

Universidad Politécnica de Cartagena



Escuela Técnica Superior de Ingeniería de Telecomunicación

PRÁCTICAS DE MODELADO Y SIMULACIÓN

BOLETÍN DE ENTREGA

Trabajo simulación: Redes de colas

INTEGRANTES DEL GRUPO:

NOMBRE Y APELLIDOS	CORREO ELECTRÓNICO
Manuel Alejandro De la Torre Ruzafa	manutower9@gmail.com
Jairo Peña Iglesias	jairopi91@gmail.com

1. Proporcione el código completo del simulador de redes de colas (4 puntos) (+1 punto si implementa el cálculo de R)

```
%% ESQUELETO DE SIMULACION

listaEV = []; % Lista vacia al comienzo
% PARAMETROS DE SIMULACION
% Parametros de comprobacion (descomentar para ejecutar)
% Tiempo medio T=0.35
% A= [0.2 0.2 0.2 0.2 0.2; 0.2 0.2 0.2 0.2 0.2; 0.2 0.2 0.2 0.2 0.2; 0.2 0.2 0.2 0.2 0.2];
% tipox= [2 2 2 2]; % Exponencial
% tipos= [2 2 2 2]; % Exponencial
% param1x = [5 0 0 0];
% param2x = [0 0 0 0];
% param1s = [15 15 15 15];
% param2s = [0 0 0 0];
% k= [2 2 2 2];

% Primera prueba
A = [0.1 0.3 0.1 0.3 0.2; 0 0.2 0.3 0.3 0.2; 0.1 0.2 0.4 0 0.3; 0 0.5 0 0.1 0.4; 0 0 0 0 1];
tipox = 1;
tipos= [1 1 2 3];
param1x = 50;
param2x = 80;
param1s = [3 5 1 5];
param2s = [5 10 0 0];
k= [2 2 1 4];

% Segunda prueba
% p = 0.1;
% A = [0 1 0 0 0 0; 0 0 1 0 0 0; 0 0 0 1 0 0; 0 0 0 0 1 0; 1-p 0 0 0 0 p; 0 0 0 0 0 1];
% tipox = 2;
% tipos= [2 2 2 2 2];
% param1x = 2;
% param2x = 0;
% param1s = [5 5 5 5 5];
% param2s = [0 0 0 0 0];
% k= [5 5 5 5 5];

t_simulacion = 0.0; % Reloj de simulación
unomenosalfa = 0;
delta = 100;
cont = 0;
contb = 0;
contN = 0;
R = 0;
contr = 0;
H=10000;
pasos = 0;
z = 1;
cola = 1.0;
tcola = 0;

% ACCIONES DE INCIO: p.ej. definir estado, generar primeros eventos
numcolas = length(A)-1
X=[];
X2=[];
aux=1;

% Se proporciona ejemplo del
% Caso cola de trabajos

% TIPOS DE EVENTOS, CADA UNO UN NUMERO DIFERENTE
SALE = 0;
LLEGA = 1;
```

```

EVENTO = 2;

% ESTADO
N = zeros(1,numcolas);
fifoTiempos = cell(numcolas,1);
tllegada = 0;

% VARIABLES PARA EL CALCULO DE LOS PROMEDIOS DE INTERES
summuestrasT = 0;
nummuestrasT = 0;
summuestrasT2 = 0;
sumN = 0;
sumR = 0;
tolerativa=0.05;
calidad=0.95;

% PRIMEROS EVENTOS
[z, muestra] = aleatorio(z,tipox(cola),param1x(cola),param2x(cola));
listaEV = encolarEvento(listaEV, muestra , LLEGA, -1,cola);

[z, muestra] = aleatorio(1,tipox(1), param1x(1), param2x(1));
listaEV = encolarEvento(listaEV, muestra, EVENTO, 0, 1);

while true

    pasos = pasos + 1;
    % Salto al siguiente evento
    [listaEV, tiempo, tipo,tllegada,cola] = sgteEvento(listaEV);

    % Actualizamos el tiempo
    t_simulacion = tiempo;

    switch tipo
        case LLEGA
            N(cola) = N(cola)+1;
            sumR = sumR +1;
            if(tllegada <0)
                [z,muestra] = aleatorio(z,tipox(cola),param1x(cola),param2x(cola));
                listaEV = encolarEvento(listaEV, t_simulacion + muestra, LLEGA,-
1,cola);
                tllegada = t_simulacion;
            end
            if N(cola) <= k(cola)
                [z,muestra] = aleatorio(z,tipos(cola),param1s(cola),param2s(cola)); %
Tiempo en el recurso
                listaEV = encolarEvento(listaEV, t_simulacion + muestra, SALE,
tllegada,cola);
            else
                fifoTiempos{cola} = pushFIFO(fifoTiempos{cola},tllegada);
            end
        case SALE
            N(cola) = N(cola)-1;
            if N(cola)>=k(cola) % Otro trabajo pasa a ocupar el "procesador"
                [z,muestra] = aleatorio(z,tipos(cola),param1s(cola),param2s(cola)); %
Tiempo en el recurso
                [fifoTiempos{cola}, tcola] = popFIFO(fifoTiempos{cola});
                listaEV = encolarEvento(listaEV, t_simulacion + muestra, SALE, tcola,
cola);
            end
            [z, salida] = SalidaAleatoria(z, A(cola,:) );
            if(salida > numcolas)
                contr = contr + 1;
                if(contr < H)
                    cont = cont + 1;
                else
                    summuestrasT = summuestrasT + (t_simulacion - tllegada);
                    nummuestrasT = nummuestrasT + 1;
                    summuestrasT2= summuestrasT2 +(t_simulacion-tllegada)^2;
                end
            end
        end
    end
end

```

```

        contb = contb + 1;
        if(contb == delta*aux)
            nummuestrasT;
            X(aux) = summuestrasT / nummuestrasT;
            X2(aux) = summuestrasT2 / nummuestrasT;
            aux = aux+1;
            summuestrasT2 = 0;
            summuestrasT = 0;
            nummuestrasT = 0;
            summ = sum(X);
            summ2 = sum(X2);
            [unomenosalfa, iizqda, iderecha]=
Calidad(tolrelativa,length(X),summ,summ2);
            Ntotal = sumN/contN;
            R = sumR / contr;
        end
    end
else
    listaEV = encolarEvento(listaEV, t_simulacion, LLEGA, tllegada,salida);
end
case EVENTO
    sumN = sumN + sum(N);
    contN = contN+1;
    [z, muestra] = aleatorio(z,tipox(cola), param1x(cola), param2x(cola));
    listaEV = encolarEvento(listaEV, t_simulacion + muestra, EVENTO,
tllegada, cola);
    if(unomenosalfa >= calidad) break;
end
end
end

% Mostramos los promedios calculados
display('FIN DE LA SIMULACION');
Ttotal = summ/length(X)
pasos
unomenosalfa
iizqda
iderecha
Ntotal
R

```

2. Indique los resultados obtenidos para la configuración dada en la primera prueba (2.5 puntos)

Ttotal =

17.8899

pasos =

1069040

unomenosalfa =

0.9501

iizqda =

16.9954

iderecha =

18.7843

Ntotal =

0.2746

R =

3.8630

3. Proporcione una tabla con los resultados en función de la probabilidad “p” para la configuración del anillo dada en la segunda prueba (2.5 puntos)

p	0	0.1	0.2	0.3	0.4	0.5
Ttotal	-	15.9399	5.0926	3.3473	2.5041	2.0099
Ntotal	-	32.0203	10.1535	6.6658	4.9798	4.0229
R	-	50.1959	24.9512	16.6664	12.4984	10.0421