Problem 5.1 Let us assume that the state is \((n, m)\). Male \(i\) mates at a rate \(\lambda\) with female \(j\), and therefore it mates at a rate \(\lambda m\). Since there are \(n\) males, matings occur at a rate \(\lambda nm\). Therefore,

\[ v_{(n,m)} = \lambda nm . \]

Since any mating is equally likely to result in a female as in a male, we have

\[ P_{(n,m)}(n+1,m) = P_{(n,m)}(n,m+1) = \frac{1}{2} . \]

Problem 5.2 Let \(N_A(t)\) be the number of organisms in state \(A\) and let \(N_B(t)\) be the number of organisms in state \(B\). Then clearly \(\{(N_A(t), N_B(t)) : t \geq 0\}\) is a continuous-Markov chain with

\[ v_{(n,m)} = \alpha n + \beta m \]

and

\[ P_{(n,m)}(n-1,m+1) = \frac{\alpha n}{\alpha n + \beta m} , \quad P_{(n,m)}(n+2,m-1) = \frac{\beta m}{\alpha n + \beta m} . \]

Problem 5.3 (a) Let \(N(t)\) denote the number of transitions be \(t\). It is easy to show in this case that

\[ P(N(t) \geq n) \leq \sum_{j=n}^{\infty} e^{-Mt} \frac{(Mt)^j}{j!} \]

and thus \(P(N(t) < \infty) = 1\).

(b) Let \(X_{n+1}\) denote the time between the \(n\)-th and \((n+1)\)-st transition and let \(J_n\) denote the \(n\)-th state visited. Also if we let

\[ N(t) \triangleq \sup\{n : X_1 + \cdots + X_n \leq t\} \]

then \(N(t)\) denotes the number of transitions by \(t\). Now let \(j\) be the first recurrent state that is reached and suppose it was reached at the \(n_0\)-th transition (\(n_0\) must be finite by assumption). Let \(n_1, n_2, \cdots\) be the successive integers \(n\) at which \(J_n = j\). (Such integers exist since \(j\) is recurrent.) Set \(T_0 = X_1 + \cdots + X_{n_0}\), and

\[ T_k \triangleq X_{n_{k-1}+1} + \cdots + X_{n_k} . \]
In other words, $T_k$ denote the amount of time between the $k$-th and $(k+1)$-th visit to $j$. Therefore it follows that $\{T_k, k \geq 1\}$ forms a renewal process, and so $\sum_{k=1}^{\infty} T_k = \infty$ with probability 1. Since

$$\sum_{n=1}^{\infty} X_n = \sum_{k=0}^{\infty} T_k,$$

it follows that $\sum_{n=1}^{\infty} X_n = \infty$.

**Problem 5.12 (a)** Since

$$P_0 = \frac{1}{1/\lambda + 1/\mu} = \frac{\mu}{\lambda + \mu},$$

it follows that

$$\lim_{t \to \infty} \frac{N(t)}{t} = \frac{\alpha_0 \mu}{\lambda + \mu} + \frac{\alpha_1 \mu}{\lambda + \mu}.$$

(b) The expected total time spent in state 0 by $t$ is

$$\int_0^t P_{00}(s) ds = \frac{\mu}{\lambda + \mu} t + \frac{\lambda}{(\lambda + \mu)^2} \left(1 - e^{-(\lambda + \mu)t}\right).$$

Calling the above $E[T(t)]$ we have

$$E[N(t)] = \alpha_0 E[T(t)] + \alpha_1 (t - E[T(t)]).$$

**Problem 5.19** Conditioning on $X$, the time of the first transition yields:

$$E[T] = \int_0^t E[T|X = s] v_0 e^{-v_0 s} ds + t e^{-v_0 t}$$

$$= \int_0^t (s + E[R] + E[T]) v_0 e^{-v_0 s} ds + t e^{-v_0 t}$$

where $R$, which is independent of $X$, is the time to return to 0 once it has been left. Hence,

$$E[T] = (1 - e^{-v_0 t}) (E[R] + E[T]) + \int_0^t v_0 s e^{-v_0 s} ds + t e^{-v_0 t}.$$

Now,

$$P_0 = \frac{1/v_0}{1/v_0 + E[R]}, \quad \text{or}, \quad E[R] = \frac{1 - P_0}{v_0 P_0}$$

and substituting into the preceding gives that

$$E[T] = \frac{(e^{v_0 t} - 1) (1 - P_0)}{v_0 P_0} + \frac{e^{v_0 t} - 1}{v_0}.$$

**Problem 5.22** The analysis on page 153-154 with

$$\lambda_n = \begin{cases} \lambda & n \geq 0 \\ n \mu & 1 \leq n \leq s \\ s \mu & n > s \end{cases}$$

We need $\lambda < s \mu$. 

2
Problem 5.24 Let $X_i(t)$ denote the number of customers at server $i$, $i = 1, 2$, when there is unlimited waiting room. The, in steady state,

$$P(n_i \text{ at server } i) = 2 \prod_{i=1}^{2} \left( \frac{\lambda_i}{\mu_i} \right)^{n_i} \left( 1 - \frac{\lambda_i}{\mu_i} \right).$$

Now the model under consideration is just a truncation of the above, which is time reversible by problem 5.23. The truncation is to the set

$$A \triangleq \{ (n, m) : n = 0, m \leq N + 1 \text{ or } m = 0, n \leq N + 1 \text{ or } n > 0, m > 0, n + m \leq N + 2 \}.$$

Hence, for $(n, m) \in A$,

$$P(n, m) = C \left( \frac{\lambda_1}{\mu_1} \right)^n \left( 1 - \frac{\lambda_1}{\mu_1} \right) \left( \frac{\lambda_2}{\mu_2} \right)^m \left( 1 - \frac{\lambda_2}{\mu_2} \right).$$

Problem 5.25 In steady state it has the same probability structure as the arrival process. Hence if we include in the departure process those arrivals that do not enter, then it is a Poisson process.

Problem 5.26 (a) Follows from results of section 6.2 by writing $n'_i = D_j n$ and so $n = B_{j-1} n'$. 

(b) A Poisson process by time reversibility. If $D(0) = 0$, it is a nonhomogeneous Poisson process.

Problem 5.28 For $\underline{n} = (n_1, \cdots, n_r)$, let

$$P(\underline{n}) = C \prod_{i=1}^{r} \alpha_i^{n_i}.$$ 

If

$$\underline{n}' = (n_1, \cdots, n_i - 1, \cdots, n_j + 1, \cdots, n_r)$$

with $n_i > 0$ then

$$P(\underline{n})q(\underline{n}, \underline{n}') = P(\underline{n}')q(\underline{n}', \underline{n})$$

$$\Leftrightarrow \alpha_i \frac{\mu_i}{r-1} = \alpha_j \frac{\mu_j}{r-1}$$

$$\Leftrightarrow \alpha_i \mu_i = \alpha_j \mu_j.$$ 

So, setting $\alpha_i = 1/\mu_i$, $i = 1, \cdots, r$ and letting $C$ be such that $\sum \underline{n} P(\underline{n}) = 1$ the time reversibility equations are satisfied.

Problem Extra  Program = Calculations for the M/M/s/r=M Model following 1999 Improving paper
Ward Whitt 11/17/03

Step1 = Model Parameters
s = number of servers, $s = 100$
lambda = arrival rate, lambda = 100
mu = individual service rate, mu = 1
alpha = individual abandonment rate (exponential abandonments), alpha = 1
r = number of extra waiting spaces, r = 100

Step 2 = Key Steady-state Probabilities
ProbNoWait = Probability of not having to wait before beginning service ProbNoWait = 0.48670120172085
ProbAllBusy = Probability that all servers are busy upon arrival ProbAllBusy = 0.51329879827915
ProbServed = 0.96013900319085
ProbLoss = Probability that an arrival is lost (blocked) at arrival ProbLoss = 4.716970602792618e-019
ProbAban = Probability that a customer eventually abandons ProbAban = 0.03986099680915

Step 3 = Mean Values - Counting
MeanInQueue = 3.98609968091471
MeanBusyServers = 96.01390031908522
MeanNumberInSys = 99.99999999999993

Step 4 = Second Moments and Variances - Counting
SecondMomInQueue = 51.32697982791468
SecondMomBusyServers = 9.251450183989135e+003
SecondMomNumberInSys = 1.010000000000000e+004

Step 5 = Response-Time Moments
MeanRespTime = 0.997604954046
ConditMeanRespTimeServed = 1.03891653311596
SecMomRespTime = 2.9466069041806
VarRespTime = 1.195159129641430
ConditSecMomRespTimeServed = 3.0693698123555
ConditVarRespTimeServed = 1.98958941845386

Step 6 = Waiting-Time Moments
Step 6a = Waiting Times for Customers who are Served
MeanWaitServed = 0.03736528131360
ConditMeanWaitServed = 0.03891653311596
SecMomWaitandServed = 0.00472192207249
ConditSecMomWaitServed = 0.00491795672756
ConditVarWaitServed = 0.00340346017779

Step 6b = Waiting Times for Customers who Abandon
MeanWaitAbandon = 0.00249571549554
ConditMeanWaitAban = 0.06261046374463
SecMomWaitAbandon = 2.69589206940988e-004
ConditSecMomWaitAbandon = 0.00676121879856
ConditVarWaitAban = 0.00284114862823

Step 6c = Waiting Time for All Customers, Served or Abandoning
MeanWaitTimeAll = 0.03736528131360
ConditMeanWaitTimeAll = 0.03891653311596
SecMomWaitTimeAll = 0.00491795672756
ConditSecMomWaitTimeAll = 0.00499143099109

Step 7 = Conditional Waiting Time CCDF Values For Served Customers
meanWait = 0.05000000000000
ProbOKWaitifServed = 0.70144791424012
ConditMeanWaitIfServed = 0.05000000000000
ProbOKWaitifServed = 0.70144791424012

Step 8 = Conditional Waiting Time CDF Values For Customers Who Abandon
meanWait = 0.05000000000000
ProbOKWaitIfAbandon = 0.50953673649554
ConditMeanWaitAban = 0.06261046374463
SecMomWaitAban = 2.69589206940988e-004
ConditSecMomWaitAban = 0.00676121879856

Step 9 = Waiting Time CDF Values For All Customers
meanWait = 0.05000000000000
ProbOKWait = 0.69379814431203
ConditMeanWait = 0.05000000000000
ProbOKWait = 0.69379814431203

(a) MeanInQueue = 3.98609968091471, VarInQueue = 35.44088916172650
(b) arrival rate × ProbAban = 100 × 0.03986099680915 = 3.986
ProbNoWait = 0.48670120172085

ConditMeanWaitServed + service time = 0.03891653311596 + 1

Conditional Waiting Time CDF Values For Served Customers at \( t = 0.1 \) = ProbOK-WaitifServed = 0.84743410401854

Since the abandon rate \( \alpha \) equals the service rate \( \mu \), the model simplifies. But note that the simplification and following solution only works for the special case of \( \alpha = \mu \). First, in the view point of the number of customers in the system, the system is equivalent to \( M/M/s + r/0 \) since the departure rate from the system including the abandoned customers is \( n\mu \) even if \( n > s \) because of \( (n-s)\alpha = (n-s)\mu \) abandonment rate. However, we can anticipate that blocking is negligible with such a large waiting space. Thus the \( M/M/s + r/0 \) model should be essentially equivalent to a \( M/M/\infty \) model, for which the steady-state distribution is exactly Poisson, and approximately normal. Hence, the probability all servers are busy in the given \( M/M/s/r + M \) model is approximately equal to the probability that at least \( s \) servers are busy in the \( M/M/\infty \) model. Clearly, it is possible to compute the probability that at least \( s \) servers are busy in the \( M/M/\infty \) model very quickly. (It might not be regarded as “simple,” however. When you read, “it is easy to see that . . .,” in a paper, take that as an invitation to check very carefully.)

Now we describe how to proceed with the more special, but exact, representation in terms of the \( M/M/s + r/0 \) model: For \( n < s + r = 200 \)

\[
P_n = \frac{\lambda_0 \cdots \lambda_{n-1}}{\mu_1 \cdots \mu_n} \frac{\lambda^n}{1 + \sum_{k=1}^{s+r} \frac{\lambda_0 \cdots \lambda_{k-1}}{\mu_1 \cdots \mu_k}}
= \frac{\lambda^n}{n! \left( \sum_{k=0}^{s+r} \frac{\lambda^k}{k!} \right)}
= \frac{e^{-\lambda} \lambda^n}{n! \left( \sum_{k=0}^{s+r} e^{-\lambda} \frac{\lambda^k}{k!} \right)}
\]

and we have (for Poisson random variable \( N \) with parameter \( \lambda = 100 \))

\[
P(\text{All servers are busy}) = \sum_{n=0}^{s-1} P_n
= \frac{P(N \leq 99)}{P(N \leq 200)}
= 0.48670120172087 \text{ from MATLAB}
\]

MeanInQueue \times \text{abandon rate per customer in queue}
= 3.98609968091471 \times 1 = 3.98609968091471

(g) MeanInQueue× abandon rate per customer in queue