Escuela Técnica Superior Ingeniería de Telecomunicación

TRABAJO - MODELADO

Diego Ismael Antolinos García
Andrés Ruz Nieto

1. Código empleado

```
clc;
clear;

% Configuracion de parametros

t_prop = 0.001; %tiempo de propagacion en s
bits_fichero = 10000; % numero de bits a transmitir
R = 500000; %tasa de transmision en bits/s
p_tx_dB = 10; %potencia de transmisiÃ³n en dBm
p_tx = 10^(p_tx_dB/10)*0.001; %potencia de transmisiÃ³n en W

P_b = ProbErrorBit(R, p_tx);

Eb = 1000*p_tx/R; % energia por bit transmitido enmilijulios/bit

carga_min = 100;
carga_max = 4000;
carga_incremento = 100;

cargas = carga_min:carga_incremento:carga_max;

% ASIGNE UN VALOR A LA CABECERA DEL PAQUETE ENTRE 20 Y 120
L_cabecera = 20; % longitud cabecera en bits

T = zeros(1,length(cargas));
E = zeros(1,length(cargas));
index = 0;

for L_carga = cargas

    index = index + 1;

    % %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    % %%%%%%%% AÑADA SU CODIGO %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    n = ceil(bits_fichero/L_carga); % numero de paquetes dado
bits_fichero y L_carga
    entero = 1;
    if(mod(bits_fichero,L_carga)~=0)
        entero = 0;
    end

    % calculo de la longitud de los paquetes
    L = L_carga + L_cabecera;
    if(entero == 0)
        L_ult = mod(bits_fichero,L_carga) + L_cabecera;
    end

    % calculo de las probabilidades de error
    prob_error = 1-(1-P_b)^L;
    if(entero == 0)
        prob_error_ultimo = 1-(1-P_b)^L_ult;
    end
```

```
    % generar P
    P = zeros(n+1);
    for i = 1:n
        P(i,i)=prob_error;
        P(i,i+1)=(1-prob_error);
    end
    if(entero == 0)
        P(n,n) = prob_error_ultimo;
        P(n,n+1) = 1-prob_error_ultimo;
    end
    P(end,end) = 1;
    M = P(1:n,1:n);

    % generar g_time
    g_time = zeros(n,1);
    for i = 1:n
       g_time(i)= (L/R) + (2*t_prop);
    end
       if(entero == 0)
           g_time(n)= (L_ult/R) + (2*t_prop);
       end

    % generar g_energy
    g_energy = zeros(n,1);
    for i = 1:n
       g_energy(i)= L*Eb;
    end
       if(entero == 0)
           g_energy(n)= L_ult*Eb;
       end

    % obtener v_time
    v_time = (eye(n)-M)\g_time;

    % obtener v_energy
    v_energy = (eye(n)-M)\g_energy;

    T(index) = v_time(1);

    E(index) = v_energy(1);

end

% obtener el valor de L_carga que minimiza T

[valot_T, indice_T] = min(T);
L_carga_T_min = cargas(indice_T)

% obtener el valor de L_carga que minimiza E
[valot_E, indice_E] = min(E);
L_carga_E_min = cargas(indice_E)

plot(cargas,T);
figure;
plot(cargas,E);
```

2. Código empleado en la ampliación

```
clc;
clear;

% Configuracion de parametros

t_prop = 0.001; %tiempo de propagacion en s
bits_fichero = 10000; % numero de bits a transmitir
p_tx_dB = 10; %potencia de transmisiÃ³n en dBm
p_tx = 10^(p_tx_dB/10)*0.001; %potencia de transmisiÃ³n en W

carga_min = 100;
carga_max = 4000;
carga_incremento = 100;

cargas = carga_min:carga_incremento:carga_max;
Rvector = 100000:10000:600000;

% ASIGNE UN VALOR A LA CABECERA DEL PAQUETE ENTRE 20 Y 120
L_cabecera = 20; % longitud cabecera en bits

T = zeros(length(cargas),length(Rvector));
E = zeros(length(cargas),length(Rvector));
index1 = 0;
index2 = 0;

for L_carga = cargas
    index1 = index1 + 1;
    index2 = 0;
    for R = Rvector

        P_b = ProbErrorBit(R, p_tx);
        Eb = 1000*p_tx/R; % energia por bit transmitido en
milijulios/bit

        index2 = index2 + 1;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %%%%%%% AÑADA SU CODIGO %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        n = ceil(bits_fichero/L_carga); % numero de paquetes
dado bits_fichero y L_carga
        entero = 1;
        if(mod(bits_fichero,L_carga)~=0)
            entero = 0;
        end
        % calculo de la longitud de los paquetes
        L = L_carga + L_cabecera;
        if(entero == 0)
            L_ult = mod(bits_fichero,L_carga) + L_cabecera;
        end

        % calculo de las probabilidades de error
        prob_error = 1-(1-P_b)^L;
        if(entero == 0)
            prob_error_ultimo = 1-(1-P_b)^L_ult;
        end
```

```
        % generar P
        P = zeros(n+1);
        for i = 1:n
            P(i,i)=prob_error;
            P(i,i+1)=(1-prob_error);
        end
        if(entero == 0)
            P(n,n) = prob_error_ultimo;
            P(n,n+1) = 1-prob_error_ultimo;
        end
        P(end,end) = 1;
        M = P(1:n,1:n);

        % generar g_time
        g_time = zeros(n,1);
        for i = 1:n
           g_time(i)= (L/R) + (2*t_prop);
        end
           if(entero == 0)
               g_time(n)= (L_ult/R) + (2*t_prop);
           end

        % generar g_energy
        g_energy = zeros(n,1);
        for i = 1:n
           g_energy(i)= L*Eb;
        end
           if(entero == 0)
               g_energy(n)= L_ult*Eb;
           end

        % obtener v_time
        v_time = (eye(n)-M)\g_time;

        % obtener v_energy
        v_energy = (eye(n)-M)\g_energy;

        T(index1,index2) = v_time(1);

        E(index1,index2) = v_energy(1);
    end
end

% obtener el valor de L_carga que minimiza T
min_elemento_T = min(T(:));
[L_carga_T, R_T] = find(T==min_elemento_T);
L_carga_T_min = cargas(L_carga_T)
R_T_min = Rvector(R_T)

% obtener el valor de L_carga que minimiza E
min_elemento_E = min(E(:));
[L_carga_E, R_E] = find(E==min_elemento_E);
L_carga_E_min = cargas(L_carga_E)
R_E_min = Rvector(R_E)

figure
mesh(Rvector,cargas,T);
figure;
mesh(Rvector,cargas,E);
```

3. Datos obtenidos

a. Parte I

$$L_{carga}\ que\ minimiza\ T \rightarrow 1700\ bits$$

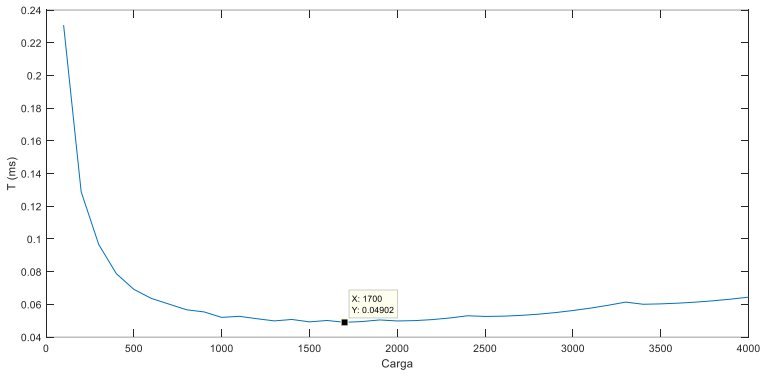
$$L_{carga}\ que\ minimiza\ E \rightarrow 300\ bits$$

b. Parte II (ampliación)

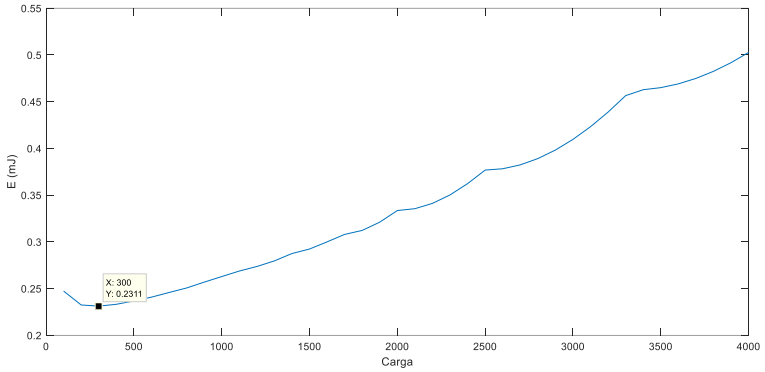
$$L_{carga}\ y\ R\ que\ minimiza\ T \rightarrow 3400\ bits\ a\ 370000\ bits/s$$

$$L_{carga}\ y\ R\ que\ minimiza\ E \rightarrow 200\ bits\ a\ 600000\frac{bits}{s}$$

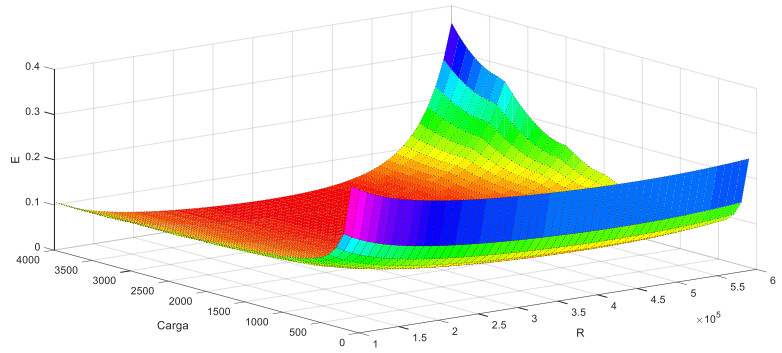
4. Gráficas obtenidas



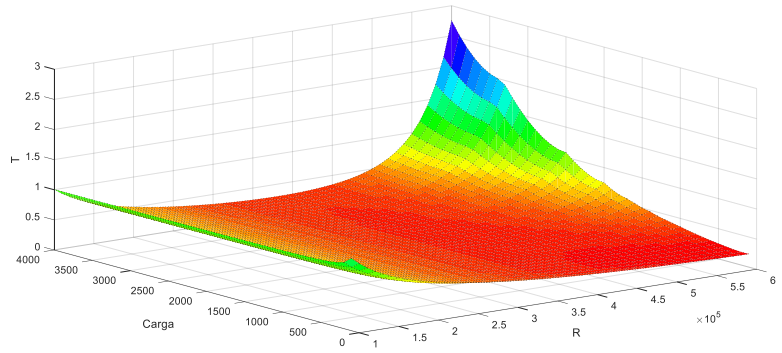
Tiempo de envío respecto la carga del paquete



Energía consumida respecto la carga del paquete



Energía respecto la carga del paquete y su velocidad de transmisión



Tiempo de envío respecto la carga del paquete y su velocidad de transmisión