Solutions for Assignment #4

KJC (10/19/04)

#1) The P matrix, set-up of equations, and solution to the equations is:

$$P := \begin{pmatrix} 0.25 & 0.75 & 0 \\ 0 & 0.10 & 0.90 \\ 0.60 & 0.40 & 0 \end{pmatrix}$$

$pi0 := 1 \quad pi1 := 0 \quad pi2 := 0$ These are guesses for Mathca

Given

$$pi0 = 0.25 \cdot pi0 + 0.60 \cdot pi2$$

$$pi1 = 0.75 \cdot pi0 + 0.10 \cdot pi1 + 0.40 \cdot pi2$$

$$1 = pi0 + pi1 + pi2$$

$$\text{Find}(pi0, pi1, pi2) = \begin{pmatrix} 0.275 \\ 0.382 \\ 0.344 \end{pmatrix}$$

The `in.dat` file for iter.c (and the execution output from iter.c) is:

```
P 3
& ---------------- column #0    &
0       0.25
2       0.60
-1      zero
& ---------------- column #1    &
0       0.75
1       0.10
2       0.40
-1      zero
& ---------------- column #2    &
1       0.90
-1      zero
& ---------------- END FLAG     &
-999    zero
```

```
------------------------------------------------- iter.c -----
  Iteration count = 10 -- Convergence mean = 0.333333
  Iteration count = 20 -- Convergence mean = 0.000109076
  Iteration count = 30 -- Convergence mean = 3.61621e-07
  Iteration count = 40 -- Convergence mean = 1.18098e-09
  Iteration count = 50 -- Convergence mean = 3.80483e-12
  Iteration count = 60 -- Convergence mean = 1.21014e-14
  Iteration count = 70 -- Convergence mean = 3.70074e-17
------------------------------------------------------------
  Results for P matrix - size = 3 and tolerance = 1e-15
    Pi[0] = 0.274809
    Pi[1] = 0.381679
    Pi[2] = 0.343511
------------------------------------------------------------
```

#2) The Q matrix, set-up of equations, and solution to the equations is:

$$Q := \begin{pmatrix} -15 & 15 & 0 \\ 0 & -18 & 18 \\ 12 & 8 & -20 \end{pmatrix}$$

pi0 := 1 pi1 := 0 pi2 := 0 These are guesses for Mathca

Given

$0 = -15 \cdot pi0 + 12 \cdot pi2$

$0 = 15 \cdot pi0 - 18 \cdot pi1 + 8 \cdot pi2$

$1 = pi0 + pi1 + pi2$

Find(pi0, pi1, pi2) =

The `in.dat` file for iter.c (and the execution output from iter.c) is:

```
Q 3
& ---------------- column #0    &
0       -15
2        12
-1      zero
& ---------------- column #1    &
0        15
1       -18
2         8
-1      zero
& ---------------- column #2    &
1        18
2       -20
-1      zero
& ---------------- END FLAG     &
-999    zero
------------------------------------------------- iter.c -----
  Iteration count = 10 -- Convergence mean = 0.333333
  Iteration count = 20 -- Convergence mean = 0.00384069
  Iteration count = 30 -- Convergence mean = 0.000337891
<SNIP SNIP>
  Iteration count = 120 -- Convergence mean = 3.2474e-14
  Iteration count = 130 -- Convergence mean = 2.16493e-15
  Iteration count = 140 -- Convergence mean = 2.40548e-16
-----------------------------------------------------------
  Results for Q matrix - size = 3 and tolerance = 1e-15
    Pi[0] = 0.274809
    Pi[1] = 0.381679
    Pi[2] = 0.343511
-----------------------------------------------------------
```

#3) This chain is not ergodic, so we cannot directly solve it. State 0 is obviously transient. Once in states 1 and 2, or 3 and 4 the probability of being in 1 or 2 (or 3 or 4) is 0.50 for each state. The probability of being in states 1 and 2 is 0.25 and the probability of being in states 3 and 4 is 0.75. So, $\pi 0 = 0$, $\pi 1 = 0.25 \cdot 0.50 = 0.125$, $\pi 2 = \pi 1 = 0.125$, $\pi 3 = 0.75 \cdot 0.50 = 0.375$, and $\pi 4 = \pi 3 = 0.375$.

#4) This is an Erlang-B problem. We want the blocking probability for λ = 100 calls per hour and μ = 12 calls per hour (this is a mean service time of 5 minutes) to be less than 1%. Using erlang.c repeatedly we see:

```
------------------------------------------------ erlang.c -----
lambda = 100.000000  mu = 12.000000  num_servers = 15
rho = 0.555556
Pr[block] (ErlangB) = 0.012072
Pr[queue] (ErlangC) = 0.026758
---------------------------------------------------------------
------------------------------------------------ erlang.c -----
lambda = 100.000000  mu = 12.000000  num_servers = 16
rho = 0.520833
Pr[block] (ErlangB) = 0.006248
Pr[queue] (ErlangC) = 0.012952
---------------------------------------------------------------
```

Thus, 16 operators are needed.
