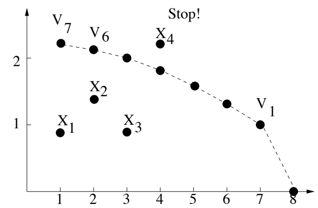
ECE 467 COMMUNICATION NETWORK ANALYSIS **FALL 2006**
PROBLEM SET 7 SOLUTIONS

1. Illustration of dynamic programming – a stopping time problem
(a) Consider the game for a possible $n+1$ observations. After seeing X_1, the player can either sop and receive reward X_1, or continue and receive expected reward V_n. Thus, $V_{n+1} = E[X_1 \vee V_n] = \int_0^\infty (x \vee V_n)e^{-x}dx = \int_0^{V_n} V_n e^{-x}dx + \int_{V_n}^\infty xe^{-x}dx = V_n + e^{-V_n}$.
(b) The optimal policy is threshold type. For $1 \leq k \leq n-1$, the player should stop after observing X_k if $X_k \geq V_{n-k}$. The rule for $n = 8$ is pictured.



(c)

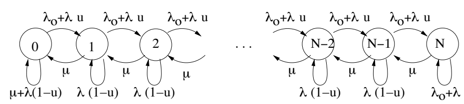
n	V_n	$1 + \frac{1}{2} + \cdots + \frac{1}{n}$
1	1.00000	1.00000
2	1.36788	1.50000
3	1.62253	1.83333
4	1.81993	2.08333
5	1.98196	2.28333
6	2.11976	2.45000
7	2.23982	2.59286
8	2.34630	2.71786
9	2.44202	2.82897
10	2.52901	2.92897
20	3.12883	3.59774
30	3.49753	3.99499

2. Comparison of four ways to share two servers
(a) *System 1:* Each of the two subsystems is an M/M/1 queue, so that N_1 is twice the mean number in an $M/M/1$ queue with arrival rate λ and departure rate μ: $N_1 = \frac{2\rho}{1-\rho}$.
System 2: Each of the two subsystems is G/M/1 where the interarrival distribution is the same as the sum of two exponential random variables with parameter 2λ. We will analyze such a G/M/1 queue. The Laplace transform of the interarrival distribution is $A^*(s) = (\int_0^\infty e^{-st}2\lambda e^{-2\lambda t}dt)^2 = (\frac{2\lambda}{s+2\lambda})^2$. We seek the solution σ in the range $0 \leq \sigma < 1$ of the equation $\sigma = A^*(\mu - \mu\sigma)$, or $\sigma^3 - 2(1+2\rho)\sigma^2 + (1+4\rho(1+\rho))\sigma - 4\rho^2 = 0$. Since $\sigma = 1$ is a solution, this equation can be written as $(\sigma - 1)(\sigma^2 - (1+4\rho)\sigma + 4\rho^2) = 0$ which has solutions $\sigma = 1$ and $\sigma = \frac{(1+4\rho)\pm\sqrt{1+8\rho}}{2}$. Thus, there is a solution $\sigma \in [0, 1)$ if and only if $0 \leq \rho < 1$, and it is given by $\sigma = \frac{1+4\rho-\sqrt{1+8\rho}}{2}$. Therefore, $N_2 = \frac{2\rho}{1-\sigma} = \frac{4\rho}{1-4\rho+\sqrt{1+8\rho}}$.
System 3 The third system is an $M/M/2$ queueing system. The number in the system is a birth-death Markov process with arrival rates $\lambda_k = 2\lambda$ for all $k \geq 0$ and death rates $\mu_k = (k \wedge 2)\mu$ for $k \geq 1$. Let $\rho = \frac{\lambda}{\mu}$. The usual solution method for birth-death processes yields that $p_1 = \frac{2\lambda}{\mu}p_0$ and $p_k = 2\rho^k p_0$ for $k \geq 1$. The process is positive recurrent if and only if $\rho < 1$, and for such ρ we find $p_0 = (1 + 2(\rho + \rho^2 + \cdots)) = \frac{1-\rho}{1+\rho}$ and $N_3 = \frac{2\rho}{1-\rho^2}$, which is smaller than N_1 by a factor $1 + \rho$.
(b) For $0 < \rho < 1$, $N_3 < N_4 < N_2 < N_1$. Here is the justification. First, we argue that $N_3 < N_4$, using the idea of stochastic domination. Indeed, let there be three independent Poisson streams: an arrival stream A with rate 2λ, and for $i = 1, 2$ a potential departure stream D_i with departure rate μ_i. Consider systems

3 and 4 running, using these same three processes. If stream D_i has a jump at time t, then there is a departure from system 3 if queue i is not empty, and there is a departure from system 4 if the entire system is not empty. By induction on the number of events (arrival or potential departure events) we see that the number of customers in system 3 is always less than or equal to the number of customers in system 4, and is occasionally strictly less. Thus, $N_3 < N_4$. We argue using dynamic programming that system 4, using send to the shorter queue, is using the optimal dynamic routing policy. Thus, N_4 is less than N_1 and N_2. Finally, we can compare the formulas found above to see that $N_1 > N_2$. (It can be shown that as $\rho \to 1$, $(1-\rho)N_1 \to \frac{2}{3}$, $(1-\rho)N_2 \to \frac{2}{3}$, $(1-\rho)N_3 \to 1$, and $(1-\rho)N_4 \to 1$.)

3. A stochastic dynamic congestion control problem

We take the state space to be $\mathcal{S} = \{0, 1, \ldots, K\}$, where a state represents the number of customers in the system. Let α, with $\alpha > 0$, denote the discount rate. The interevent times are exponentially distributed with parameter $\gamma = \lambda + \lambda_o + \mu$, and the control set is given by $U = [0, 1]$. The one step transition probabilities are given by (the quantities in the diagram should be divided by γ):



$$P(u) = \frac{1}{\gamma} \begin{pmatrix} \mu + \lambda(1-u) & \lambda_o + \lambda u & & & & & \\ \mu & \lambda(1-u) & \lambda_o + \lambda u & & & & \\ & \mu & \lambda(1-u) & \lambda_o + \lambda u & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & & \mu & \lambda(1-u) & \lambda_o + \lambda u \\ & & & & & \mu & \lambda_o + \lambda \end{pmatrix}.$$

The cost is given by

$$\begin{aligned} \text{cost} &= E \sum_{t>0} \left[cI_{\{x_{t-}=K \text{ and an arrival occurs at time } t\}} - rI_{\{\triangle x_t = -1\}} \right] e^{-\alpha t} \\ &= E \int_0^\infty e^{-\alpha t} g(x_t, u_t) dt - \frac{r\mu}{\alpha} \end{aligned}$$

where $g(x,u) = c(\lambda_o + \lambda u) I_{\{x=K\}} + r\mu I_{\{x=0\}}$. Here we decided that u can be nonzero even when the system is full. It is optimal to take $u_n(K) = 0$ however, because admitting customers when the system is full incurs a cost with no change in state. Therefore, we could have assumed that $u_n(K) = 0$ and taken $g(x,u) = c\lambda_o I_{\{x=K\}} + r\mu I_{\{x=0\}}$. We will drop the constant term, and instead take the cost to be

$$\text{cost} = E \int_0^\infty e^{-\alpha t} g(x_t, u_t) dt$$

(b) For the discrete-time equivalent model, the one-step discount factor is $\beta = \frac{\alpha}{\alpha+\gamma}$ and the value update equations are given by

$$V_{n+1}(x) = \min_{0 \le u \le 1} \left[g(x,u) + \frac{\beta}{\gamma} \{ \mu V_n((x-1)_+) + \lambda_o V_n((x+1) \wedge K) + \lambda(1-u) V_n(x) + \lambda u V_n((x+1) \wedge K) \} \right]$$

In particular, for $x = K$,

$$V_{n+1}(x) = \min_{0 \le u \le 1} \left[c(\lambda_o + \lambda u) + \frac{\beta}{\gamma} \{ \mu V_n(K-1) + (\lambda_o + \lambda) V_n(K) \} \right]$$

so we see that indeed $u = 0$ is optimal when $x = K$. Thus, using $g(x,u) = c\lambda_o I_{\{x=K\}} + r\mu I_{\{x=0\}}$. we have that

$$V_{n+1}(x) = g(x) + \frac{\beta}{\gamma}\left[\mu V_n((x-1)_+) + \lambda_o V_n((x+1)\wedge K) + \lambda \min\left\{V_n(x), V_n((x+1)\wedge K)\right\}\right]$$

(c) When there are n steps-to-go and the current state is x, an optimal control is given by

$$u_n(x) = \begin{cases} 0 & \text{if } V_n(x) \leq V_n((x+1)\wedge K) \\ 1 & \text{if } V_n(x) > V_n((x+1)\wedge K) \end{cases}$$

(d) Let us prove that for each $n \geq 1$ there exists a threshold τ_n such that $u_n(x) = I_{\{x \leq \tau_n\}}$. It is enough to show that the following is true for each $n \geq 1$:

(a) V_n is convex, i.e. $V_n(1) - V_n(0) \leq V_n(2) - V_n(1) \leq \cdots \leq V_n(K) - V_n(K-1)$
(b) $-\frac{r\gamma}{\beta} \leq V_n(1) - V_n(0)$
(c) $V_n(K) - V_n(K-1) \leq \frac{c\gamma}{\beta}$

To complete the proof, we prove (a)-(c) by induction on n. The details are very similar to those for the $M_{\text{controlled}}/M/1$ example in the notes, and are omitted.

(e) For an exact formulation, we need to describe a state space. If FCFS service order is assumed, it is not enough to know the number of customers of each type in the system, so that the state space would be the set of sequences from the alphabet $\{1,2\}$ with length between 0 and K. We expect the optimal control to have some monotonicity properties, but it would be hard to describe the control because the state space is so complex. If instead, the service order were pure preemptive resume priority to customers of one type, or processor sharing, then the state space could be taken to be $\{(n_1, n_2) : n_1 \geq 0, n_2 \geq 0, \text{ and } n_1 + n_2 \leq K\}$. We expect the optimal control to have a switching curve structure.

4. Control dependent cost

(a)

$$\begin{aligned} E_x \int_0^{t_n} g(X(t), u(t)) e^{-\alpha t} dt &= \sum_{k=0}^{n-1} E_x \int_{t_k}^{t_{k+1}} g(X(t), u(t)) e^{-\alpha t} dt \\ &= \sum_{k=0}^{n-1} E_x \int_{t_k}^{t_{k+1}} g(X(t_k), w_k) e^{-\alpha t} dt \\ &= \sum_{k=0}^{n-1} E_x [g(X(t_k), w_k) \int_{t_k}^{t_{k+1}} e^{-\alpha t} dt] \\ &= \sum_{k=0}^{n-1} E_x [g(X(t_k), w_k)] E_x [\int_{t_k}^{t_{k+1}} e^{-\alpha t} dt] \\ &= \frac{1-\beta}{\alpha} E_x \sum_{k=0}^{n-1} \beta^k g(X(t_k), w_k) \end{aligned}$$

where we used the fact that

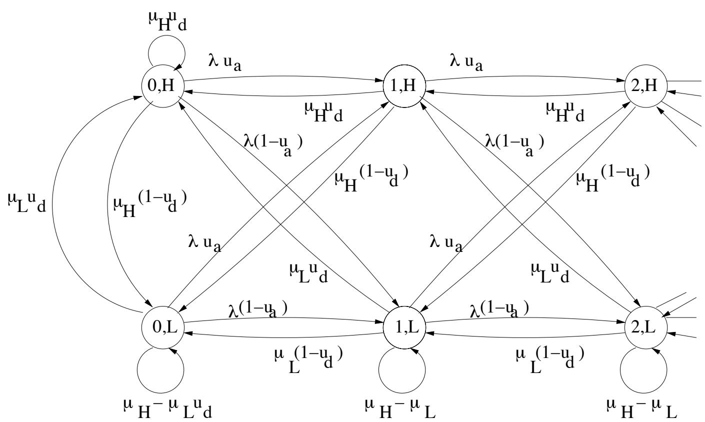
$$E_x[\int_{t_k}^{t_{k+1}} e^{-\alpha t} dt] = E_x[\frac{1}{\alpha}(e^{-\alpha t_k} - e^{-\alpha t_{k+1}})] = \frac{1}{\alpha}[\beta^k - \beta^{k+1}] = \frac{(1-\beta)\beta^k}{\alpha}.$$

(b) The only change is that g should be added to the cost before the minimization over u, yielding:

$$V_{n+1}(x) = \inf_{u \in \mathcal{U}} \left[g(x,u) + \beta \sum_y p_{xy}(u) V_n(y) \right].$$

5. A dynamic server rate control problem with switching costs

(a) We use the state space $\mathcal{S} = \{(l, \theta) : l \in \mathbb{Z}_+, \theta \in \{H, L\}\}$, where for a given state $x = (l, \theta)$, l denotes the number of customers in the system and θ denotes the state of the server. Let $\gamma = \lambda + \mu_H$, which is the maximum event rate that we shall use. (Another natural choice would be $\gamma = \lambda + \mu_L + \mu_H$, and then the self loop transition probabilities would be increased in the one step transition probabilities.) We take the control values $u = (u_a, u_d)$ to be in the set $\mathcal{U} = [0, 1]^2$, where u_a is the probability the server is in the high state after the next event, given the next event is an arrival, and u_d is the probability the server is in the high state after the next event, given the next event is a departure. The following diagram gives the transition probabilities (the quantities shown should be divided by γ).[1]



The total costs can be described as follows.

$$\text{cost due to use of high service rate and queueing} = E_x \left[\int_0^T \{c_H I_{\theta_t = H\}} + c_W l_t\} e^{-\alpha t} dt \right]$$

$$\begin{aligned} \text{cost due to switching} &= E_x \left[\sum_{t \leq T} c_S I_{\{\theta_{t-} = L, \theta_t = H\}} e^{-\alpha t} \right] \\ &= E_x \left[\int_0^T c_S I_{\{\theta_{t-} = L\}} [\lambda u_a(t) + \mu_L u_d(t)] e^{-\alpha t} dt \right] \end{aligned}$$

Thus, the cost is captured by taking

$$g(x, u) = g((l, \theta), (u_a, u_d)) = c_H I_{\{\theta = H\}} + c_W l \ + \ c_S I_{\{\theta = L\}} [\lambda u_a + \mu_L u_d]$$

(b) We first consider $V_{n+1}(x)$ for states $x = (l, \theta)$ with $\theta = H$. Since $g((l, H), u) = c_H + c_W l$, which doesn't depend on u, the backwards recursion is:

$$\begin{aligned} V_{n+1}(l, H) = c_H + c_W l + \beta \min_{u \in [0,1]^2} \Big\{ & \frac{\lambda u_a}{\gamma} V_n(l+1, H) + \frac{\lambda(1-u_a)}{\gamma} V_n(l+1, L) \\ & + \frac{\mu_H u_d}{\gamma} V_n((l-1)_+, H) + \frac{\mu_H(1-u_d)}{\gamma} V_n((l-1)_+, L) \Big\} \end{aligned}$$

[1] Variations of these equations are possible, depending on the exact assumptions. We chose to allow a server state change at potential departure times, even if the potential departures are not used.

The analogous equation for $\theta = L$ is

$$V_{n+1}(l, L) = \min_{u\in[0,1]^2} c_W l + c_S\{\lambda u_a + \mu_L u_d\} + \beta \left\{ \frac{\lambda u_a}{\gamma} V_n(l+1, H) + \frac{\lambda(1-u_a)}{\gamma} V_n(l+1, L) \right.$$
$$\left. + \frac{\mu_L u_d}{\gamma} V_n((l-1)_+, H) + \frac{\mu_L(1-u_d)}{\gamma} V_n((l-1)_+, L) + \frac{\mu_H - \mu_L}{\gamma} V_n(l, L) \right\}$$

Solving for u in the above equations and setting $\delta = \frac{c_S\gamma}{\beta}$ yields the following simpler version of the dynamic programming backwards recursion:

$$\begin{aligned} V_{n+1}(l, H) &= c_H + c_W l + \frac{\beta\lambda}{\gamma} \min\{V_n(l+1, H), V_n(l+1, L)\} \\ &\quad + \frac{\beta\mu_H}{\gamma} \min\{V_n((l-1)_+, H), V_n((l-1)_+, L)\} \end{aligned} \tag{1}$$

$$\begin{aligned} V_{n+1}(l, L) &= c_W l + \frac{\beta\lambda}{\gamma} \min\{\delta + V_n(l+1, H), V_n(l+1, L)\} \\ &\quad + \frac{\beta\mu_L}{\gamma} \min\{\delta + V_n((l-1)_+, H), V_n((l-1)_+, L)\} + \frac{\beta(\mu_H - \mu_L)}{\gamma} V_n(l, L) \end{aligned} \tag{2}$$

The optimal controls can be succinctly described as follows. If the current switch state is θ, and if there will be l' customers in the system after the next arrival or potential departure, then the optimal server state after such arrival or potential departure is given by:

$$\begin{array}{lll} \text{If } \theta = H \text{ then:} & V_n(l', L) > V_n(l', H) & \Leftrightarrow \text{ new server state H} \\ \text{If } \theta = L \text{ then:} & V_n(l', L) > V_n(l', H) + \delta & \Leftrightarrow \text{ new server state H} \end{array} \tag{3}$$

The function $V_n(l, \theta)$ represents the cost-to-go for initial state (l, θ), with the understanding that it is not possible to switch the server state until the first event time. Let $W_n(l, \theta)$ denote the cost-to-go, given the initial state is (l, θ), but assuming that the server state can be changed at time zero. Then the dynamic programming equations become:[2]

$$W_n(l, \theta) = \min\{V_n(l, L), V_n(l, H) + \delta I_{\{\theta=L\}}\} \tag{4}$$

$$\begin{aligned} V_{n+1}(l, \theta) &= c_H I_{\{\theta=H\}} + c_W l + \frac{\beta\lambda}{\gamma} W_n(l+1, \theta) + \frac{\beta\mu_L}{\gamma} W_n((l-1)_+, \theta) \\ &\quad + \frac{\beta(\mu_H - \mu_L)}{\gamma} \left\{ I_{\{\theta=H\}} W_n(l-1, H) + I_{\{\theta=L\}} W_n(l, L) \right\} \end{aligned} \tag{5}$$

with the initial conditions $V_0 \equiv 0$ and $W_n \equiv 0$.

In the special case that there is no switching cost, $\delta = 0$. In that case, $W_n(l, \theta)$ does not depend on θ, so we write it as $W_n(l)$. Then the dynamic programming equations become:

$$W_n(l) = \min\{V_n(l, L), V_n(l, H)\} \tag{6}$$

$$\begin{aligned} V_{n+1}(l, \theta) &= c_H I_{\{\theta=H\}} + c_W l + \frac{\beta\lambda}{\gamma} W_n(l+1) + \frac{\beta\mu_L}{\gamma} W_n((l-1)_+) \\ &\quad + \frac{\beta(\mu_H - \mu_L)}{\gamma} \left\{ I_{\{\theta=H\}} W_n(l-1) + I_{\{\theta=L\}} W_n(l) \right\} \end{aligned} \tag{7}$$

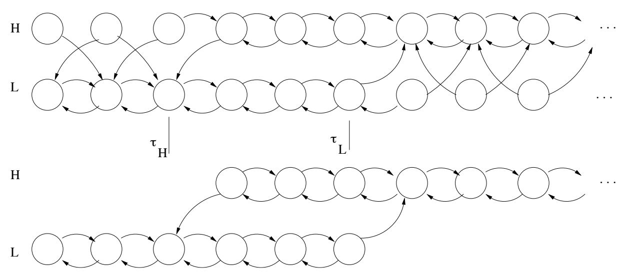
In the case of no switching costs, it is rather simple to show by induction on n that W_n is a convex, nondecreasing function on $\mathbb{Z}_+$ and V_n satisfies properties 1-5 in the original solutions. As noted in the original solutions, property 3 for V_n implies the conjectured threshold structure.

[2] In this version, we allow a change of server state after any event. Thus, the functions V_n are slightly different than the ones above.

(c) We conjecture that the optimal control has a threshold behavior, because when there are more customers in the system, it is more valuable to quickly serve the customer in the server, thereby reducing the waiting times of all remaining customers. Since there is a switching cost, however, it is better to be a bit reluctant to switch, because excessive switching is costly. This suggests that the threshold for switching from H to L should be smaller than the threshold for switching from L to H. Specifically, we expect there are two thresholds, τ_H and τ_L, with $\tau_H \leq \tau_L$, so that the optimal control law (3) becomes

$$l' > \tau_\theta \Leftrightarrow \text{ new server state H} \tag{8}$$

This leads to the nonzero transition probabilities shown. The transient states are included in the first diagram and are omitted in the second.



We haven't found a proof of the conjectured threshold structure in general, but we here sketch a proof in the case of zero switching cost ($\delta = 0$). Consider the following properties of a function V on $\mathbb{Z}_+ \times \{L, H\}$:

1. $V(l, \theta)$ is nondecreasing in l for $\theta = L$ and for $\theta = H$.
2. $V(l, \theta)$ is convex in l for $\theta = L$ and for $\theta = H$.
3. $V(l+1, L) - V(l+1, H) - V(l, L) + V(l, H) \geq 0$ for $l \geq 0$.
4. $V(l+2, H) - V(l+1, H) - V(l+1, L) + V(l, L) \geq 0$ for $l \geq 0$.
5. $V(l+2, L) - V(l+1, L) - V(l+1, H) + V(l, H) \geq 0$ for $l \geq 0$.

We write property 1.H to denote property 1 for $\theta = 1$. Properties 1.L, 2.H, and 2.L are defined analogously. The following figure gives a graphical representation of the five properties.

Property:	1.L	1.H	2.L	2.H	3	4	5
H		− +		+ 2− +	+ −	− +	+ −
L	− +		+ 2− +		− +	+ −	− +

For each property, the figure indicates the type of linear combinations of values of V which should be nonnegative. The collection of properties 1-5 are nearly symmetric in H and L. Property 5 is obtained by swapping L and H in property 4. Thus, property 3 is the only part of properties 1-5 that is not symmetric in L and H.

Properties 3 and 4 imply property 2 (both 2.L and 2.H). This can be seen graphically by sliding the diagram for property 4 to the left to overlap the diagram for property 3, and adding. Similarly, properties 2.L and 3 imply property 5. Thus, properties 3 and 4 together imply properties 2 and 5. So to prove a function has properties 1-5, it suffices to prove it has properties 1, 3, and 4.

Property 3 is the one connected to the threshold structure. Another way to state property 3 is that $V(l, H) - V(l.L)$ is nonincreasing in l. That is, as l increases, given there are l customers in the system, it becomes increasingly preferable to be in state H. More specifically, if V_n satisfies property 3, and if $\tau_L = \max\{l : V(l, H) - V(l, L) \geq 0\}$ and $\tau_H = \max\{l : V(l, H) + \delta - V(l, L) \geq 0\}$, then (3) is equivalent to (8).

It remains to show that V_n has properties 1-5 for all $n \geq 0$. To do so, it can be proved by induction on n that V_n has properties 1-5 for all $n \geq 0$, and W_n is convex and nondecreasing on $\mathbb{Z}_+$. For the base case, observe that the function V_0 given by $V_0 \equiv 0$ has properties 1-5. For the general induction step, it can be easily shown that if V_n has properties 1-5, then W_n is convex, nondecreasing. And given that W_n is convex, nondecreasing, it can be shown that V_{n+1} has properties 1-5. As mentioned above, it suffices to establish that V_{n+1} has properties 1,3, and 4. Unfortunately, this approach doesn't seem to work in case $\delta > 0$.

(d) First, if $c_S = 0$ then the cost-to-go does not depend on the server state, and in particular the thresholds should be equal: $\tau_L = \tau_H$. Suppose a single customer is served, with no other customers waiting or arriving. If the server is put into the high state, and if t_H denotes the time the customer departs, then the average cost is $E[\int_0^\infty e^{-\alpha t} dt](c_H + c_W) = E[\frac{1-e^{-\alpha t_H}}{\alpha}](c_H + c_W) = \frac{c_H + c_W}{\alpha + \mu_H}$. Similarly, if the server is put into the low state during the service interval, then the cost is $\frac{c_W}{\alpha + \mu_L}$. The given inequality implies that it costs less to use the high rate, even if there is only one customer in the system over all time. But using the high rate for a given server helps decrease the costs for any other customers that might be in the system, so that it is optimal to always use the high rate (when the system is not emput), under the conditions given in the problem.