{bmatrix} 0 & 10\\ 0 & 10 \end{bmatrix}$$
$$Q(x=2) = \begin{bmatrix} 0 & 5\\ 0 & 5 \end{bmatrix}$$

Then the expected "cost" of making decision x_k at state s_k is $C_{s_k,x_k} = \sum_{j=0}^{3} q_{s_k,j} p_{s_k,j}(x_k)$.

$$C_{01} = q_{00}(1)p_{00}(0) + q_{01}(1)p_{01}(0) = 2$$

$$C_{02} = q_{00}(2)p_{00}(2) + q_{01}(2)p_{01}(2) = 3$$

$$C_{11} = q_{10}(1)p_{10}(1) + q_{11}(1)p_{11}(1) = 5$$

$$C_{12} = q_{10}(2)p_{10}(2) + q_{11}(2)p_{11}(2) = 4$$

- (a) Starting policy:
 - If $s_k = 0$, make decision $x_k = 2$.
 - If $s_k = 1$, make decision $x_k = 2$.

Use the policy iteration algorithm. Let $v_1 = 0$, then the value determination equations

$$g + v_0 = 3 + 0.4v_0 + 0.6v_1$$

$$g + v_1 = 4 + 0.2v_0 + 0.8v_1$$

gives g = 15/4, $v_0 = -5/4$ and $v_1 = 0$.

To find out if it is optimal we do one step of the policy iteration. For i = 0

$$\max_{k=1,2} \{ C_{0k} + (p_{00}(k)v_0 + p_{01}(k)v_1 + p_{02}(k)v_2) \} =$$

$$= \max\{C_{01} + (p_{00}(1)v_0 + p_{01}(1)v_1), C_{02} + (p_{00}(2)v_0 + p_{01}(2)v_1)\}$$
$$= \max\{\underbrace{2 + 0.6 * (-5/4)}_{5/4}, \underbrace{3 + 0.4 * (-5/4)}_{5/2}\} = 5/2 \text{ for } k = 2.$$

For i = 1

$$\min_{k=1,2} \{ C_{1k} + (p_{10}(k)v_0 + p_{11}(k)v_1) \} =$$

$$= \max\{C_{11} + (p_{10}(1)v_0 + p_{11}(1)v_1), C_{12} + (p_{10}(2)v_0 + p_{11}(2)v_1)\}$$
$$= \max\{\underbrace{5 + 0.5 * (-5/4)}_{35/8}, \underbrace{4 + 0.8 * (-5/4)}_{3}, \} = 35/8 \text{ for } k = 1.$$

The starting policy is not optimal, it is better to use the updated policy Updated policy:

If $s_k = 0$, make decision $x_k = 2$. If $s_k = 1$, make decision $x_k = 1$.

- (b) Starting policy:
 - If $s_k = 0$, make decision $x_k = 2$.

If $s_k = 1$, make decision $x_k = 2$.

Use the policy iteration algorithm. The value determination equations

$$V_0 = 3 + \alpha (0.4V_0 + 0.6V_1)$$
$$V_1 = 4 + \alpha (0.2V_0 + 0.8V_1)$$

where $\alpha = 0.8$ gives $V_0 = 5.7$ and $V_1 = 1.87$.

To find out if it is optimal we do one step of the policy iteration. For i = 0

$$\max_{k=1,2} \{ C_{0k} + \alpha (p_{00}(k)V_0 + p_{01}(k)V_1) \} =$$

$$= \max\{C_{01} + \alpha(p_{00}(1)V_0 + p_{01}(1)V_1), C_{02} + \alpha(p_{00}(2)V_0 + p_{01}(2)V_1)\}$$

 $= \max\{\underbrace{2+0.8*(0.8*5.7+0.2*1.87)}_{5.9}, \underbrace{3+0.8*(0.4*5.7+0.6*1.87)}_{5.7}\} = 5.9 \text{ for } k = 1.$

For i = 1

$$\max_{k=1,2} \{ C_{1k} + \alpha (p_{10}(k)V_0 + p_{11}(k)V_1) \} =$$

$$= \max\{C_{11} + \alpha(p_{10}(1)V_0 + p_{11}(1)V_1), C_{12} + \alpha(p_{10}(2)V_0 + p_{11}(2)V_1)\}$$
$$= \max\{\underbrace{5 + 0.8 * (0.5 * 5.7 + 0.5 * 1.87)}_{8}, \underbrace{4 + 0.8 * (0.2 * 5.7 + 0.8 * 1.87)}_{6.1}\} = 8 \text{ for } k = 1.$$

The starting policy is not optimal, it is better to use the updated policy Updated policy:

If $s_k = 0$, make decision $x_k = 1$. If $s_k = 1$, make decision $x_k = 1$.