clc;

delete('Trace\_SIR.xlsx');

%We need to repeat 700times

counter = 1;

%We have to ignore warm Up time data

Warm\_Up = 250;

while counter<=700

%Servers' situations

S = zeros(5,1);

%Num of messages serviced by servers

M = zeros(5,1);

%Idle time from last service

I = zeros(5,1);

%Total idle time

Idle = zeros(5,1);

%Total idle time for output

Idle\_out = zeros(5,1);

%Simulation Clock

clock = 0;

%time of previous event

P\_clock = 0;

%Departures related to each server

%If a server is idle his/her departure is inf

%Servicing meesage Departure

Departure = inf(5,1);

%Last message departure

L\_Departure = zeros(5,1);

%Mean of service times

M\_s = [6;8;10;12;14];

%Mean of arrival times

M\_a = 2.8;

%Arrival Time

Arr = 0;

%How many messages did we have till this time

Num = 0;

%Which message relates to ith server

SM = zeros(5,1);

%End of simulation

E = 1000;

%Counter for trace rows

n = 1;

%Necessary Matrices for trace :

%CLOCK

%Event

%Num\_m

%lose

%Chosen\_Server

%L1(t)

%L2(t)

%L3(t)

%L4(t)

%L5(t)

%FEL\_D1

%FEL\_D2

%FEL\_D3

%FEL\_D4

%FEL\_D5

%FEL\_A

%FEL\_E

%U1(t)

%U2(t)

%U3(t)

%U4(t)

%U5(t)

%I1(t)

%I2(t)

%I3(t)

%I4(t)

%I5(t)

clock = expo(M\_a);

Event = "Arrival";

while clock<=E

if counter == 1

%Creating trace

CLOCK(n,1) = clock;

EVENT(n,1) = Event;

end

if Event == "Arrival"

%Increase Num of Messages

Num = Num+1;

%Find availabe servers

Ava = find(S==0);

%If there is no available server, increase lost messages

if isempty(Ava)== 1

if counter ==1

lose(n,1) = "Lost";

Chosen\_Server(n,1) = "-";

end

%Generate next arrival time

Arr = clock + expo(M\_a);

%Update clock

P\_clock = clock;

min\_d = min(min(Departure));

if min\_d<= Arr

Event = "Departure";

clock = min\_d;

else

Event = "Arrival";

clock = Arr;

end

%If we have available servers

else

%Idle times from last service

I(Ava,1) = clock - L\_Departure(Ava,1);

%Choose one of the servers

%Choose min of idle times of available servers

min\_I = min(min(I(Ava,1)));

Chosen\_s = find(I==min\_I);

S(Chosen\_s(1,1),1) = 1;

SM(Chosen\_s(1,1),1) = Num;

%Servers' total idle time

Idle(Ava,1) = Idle(Ava,1)+ (clock - P\_clock);

if clock>=Warm\_Up

%Incerase serever's messages

M(Chosen\_s(1,1),1) = M(Chosen\_s(1,1),1)+1;

%Servers' total idle time

Idle\_out(Ava,1) = Idle\_out(Ava,1)+ (clock - P\_clock);

end

%Generate service time & departure

Service\_time = expo(M\_s(Chosen\_s(1,1),1));

Departure(Chosen\_s(1,1),1) = clock+Service\_time;

%Generate next arrival time

Arr = clock + expo(M\_a);

%Update clock

P\_clock = clock;

min\_d = min(min(Departure));

if min\_d<= Arr

Event = "Departure";

clock = min\_d;

else

Event = "Arrival";

clock = Arr;

end

if counter ==1

lose(n,1) = "Received";

Chosen\_Server(n,1) = Chosen\_s(1,1);

end

end

if counter == 1

%Update trace's matrices

Num\_m(n,1) = Num;

FEL\_D1(n,1)= Departure(1,1);

FEL\_D2(n,1)= Departure(2,1);

FEL\_D3(n,1)= Departure(3,1);

FEL\_D4(n,1)= Departure(4,1);

FEL\_D5(n,1)= Departure(5,1);

FEL\_E(n,1) = 1000;

FEL\_A(n,1) = Arr;

L1(n,1) = S(1,1);

L2(n,1) = S(2,1);

L3(n,1) = S(3,1);

L4(n,1) = S(4,1);

L5(n,1) = S(5,1);

U1(n,1) = M(1,1)/sum(M,'all');

U2(n,1) = M(2,1)/sum(M,'all');

U3(n,1) = M(3,1)/sum(M,'all');

U4(n,1) = M(4,1)/sum(M,'all');

U5(n,1) = M(5,1)/sum(M,'all');

I1(n,1) = Idle\_out(1,1);

I2(n,1) = Idle\_out(2,1);

I3(n,1) = Idle\_out(3,1);

I4(n,1) = Idle\_out(4,1);

I5(n,1) = Idle\_out(5,1);

end

elseif Event == "Departure"

%Find availabe servers

Ava = find(S==0);

%Servers' total idle time

Idle(Ava,1) = Idle(Ava,1)+ (clock - P\_clock);

if clock>=Warm\_Up

%Servers' total idle time

Idle\_out(Ava,1) = Idle\_out(Ava,1)+ (clock - P\_clock);

end

%Find server

server\_num = find(Departure==clock);

S(server\_num(1,1),1) = 0;

L\_Departure(server\_num(1,1),1) = Departure(server\_num(1,1),1);

Departure(server\_num(1,1),1) = inf;

%Update clock

P\_clock = clock;

min\_d = min(min(Departure));

if min\_d<= Arr

Event = "Departure";

clock = min\_d;

else

Event = "Arrival";

clock = Arr;

end

if counter == 1

%Update trace's matrices

Num\_m(n,1) = SM(server\_num,1);

FEL\_D1(n,1)= Departure(1,1);

FEL\_D2(n,1)= Departure(2,1);

FEL\_D3(n,1)= Departure(3,1);

FEL\_D4(n,1)= Departure(4,1);

FEL\_D5(n,1)= Departure(5,1);

FEL\_E(n,1) = 1000;

FEL\_A(n,1) = Arr;

Chosen\_Server(n,1) = "-";

lose(n,1) = "-";

L1(n,1) = S(1,1);

L2(n,1) = S(2,1);

L3(n,1) = S(3,1);

L4(n,1) = S(4,1);

L5(n,1) = S(5,1);

U1(n,1) = M(1,1)/sum(M,'all');

U2(n,1) = M(2,1)/sum(M,'all');

U3(n,1) = M(3,1)/sum(M,'all');

U4(n,1) = M(4,1)/sum(M,'all');

U5(n,1) = M(5,1)/sum(M,'all');

I1(n,1) = Idle\_out(1,1);

I2(n,1) = Idle\_out(2,1);

I3(n,1) = Idle\_out(3,1);

I4(n,1) = Idle\_out(4,1);

I5(n,1) = Idle\_out(5,1);

end

end

if counter == 1

%Going to next row in trace

n = n+1;

end

end

if counter == 1

CLOCK(n,1) = E;

EVENT(n,1) = "End";

Num\_m(n,1) = "-";

Chosen\_Server(n,1) = "-";

L1(n,1) = "-";

L2(n,1) = "-";

L3(n,1) = "-";

L4(n,1) = "-";

L5(n,1) = "-";

lose(n,1) = "-";

FEL\_D1(n,1) = "-";

FEL\_D2(n,1) = "-";

FEL\_D3(n,1) = "-";

FEL\_D4(n,1) = "-";

FEL\_D5(n,1) = "-";

FEL\_E(n,1) = "-";

FEL\_A(n,1) = "-";

U1(n,1) = U1(n-1,1);

U2(n,1) = U2(n-1,1);

U3(n,1) = U3(n-1,1);

U4(n,1) = U4(n-1,1);

U5(n,1) = U5(n-1,1);

I1(n,1) = I1(n-1,1);

I2(n,1) = I2(n-1,1);

I3(n,1) = I3(n-1,1);

I4(n,1) = I4(n-1,1);

I5(n,1) = I5(n-1,1);

T1 = table(CLOCK,EVENT,Num\_m,Chosen\_Server,L1,L2,L3,L4,L5,lose,FEL\_D1,FEL\_D2,FEL\_D3,FEL\_D4,FEL\_D5,FEL\_A,FEL\_E,U1,U2,U3,U4,U5,I1,I2,I3,I4,I5);

fileName='Trace\_SIR.xlsx';

writetable(T1,fileName);

winopen(fileName);

end

Ser\_u = M/sum(M,'all');

utilization(:,counter) = Ser\_u;

Idleness(:,counter) = Idle\_out;

counter = counter+1;

end

Title = ["Min";"Max";"Mean";"Std"];

i = 1;

while i<=5

U\_Server(1,i) = min(min(utilization(i,:)));

U\_Server(2,i) = max(max(utilization(i,:)));

U\_Server(3,i) = mean(utilization(i,:));

U\_Server(4,i) = sqrt(var(utilization(i,:)));

i = i+1;

end

i = 1;

while i<=5

I\_Server(1,i) = min(min(Idleness(i,:)));

I\_Server(2,i) = max(max(Idleness(i,:)));

I\_Server(3,i) = mean(Idleness(i,:));

I\_Server(4,i) = sqrt(var(Idleness(i,:)));

i = i+1;

end

seperator = ["\*";"\*";"\*";"\*"];

T2 = table(Title,U\_Server,seperator,I\_Server);

fileNamee = 'IU\_SIR.xlsx';

writetable(T2,fileNamee);

winopen(fileNamee);

function e = expo(beta)

%beta = mean of distribution

e = (-1\*log(rand(1))\*beta);

end