

Apellidos, Nombre: García Martínez, Máximo
Apellidos, Nombre:

1.Planteamiento y órbita inicial:

Adjuntar código de apolo().

```
function sp = apolo(t,s)
    % s = vector estado con posición y velocidad s(1:2) y velocidad s(7:8) de la nave
    %      (12x1)      posición s(3:4) y velocidad s(9:10) de la LUNA
    %                  posición s(5:6) y velocidad s(11:12) de la TIERRA

    global motor

    GM1 = 3.99e+005; GM2 = 4.90e+003;

    % Extraer del vector de entrada s la posición y velocidad de la nave
    rn=s(1:2); rL=s(3:4); rT=s(5:6);
    vn=s(7:8); vL=s(9:10); vT=s(11:12);

    % Vector posición tierra y luna respecto a nave.
    r1 = rn - rT; r2 = rn - rL;
    % distancia entre luna y tierra
    rTL = rL - rT;

    % Establecemos velocidades
    sp(1:6) = s(7:12);

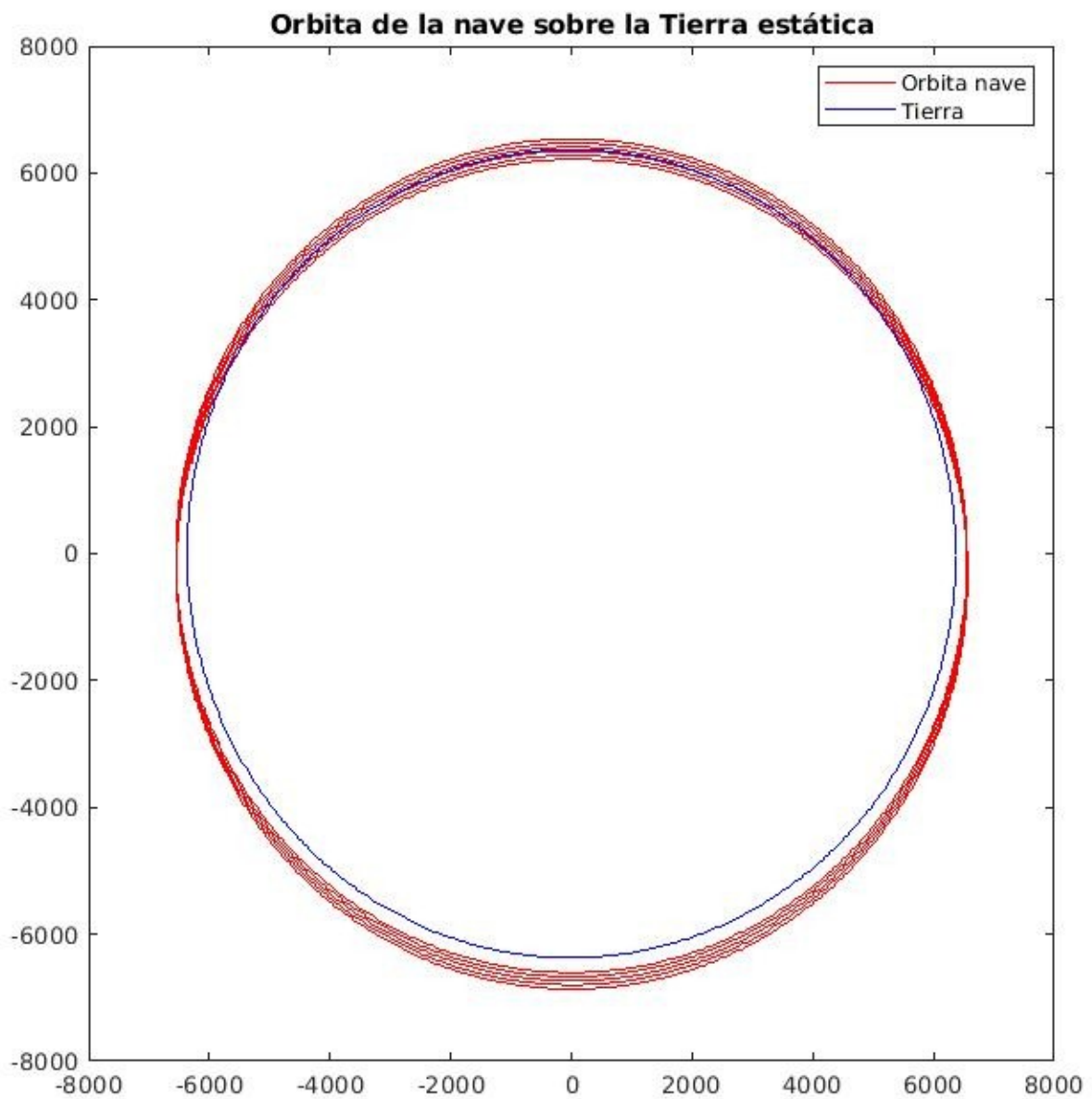
    % Calculo aceleraciones nave
    sp(7:8) = -(GM1 * r1)/norm(r1)^3 - (GM2 * r2 / norm(r2)^3);

    % Calculo aceleracion Luna
    sp(9:10) = -(GM1 * rTL)/norm(rTL)^3;

    % Calculo aceleracion Tierra
    sp(11:12) = -(GM2 * -rLT)/norm(-rLT)^3;

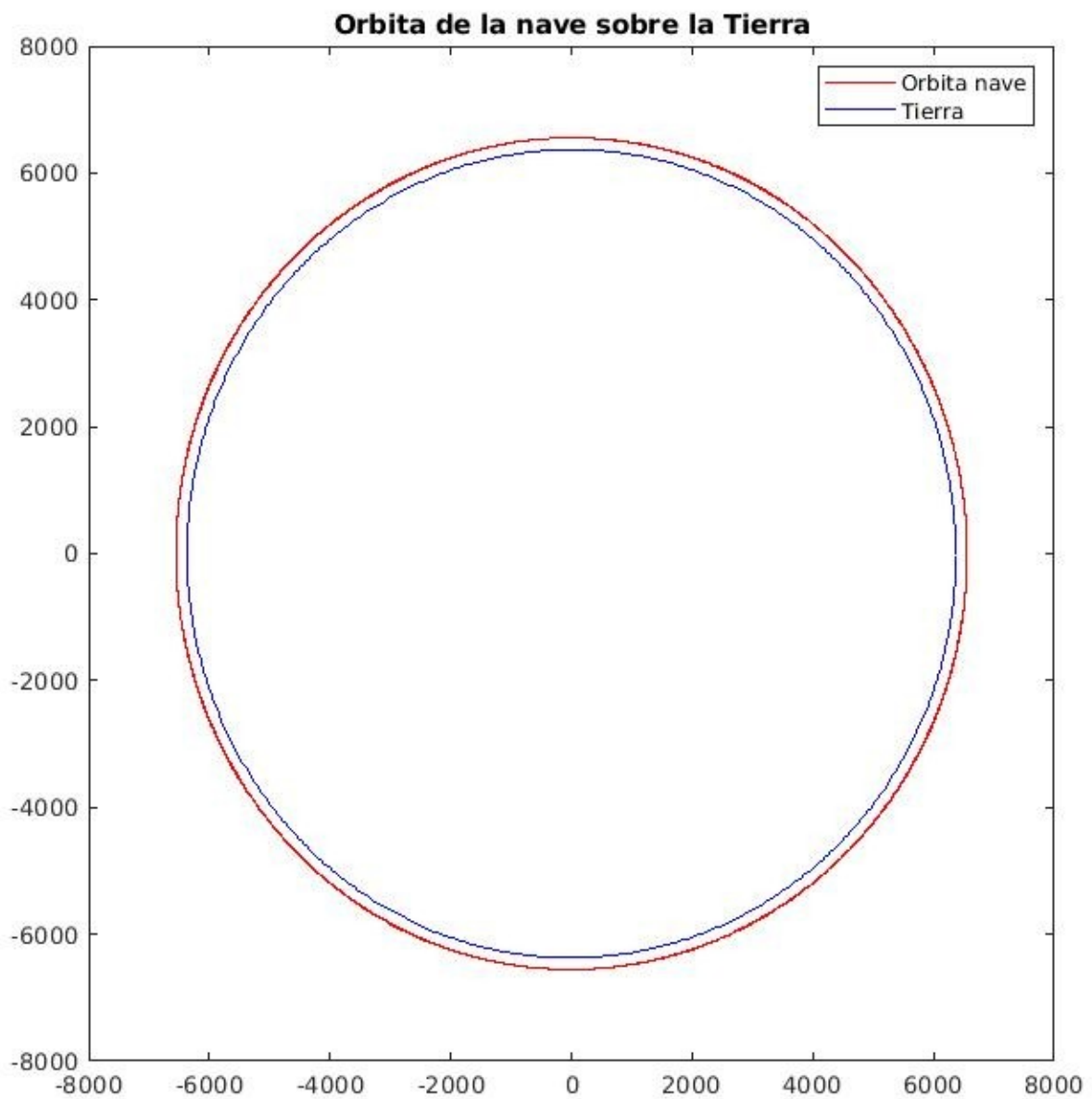
    % Devuelve en sp las velocidades y aceleraciones de las variables de estado
end
```

Adjuntad el gráfico. ¿Cuál parece ser el problema?



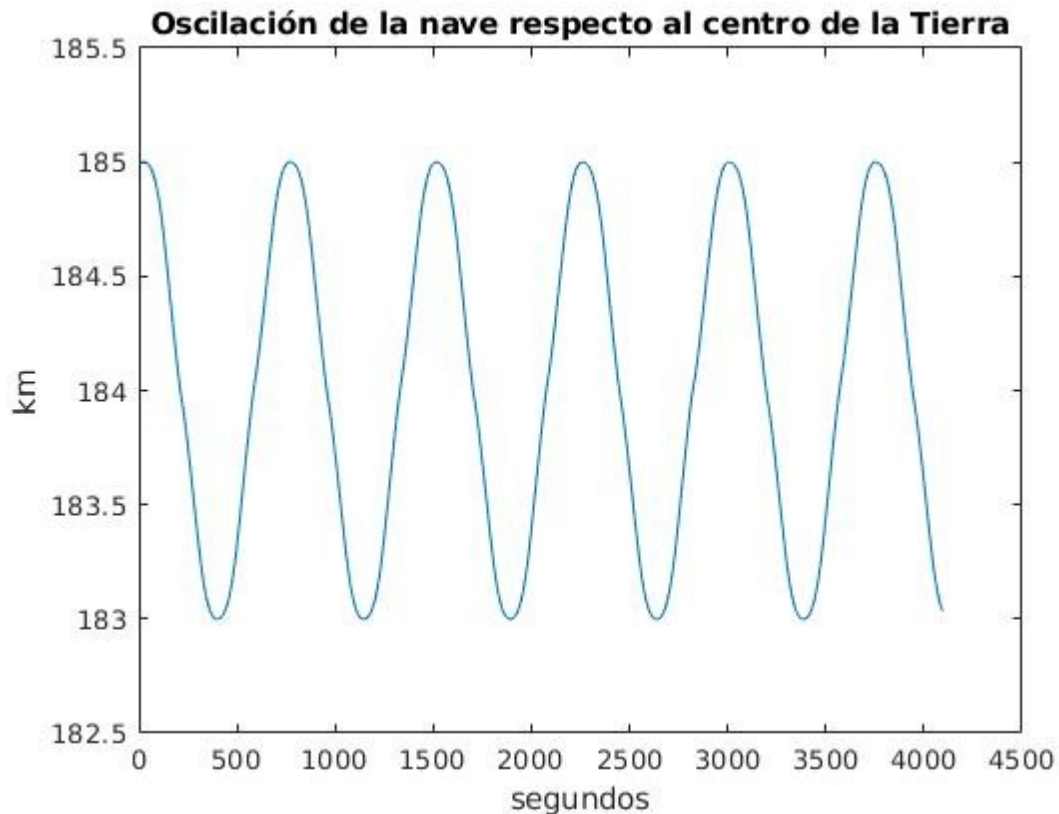
No es una órbita estable. Esto es debido a qué la Tierra está quieta cuando en realidad no es así, se debería estar moviendo. La línea en color rojo es la nave que se mueve alrededor de la Tierra a la vez que la Tierra se mueve alrededor del Sol.

Adjuntad nueva gráfica. ¿Es ahora una órbita estable?



Ahora es una órbita estable.

Adjuntad gráfica/código usado. ¿Es una órbita circular perfecta? ¿Entre qué alturas oscila?



La nave oscila entre 183km y 185km sobre el nivel del mar.

Determinar sobre la gráfica cuánto tiempo tarda en dar la nave una vuelta a la Tierra.

Un período de la gráfica anterior es igual a una vuelta a la Tierra, es decir, la nave tarda unos 880 segundos (14,67 minutos) en dar una vuelta.

2. Fase de TLI (Trans Lunar Injection)

Adjuntad vuestro código de la función apolo() añadiendo el empuje del motor en función del valor de motor y con la nueva componente (masa de la nave) en el vector de estado.

```
function sp = apolo(t,s)
    % s = vector estado con posición y velocidad de la nave
    %      (12x1)      posición y velocidad de la LUNA
    %      posición y velocidad de la TIERRA

    global motor

    GM1 = 3.99e+005; GM2 = 4.90e+003;

    % Extraer del vector de entrada s la posición y velocidad de la nave
    rn=s(1:2); rL=s(3:4); rT=s(5:6);
    vn=s(7:8); vL=s(9:10); vT=s(11:12);
    m=s(13);

    % Vector posición tierra y luna respecto a nave.
    r1 = rn - rT; r2 = rn - rL;
    % distancia entre luna y tierra
    rTL = rL - rT; rLT = -rTL;

    % Reservamos el espacio de sp
```

```

sp=0*s;

% Establecemos velocidades
sp(1:6) = s(7:12);

% Calculo aceleraciones nave
sp(7:8) = -GM1 * r1/norm(r1)^3 - GM2 * r2 / norm(r2)^3;

% Calculo aceleracion Luna
sp(9:10) = -(GM1 * rTL)/norm(rTL)^3;

% Calculo aceleracion Tierra
sp(11:12) = -(GM2 * rLT)/norm(rLT)^3;

% Calculamos el valor de la masa si el motor esta encendido.
% Si el motor esta encendido, estara quemando combustible y por
% lo tanto perdiendo masa
if motor == 1
    F = 1000;
    sp(7:8) = sp(7:8) + (F/m)*((vn-vT)/(norm(vn-vT)));
    sp(13) = -240;
else
    sp(13) = 0;
end

% Devolver en sp las velocidades y aceleraciones de las variables de estado
end

```

Código de las 3 llamadas a ode45 en el programa principal.

```

[T1 S1]=ode45(@apolo,[0 T0],S0,opt);T1=T1';S1=S1';
motor=1;
[T2 S2]=ode45(@apolo,[T0 T0+dT0],S1(:,end),opt);T2=T2';S2=S2';
motor=0;
[T3 S3]=ode45(@apolo,[T0+dT0 TF],S2(:,end),opt);T3=T3';S3=S3';

```

Una vez modificada apolo() y añadidas las llamadas anteriores a ode45 en el programa podemos experimentar para determinar los valores de T0 y dT0 necesarios para conseguir una buena trayectoria en 8 pasando por detrás de la Luna y que nos traiga de vuelta.

Ángulo de reentrada.

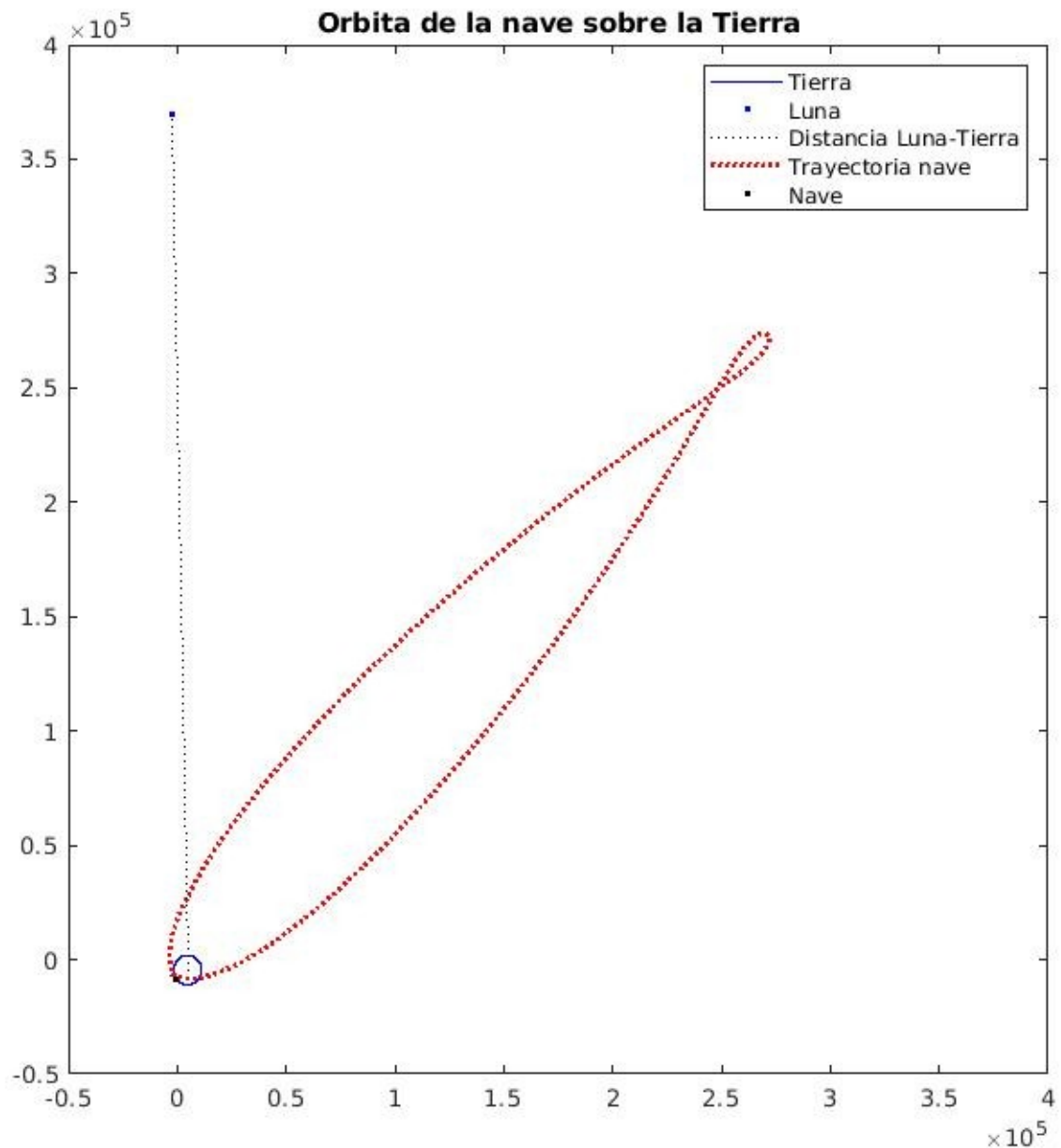
Buscar una trayectoria que vuelva a la Tierra de forma que la función de eventos detenga el proceso. **Valores encontrados para T0 y dT0. Adjuntar la gráfica de la trayectoria. ¿Cuál ha sido la duración del viaje (en segundos y en días/horas/minutos)?**

Los valores encontrados para T0 y dT0 son:

```

T0=3250;
dT0=310;

```



La duración del viaje en total han sido 614242 segundos, lo que equivale a 7 días, 2 horas, 37 minutos.

¿Velocidad de la nave con respecto a la Tierra en la reentrada (en km/s y en km/hora)?

Usando las velocidades de x e y de la nave y de la Tierra y con la fórmula de la velocidad relativa se obtiene que la nave tiene una velocidad respecto a la Tierra de 11,9896km/s o 43.162km/h.

Fórmula de la velocidad:

Velocidad relativa Nave-Tierra = $S_3(11, \text{end}) + S_3(12, \text{end}) - S_3(8, \text{end}) + S_3(7, \text{end})$

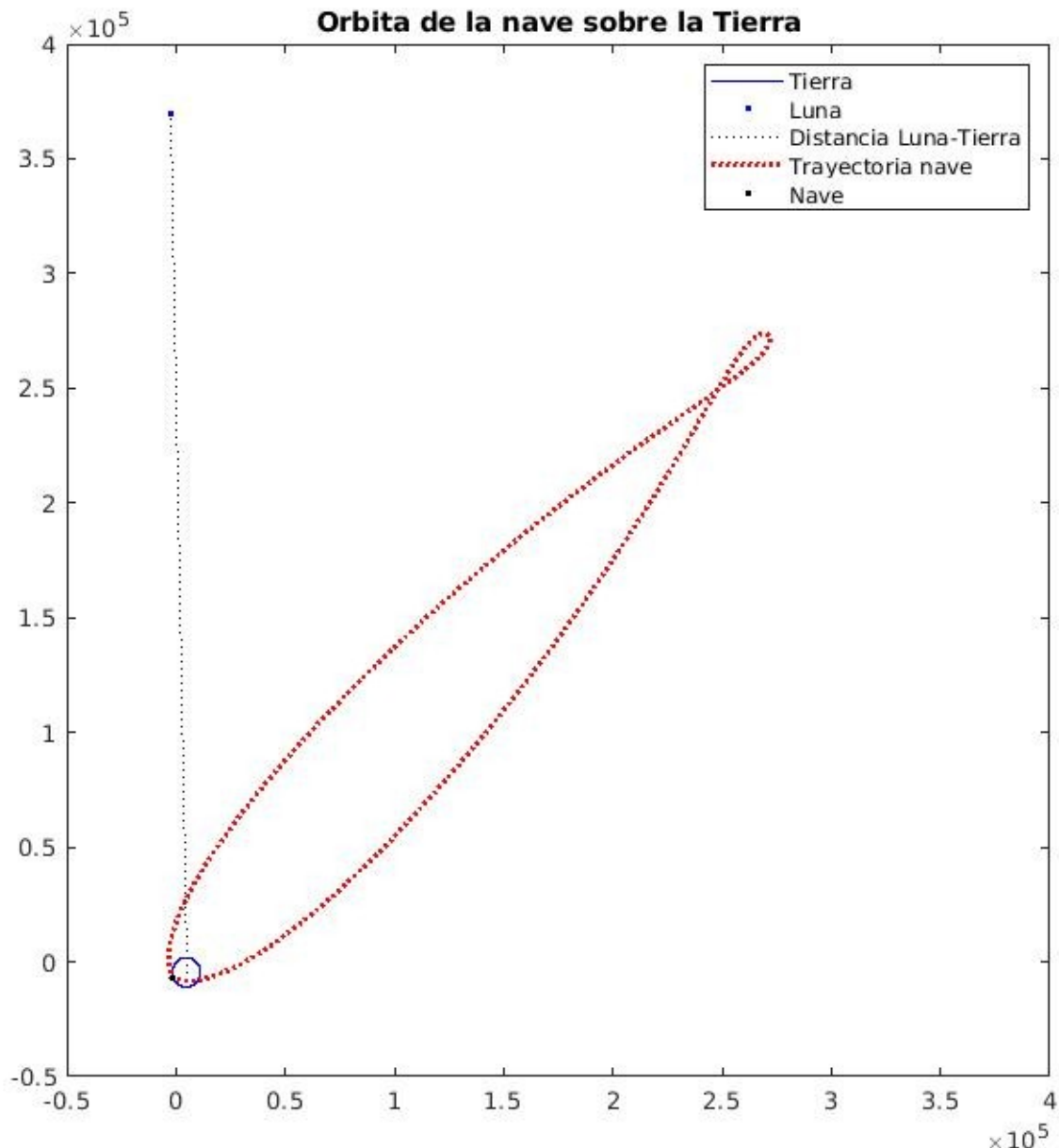
Adjuntar vuestro código de la función phi() y valor del ángulo para la trayectoria anterior.

```
function fi=phi(S)
    xn = S(1)-S(5); yn = S(2)-S(6);
    vx = S(7)-S(11); vy = S(8)-S(12);
    cos = (xn*vy - yn*vx)/(norm([xn -yn]) * norm(S(7:8)));
    fi = acosd(cos);
```

end

El valor del ángulo es de 2.3424 grados.

Mejores valores encontrados para T_0 y dT_0 . Ángulo de reentrada obtenido.
Gráfica de la trayectoria obtenida para vuestra mejor combinación de T_0 y dT_0 .
Para T_0 igual a 3250.1364 y dT_0 igual a 310, el grado obtenido ha sido 6.5079.
Gráfica:



**Adjuntad fichero x_apolo.m junto con el fichero de respuestas en un .rar o similar.
El fichero x_apolo debe incluir todas las funciones pedidas (apolo, phi, ...), de forma que sea autosuficiente para ejecutarse.**

----- Hasta aquí 45% de la nota -----

3. Optimización para conseguir objetivos de la misión (25%)

¿Distancia mínima de vuestra trayectoria?

La distancia mínima entre la nave y la Luna es de 5.304387e+03 km

Repetir para $T_0=3245$, $dT_0=310$ y volcar los resultados obtenidos.

Distancia mínima entre la nave y la Tierra: 1.035681e+04

Ángulo: 2.358396e+00

Distancia mínima entre la nave y la Luna: 4.351346e+03

Coste: 1.227813e+08

Adjuntad código de la función `opt_trayectoria`.

```
function h = vector_h(p1x,p1y,p2x,p2y)
    x = p1x - p2x; y = p1y - p2y;
    h = sqrt(x.^2 + y.^2);
end

function [min_h idx] = altura_minima(p1x,p1y,p2x,p2y)
    h = vector_h(p1x,p1y,p2x,p2y);
    N = length(h);
    idx = N;
    min_h = h(idx);
    for k=2:N
        if h(k) < h(k-1) && h(k) < min_h
            min_h = h(k);
            idx = k;
        end
    end
end

function coste=opt_reentrada(x)
    global motor
    global cont
    % Partir de las condiciones iniciales S0,
    % Resolver las tres etapas [0,T0] + [T0,T0+dT], [T0+dT, 8 dias]
    % A partir de la trayectoria de la última etapa evaluar función de coste
    cont = cont + 1;

    opt=odeset('RelTol',1e-6,'events',@vuelta,'Refine',8); % SIN GRAFICOS

    T0=x(1);dT0=x(2);TF=8*24*3600;

    [S0, ~] = preparar_trayectoria();
    [~, S1]=ode45(@apolo,[0 T0],S0,opt);S1=S1';
    motor=1;
    [~, S2]=ode45(@apolo,[T0 T0+dT0],S1(:,end),opt);S2=S2';
    motor=0;
    [T, S3]=ode45(@apolo,[T0+dT0 TF],S2(:,end),opt);S=S3';

    % Obtenemos datos respecto a la Tierra
    [min_h idx] = altura_minima(S(1,:),S(2,:),S(5,:),S(6,:));
    ang = phi(S(:,idx));
    coste = (min_h - 120)^2 + (ang - 6.5)^2;

    % Obtenemos datos respecto a la Luna
    [min_h_lunar idx] = altura_minima(S(1,:),S(2,:),S(3,:),S(4,:));
```



```

coste = coste + (min_h_lunar-110)^2;

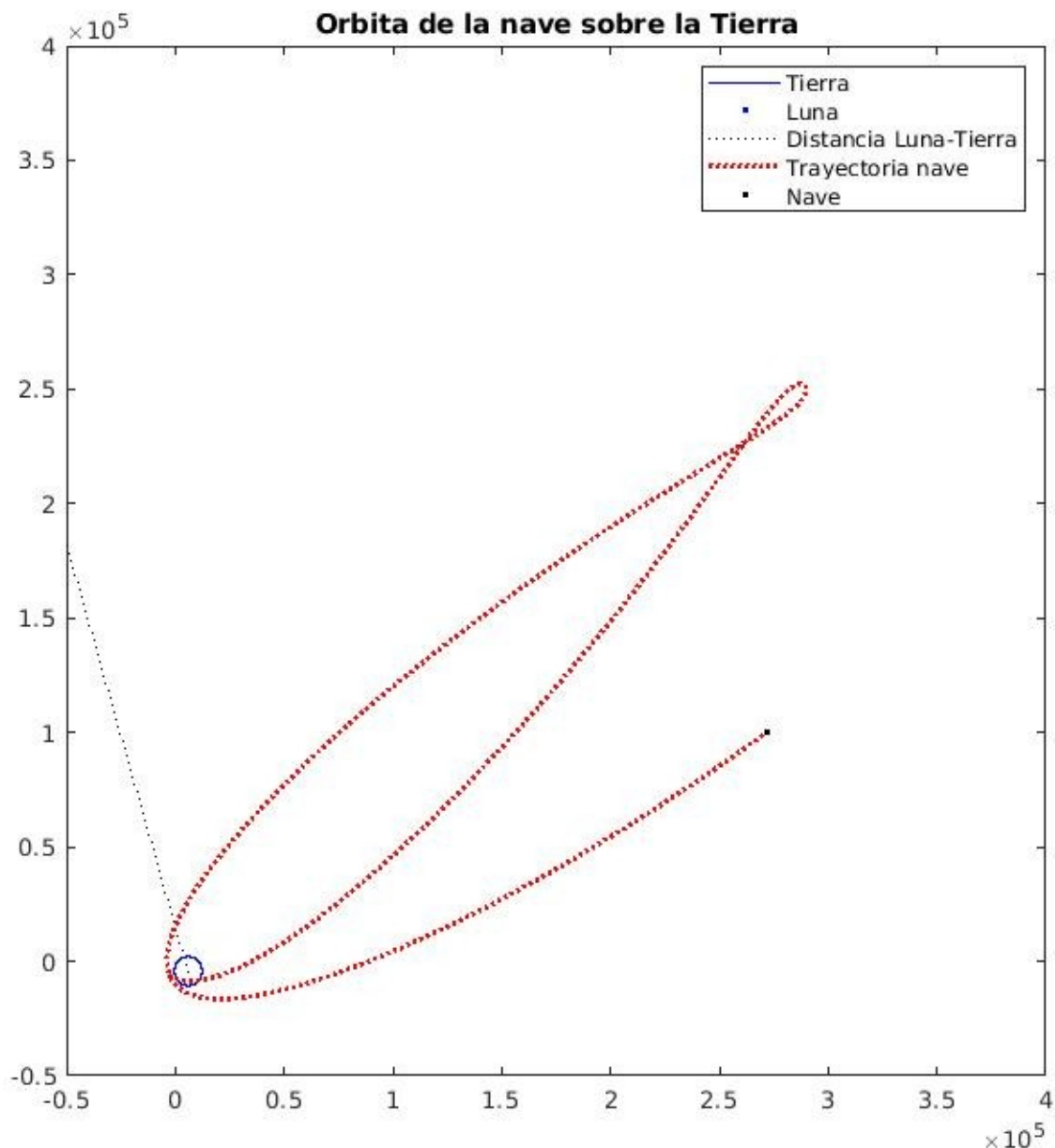
fprintf("Distancia minima entre la nave y la Tierra: %d\n", min_h);
fprintf("Distancia minima entre la nave y la Luna: %d en el instante %d\n", min_h_lunar, T(idx));
fprintf("Angulo: %d\n", ang);
fprintf("Coste: %d\n", coste);
end

```

Indicad vuestros datos de partida (T_0, dT) y los parámetros T_0 y dT obtenidos tras el proceso de optimización (con 4 decimales). ¿Cuántas iteraciones necesitáis?

Con $T_0=3250.1364$ y $dT_0= 310$, `fminsearch` realiza 596 iteraciones y los valores que encuentra óptimos para T_0 y dT_0 son $3.2048e+03$ y $3.10401e+02$ respectivamente.

Si aplicamos dichos datos al algoritmo inicial y dibujamos el recorrido de la nave, observamos que estos datos son erróneos porque no llega a realizar la reentrada en la Tierra.



No he podido encontrar el error en el algoritmo.

¿A qué distancia mínima de la Luna pasa vuestra trayectoria? ¿En qué instante de tiempo se alcanza dicha distancia?

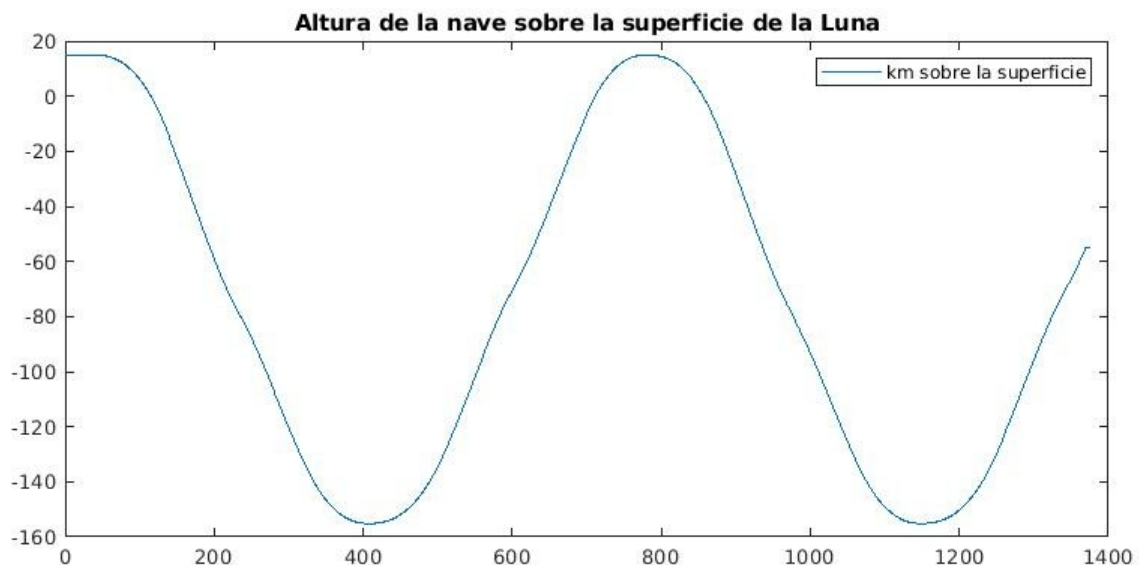
Distancia minima entre la nave y la Luna: $3.031473e+03$ en el instante $2.730701e+05$

El siguiente apartado es independiente de los anteriores. Aunque no hayáis implementado la parte de optimización podéis continuar haciendo la práctica.

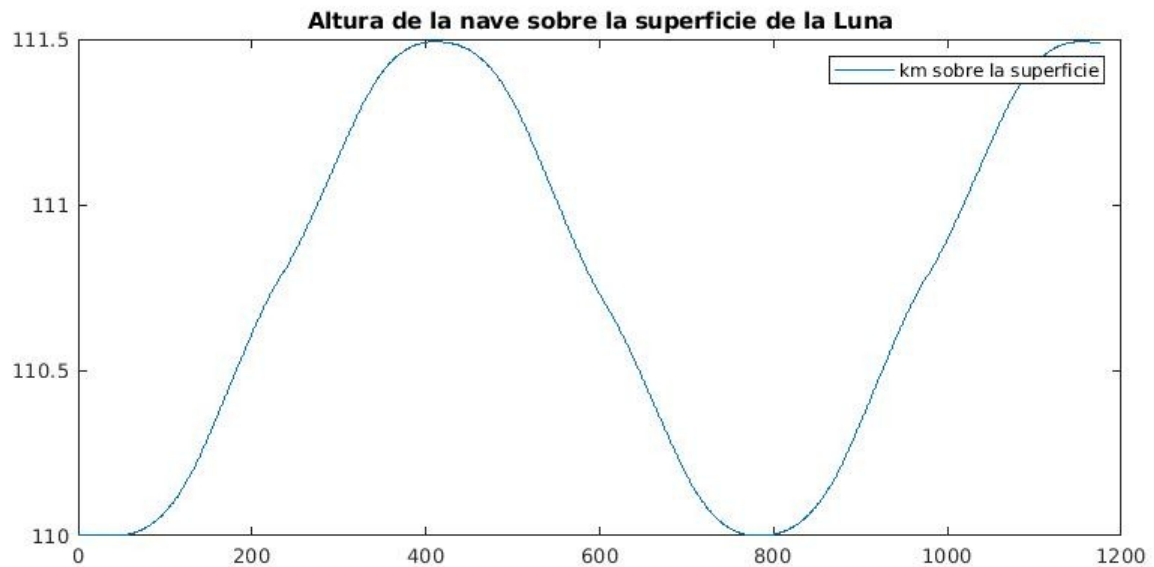
4. Descenso Lunar (30%):

Adjuntad el gráfico ¿Entre qué alturas oscila? ¿Cuánto tarda la nave en completar una órbita alrededor de la Luna?

Según el enunciado de la práctica (y el correo corrigiendo la errata cambiando la posición de x por y) la nave debería estar a 15km sobre la superficie lunar. Si usamos ese dato, obtenemos la siguiente variación de alturas:



Como se puede observar la nave toma valores negativos en algunas regiones (se choca contra la superficie). Si suponemos que en un inicio la nave se encuentra a 110km sobre la superficie lunar, tenemos que:



En este caso, la nave oscila entre 110km y 111.5km. Y el tiempo que tarda en dar una vuelta es de 782 segundos.

Adjuntad el código de descenso() con las diferentes opciones del motor.

```
function sp=descenso(t,s)
    global motor Rm
    sp=0*s;

    r= s(1:2); v=s(3:4); m=s(5);

    GM2 = 4.90e+003;

    % Calculo aceleracion gravitatoria
    sp(1:2) = v;
    sp(3:4) = -GM2 * r / (norm(r))^3;
    sp(5) = 0;

    % Modificar aceleraciï½n segun estado del motor: 1,2,3
    switch(motor)
        case 1
            p = .25;
            E = p * 45;
            G = -15*p;

            % Se resta porque el motor hace fuerza justo al sentido
            % contrario a la velocidad
            sp(3:4) = sp(3:4) - (E/m)*(v/(norm(v)));

            %Gasto del motor
            sp(5) = G;
        case 2
            p=1;
            E = p * 45;
            G = -15 * p;
            up = r/norm(r);
            d = v + 0.14 * up;
            sp(3:4) = sp(3:4) - (45/m) * (d/norm(d));
            sp(5) = G;
        case 3
```

```

vd = -r/norm(r);
h = sqrt(sp(1)^2 + sp(2)^2) - Rm;
a = -sp(3:4) - 2 .* vd/h .* v;
E = m * norm(a);
p = E/45;
a = a/p;
G = -15;
sp(5) = G;
end
% Devolver vel, aceleración y gasto combustible en vector sp
end

```

Adjuntad la función eventos una vez completada.

```

function [val,term,dir]=evento_altura(t,s)
% Detecta cuando la altura sobre la Luna llega a un valor h_stop
global Rm h_stop
val = norm(s(1:2)) - Rm - h_stop; % negativo si llegamos al limite
term = 1; % para la iteracion si se cumple condicion
dir = -1; % Activar si cruce por cero es en sentido negativo (bajando)
end

```

Etapas 1)

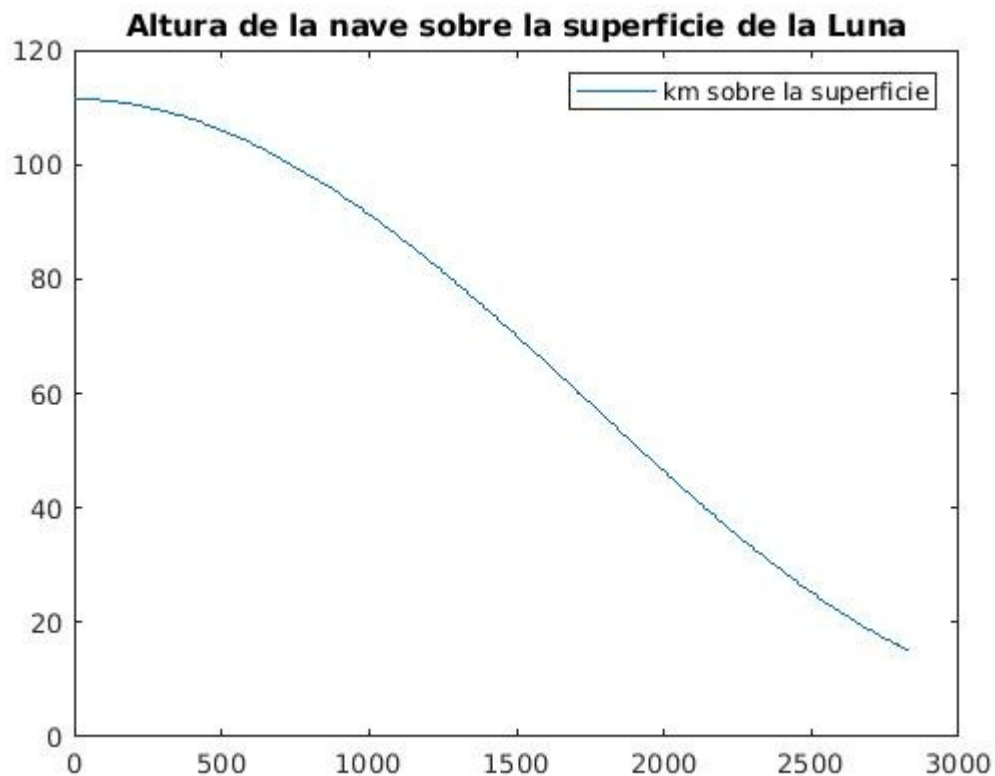
Tiempo (segundos) en el que la nave alcanza los 15 km de altura. Duración de esta fase.

Hacer una gráfica de la altura de la trayectoria durante esta fase. Adjuntadla.

Módulo de la velocidad en km/s y km/h al llegar a ese punto.

La nave tarda 2833 segundos, es decir, unos 47 minutos en alcanzar los 15km desde que apago los motores.

La gráfica de la altura se muestra a continuación:



La velocidad de la nave es 1750 km/s o 6300000 km/h

Etapa 2)

Tiempo (segundos) al final de esta fase (2.25 km de altura). Duración de la fase.

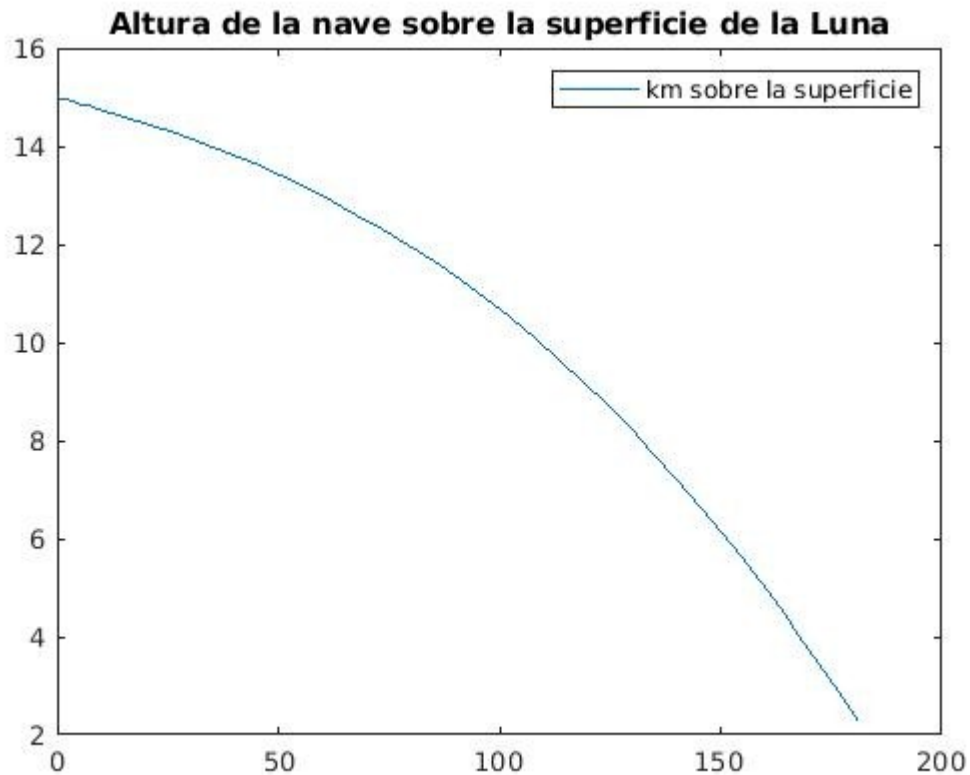
Gráfica de la altura de la trayectoria en esta fase.

Velocidad horizontal y vertical en km/h al llegar a ese punto.

% de combustible restante en ese momento.

La nave tarda 181 segundos en alcanzar los 2.250000e+00 km

La gráfica de la altura:



Velocidad vertical: 1.668828e+03 km/s de la nave es 1737 km/s o 6253200 km/h
Fuel restante 65 %

Etapa 3)

Tiempo (segundos) al final de esta fase. Duración de esta fase.

Gráfica de la altura de la trayectoria durante esta fase.

Velocidad horizontal y vertical en km/h al final de esta fase.

Porcentaje de combustible restante en ese momento (de los 8200 kg iniciales).

La nave tarda 14 segundos en alcanzar los 150 metros.

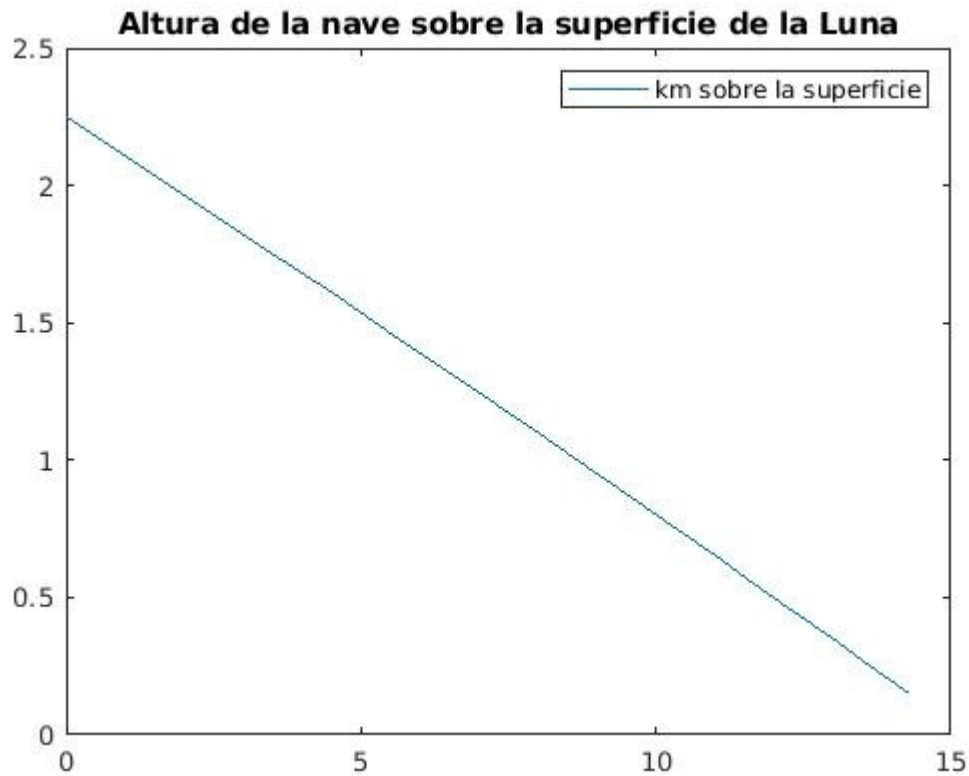
La velocidad de la nave es 1735 km/s o 6246000 km/h.

Velocidad horizontal: -4.673767e+02 km/s

Velocidad vertical: 1.671019e+03 km/s

Fuel restante 63 %

Gráfica:



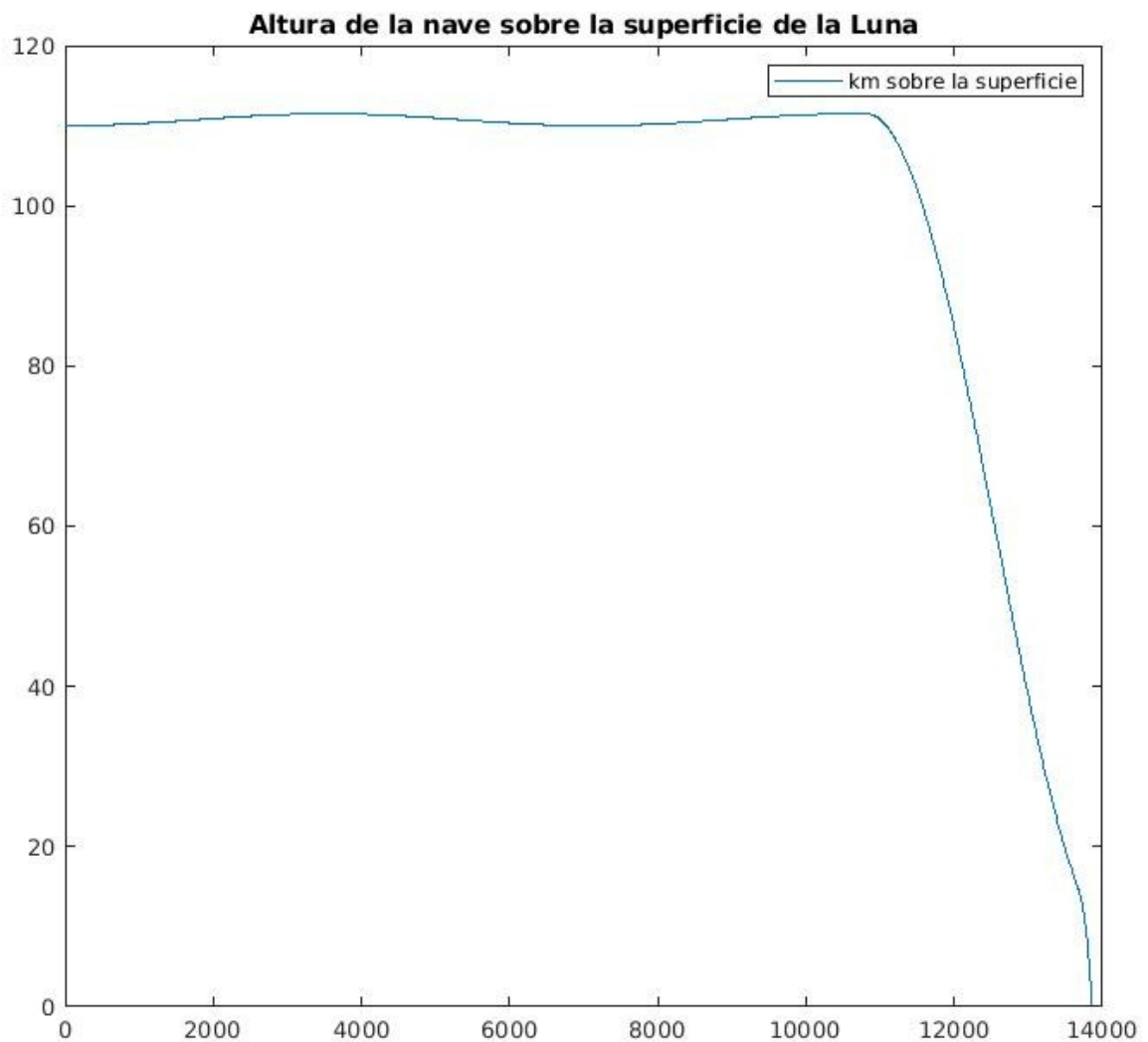
Combustible restante en kg al final de la fase 3 (150 mt).

Cantidad de fuel restante es $5.155287e+03$

Durante esta 3ª fase, de media, ¿cuál ha sido el % de empuje usado por el motor?

Sabiendo que en esos instantes finales el motor puede "sostener" la nave usando un 25% de su empuje total, ¿de cuántos minutos dispone el piloto para buscar un buen sitio para aterrizar?

Tras ejecutar cada etapa, guardar los tiempos y alturas de cada una de ellas y al terminar superponer en una sola gráfica la altura de la nave en función del tiempo de descenso durante las 3 fases del vuelo. Pintad las tres fase con colores distintos (negro 'k', azul 'b' y rojo 'r'). [Adjuntad la gráfica resultante.](#)



Además del fichero doc con las respuestas adjuntad (en un .rar o similar) con la entrega vuestros programas `x_apolo.m` y `x_dencenso.m`.

Ambos programas deben ser autosuficientes (deben contener todas las funciones auxiliares con las ecuaciones diferenciales, funciones de eventos, funciones auxiliares, etc.). Para comprobar que son autosuficientes comprobad que se ejecutan correctamente después de hacer un clear, desde un directorio sin ninguna otra función.