**Universidad Politécnica de Cartagena**

**Escuela Técnica Superior de Ingeniería de Telecomunicación**

**PRÁCTICAS DE MODELADO Y SIMULACIÓN**

# BOLETÍN DE ENTREGA

# TRABAJO SIMULACIÓN

## INTEGRANTES DEL GRUPO:

|  |  |
| --- | --- |
| NOMBRE Y APELLIDOS | CORREO ELECTRÓNICO |
| Diego Ismael Antolinos García | [diego.antolinos@edu.upct.es](mailto:diego.antolinos@edu.upct.es) |
| Andrés Ruz Nieto | andres.ruz@edu.upct.es |

**Información y metodología de evaluación:**

## Este trabajo tiene un peso total del 15% en la evaluación de la asignatura

1. El boletín debe rellenarse y convertirse a **formato PDF** para su envío a través de aula virtual, en las fechas dispuestas en el calendario
2. La evaluación se basará en las **respuestas del boletín** y también se contempla la posibilidad de realizar **entrevistas online** individuales para verificar la autoría de las mismas.
3. **Proporcione el modelo completo de simulación: estado, eventos, pseudocódigos, etc. (5 puntos)**
4. **Estado**

Nuestro estado está formado por las siguientes variables:

* numpaquete: Indica el último paquete que se ha generado.
* numpaquetes\_entregados: Indica el número del último paquete entregado
* n\_paquetes y t\_paquetes: Contiene el número y tiempo de generación de los paquetes que se han recibido pero no se han podido entregar por falta de paquetes anteriores.
* numeropaqueteenviado: Número del paquete que se envía.

1. **Eventos**

Nuestro programa está formado por dos eventos TX y RX.

Los eventos TX y RX tendrán los siguientes parámetros adicionales:

* Tiempo en el que ha generado el paquete. (*t\_simulacion + tsiguientetx*)
* Número del paquete.
* Tiempo de retardo en red (en el caso de TX será 0).

1. **Pseudo-código**

* **Evento TX**

Definimos una variable para poder calcular el retardo total en la red (*tiempodeenvio*)

Generaremos un tiempo de retardo en red aleatorio, si este es mayor que el tiempo máximo de retardo en red, añadiremos el valor del tiempo máximo de retardo en red a *tiempodeenvio.* Y volveremos a generar otro tiempo.Este proceso lo realizaremos tantas veces como sea necesario hasta que el tiempo generado sea menor que el tiempo máximo de retardo en red. Este tiempo también lo sumaremos a nuestra variable *tiempodeenvio.* Podemos ver una representación gráfica en el anexo.

Posteriormente, ya podremos generar el evento de recepción.

A continuación, generamos un evento de transmisión con un tiempo aleatorio.

* **Evento RX**

Cuando recibimos un paquete, guardamos su número y tiempo de generación en un array. A través de una variable que cuenta el número de paquetes que han sido entregados, podremos realizar la comprobación de si un paquete que recibimos puede ser entregado o debe ser guardado.

Si esta variable es igual al número del paquete que acabamos de recibir, buscaremos el índice de este número en nuestros arrays y eliminaremos los datos de este paquete. Después buscaremos los paquetes consecutivos que se encuentran guardados y que ya pueden ser entregados, para ello nuestra variable que cuenta el número de paquetes que han sido entregados irá incrementándose con la entrega de cada paquete, cuando esta no coincida con el siguiente número consecutivo n-ésimo a entregar dejará de procesar paquetes. Los datos del paquete que se va entregando son eliminados de nuestros arrays, antes de ello obtendremos la muestra de dicho paquete que contendrá la diferencia del tiempo actual y el tiempo en el que se creó el paquete.

1. **Proporcione el código MATLAB del simulador (3 puntos)**

%% ESQUELETO DE SIMULACION

listaEV = []; % Lista vacia al comienzo

t\_simulacion = 0.0; % Reloj de simulación

% ACCIONES DE INCIO: p.ej. definir estado, generar primeros eventos

% Se proporciona ejemplo del

% Caso cola de trabajos

t\_muestra = 1; %Seg de cada muestra

% TIPOS DE EVENTOS, CADA UNO UN NUMERO DIFERENTE

TX = 0;

RX = 1;

% ESTADO

N = 0;

% PARAMETROS DE SIMULACION

%1 -> Fishman-Moore

%2 -> Kobayashi

%3 -> Coveyou-McPherson

%4 -> glibc

%5 -> MMIX

generadorZ = 2;

Z = 1;

%VALIDACION PROFESOR

calidadobjetivo = 0.95; % Suele estar entre el 90% y el 99%

tolrelativa = 0.01; % Suele estar entre el 1% y el 10%

TEST = 100; % Cada cuantas muestras comprobamos calidad

H = 50000;

D = 10;

tipoX = 2;

param1X = 1;

param2X = 0;

tipoE = 2;

param1E = 0.5;

param2E = 0;

Tmax = 1.5;

%ESCENARIO 1

% calidadobjetivo = 0.95; % Suele estar entre el 90% y el 99%

% tolrelativa = 0.01; % Suele estar entre el 1% y el 10%

% TEST = 100; % Cada cuantas muestras comprobamos calidad

% H = 50000;

% D = 10;

%

% tipoX = 4;

% param1X = 0.3;

% param2X = 0;

%

% tipoE = 1;

% param1E = 0;

% param2E = 1;

%

% Tmax = 0.8;

%ESCENARIO 2

% calidadobjetivo = 0.99; % Suele estar entre el 90% y el 99%

% tolrelativa = 0.01; % Suele estar entre el 1% y el 10%

% TEST = 100; % Cada cuantas muestras comprobamos calidad

% H = 50000;

% D = 100;

% tipoX = 2;

% param1X = 100;

% param2X = 0;

% tipoE = 1;

% param1E = 0;

% param2E = 1;

% Tmax = 0.5;

summuestrasT = 0;

muestrasT = 0;

summuestrasTcuadrado = 0;

nummuestrasT\_bloque = 0;

summuestrasT\_bloque = 0;

nt = 0;

numpaquete = 0; % Parte del estado

numpaquetes\_entregados = 0; % Parte del estado

n\_paquetes = []; % Parte del estado

t\_paquetes = []; % Parte del estado

% PRIMEROS EVENTOS

[Z,taux] = aleatorio(Z,tipoX,param1X,param2X,generadorZ);

listaEV = encolarEvento(listaEV, taux, TX, taux, numpaquete,0);

numpaquete = numpaquete + 1;

pasos = 0;

acabo = false;

while true

pasos = pasos + 1;

[listaEV, tiempo, tipo, tgeneracionpaquete, numeropaqueteenviado, taire] = sgteEvento(listaEV);

% Actualizamos el tiempo

t\_simulacion = tiempo;

switch tipo

case TX

tiempodeenvio = 0; % Parte del estado

[Z,taux] = aleatorio(Z,tipoE,param1E,param2E,generadorZ); % Tiempo de envio

tiempodeenvio = tiempodeenvio + taux;

while taux > Tmax

tiempodeenvio = tiempodeenvio - taux;

tiempodeenvio = tiempodeenvio + Tmax;

[Z,taux] = aleatorio(Z,tipoE,param1E,param2E,generadorZ); % Tiempo de envio

tiempodeenvio = tiempodeenvio + taux;

end

listaEV = encolarEvento(listaEV, t\_simulacion + tiempodeenvio, RX, tgeneracionpaquete, numeropaqueteenviado, tiempodeenvio);

[Z,tsiguientetx] = aleatorio(Z,tipoX,param1X,param2X,generadorZ);

listaEV = encolarEvento(listaEV , t\_simulacion + tsiguientetx, TX , t\_simulacion + tsiguientetx, numpaquete , 0);

numpaquete = numpaquete + 1;

case RX

n\_paquetes = [n\_paquetes, numeropaqueteenviado];

t\_paquetes = [t\_paquetes, tgeneracionpaquete];

if numpaquetes\_entregados == numeropaqueteenviado

indice = find (n\_paquetes == numeropaqueteenviado);

while ~isempty(indice)

nt = nt + 1;

tiempoairepaquete\_num = t\_paquetes(indice);

if nt > H

nummuestrasT\_bloque = nummuestrasT\_bloque + 1;

if nummuestrasT\_bloque == D

summuestrasT = summuestrasT + (t\_simulacion - tiempoairepaquete\_num);

muestrasT = muestrasT + 1;

summuestrasTcuadrado = summuestrasTcuadrado + (t\_simulacion - tiempoairepaquete\_num)^2;

nummuestrasT\_bloque = 0;

summuestrasT\_bloque = 0;

if ~mod(muestrasT,TEST)

[unomenosalfa, intizqda, intderecha] = calidad(tolrelativa, muestrasT, summuestrasT, summuestrasTcuadrado);

if(unomenosalfa >= calidadobjetivo)

acabo = true;

break;

end

end

end

end

n\_paquetes(indice) = [];

t\_paquetes(indice) = [];

numpaquetes\_entregados = numpaquetes\_entregados + 1; %Aumentamos para encontrar paquetes consecutivos

indice = find (n\_paquetes == numpaquetes\_entregados);

end

if acabo

break;

end

end

end

end

% Mostramos los promedios calculados

disp('####################');

disp('FIN DE LA SIMULACION');

disp('####################');

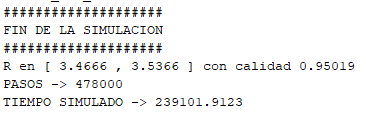
disp(['R en [ ',num2str(intizqda),' , ',num2str(intderecha),' ] con calidad ',num2str(unomenosalfa)]);

disp(['PASOS -> ',num2str(pasos)]);

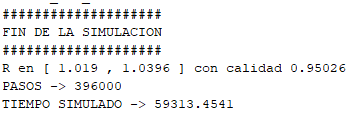
disp(['TIEMPO SIMULADO -> ',num2str(t\_simulacion)]);

1. **Proporcione los resultados en los escenarios solicitados (2 puntos)**

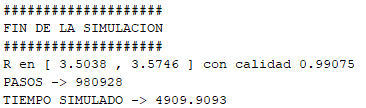
CONFIGURACION PROFESOR

****

CONFIGURACION 1



CONFIGURACION 2



1. **ANEXO**

**Imagen que contiene texto, mapa

Descripción generada automáticamente**

1. tiempoenvio = 6
2. tiempoenvio – 6 + 5 = 5
3. tiempoenvio + 7 = 12
4. tiempoenvio – 7 + 5 = 10
5. tiempoenvio + 3 = 13

Este procedimiento se encuentra implementado en nuestro código mediante un bucle while en el que se compara *taux* y *tmax*.