Appendix A

Definition 1 *The Kronecker product of $A \in M_{m,n}(F)$ and $B \in M_{p,q}(F)$ is denoted by $A \otimes B$ and is defined to be the block matrix*

$$A \otimes B = \begin{bmatrix} a_{11}B & \cdots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \cdots & a_{mm}B \end{bmatrix} \in M_{mp,nq}(F).$$

We also mention another Kronecker operation, the Kronecker sum, $A \oplus B$ is defined by square matrices A and B and is given by

$$A \oplus B \triangleq A \otimes I_m + I_n \otimes B,$$

where $A \in M_n$ and $B \in M_m$. Thus, $A \otimes I_m, I_n \otimes B$, and their sum are in M_{mn}.

NATIONAL CHENGCHI UNIVERSITY LIBRARY

Appendix B

Following is the code of the program.

```
function []=productform
format short %The precision is 16 bits
clear %Clearing Variables
fprintf('\n\n\n\n  ********This is a Ck/Cm/1/N Queueing system. ********')
n=1;
while(n==1)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                 %
%         Input the parameters,N,k,m,lambda,mu,pro1,pro2         %
%                                                                 %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
no=input('\n\n     Please input the capacity size of Queueing System =');
ap=input('\n     Please input the number of phases of arrival process =');
sp=input('\n     Please input the number of phases of service process =');
lambda=input('\n     Please input the arrival rate at each phase =');
mu=input('\n     Please input the service rate at each phase =');
pro1=input('\n Please input the branching probability of arrival process  =');
pro2=input('\n Please input the branching probability of service process  =');
if((no>0))
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                 %
%                            Define rho                           %
%                                                                 %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
alambda=0;
for i=1:(ap)
   Arate=0;
   for j=1:(i)
      Arate=Arate+1/(lambda(j));
   end
```

```
   Aprod=1;
   for r=1:(i-1)
      Aprod=Aprod*(1-pro1(r));
   end
   alambda=alambda+Arate*Aprod*pro1(i);
end
lambd=1/alambda;

smu=0;
for i=1:(sp)
   Brate=0;
   for j=1:(i)
      Brate=Brate+1/(mu(j));
   end
   Bprod=1;
   for r=1:(i-1)
      Bprod=Bprod*(1-pro2(r));
   end
   smu=smu+Brate*Bprod*pro2(i);
end
muu=1/smu;

rho=lambd/muu;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                 %
%                  Define:T1,T2,r1,r2,tau1,tau2,R1                %
%                                                                 %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for i=1:ap
   for j=1:ap
      if i==j
         T1(i,j)=-lambda(i);
      elseif (i+1)==j
```

國立政治大學圖書館 NATIONAL CHENGCHI UNIVERSITY LIBRARY

```
        T1(i,j)=(1-pro1(i))*lambda(i);
      else  T1(i,j)=0;
      end
   end
end

for i=1:sp
   for j=1:sp
      if i==j
         T2(i,j)=-mu(i);
      elseif (i+1)==j
         T2(i,j)=(1-pro2(i))*mu(i);
      else  T2(i,j)=0;
      end
   end
end

r1=-T1*ones(length(T1(:,1)),1);
r2=-T2*ones(length(T2(:,1)),1);

for i=1:length(T1(:,1))
    if i==1
        tau1(1)=1;
    else tau1(i)=0;
    end
end
for i=1:length(T2(:,1))
    if i==1
        tau2(1)=1;
    else tau2(i)=0;
    end
end
```

國立政治大學圖書館 NATIONAL CHENGCHI UNIVERSITY LIBRARY

```
for i=1:length(T1(:,1))
    for j=1:length(T1(:,1))
        if i==j
           R1(i,j)=r1(i);
        else  R1(i,j)=0;
        end
    end
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                  %
%          Constructing symbolic objective, and define             %
%       Laplace-transform of arrival(service) distribution         %
%                                                                  %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
syms x
f1=0;
for j=1:ap
   prod=1;
   for r=1:j-1
      prod=prod*(1-pro1(r))*lambda(r)/(x+lambda(r));
   end
   f1=f1+(pro1(j)*lambda(j)/(x+lambda(j)))*prod;
end

f2=0;
for j=1:sp
   prods=1;
   for r=1:j-1
      prods=prods*(1-pro2(r))*mu(r)/(-x+mu(r));
   end
   f2=f2+(pro2(j)*mu(j)/(-x+mu(j)))*prods;
end
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                 %
%                  If rho=1, we adjust the (2.11)                 %
%                                                                 %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if (rho==1)
  if ((no>0)&(ap>0)&(sp>0))
      y=f1*f2-1.000001;
      XX=double(solve(y));
  end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                 %
%           Pick out the roots which we need when rho=1           %
%                                                                 %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
X=XX;
else
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                 %
%                          Solve: f1*f2=1                         %
%                                                                 %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if ((no>0)&(ap>0)&(sp>0))
      y=f1*f2-1;
      XX=double(solve(y));
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                 %
%                 Pick out the roots which we need                %
%                                                                 %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
n=1;
for i=1:length(XX)
```

```
   if ((rho<1)&(real(XX(i)) > 0))
      X(n)=XX(i);
      n=n+1;
   end
   if ((rho>1)&(real(XX(i)) < 0))
      X(n)=XX(i);
      n=n+1;
   end
end
end
X;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                 %
%            Define: walpha,ualpha,valpha,W=w-alpha-n             %
%                                                                 %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for i=1:length(X)
   walpha(i)=0;
   for j=1:ap
      prod=1;
      for r=1:j-1
         prod=prod*(1-pro1(r))*lambda(r)/(X(i)+lambda(r));
      end
      walpha(i)=walpha(i)+(pro1(j)*lambda(j)/(X(i)+lambda(j)))*prod;
   end
   walpha(i);
end

for k=1:length(X)
ualpha(k,:)=(X(k)/(walpha(k)-1))*((tau1)*inv(T1-X(k)*eye(length(T1(1,:)))));
end

for k=1:length(X)
```

NATIONAL CHENGCHI UNIVERSITY LIBRARY

```
valpha(k,:)=(X(k)*walpha(k)/(walpha(k)-1))*(tau2)*inv(T2+X(k)*eye(length(T2(1,:))));
end

for i=1:length(X)
    for j=1:(no-1)
    W(i,:,j)=((walpha(i))^(j-1))*kron(ualpha(i,:),valpha(i,:));
    end
end


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                  %
%                   Define:A0,B0,C0,A,B,C,B1                       %
%                                                                  %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
A0=kron(r1*tau1,tau2);
B0=T1;
C0=kron(eye(length(T1(:,1))),r2);
A=kron(r1*tau1,eye(length(T2(:,1))));
B=kron(T1,eye(sp))+kron(eye(ap),T2);
C=kron(eye(length(T1(:,1))),r2*tau2);
B1=kron((T1+R1),eye(sp))+kron(eye(ap),T2);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                  %
%                   Define:the T matrix of p'T=b                   %
%                                                                  %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for i=1:(ap+length(X)+ap*sp)
    for j=1:(ap+3*ap*sp+1)
        if ((i<=ap) & (j<=ap))
            T(i,j)=B0(i,j);
        elseif ((i<=ap) & (j<=(ap+ap*sp)))
            T(i,j)=A0(i,(j-ap));
```

```
        elseif ((i<=ap)&(j<=(ap+3*ap*sp)))
            T(i,j)=0;
        elseif ((i<=ap)&(j==(ap+3*ap*sp+1)))
            T(i,j)=1;
        elseif ((i<=(ap+length(X)))&(j<=ap))
            T(i,1:ap)=W(i-ap,:,1)*C0;
        elseif ((i<=(ap+length(X)))&(j<=(ap+ap*sp)))
            T(i,(ap+1):(ap+ap*sp))=W(i-ap,:,1)*B+W(i-ap,:,2)*C;
        elseif ((i<=(ap+length(X)))&(j<=(ap+2*ap*sp)))
            T(i,(ap+ap*sp+1):(ap+2*ap*sp))=W(i-ap,:,no-2)*A+W(i-ap,:,no-1)*B;
        elseif ((i<=(ap+length(X)))&(j<=(ap+3*ap*sp)))
            T(i,(ap+2*ap*sp+1):(ap+3*ap*sp))=W(i-ap,:,no-1)*A;
        elseif ((i<=(ap+length(X)))&(j==(ap+3*ap*sp+1)))
            summ=0 ;
               for u=1:ap*sp
                  for v=1:(no-1)
                  summ=summ+W(i-ap,u,v);
                  end
               end
             T(i,j)=summ;
        elseif ((i<=(ap+length(X)+ap*sp))&(j<=(ap+ap*sp)))
             T(i,j)=0;
        elseif ((i<=(ap+length(X)+ap*sp))&(j<=(ap+2*ap*sp)))
             T(i,j)=C(i-ap-length(X),j-ap-ap*sp);
        elseif ((i<=(ap+length(X)+ap*sp))&(j<=(ap+3*ap*sp)))
             T(i,j)=B1(i-ap-length(X),j-ap-2*ap*sp);
        else   ((i<=(ap+length(X)+ap*sp))&(j==(ap+3*ap*sp+1)))
             T(i,j)=1;
        end
    end
end
T;
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                  %
%           Solve p'T=b by the Cholesky factorization              %
%                                                                  %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
U=T*T';
for i=1:(length(T(1,:)))
   if i==length(T(1,:))
      b(i)=1;
   else  b(i)=0;
   end
end
q=b*T';
R=chol(U);
z(1)=q(1)/R(1,1);
for i=2:length(R(:,1))
     sumsum=0;
     for j=1:(i-1)
     sumsum=sumsum+z(j)*R(j,i);
     end
     z(i)=(q(i)-sumsum)/R(i,i);
end
m=length(R(1,:));
c(m)=z(m)/R(m,m);
for k=1:(m-1)
    i=m-k;
    sumsums=0;
    for p=1:k
    j=m-p+1;
    sumsums=sumsums+c(j)*R(i,j);
    end
    c(i)=(z(i)-sumsums)/R(i,i);
end
```

國立政治大學圖書館 NATIONAL CHENGCHI UNIVERSITY LIBRARY

```
p=c';
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                  %
%         Obtain the unboundary stationary probabilities          %
%                                                                  %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for i=1:(no-1)
   ssum=0;
   for u=1:length(X)
      ssum=ssum+p(u+ap)*W(u,:,i);
   end
   P(i,:)=ssum;
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                  %
%             Compute the system-size probability pi              %
%                                                                  %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for i=1:(no+1)
   if i==1
     suum=0;
     for k=1:ap
        suum=suum+p(k);
     end
     pi(i)=suum;
   elseif i<(no+1)
     suuuum=0;
     for k=1:(ap*sp)
        suuuum=suuuum+P((i-1),k);
     end
     pi(i)=suuuum;
   else i==(no+1)
     suuum=0;
```

NATIONAL CHENGCHI UNIVERSITY LIBRARY

```
      for k=(ap+length(X)+1):(ap+length(X)+ap*sp)
         suuum=suuum+p(k);
      end
    pi(i)=suuum;
    end
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                 %
%    Solve: Lq,Ls,Ws,Wq, and print out Ls,Lq,Wq,Ws,p0,pN,pi       %
%                                                                 %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
summm=0;
for i=1:no
   summm=summm+(i-1)*pi(i+1);
end
Lq=summm;

ssummm=0;
for i=1:no
   ssummm=ssummm+(i)*pi(i+1);
end
Ls=ssummm;

Wq=Lq/(lambd*(1-pi(no+1)));
Ws=Ls/(lambd*(1-pi(no+1)));
fprintf('\n     The expected system size is %10.4f',Ls)
fprintf('\n     The expected queue size is %10.4f',Lq)
fprintf('\n     The expected time spent in queue is %10.4f',Wq)
fprintf('\n     The expected time spent in system is %10.4f',Ws)
fprintf('\n     The idle probability of system is %1.6f',pi(1))
fprintf('\n     The probability that a customer is lost = %1.6f',pi(no+1))
fprintf('\n     The stationary probability is')
pi
```

```
nn=10000;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                 %
%   The following section is the probability of each phase of n   %
%   in the system                                                 %
%                                                                 %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fprintf('\n\n The following section is the probability of each phase of n in the system.'
fprintf('\n    **********If you input n=10000,then it ends the program.**********')
      j=0;
      while(j<nn)
         i=input('\n\n please input n=');
         if ((i<0)|(i>10000))
            fprintf('\n  The data is error !')
         elseif(i<=no)
            if (i==0)
               pn=(p(1:ap))';
            elseif(i==no)
               pn=(p((ap+length(X)+1):(ap+length(X)+ap*sp)))';
            else
               pn=P(i,:);
            end
            fprintf('\n The probability of %d  in system is  ',i)
            pn
         else
            fprintf('\n The probability of %d  in system is 0 ',i)


         end
        if (i==nn)
            fprintf('\n\n ************End.***************')
            break
        end
        j=i;
```

```
    end
else
  fprintf('\n      Your datas are error !')
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                    %
%                 Choose "continuous" or "end"                       %
%                                                                    %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
  fprintf('\n\n  The Ck/Cm/1/N problem is  (1)continuous (0)end'  )
  n=input('\n\n        Please choose:');
  if(n==0)
      fprintf('\n *******The Ck/Cm/1/N  problem is end!**********')
      break
  end
end
```

國立政治大學圖書館 NATIONAL CHENGCHI UNIVERSITY LIBRARY