# Model Adjustments

The model was adjusted slightly to the following:

```
f = @(t,x) [x(2);
        g + x(8)*cos(deg2rad(x(5))) / (M+x(7)) - c_d*x(2)*abs(x(2))*exp(-x(1)/10400);
        x(4);
        x(8)*sin(deg2rad(x(5))) / (M+x(7));
        x(6);
        k/(J+L^2*x(7))*x(9);
        -1/eta*(abs(x(8))+abs(x(9)));
        0;
        0];
```

abs() methods surround the inputs on the fuel state variable so that moving in either "negative" direction still decreases the amount of fuel in the system, instead of increasing it.

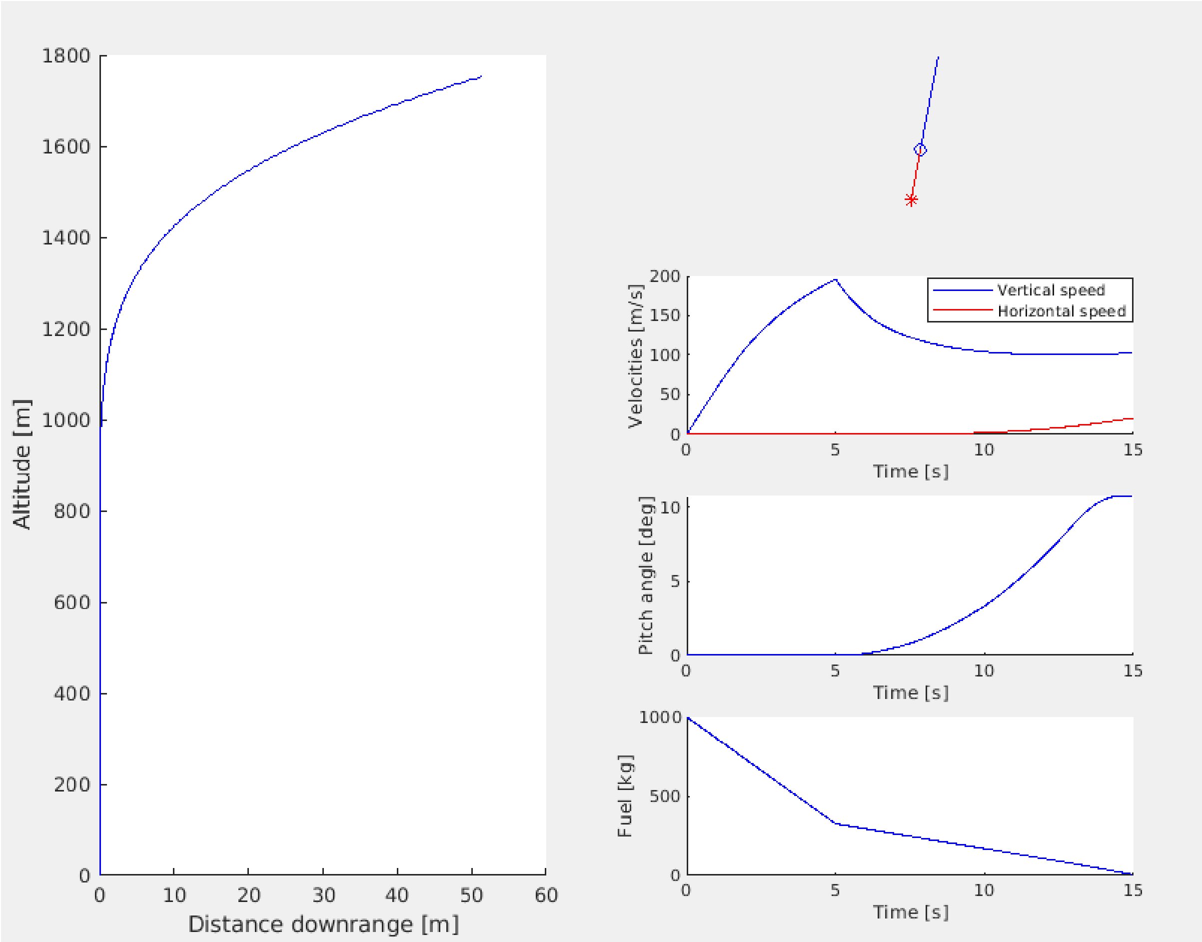
# Open Loop Control

To get a gauge for what we were aiming for in terms of system output, open loop control was implemented by first changing the inputs manually at certain time intervals.

```
if t < 5
        u1 = 135000;
        u2 = -1;
elseif t >= 13
        u1 = 30000;
        u2 = -5000;
else
        u1 = 30000;
        u2 = 1200;
End
```

This open loop control resulted in the below plot on the following page and the following vector.

```
[1752.47188667336;
 102.669466924392;
 51.4905670407928;
 20.9242177009967;
 10.6788061255270;
 -0.542669970767923;
 5.39500000000077]
```




The vector shows that this open loop controller is capable of closely meeting the requirements, it is just slightly off what we are aiming for.
The resulting output plots show that what is effective is a faster burst of fuel to get the height and then a slower burst of the main thruster to introduce the side thruster so that we can adjust for the angle and horizontal distance and velocity that is desired.

## Closed Loop Control Attempt One

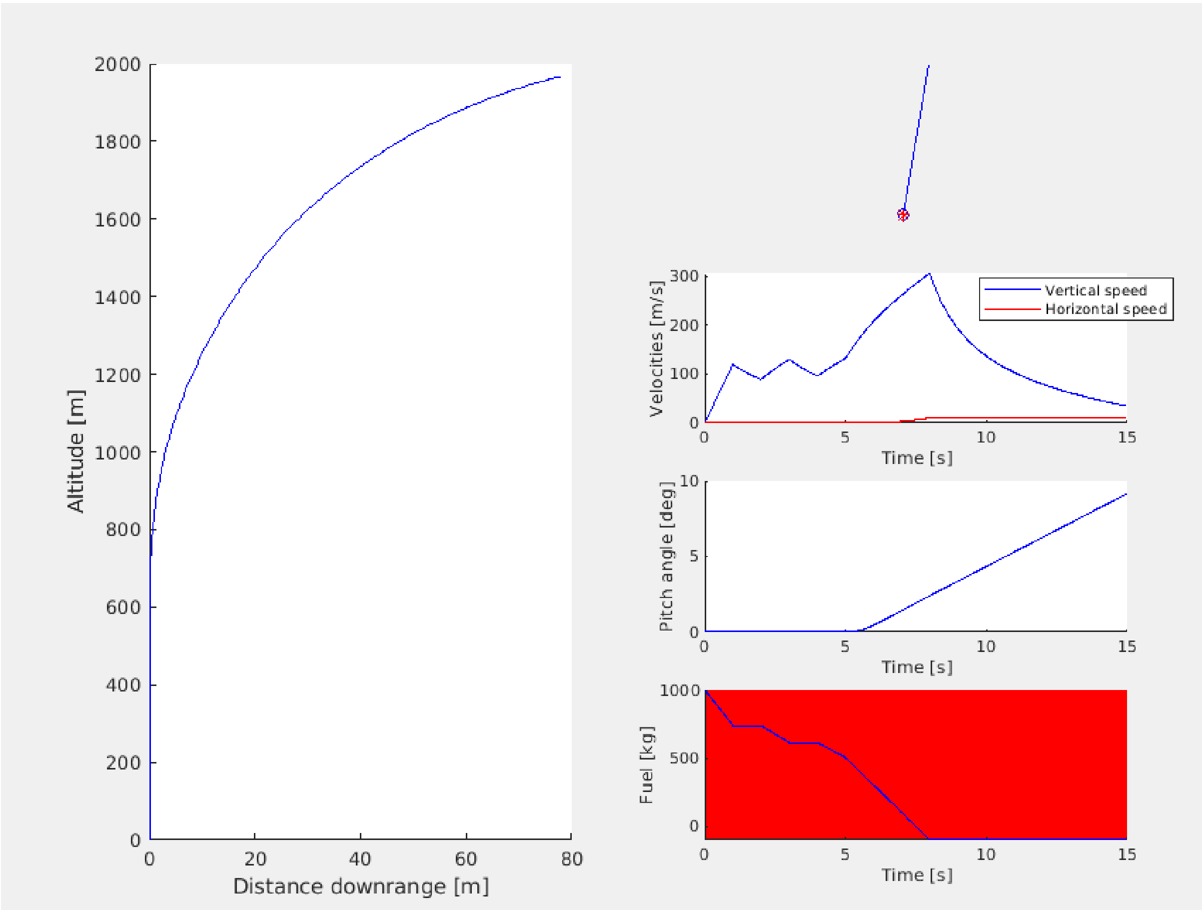
To preface this section, the closed loop controller used here was in fact not closed loop as I forgot to subtract the target from the current position.
Doing $u =- Kx$ and not $u =- K(x - x_{desired})$. However, it was possible to achieve most of the requirements through this method and too much time was spent on this bit. Also there are some fun erratic plots generated.

This iteration of attempting to solve this problem first started off with an initial open loop burst of the main thruster to gain some initial height and make the system more controllable as more state variables were in motion. This was important to do as it was found early on when attempting to control the rocket that it was more difficult to control and also reached lower heights if a closed loop approach was used from the start due to the majority of the A matrix being 0, as the inputs were tied to every state variable.

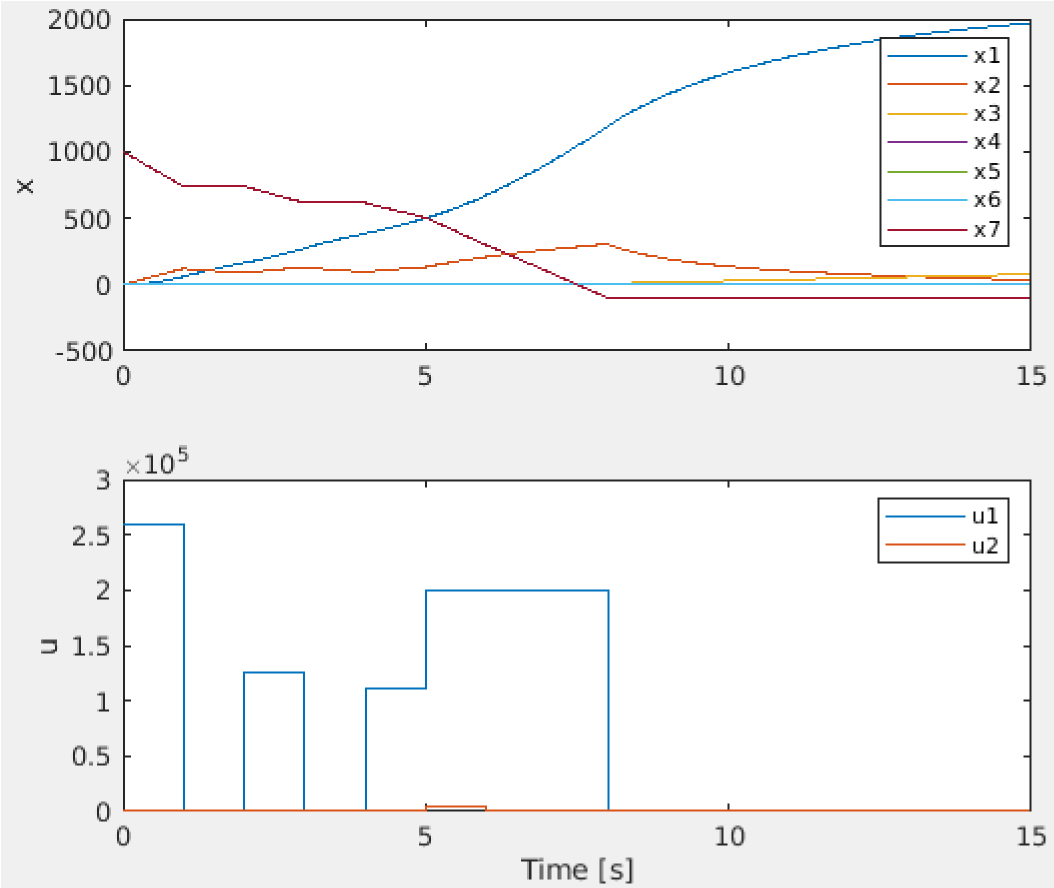
After the open loop height gain, the system was linearised at point t, which was done by creating the jacobians df/dx and dg/du to and then substituting in the current x vector (and u vector) to create the A and B matrices of this system.

From here it was decided that the LQR method would be used. So the next things created were Q and R matrices. The Q matrix was made up as a diagonal matrix with one on the x vector squared as its diagonal, with accompanying boundaries found through tuning of the system. The R matrix was made up in a similar way except it used the u vector instead of the x vector.

With these matrices made up the lqr method in matlab could be used to generate a K matrix. Which was then used to generate the input using $u = -Kx$. and the resulting plot is shown below




From this we can see where the control kicks in at around t=5 as the angle ramps up and the fuel gets immediately used up at a rate of 200kg per second, causing the rocket to top out its speed and then coast to an apogee that happens to have the correct angle and horizontal velocity and position, but not the correct angular velocity.

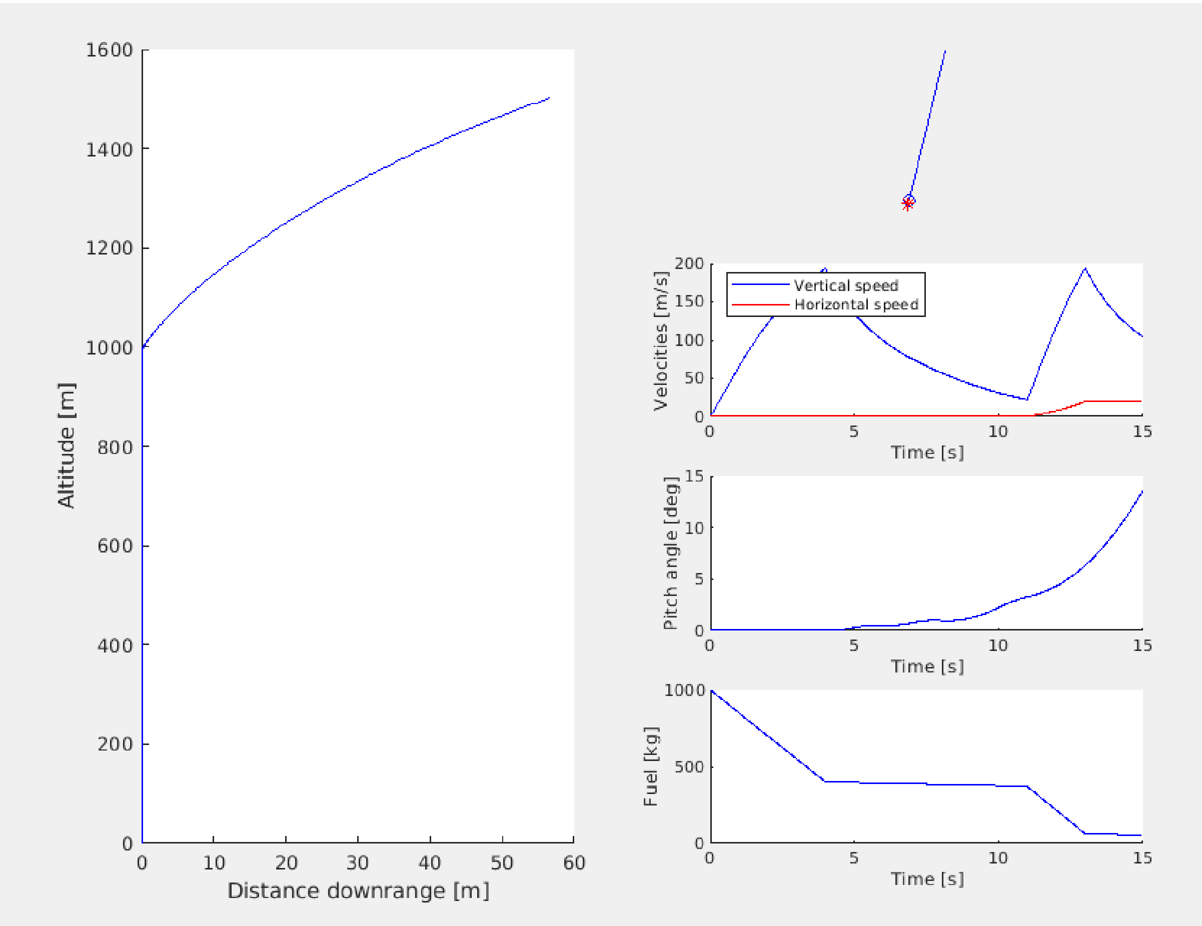



Presented here is the inputs for the rocket, and from here once the control kicks in, the side thruster is briefly pulsed and then it can't be used to slow down as the fuel gets used up. However, as mentioned this was not closed loop control so it was not capable of properly navigating towards the target and was just another method of open loop control.

## Closed Loop Control Attempt 2

Once the previous attempt was completed, it was realised where the mistake had been made and a proper closed loop controller using LQR was attempted. It was set up in the same way with an initial open loop control of the main thruster to gain height and vertical velocity, making it easier to meet those requirements and go as fast as possible. The only difference was that the target was subtracted from the current state vector when calculating the input.

Presented below is the final output vector and the final plot
1502.64043705452
103.627642496431
56.7432642308387
20.0922415493660
13.6307714428695
4.77168778712531
51.8035312046787




From these vectors we can see that the controller was capable of meeting all but two of the requirements, being the angle and angular velocity, and that is because this system was incredibly hard to tune and once you changed one thing slightly, it could have a significant effect on the rest of the system. Which is just the nature of coupled 7th order systems.

The first open loop stage can be seen  to happen when t < 5 which uses up more than half of the fuel but gets the rocket to just under 1km in altitude. The main thruster then turns off and the vertical velocity drops as the rocket adjusts the angle that it is pointing, before engaging the main thruster again. Once this happens it doesn't really properly adjust the angle which causes it to overshoot, which I haven't been able to solve with the controller. The engagement of the main thruster gives the rocket the horizontal speed it requires in addition to being the final burst of speed to get the rocket to go as fast as it can vertically. While this happens the fuel gets very close to zero.

As the regulator is a P controller but for state space control, we can expect the while the angle has overshot the target, that it will oscillate around the angle that we are aiming for. So it is possible that changing the angle earlier in the flight would be beneficial to allow this settling. However, that would affect the horizontal velocity and the distance travelled there. So a faster controller would serve to better improve this as it could reach close to the target faster before settling.