PS 5 Problem 1 Part 1

Wednesday, May 8, 2019 10:54 AM

Objective: $\underset{u(\cdot)}{\text{maximize}}\ y(t_f)$

$$\Rightarrow \min_{u(\cdot)} J = -y(t_f)$$

$$\therefore h = -y(t_f) \qquad g = 0 \qquad \left[J = h + \int g\,dt\right]$$

dynamics: $\dot{y}(t) = v(t) \qquad \dot{v}(t) = \frac{u(t)}{m(t)} - g$

$$\dot{m}(t) = -bu(t)$$

PART 1

$$\mathcal{H} = g + p_y \dot{y} + p_v \dot{v} + p_m \dot{m}$$

$$= 0 + p_y v + p_v\left(\frac{u}{m} - g\right) + p_m(-bu)$$

(Omitting dep on t for clarity)

By NOC,

$$\boxed{\begin{aligned} \dot{p}_y &= -\frac{\partial \mathcal{H}}{\partial y} = 0 \\ \dot{p}_v &= -\frac{\partial \mathcal{H}}{\partial v} = -p_y \\ \dot{p}_m &= \frac{-\partial \mathcal{H}}{\partial m} = -\left(-\frac{p_v u}{m^2}\right) \end{aligned}}$$

$$\Rightarrow p_y = \text{const} = c_1$$

$$\Rightarrow p_v = -c_1 t + c_2$$

$$\Rightarrow \dot{p}_m(t) = \frac{(-c_1 t + c_2)u(t)}{m(t)^2}$$

Part 2

Wednesday, May 8, 2019 9:04 PM

PART 2

Final BC: $\left[\left(p_y(t_f), p_v(t_f), p_m(t_f)\right) - \nabla h\left(y(t_f), v(t_f), m(t_f)\right)\right]$

$\cdot \ker \nabla F\left(y(t_f), v(t_f), m(t_f)\right) = 0$

$F(y, v, m) := m - m_f$

$\Rightarrow \nabla F = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$

Let $w \in \ker \nabla F \Rightarrow (w_1 \;\; w_2 \;\; w_3)\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = 0$

$\Rightarrow w_3 = 0$

$\Rightarrow \ker \nabla F = \{ w \in \mathbb{R}^3 : w_3 = 0 \}$

$h = -y(t_f)$

$\Rightarrow \nabla h = \begin{bmatrix} -1 \\ 0 \\ 0 \end{bmatrix}$

$$\Rightarrow \overbrace{\begin{bmatrix} p_y(t_f) + 1 \\ p_v(t_f) \\ p_m(t_f) \end{bmatrix}}^{\begin{matrix} p_y \\ p_v \\ p_m \end{matrix} - \nabla h} \cdot \overbrace{\begin{bmatrix} w_1 \\ w_2 \\ 0 \end{bmatrix}}^{\ker \nabla F} = 0$$

$\Rightarrow w_1 \left(p_y(t_f) + 1\right) + w_2 \left(p_v(t_f)\right) = 0 \quad \forall \; w_1 \; w_2$

$\Rightarrow \boxed{p_y(t_f) = -1 \quad \text{and} \quad p_v(t_f) = 0}$

Part 3

Wednesday, May 8, 2019 9:04 PM

PART 3

$$\phi(t) = \frac{p_v(t)}{m(t)} - p_m(t)\, b$$

$$\Rightarrow \dot{\phi}(t) = \frac{\dot{p}_v(t)\, m(t) - \dot{m}(t) p_v(t)}{m(t)^2} - b\,\dot{p}_m(t)$$

From previous parts,

$$p_y(t) = c_1 \qquad p_y(t_f) = -1 \quad \Rightarrow c_1 = -1 \quad p_y(t) = -1$$

$$p_v(t) = -c_1 t + c_2 = t + c_2 \qquad p_v(t_f) = 0$$

$$\Rightarrow c_2 = -t_f \quad \Rightarrow p_v(t) = t - t_f$$

$$\dot{p}_m(t) = \frac{p_v(t)\, u(t)}{m(t)^2} = \frac{(t - t_f)\, u(t)}{m(t)^2}$$

$$\dot{\phi}(t) = \frac{\underbrace{\dot{p}_v(t)}_{-p_y(t)}\, m(t) - \underbrace{\dot{m}(t)}_{-bu(t)} p_v(t)}{m(t)^2} - b\,\underbrace{\dot{p}_m(t)}_{\frac{p_v(t)u(t)}{m(t)^2}}$$

$$\Rightarrow \dot{\phi}(t) = \frac{-p_y(t)\, m(t) + b u(t) p_v(t) - b p_v(t) u(t)}{m(t)^2}$$

$$\Rightarrow \boxed{\dot{\phi}(t) = -\frac{p_y(t)}{m(t)}}$$

We know $p_y(t) = -1 \quad \Rightarrow \dot{\phi}(t) = \frac{1}{m(t)}$

$\because m(t) > 0$ (mass can't be negative) $m(0) = m_0 \quad m(t_f) = m_f$

$\Rightarrow \dot{\phi}(t) > 0$

$\therefore$ $\boxed{\phi(t) \text{ can't be zero on any non-zero time interval } \because \dot{\phi}(t) > 0}$

Part 4

Wednesday, May 8, 2019 9:05 PM

PART 4

$\because \phi(t) \neq 0$ on any non-zero time interval,

$$u^*(t) = \arg\min_u \mathcal{H}(p_y, p_v, p_m, y, v, m)$$

$$= \arg\min_u \left(p_y v + p_v\left(\frac{u}{m} - g\right) + p_m(-bu) \right)$$

$$= \arg\min_u \left(\underbrace{-\overset{p_y(t)}{1}\, v(t)}_{\text{indep of } u(t)} + u(t)\left(\frac{p_v(t)}{m(t)} - b p_m(t)\right) \underbrace{- g\,(t - t_f)}_{\text{indep of } u(t)} \right)$$

$$= \arg\min_u \big(u(t)\, \phi(t) \big)$$

$0 \le u(t) \le u_{max}$

$$\therefore \boxed{u^*(t) = \begin{cases} 0 & \text{if } \phi(t) > 0 \\ u_{max} & \text{if } \phi(t) < 0 \end{cases}}$$

Part 5

Wednesday, May 8, 2019 9:05 PM

PART 5

Given $\dot{v}(0) > 0 \Rightarrow \frac{u(0)}{m_0} - g > 0$

$\Rightarrow u(0) > m_0 g$

$u^*(t)$ only takes 2 values — 0 or u_{max}.

$\because u_{max} > m_0 g$ & $u(0)$ can't be 0

$\Rightarrow \boxed{u^*(0) = u_{max}}$

From previous part, we know u_{max} is optimal only if $\phi(t) < 0$

$\Rightarrow \boxed{\phi(0) < 0}$

PART 6

We know $\phi(t) < 0$ for $t=0$

We also know $\dot{\phi}(t) = \frac{1}{m(t)} > 0$

$\because$ Final time is free $\Rightarrow \delta t_f$ arbitrary

$$\Rightarrow \left[\mathcal{H} + \frac{\partial h}{\partial t}\right]_{t_f} = 0$$

$$\Rightarrow \underbrace{p_y}_{p_y(t_f)=-1} v + \underbrace{p_v}_{p_v(t_f)=0}\left(\frac{u}{m} - g\right) + p_m(-bu) + \left(-\underbrace{\frac{dy}{dt}}_{\frac{dy}{dt} = v(t)}\right)\Bigg|_{t_f} = 0$$

$$\Rightarrow -\cancel{v(t_f)} + 0 - \underbrace{b}_{b>0} u(t_f)\, p_m(t_f) - \cancel{v(t_f)} = 0$$

$$\Rightarrow u(t_f)\, p_m(t_f) = 0$$

$\Rightarrow$ either $u(t_f) = 0 \Rightarrow \phi(t_f) > 0$

or $p_m(t_f) = 0$

$$\Rightarrow \phi(t_f) = \frac{p_v(t_f)}{m(t_f)} - p_m(t_f)\, b$$
$$= 0 - 0 \quad = 0$$

A) if $p_m(t_f) = 0 \Rightarrow t_{sw} = t_f \Rightarrow \phi(t_f) = 0$

and $\phi(t) < 0$ for $t \in [0, t_{sw}]$

$\phi(t) > 0$ for $t \in (t_{sw}, t_f]$

$\because$ $\phi(t)$ can't be 0 for a finite time interval

B) If $p_m(t_f) \neq 0$ and $u(t_f) = 0$

$\Rightarrow \phi(t_f) > 0$

$\Rightarrow \exists\, t_{sw} \in (0, t_f)$

$\phi(t) < 0$ for $t \in [0, t_{sw}]$

$\phi(t) > 0$ for $t \in (t_{sw}, t_f]$

$\because$ $\phi(0) < 0$ and $\dot{\phi}(t) > 0$

$\therefore$ In both cases, $\exists\, t_{sw} \in (0, t_f]$ such that $\phi(t) < 0$ for $t \in [0, t_{sw}]$
$\phi(t) > 0$ for $t \in (t_{sw}, t_f]$

Note: another way of showing same result

$\because$ Final time is free δt_f arbitrary

$$\Rightarrow \mathcal{H} + \frac{\partial h}{\partial t}\Big|_{t_f} = 0$$

$$\Rightarrow \left(p_y v + p_v\left(\frac{u}{m} - g\right) + p_m(-bu)\right) - \frac{\partial y(t_f)}{\partial t} = 0$$

$$\Rightarrow \left(-1\, v(t) + u(t)\left(\frac{p_v(t)}{m(t)} - b p_m(t)\right) - g(t - t_f)\right)\Big|_{t_f} - \dot{y}(t_f) = 0$$

$$\Rightarrow -v(t_f) + u(t_f)\,\phi(t_f) - \underbrace{\dot{y}(t_f)}_{\dot{y} = v} = 0$$

$$\Rightarrow u(t_f)\,\phi(t_f) = 0$$

$\therefore$ Either $\phi(t_f) = 0$ momentarily or $u(t_f) = 0 \Rightarrow \phi(t_f) > 0$

However, because $\mathcal{H}$ does not depend explicitly on time, we also have the NOC

$$\mathcal{H}(x^*, u^*, p^*) = 0 \quad \forall\, t \in [t_0, t_f]$$

$$\Rightarrow p_y(t)\, v_0 + p_v(t)\left(\frac{u(t)}{m(t)} - g\right) + p_m(t)\left(-b u(t)\right) = 0$$

$p_y(t) = -1$ $\quad$ $p_v(t) = t - t_f$

$$\Rightarrow -1\, v(t) + u(t)\left(\frac{p_v(t)}{m(t)} - b p_m(t)\right) - (t - t_f)\, g = 0$$

$$\Rightarrow -v(t) + u(t)\,\phi(t) - (t - t_f)\, g = 0$$

$$\Rightarrow u(t)\,\phi(t) = v(t) + (t - t_f)\, g$$

At $t = t_f$ $\quad \mathcal{H} + \frac{\partial h}{\partial t} = 0 \Rightarrow \mathcal{H}(t_f) + (-\dot{y}) = 0$
$\Rightarrow v(t_f) = 0$

$$\Rightarrow u(t_f)\,\phi(t_f) = 0$$

$\therefore$ Either $\phi(t_f) = 0$ momentarily or $u(t_f) = 0 \Rightarrow \phi(t_f) > 0$

Part 7

Wednesday, May 8, 2019 9:06 PM

PART 7

$u^*(0) = u_{max}$

$\forall\ t \in [0, t_{sw}] \quad \phi(t) < 0 \Rightarrow u^*(t) = u_{max}$

$\forall\ t \in (t_{sw}, t_f] \quad \phi(t) > 0 \Rightarrow u^*(t) = 0$

$\therefore \forall\ t \in (t_{sw}, t_f] \quad \dot{m}(t) = -bu^*(t) = 0$

$\Rightarrow m(t_{sw}) = m(t_f) = m_f$

$\forall\ t \in [0, t_{sw}] \quad \dot{m}(t) = -bu^*(t) = -bu_{max} = \text{const}$

$\Rightarrow m(t_{sw}) = m(0) + (-bu_{max})(t_{sw})$

$$\Rightarrow t_{sw} = \frac{m(t_{sw}) - m_0}{-bu_{max}}$$

$$\Rightarrow \boxed{t_{sw} = \frac{m_0 - m_f}{bu_{max}}}$$

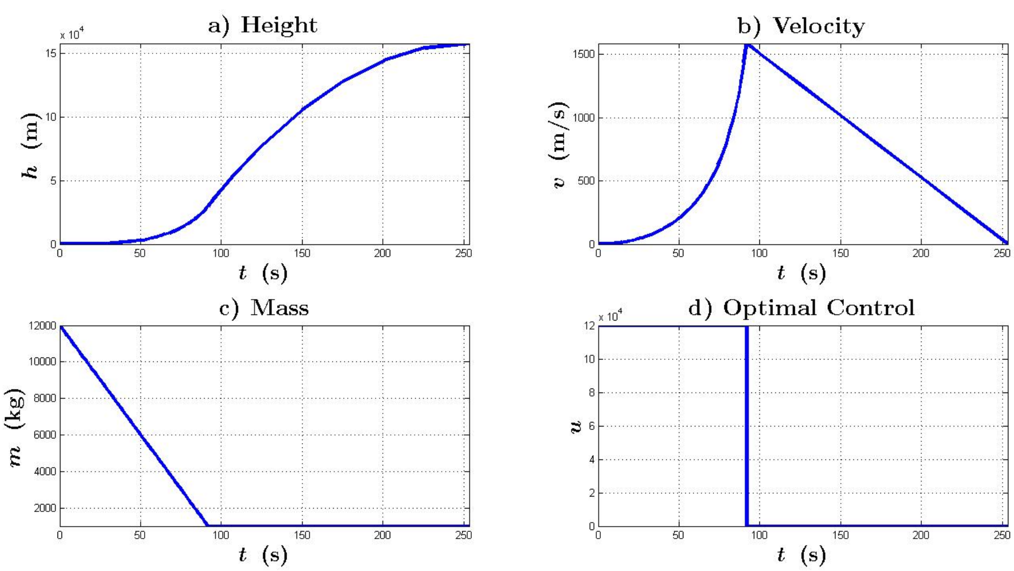
Problem 2 Part 1 – Dichotomy solver

```
% Exaclty solving Goddard's problem.
% From the homework, we know that Goddard's problem is solved
% with switching time tSw = (m0 - mf)/(b*uMax). We want to
% discover what is the optimal final time tf that maximizes
% the final height h(tf). For this, we implement a dichotomic
% search on tf with the following idea: the final time tf
% that maximizes h(tf) is the time for which the time derivative
% of h(t) at tf is zero, i.e., 0 = h'(tf) = v(tf). Then, we
% seek tf as the zero for v(tf), where the velocity v arises
% from integrating the rocket dynamics with the optimal control
% given in the homework, i.e., u(t) = uMax if t <= tSw and
% u(t) = 0 otherwise.

clear all; clf; clc; format long;

global g; g = 9.81;
global b;
global uMax;
global h0; h0 = 0.;
global v0; v0 = 0.;
global m0;
global mf;

% Scenario: recall that we must satisfy uMax > m0*g
m0 = 12000; mf = 1000;
b = 1e-3; uMax = 1.2e5;
```

```
% Parameters for the dichotomic search.
% For given initial times tA, tB such that v(tA) > 0, v(tB) < 0,
% we iteratively evaluate v at tMed = (tA + tB)/2 until we find
% v(tMed) = 0. Therefore: tf = tMed.
tA = 1.;
tB = 500.;
dichotomyFuncTA = dichotomyFunc(tA);
dichotomyFuncTB = dichotomyFunc(tB);
tMed = (tA + tB)/2.;
dichotomyFuncTMed = dichotomyFunc(tMed);
iterDichotomy = 1;
iterDichotomyMax = 1000;
epsDichotomy = 1e-1;

if dichotomyFuncTA < 0 || dichotomyFuncTB > 0
    fprintf('Wrong guess times tA and tB! Choose them such that: v(tA) > 0 and v(tB) < 0...\n',iterDichotomy);
else
    % Classical dichotomic/binary/bisection search
    while ( abs(dichotomyFuncTMed) > epsDichotomy && iterDichotomy < iterDichotomyMax )
        % TODO: Implement dichotomic search. See initialization in
        % lines 32-37 for reference.
        if dichotomyFuncTMed > 0 % search between tMed and tB
           tA = tMed;
        else % search between tA and tMed
           tB = tMed;
        end
        dichotomyFuncTA = dichotomyFunc(tA);
        dichotomyFuncTB = dichotomyFunc(tB);
        tMed = (tA + tB)/2.;
        dichotomyFuncTMed = dichotomyFunc(tMed);
        iterDichotomy = iterDichotomy + 1;
    end
    tf = tMed;

    % Optimal switching time.
    tSw = (m0 - mf)/(b*uMax);
    if tSw > tf % Verifying that: 0 < tSw <= tf
        tSw = tf;
    end
```

```matlab
    % Plotting
    fprintf('Switching time tSw = %f\n',tSw);
    fprintf('Final time tf = %f\n',tf);
    options = odeset('AbsTol',1e-9,'RelTol',1e-9);
    [t,x] = ode113(@(t,x) Xdyn(t,x,tf), [0 tf], [h0;v0;m0], options);
    subplot(221); plot(t,x(:,1),'linewidth',3);
    title('\textbf{a) Height}','interpreter','latex','FontSize',22,'FontWeight','bold');
    xlabel('\boldmath{$t$} \ \textbf{(s)}','interpreter','latex','FontSize',20,'FontWeight','bold');
    ylabel('\boldmath{$h$} \ \textbf{(m)}','interpreter','latex','FontSize',20,'FontWeight','bold');
    xlim([-inf inf]);
    ylim([-inf inf]);
    grid on;
    subplot(222); plot(t,x(:,2),'linewidth',3) ;
    title('\textbf{b) Velocity}','interpreter','latex','FontSize',22,'FontWeight','bold');
    xlabel('\boldmath{$t$} \ \textbf{(s)}','interpreter','latex','FontSize',20,'FontWeight','bold');
    ylabel('\boldmath{$v$} \ \textbf{(m/s)}','interpreter','latex','FontSize',20,'FontWeight','bold');
    xlim([-inf inf]);
    ylim([-inf inf]);
    grid on;
    subplot(223); plot(t,x(:,3),'linewidth',3) ;
    title('\textbf{c) Mass}','interpreter','latex','FontSize',22,'FontWeight','bold');
    xlabel('\boldmath{$t$} \ \textbf{(s)}','interpreter','latex','FontSize',20,'FontWeight','bold');
    ylabel('\boldmath{$m$} \ \textbf{(kg)}','interpreter','latex','FontSize',20,'FontWeight','bold');
    xlim([-inf inf]);
    ylim([-inf inf]);
    grid on;
    control = zeros(size(t));
    for i = 1:size(t)
        if t(i) <= tSw % Optimal control from our optimal policy
            control(i) = uMax;
        end
    end
    subplot(224); plot(t,control,'linewidth',3);
    title('\textbf{d) Optimal Control}','interpreter','latex','FontSize',22,'FontWeight','bold');
    xlabel('\boldmath{$t$} \ \textbf{(s)}','interpreter','latex','FontSize',20,'FontWeight','bold');
    ylabel('\boldmath{$u$}','interpreter','latex','FontSize',20,'FontWeight','bold');
    xlim([-inf inf]);
    ylim([-inf inf]);
    grid on;
end
```

```
Switching time tSw = 91.666667
Final time tf = 253.302002
```



PS 5 Problem 2 Part 2 Parts a and b

Wednesday, May 8, 2019 3:59 PM

Part a

repeating steps from P1

$$\dot{p}_y(t) = -\frac{\partial H}{\partial y} = 0 \quad \Rightarrow p_y(t) = \text{const} = c_1$$

From final BCs $\rightarrow$ $\because p_y(t_f) = -1 \quad \Rightarrow p_y(t) = -1$

$$\dot{p}_v(t) = -\frac{\partial H}{\partial v} = -p_y(t) \quad \Rightarrow p_v(t) = -c_1 t + c_2$$

$$= +t + c_2$$

$\because p_v(t_f) = 0 \quad \Rightarrow c_2 = -t_f$

$$\boxed{\begin{array}{ll} \Rightarrow p_v(t) = t - t_f & p_y(t) = -1 \\ \therefore \ p_v(0) = -t_f & p_y(0) = -1 \end{array}}$$

Part b

$\because$ final time is arbitrary & H does not explicitly depend on t, $\mathcal{H} = 0 \ \forall t$

$$\mathcal{H}(t) = 0$$

$$\Rightarrow \underbrace{p_y(t)}_{p_y(t)=-1} v(t) + \underbrace{p_v(t)}_{p_v(t)=t-t_f}\left(\frac{u(t)}{m(t)} - g\right) + p_m(t)\left(-b\underbrace{u(t)}_{u(0)=u_{max},\ u(t_f)=0}\right) = 0$$

@ $t=0$

$$\Rightarrow -1\, v_0 + (-t_f)\left(\frac{u_{max}}{m_0} - g\right) + p_m(0)(-b u_{max}) = 0$$

$$\Rightarrow \boxed{p_m(0) = \frac{-1}{b u_{max}}\left[v_0 + t_f\left(\frac{u_{max}}{m_0} - g\right)\right]}$$

Problem 2 Part 2 Part c– Shooting method solver

I implemented the following functions:

```
% Adjoints equations related to our rocket.

function zdot = Zdyn(t,z)

global g;
global b;
global uMax;

v = z(2);
m = z(3);
py = z(4);
pv = z(5);
pm = z(6);

% Compute phi
phi = pv/m - pm*b;
% Use phi to compute control action
if phi > 0
    uStar = 0;
else
    uStar = uMax;
end
% Rocket dynamics and adjoint equations
yDot = v;
vDot = uStar/m - g;
mDot = -b*uStar;
pyDot = 0;
pvDot = -py;
pmDot = pv*uStar/(m^2);

zdot = [yDot; vDot; mDot; pyDot; pvDot; pmDot];
```

```
% Hamiltonian related to the Goddard's problem.

function H = hamiltonianFunc(y,v,m,py,pv,pm)

global g;
global b;
global uMax;

% Compute phi
phi = pv/m - pm*b;

% Use phi to compute control action
if phi > 0
    uStar = 0;
else
    uStar = uMax;
end

% Return Hamiltonian H(y,v,m,ph,pv,pm)
H = py*(v) + pv*(uStar/m - g) +  pm*(-b*uStar);
```

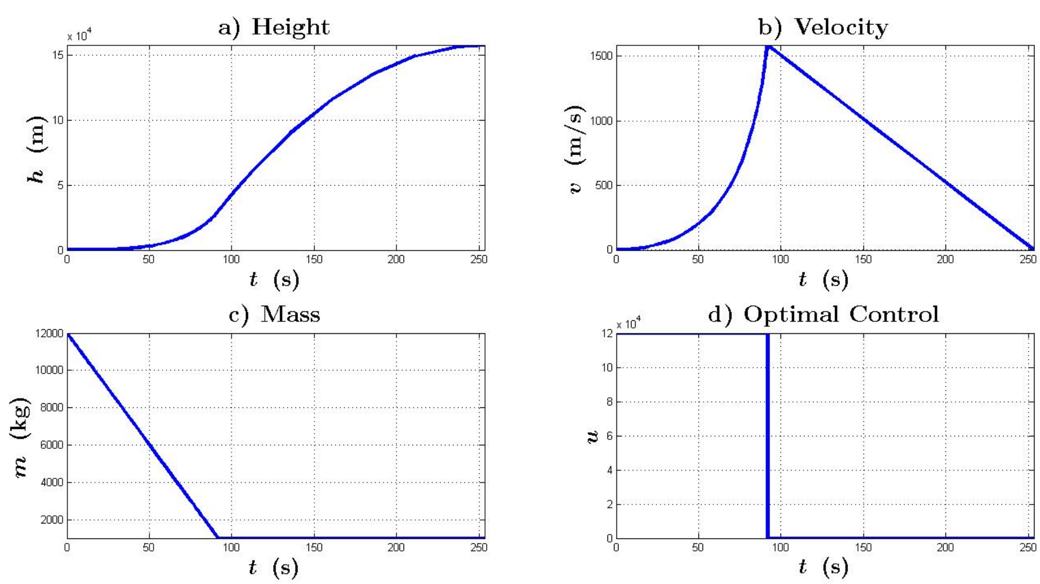
Results:

```
                                         Norm of      First-order   Trust-region
 Iteration  Func-count     f(x)          step         optimality    radius
     0          5      0.000198371                        33.5               1
     1         10      2.89571e-08     0.00202262        0.289               1
     2         15      3.77863e-09    1.34771e-06       0.0905               1
     3         16      3.77863e-09     0.00488286       0.0905               1
     4         17      3.77863e-09     0.00122072       0.0905         0.00122
     5         18      3.77863e-09    0.000305179       0.0905        0.000305
     6         19      3.77863e-09    7.62947e-05       0.0905        7.63e-05
     7         20      3.77863e-09    1.90737e-05       0.0905        1.91e-05
     8         21      3.77863e-09    4.76842e-06       0.0905        4.77e-06
     9         22      3.77863e-09    1.19211e-06       0.0905        1.19e-06
    10         23      3.77863e-09    2.98026e-07       0.0905        2.98e-07
    11         28      2.62575e-09    7.45066e-08       0.0125        7.45e-08
    12         33      1.68146e-09    7.45066e-08       0.0037        7.45e-08
    13         38      3.55833e-10    1.86266e-07       0.0134        1.86e-07
    14         43      1.31491e-11    1.76786e-07      0.00639        4.66e-07
    15         48      4.15071e-13    2.33601e-08      0.00116        4.66e-07
    16         49      4.15071e-13    4.15036e-09      0.00116        4.66e-07
    17         50      4.15071e-13    1.03759e-09      0.00116        1.04e-09
    18         51      4.15071e-13    2.59397e-10      0.00116        2.59e-10
    19         52      4.15071e-13    6.48493e-11      0.00116        6.48e-11
    20         53      4.15071e-13    1.62123e-11      0.00116        1.62e-11
    21         54      4.15071e-13    4.05308e-12      0.00116        4.05e-12
    22         55      4.15071e-13    1.01327e-12      0.00116        1.01e-12

Equation solved, fsolve stalled.

fsolve stopped because the relative size of the current step is less than the
selected value of the step size tolerance squared and the vector of function values
is near zero as measured by the default value of the function tolerance.



EXITFLAG =

     2

Switching time tSw = 91.666667
Final time tf = 253.303430
```

Switching time tSw = 91.666667

Final time tf = 253.303430

These are the same results as from the dichotomy approach.



Problem 2 Part 2 Part d– Sensitivity of guess

tf guess	Convergence reached?	# of iterations
`tf = 253.302`	**Yes**	**22**
`tf = 255`	**Yes**	**28**
`tf = 270`	**No**	**n/a**

This shows that the shooting method is very sensitive to our guess of tf.