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:

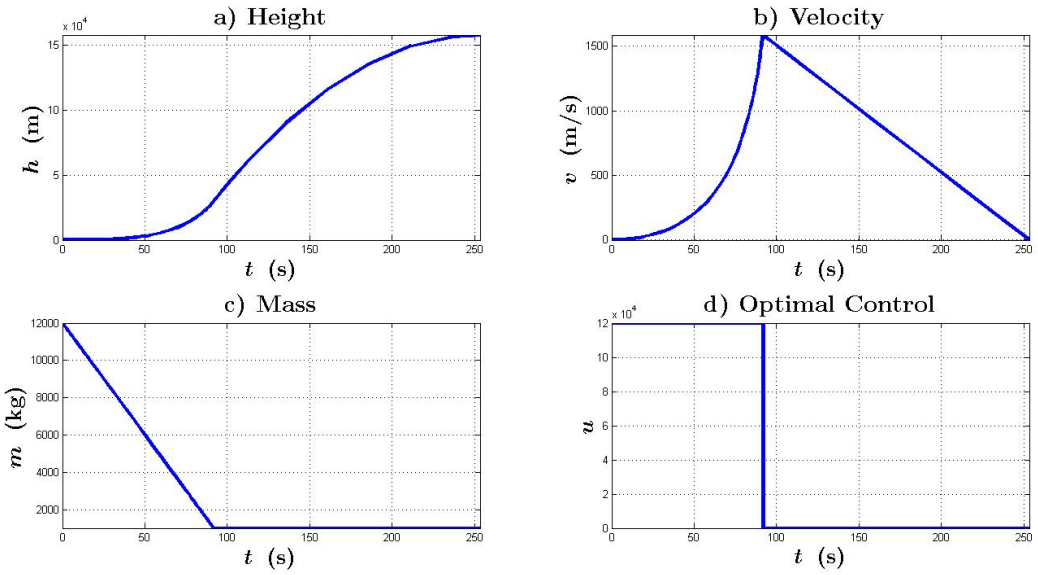
Iteration	Func-count	f(x)	Norm of step	First-order optimality	Trust-region radius
0	5	0.000198371		33.5	1
1	10	2.89571e-08	0.00202262	0.289	1
2	15	3.77863e-09	1.34771e-06	0.0905	1
3	16	3.77863e-09	0.00488286	0.0905	1
4	17	3.77863e-09	0.00122072	0.0905	0.00122
5	18	3.77863e-09	0.000305179	0.0905	0.000305
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.