Voyage to Neptune

Team 160

Problem A

Abstract

Our spacecraft, the *Poseidon*, is traveling at 30 km/s upon reaching the outer atmosphere of Neptune. We define the radius of Neptune, R_N, as the distance from the center of mass at which pressure is equal to 1 bar; that is, $R_N \approx 24764$ km. We define the outer edge of the atmosphere to be at an altitude of approximately 700 km [1]. We call this altitude the atmospheric radius, $R_{\text{atm}} \approx 25464$ km. To clarify, the *Poseidon* is traveling at 30 km/s at a distance of 25464 km from the center of mass of Neptune. We wish to enter a circular orbit. Specifically, we wish to enter an orbit of relatively small radius, i.e., radius approximately equal to R_{atm}. To achieve this orbit, the *Poseidon* must decelerate significantly. If the only forces acting on the spacecraft are gravity and thrust, then the amount of fuel required to decelerate sufficiently is a significant percentage of the initial mass, $m_0 = 2500$ kg. We calculate a rough estimate of this percentage in Section 1. Another way to achieve the appropriate deceleration is to enter the Neptunian atmosphere where the resulting drag will decrease the velocity of the spacecraft at no fuel cost. This is, however, a potentially dangerous maneuver. If the drag force is too strong, the *Poseidon* might decelerate too quickly and be unable to escape the Neptunian atmosphere. On the other hand, if the drag force is too weak, the *Poseidon* will exit the Neptunian atmosphere without a significant decrease in velocity leaving the original problem unsolved. In Section 2.1 we derive the equations of motion for a spacecraft experiencing drag and gravitational attraction in the atmosphere of Neptune. As the outer regions of Neptune are better understood than the inner regions, our equations of motion are only valid between R_N and R_{atm}. Consequently, we require the *Poseidon* to remain at distances above R_N. Additionally, we require the exit speed, v_e, to be such that the spacecraft is able to enter a circular orbit at R_{atm} at a fuel cost of at most 20% of m_0. We compute the trajectories numerically. We find that the acceptable range of v_e does not allow for a trajectory falling below R_N; that is, although only valid between R_N and R_{atm}, our equations of motion suffice for this problem. Now, the trajectories depend on the angle, θ, at which the *Poseidon* enters the Neptunian atmosphere. We solve for the range of θ in which the *Poseidon* is able to emerge from the atmosphere with a velocity that allows for a transition to circular orbit at R_{atm} at a fuel cost of at most $0.2m_0$. We conclude

$$8.8296^\circ < \theta < 8.8549^\circ$$

yielding an allowed margin of error of 0.0253°. Consequently, we would advise against the employment of such a maneuver.

1 No Atmospheric Entry

How much fuel would be required to enter a circular orbit without entering the Neptunian atmosphere? Now, the *Poseidon* is traveling at $v_0 = 30$ km/s when it is at R_{atm} from the center of mass of Neptune. Let us evaluate the energy, E, of the *Poseidon* at this moment.

$$E = \frac{1}{2}mv_0^2 - \frac{GmM}{R_{\text{atm}}} \approx 4.54 \times 10^{11} \text{ J}. \tag{1.1}$$

That is, $E > 0$. If the *Poseidon* does not fire its boosters nor enter the Neptunian atmosphere, then it will continue traveling in a hyperbolic orbit about the center of mass of Neptune. Then, the minimum radius of this orbit is R_{atm}. A deceleration is required in order to enter a circular orbit. We only consider a change of orbit at the point at which the two orbits are tangent to one another as shown in Figure 1.1.

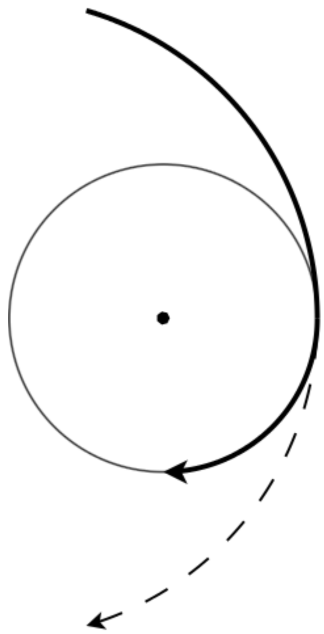


Figure 1.1: The spacecraft would have continued along a hyperbolic trajectory but instead enters a circular orbit at the point at which the two orbits are tangent to one another.

The spacecraft must effect a backward thrust in order to decelerate into a circular orbit of radius R_{atm}. We will calculate the amount of fuel required to achieve this.

For circular motion at radius R_{atm}, we require a final velocity, v_f, given by

$$\frac{mv_f^2}{R_{\text{atm}}} = \frac{GmM}{R_{\text{atm}}^2}. \tag{1.2}$$

That is,

$$v_f = \sqrt{\frac{GM}{R_{\text{atm}}}} \approx 16.38 \text{ km/s}. \tag{1.3}$$

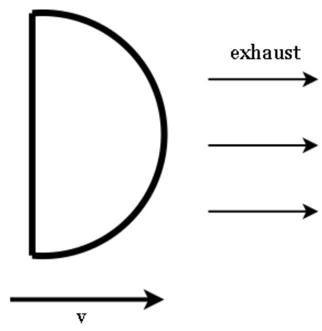


Figure 1.2: The spacecraft releases fuel in the forward direction which effects a backward thrust.

So we require a change in velocity $\Delta v \approx 13.62$ km/s.

The spacecraft expels fuel in the forward direction in order to decelerate as shown in Figure 1.2. We derive the rocket equation to describe the mass loss required to achieve a change in velocity of Δv. At some time, t, let $P(t)$ be the momentum of the *Poseidon* together with its remaining fuel. Let $m(t)$ be the mass and $v(t)$ be the velocity. Then,

$$P(t) = m(t)v(t). \tag{1.4}$$

At a later time, Δt, the *Poseidon* has ejected fuel of mass Δm at velocity v_{ex} and changed velocity by Δv. Then,

$$P(t + \Delta t) = (m - \Delta m)(v - \Delta v) + \Delta m(v + v_{\text{ex}}). \tag{1.5}$$

Then,

$$\Delta P = P(t + \Delta t) - P(t) = -m\Delta v + v_{\text{ex}}\Delta m + \Delta m \Delta v, \tag{1.6}$$

and this becomes

$$\dot{P} = -m\dot{v} - v_{\text{ex}}\dot{m}. \tag{1.7}$$

Now $\dot{P}$ is the component of the force on the *Poseidon* parallel to v. If we assume that the fuel ejection occurs in a short period of time during which the force of gravity due to Neptune is approximately perpendicular to v, then $\dot{P} \approx 0$. Then we have

$$m\dot{v} = v_{\text{ex}}\dot{m}. \tag{1.8}$$

We can solve this differential equation by separation of variables.

$$\int \dot{v}dt = v_{\text{ex}} \int \frac{\dot{m}}{m} dt. \tag{1.9}$$

If we integrate from initial velocity, v_0, to final velocity, v_f, and from initial mass, m_0, to final mass, m_f, we have

$$v_f - v_0 = v_{\text{ex}} \ln \frac{m_f}{m_0}. \tag{1.10}$$

That is,

$$-\Delta v = v_{\text{ex}} \ln \frac{m_f}{m_0}. \tag{1.11}$$

Solving for m_f yields

$$m_f = m_0 e^{-\Delta v / v_{\text{ex}}}. \tag{1.12}$$

We choose a typical exhaust speed $v_{\text{ex}} = 4.5$ km/s [3]. This yields

$$m_f \approx 121.19 \text{ kg}. \tag{1.13}$$

That is, the *Poseidon* must shed approximately 2378.81 kg or 95.15% of its initial mass.

2 Atmospheric Entry

Now we consider the case in which the *Poseidon* enters the Neptunian atmosphere. We begin with a derivation of the equations of motion, the greater part of which is devoted to an estimation of the drag force. The corresponding trajectories were computed numerically, and in Section 2.2 we describe our approach to this computation. In Section 2.3, we estimate the range of exit velocities which allow the *Poseidon* to enter a circular orbit at R_{atm} at a relatively low fuel cost. Finally, we define the allowed margin of error in Section 2.4.

2.1 Equations of Motion

When the *Poseidon* enters the atmosphere of Neptune, the forces acting on it are gravity and drag. Let the center of mass of Neptune be the origin of a polar coordinate system, and let $\mathbf{r}$ be the position vector of the *Poseidon*. Let $\hat{\mathbf{r}}$ be the unit vector parallel to $\mathbf{r}$ and let r be the magnitude. That is,

$$\mathbf{r} = r\hat{\mathbf{r}}. \tag{2.1}$$

Let M be the mass of Neptune, m the mass of the *Poseidon* and G the gravitational constant ($G \approx 6.67 \times 10^{-11}$ N·m^2/kg^2). Then the force of gravity, $\mathbf{F}_g$, on the *Poseidon* is given by

$$\mathbf{F}_g = -\frac{GmM}{r^2}\hat{\mathbf{r}}. \tag{2.2}$$

Let $\mathbf{v} = \dot{\mathbf{r}}$, and let $\hat{\mathbf{v}}$ be the unit vector parallel to $\mathbf{v}$. Let ρ be the density of the Neptunian atmosphere, A the cross-sectional area of the *Poseidon* perpendicular to $\mathbf{v}$, and c_d the coefficient of quadratic drag. Then, the drag force, $\mathbf{F}_d$, is given by

$$\mathbf{F}_d = -c_d \frac{\rho A \mathbf{v}^2}{2}\hat{\mathbf{v}}. \tag{2.3}$$

To find ρ, recall the ideal gas law

$$PV = Nk_BT, \tag{2.4}$$

where P is pressure, V is volume, N is the number of molecules, k_B is the Boltzmann constant ($k_B \approx 1.38 \times 10^{-23}$ J/K) and T is temperature. Let μ be the mass of each molecule. Then the mass density, ρ, is given by

$$\rho = \frac{N\mu}{V} = \frac{P\mu}{k_BT}. \tag{2.5}$$

In the Neptunian atmosphere, it has been found that $\log_{10} P$ decreases approximately linearly as altitude increases. Let h be altitude. Now, the radius of Neptune is defined as the distance from the center of mass at which $P = 1$ bar $= 10^5$ Pa. This corresponds to $h = 0$. Additionally, we notice that with $h \approx 600$ km we have $P \approx 10^{-4}$ mbar $= 0.01$ Pa [1]. Fixing these points, we approximate the dependence as

$$P(h) = \alpha e^{-h/\beta}, \tag{2.6}$$

where $\alpha = 10^5$ Pa and $\beta \approx 3.72 \times 10^4$ m.

It has also been found that in the Neptunian atmosphere T increases approximately linearly with h. We notice that with $h \approx 0$ km we have $T \approx 50$ K; with $h \approx 700$ km we have $T \approx 260$ K [1]. Fixing these points, we approximate the dependence as

$$T(h) = ah + b, \tag{2.7}$$

where $a \approx 0.3 \times 10^{-3}$ K/m and $b \approx 50$ K.

The atmosphere of Neptune is about 80% hydrogen (H_2) and 19% helium (He) [6]. To simplify our calculations, we assume the atmosphere is composed entirely of H_2. Let M_{H_2} be the mass of one molecule of H_2. Then,

$$\mu = M_{H_2} \approx 2 \text{ u} \approx 3.32 \times 10^{-27} \text{ kg}. \tag{2.8}$$

From (2.5), (2.6), (2.7) and (2.8) we have

$$\rho(h) = \frac{M_{H_2} P(h)}{k_B T(h)}. \tag{2.9}$$

We assume the perpendicular cross-section of the *Poseidon* is circular with diameter approximately 3 m. We choose this value as it is comparable to the size of Voyager 2 [7]. Then,

$$A = \pi (1.5 \text{ m})^2 \approx 7.07 \text{ m}^2. \tag{2.10}$$

We approximate the shape of the *Poseidon* as half-spherical yielding a drag coefficient

$$c_d \approx 0.4.[2]. \tag{2.11}$$

Define

$$\sigma \equiv c_d \frac{M_{H_2} A}{2k_B}. \tag{2.12}$$

From (2.8), (2.10) and (2.11) we have

$$\sigma \approx 3.40 \times 10^{-4}. \tag{2.13}$$

Then

$$\mathbf{F}_d = -\sigma \mathbf{v}^2 \frac{P(h)}{T(h)} \hat{\mathbf{v}}. \tag{2.14}$$

We make a change of variables to Cartesian coordinates as shown in Figure 2.1. Without loss of generality, we assume the velocity of the *Poseidon* has a non-negative component in

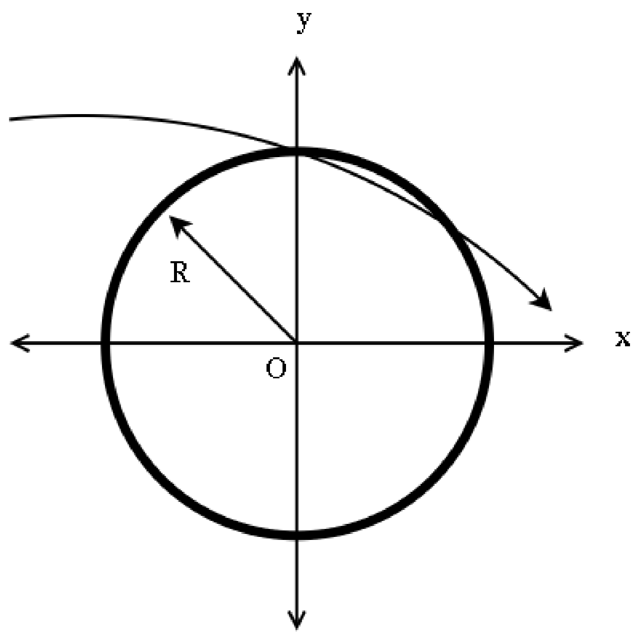


Figure 2.1: The spacecraft enters the atmosphere at $(x, y) = (0, R)$ where R is the distance from the center of mass of Neptune to the edge of the outer atmosphere.

the positive x direction. Let $\hat{\imath}$ be the unit vector parallel to the x-axis, and let $\hat{\jmath}$ be the unit vector parallel to the y-axis. Then

$$\mathbf{r} = x\hat{\imath} + y\hat{\jmath}, \qquad \dot{\mathbf{r}} = \dot{x}\hat{\imath} + \dot{y}\hat{\jmath}, \qquad \ddot{\mathbf{r}} = \ddot{x}\hat{\imath} + \ddot{y}\hat{\jmath}. \tag{2.15}$$

Additionally, we have

$$r = \sqrt{x^2 + y^2}, \qquad \hat{r} = \frac{\mathbf{r}}{r} = \frac{x\hat{\imath} + y\hat{\jmath}}{\sqrt{x^2 + y^2}}, \tag{2.16}$$

$$|\mathbf{v}| = \sqrt{\dot{x}^2 + \dot{y}^2}, \qquad \hat{v} = \frac{\mathbf{v}}{|\mathbf{v}|} = \frac{\dot{x}\hat{\imath} + \dot{y}^2\hat{\jmath}}{\sqrt{\dot{x}^2 + \dot{y}^2}}. \tag{2.17}$$

Then, (2.2) and (2.14) become

$$\mathbf{F}_g = -\frac{GmM}{(x^2 + y^2)^{3/2}}(x\hat{\imath} + y\hat{\jmath}), \qquad \mathbf{F}_d = -\sigma\frac{P(h)}{T(h)}\sqrt{\dot{x}^2 + \dot{y}^2}(\dot{x}\hat{\imath} + \dot{y}\hat{\jmath}). \tag{2.18}$$

And, finally, the equations of motion:

$$\begin{cases} m\ddot{x} = & -\dfrac{GmMx}{(x^2 + y^2)^{3/2}} - \sigma\dot{x}\dfrac{P(h)}{T(h)}\sqrt{\dot{x}^2 + \dot{y}^2} \\ m\ddot{y} = & -\dfrac{GmMy}{(x^2 + y^2)^{3/2}} - \sigma\dot{y}\dfrac{P(h)}{T(h)}\sqrt{\dot{x}^2 + \dot{y}^2}. \end{cases} \tag{2.19}$$

2.2 Simulation Using MatLab

In order to simulate the trajectory of the spacecraft as it moves through Neptune's atmosphere, the equations of motion were numerically integrated using Euler's method. Specifically, time was discretely incremented in steps of $\Delta t = 0.001$ s, and at each time increment the following iterations occurred:

$$\ddot{x}(t_{k+1}) = f(x(t_k), \dot{x}(t_k)),$$

$$\dot{x}(t_{k+1}) = \dot{x}(t_k + \Delta t) = \dot{x}(t_k) + (\Delta t)\ddot{x}(t_k),$$
$$x(t_{k+1}) = x(t_k + \Delta t) = x(t_k) + (\Delta t)\dot{x}(t_k),$$

where f is some function of x and $\dot{x}$.

These calculations were performed using MATLAB code written for this problem (see Appendix). In the code, the initial position of the spacecraft is set at $(x_0,\, y_0) = (0,\, R_{\text{atm}})$ and an entrance angle (below the horizontal) is specified by the user to give the starting velocity in each direction.

The code is composed so that, when the spacecraft is within the outer atmosphere, both drag and gravity act upon it, whereas when it is outside the outer atmosphere, only gravity is has an effect. The simulation does not model what happens after the spacecraft moves out of the outer atmosphere towards the center of Neptune. For each entrance angle, the program calculates the velocity at which it leaves the outer atmosphere (exit velocity), the angle between the exit direction and the tangent line at the point of exit (exit angle), and the time elapsed before exit (exit time). Note that the spacecraft may leave the outer atmosphere into space or into Neptune.

2.3 Estimating the Exit Velocity

Within an appropriate range of incident angle, θ, the spacecraft will exit at a dampened velocity. We will see in Section 3 that the exit angle is relatively small (less than 10°). Thus, the direction the *Poseidon* is traveling upon exiting the Neptunian atmosphere is roughly parallel to the desired circular orbit at radius R_{atm}. Define this horizontal component of the velocity to be v_e. If this exit velocity is different from the final velcity v_f we need to stay in this orbit, the spacecraft needs to fire its boosters in order to make a speed adjustion. Here two cases might appear: if v_e is greater than v_f, then the spacecraft needs to fire its booster forward in order to slow down further. However, the speed v_e is already smaller than the entrance velocity 30 km/s, the spacecraft now needs to use less fuel to slow down to v_f. On the other hand, if v_e is smaller, but not significantly smaller than v_f, then the spacecraft could fire its booster backwards in order to catch up v_f.

Since the purpose of the trajectory we designed is to save a significant amount of fuel, we can require the spacecraft to use no more than a designated protion of its fuel, and obtain the allowed exit speeds that can be adjusted to v_f. However, a "significant" saving of fuel is not quantitative enough. For a fair estimation we decided to set 20% as the allowed mass loss of the spacecraft, i.e.,

$$m_f = .8m_0. \tag{2.20}$$

With the rocket equation (1.11) we can calculate the maximum exit speed $v_{\max}$ and the minimum exit speed $v_{\min}$ allowed. According to the equation (1.11), $v_{\max}$ satisfy the following relations:

$$v_f - v_{\max} = v_{\text{ex}} \ln \frac{m_f}{m_0}. \tag{2.21}$$

That is,

$$\begin{aligned} v_{\max} &= v_f + v_{\text{ex}} \ln \frac{m_0}{m_f} \\ &= 16.38 + 4.5 \ln(10/8) \text{ km/s} \\ &= 17.3841 \text{ km/s}. \end{aligned} \tag{2.22}$$

According to the rocket equation, $v_{\min}$ satisfies the following relations:

$$v_f - v_{\min} = v_{\text{ex}} \ln \frac{m_0}{m_f}. \tag{2.23}$$

That is,

$$\begin{aligned} v_{\min} &= v_f - v_{\text{ex}} \ln \frac{m_0}{m_f} \\ &= 16.38 - 4.5 \ln(10/8) \text{ km/s} \\ &= 15.3759 \text{ km/s}. \end{aligned} \tag{2.24}$$

2.4 The Allowed Margin of Error

Consider an entrance angle, θ_a, such that the corresponding exit speed is bounded by $v_{\min}$ and $v_{\max}$. Then θ_a is an allowed entrance angle. Let $\theta_{\max}$ be the least upper bound on θ_a, and let $\theta_{\min}$ be the greatest lower bound on θ_a. Then the margin of error, $\Delta\theta_a$, is given by

$$\Delta\theta_a = \theta_{\max} - \theta_{\min}. \tag{2.25}$$

3 Results

In Section 2.3, we found upper and lower bounds on an acceptable exit velocity. In particular,

$$v_{\max} \approx 17.384 \text{ km/s}, \qquad v_{\min} \approx 15.376 \text{ km/s}. \tag{3.1}$$

Now each trajectory maps an entrance angle, θ, to an exit velocity, v_e. We are interested in the entrance angles corresponding to exit speeds $v_{\max}$ and $v_{\min}$, respectively. Now, exit speeds are associated with exit angles. Then for some exit speed, the exit angle is $0°$. We call this the critical velocity, v_c, and we call the associated entrance angle the critical angle, θ_c. We show below that exit velocity is directly proportional to exit angle. That is, if $v_e > v_c$, then the *Poseidon* will escape the atmosphere; otherwise, the *Poseidon* cannot escape. Additionally, if $v_c > v_{\min}$, then no trajectory allows the spacecraft to exit the atmosphere with velocities between $v_{\min}$ and v_c. Now, when the acceptable exit velocity is at a maximum, θ is at a minimum. We call this angle $\theta_{\min}$. Similarly, when the acceptable exit velocity is at a minimum, we call the corresponding entrance angle $\theta_{\max}$. Then, $\theta_{\min}$ corresponds to $v_{\max}$. As we will see, $v_c > v_{\min}$; consequently, $\theta_{\max}$ does not correspond to $v_{\min}$ but instead to v_c.

3.1 Bounds on θ_{min}

Figure 1 (p. 16) shows a trajectory with $\theta = 8.8295°$ and $v_e > v_{\mathrm{max}}$. On the other hand, Figure 2 (p. 17) shows a trajectory with $\theta = 8.8296°$ and $v_e < v_{\mathrm{max}}$. We conclude then that

$$8.8295° < \theta_{\mathrm{min}} < 8.8296°. \tag{3.2}$$

3.2 Bounds on θ_{max}

Figures 3 (p. 18) and 4 (p. 19) show the boundary between trajectories that escape and trajectories that do not escape. With $\theta = 8.8549°$, the *Poseidon* escapes the atmosphere with $v_e \approx 16.01$ km/s. That is, $v_e > v_{\mathrm{min}}$. Then we see that $v_c < 16.01$ km/s. With $\theta = 8.8550°$, the *Poseidon* does not emerge from the Neptunian atmosphere. Therefore,

$$8.8549 < \theta_{\mathrm{max}} < 8.8550. \tag{3.3}$$

3.3 Extreme Trajectories

Figure 5 (p. 20) shows a trajectory passing through the Neptunian atmosphere with negligible deceleration. The *Poseidon* enters the atmosphere at an angle of 7° and exits with a velocity of 29.88 km/s. That is, the decrease in velocity is less than 0.5%.

4 Uncertainty Estimation

First of all, there is an inevitable uncertainty involved due to limited precision of the computer numerical simulations. Different time steps in the Euler method will result in different levels of precision of our trajectory. The level of precision of our simulated trajectories can be estimated in the following standard way [4].

In our simulation, the time step chosen was $\Delta t = 0.001$ s. We first ran a simulation with a specific initial condition (with a fixed entrance angle). Then we ran a simulation with the same initial condition with a time step only half as big($\Delta t = 0.0005$ s) and compared the two results to see which digits were in agreement. In this way, we were able to assess the effect of the time step on our result and determined those digits in agreement to be accurate. For example, in one following simulation we selected entrance angle $\theta = 8.84°$ (see Figure 6 on p. 21). Notice the exit speed of the two simulations at the two time steps: 16876.8624 m/s and 16876.5332 m/s. Because the first five digits agree, we take those to be the correct values.

The longest trajectory in our simulation was the one with the critical entrance angle. Since the simulation was longer, error had longer to accumulate and our final result was accordingly less precise. Repeating the analysis above, we found four significant figures for our final velocity result. Because the uncertainty is greatest in this critical case, we conclude that we have a precision of about 0.001 km/s for the exit speed through out the simulation. Using the same technique, we conclude we have a precision of about 1 s for the exit time and a precision greater than 0.1° for the exit angle in all our calculations.

We also need to note that there are many other factors that may cause extra uncertainties we haven't taken into account. For example, the rotation of Neptune, the winds in the atmosphere (which can reach speeds up to 600 m/s[5]). Additionally, we assumed the shape of the *Poseidon* to be relatively simple, and we made approximations regarding the temperature and density of the Neptunian atmosphere. These approximations can be improved in a more detailed analysis.

5 Conclusion

From (3.2) and (3.3), we determine that the range of entrance angles which allow the *Poseidon* to emerge from the Neptunian atmosphere with a significantly decreased velocity to be

$$8.8296^\circ < \theta < 8.8549^\circ. \tag{5.1}$$

where we have chosen the upper bound on θ_{min} as the lower bound on θ and the lower bound on θ_{max} as the upper bound. Then the allowed margin of error is

$$\Delta\theta = 0.0253^\circ. \tag{5.2}$$

If the *Poseidon* can achieve a trajectory in this range, then we estimate the amount of fuel required to enter a circular orbit at R_{atm} to be at most 500 kg or 20% of the initial mass. This can be compared with our estimate from Section 1 of the amount of fuel required to change orbits without entering the Neptunian atmosphere: approximately 95% of the initial mass. As $\Delta\theta$ is so small, we would advise against relying on the drag resistance of the Neptunian atmosphere to decelerate the *Poseidon*.

We should mention that we have assumed that the desired circular orbit has radius approximately equal to R_{atm}. Our analysis can be extended to consider orbits of larger (or smaller) radii. This will produce a much larger allowed margin of error.

Appendix

Below we provide the MATLAB code used to generate the figures and data of Section 3.

constants and initializations

```
tstep = 0.001; %time step for Euler's method, s

mR = 2500; %kg; mass of spacecraft
mN = 1.02*10^26; %kg; mass of Neptune
rN = 24764000; % m; radius of Neptune
hatm = 700000; %m; height of Neptune's atmosphere.
G = 6.67300*10^-11; %m^3 kg^-1 s^-2; gravitational constant

angle = 8.84;
%degrees; angle (below horizontal) of rocket's entrance to the atmosphere
```

vector equations

```
time = 1000; %time of simulation
tlength = time/tstep;

x = zeros(1,tlength); x(1) = 0; % m
y = zeros(1,tlength); y(1) = rN + hatm; % m

xdot = zeros(1,tlength); xdot(1) = cosd(angle)*30000; % m/s
ydot = zeros(1,tlength); ydot(1) = -sind(angle)*30000; % m/s

xddot = zeros(1, tlength); % m/s^2
yddot = zeros(1,tlength); % m/s^2

k = 1; % index variable
outpoint = 0; % atmosphere exit toggle
inpoint = 0; % Neptune entrance toggle

while k <= tlength && inpoint == 0


 % inside outer atmosphere
 if (sqrt(x(k)^2 + y(k)^2)) <= rN+hatm && (sqrt(x(k)^2 + y(k)^2)) >= rN;

 c = 68.57 * (exp(-(sqrt(x(k)^2+y(k)^2) - rN)/(3.72*10^4)))/
     (0.0003*(sqrt(x(k)^2+y(k)^2) - rN) + 50); %drag force

 xddot(k) = (1/mR)*((-G*mN*mR*x(k) )/(x(k)^2+y(k)^2)^(3/2) -
            c*sqrt((xdot(k)^2 + ydot(k)^2))*xdot(k));
 yddot(k) = (1/mR)*((-G*mN*mR*y(k) )/(x(k)^2+y(k)^2)^(3/2) -
            c*sqrt((xdot(k)^2 + ydot(k)^2))*ydot(k));

 xdot(k+1) = xdot(k) + tstep*xddot(k);
 ydot(k+1) = ydot(k) + tstep*yddot(k);

 x(k+1) = x(k) + tstep*xdot(k);
 y(k+1) = y(k) + tstep*ydot(k);


 % in space
 elseif (sqrt(x(k)^2 + y(k)^2)) > rN+hatm

 xddot(k) = (1/mR)*((-G*mN*mR*x(k) )/(x(k)^2+y(k)^2)^(3/2));
 yddot(k) = (1/mR)*((-G*mN*mR*y(k) )/(x(k)^2+y(k)^2)^(3/2));
```

```
 xdot(k+1) = xdot(k) + tstep*xddot(k);
 ydot(k+1) = ydot(k) + tstep*yddot(k);

 x(k+1) = x(k) + tstep*xdot(k);
 y(k+1) = y(k) + tstep*ydot(k);

  if outpoint == 0 % notes initial exit point

     outpoint = k;
     exitv = sqrt(xdot(outpoint)^2 + ydot(outpoint)^2); % exit velocity

        if sign(xdot(outpoint)*ydot(outpoint)) < 0
            % exit angle from horizontal
            exita =  -atand(ydot(outpoint)/xdot(outpoint));
        else
            exita =  180 - atand(ydot(outpoint)/xdot(outpoint));
        end
  end

 else % spacecraft contacts Neptune

        inpoint = k; % contact point
        exitv = sqrt(xdot(inpoint)^2 + ydot(inpoint)^2); % contact velocity

        if sign(xdot(inpoint)*ydot(inpoint)) < 0
            % contact angle from horizontal
            exita =  atand(ydot(inpoint)/xdot(inpoint));
        else
            exita =  180 - atand(ydot(inpoint)/xdot(inpoint));
        end
    end

    k = k + 1; % index step

end

clear xd* yd*
```

generating the circumference of Neptune and tangent line

```
a = 0:100:rN+hatm; % outer atmosphere
a2 = 0:100:rN; % surface
b = zeros(1,length(a)); % first quadrant
b2 = zeros(1,length(a2));
b3 = zeros(1,length(a)); % fourth quadrant
```

```
b4 = zeros(1,length(a2));

b(1:end) = sqrt((rN+hatm)^2 - a(1:end).^2);
b2(1:end) = sqrt((rN)^2 - a2(1:end).^2);
b3(1:end) = -sqrt((rN+hatm)^2 - a(1:end).^2);
b4(1:end) = -sqrt((rN)^2 - a2(1:end).^2);

xt = 0:50:round(max(x)*1.1); % tangent line at point of exit/contact

if outpoint ~= 0
    m = -x(outpoint)/y(outpoint);
    yt = m*xt + (x(outpoint)^2 + y(outpoint)^2)/(y(outpoint));
    angt = atand(-m);

elseif inpoint ~= 0
    m = -x(inpoint)/y(inpoint);
    yt = m*xt + (x(inpoint)^2 + y(inpoint)^2)/(y(inpoint));
    angt = atand(-m);

else
    yt = zeros(1,length(xt));
    angt = 0;
end
```

plotting the figure

```
if inpoint ~=0 % truncates simulation upon contact
    x = x(1:inpoint);
    y = y(1:inpoint);
end


h = figure(1);
set(h, 'color', 'white');

hold on

p1 = plot(a2,b2); %
set(p1,'Color','blue','LineWidth',1.5,'LineStyle', '-')

p2 = plot(a,b); %
set(p2,'Color','cyan','LineWidth',1.5,'LineStyle', '-')

p3 = plot(x,y); %
set(p3,'Color','red','LineWidth',1.5,'LineStyle', '--')
```

```
%p4 = plot(xt,yt); % tangent line/not plotted
%set(p4,'Color','green','LineWidth',1,'LineStyle', '-')

if outpoint ~= 0 % plots exit/contact point with marker
    p5 = plot(x(outpoint),y(outpoint));
    set(p5,'Color','black','LineWidth',5,'Marker', '+')
elseif inpoint ~=0
    p5 = plot(x(inpoint),y(inpoint));
    set(p5,'Color','black','LineWidth',5,'Marker', '+')
end


if outpoint ~= 0 % automatically sets axes
    axis([min(x) round(x(outpoint)*1.1) round(y(outpoint)*.9) max(y)])
elseif inpoint ~= 0
    if y(inpoint) > 0
    axis([min(x) round(x(inpoint)*1.1) round(y(inpoint)*.9) max(y)])
    else
    axis([min(x) round(max(x)*1.1) round(y(inpoint)*1.1) max(y)])
    end
else
    axis([min(x) max(x) min(y) max(y)])
end


p6 = plot(a2,b4); % plots fourth quadrant of planet (if needed)
set(p6,'Color','blue','LineWidth',1.5,'LineStyle', '-')

p7 = plot(a,b3);
set(p7,'Color','cyan','LineWidth',1.5,'LineStyle', '-')



xlabel('x - distance (m)', 'FontSize', 14, 'FontName',
       'Arial','FontWeight', 'bold')
ylabel('y - distance (m)', 'FontSize', 14,'FontName',
       'Arial','FontWeight', 'bold')


hold off


if outpoint~=0
    name0 = 'The spacecraft exits the atmosphere of Neptune';
```

```
    name1 = ['Entrance Angle: ' num2str(angle) '  Exit Time: '
            num2str(outpoint*tstep) ' s' ] ;
    name2 = ['Exit Speed:' num2str(exitv) ' m/s  Exit Angle: '
            num2str(angt - exita) '' ];

elseif inpoint ~=0
    name0 = 'The spacecraft enters uncharted areas of the atmosphere';
    name1 = ['Entrance Angle: ' num2str(angle) '  Contact Time: '
            num2str(inpoint*tstep) ' s' ] ;
    name2 = ['Contact Speed:' num2str(exitv) ' m/s  Contact Angle: '
            num2str(exita - angt) ''];

else
    name0 = '';
    name1 = 'The simulation did not run long enough.';
    name2 = ['Simulation time = ' num2str(tlength*tstep) ' s'];

end
title({name1, name2, ' '},'FontSize', 14,'FontName', 'Arial', 'FontWeight',
      'bold')
legend('Surface of Neptune','Atmosphere of Neptune','Trajectory')

filename = ['spacecraft' num2str(angle) '.png'] ;
print(h, filename, '-dpng');
```

References

[1] J. Lunine. The atmospheres of Uranus and Neptune, *Annu. Rev. Astron. Astrophys* 1993. 31:217-63

[2] J. Scott. "Drag of Cylinders & Cones". Aerospaceweb.org. June 5th 2005. http://www.aerospaceweb.org/question/aerodynamics/q0231.shtml (access time: Nov. 7th 2010)

[3] "Ion thruster - Definition". WordIQ.com. 2010. http://www.wordiq.com/definition/Ion_thruster (access time: Nov. 7th 2010)

[4] Greenspan, D. (2008) *Euler's Method, in Numerical Solution of Ordinary Differential Equations*, Wiley-VCH Verlag GmbH, Weinheim, Germany. doi: 10.1002/9783527618774.ch1

[5] Suomi, V. E.; Limaye, S. S.; Johnson, D. R. (1991). *"High Winds of Neptune: A possible mechanism"*. Science (AAAS (USA)) 251 (4996): 929932. doi:10.1126/science.251.4996.929

[6] *"Neptune"*. The Internet Encyclopedia of Science. `http://www.daviddarling.info/encyclopedia/N/Neptune.html` (Retrieved 2010-11-5).

[7] Angrum, Andrea. (November 07, 2010). *"Voyager, The Interstellar Mission"*. Jet Propulsion Laboratory. NASA. `http://voyager.jpl.nasa.gov/faq.html` (Retrieved 2010-11-5).

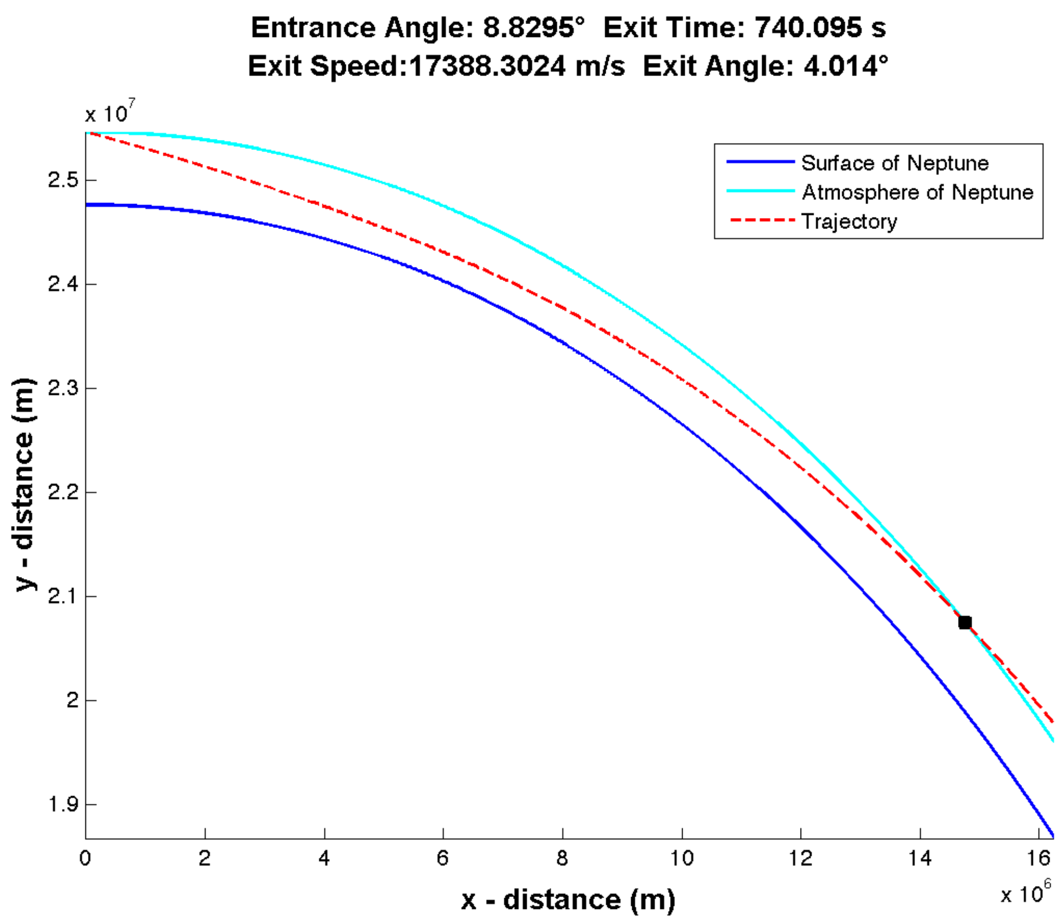


Figure 1

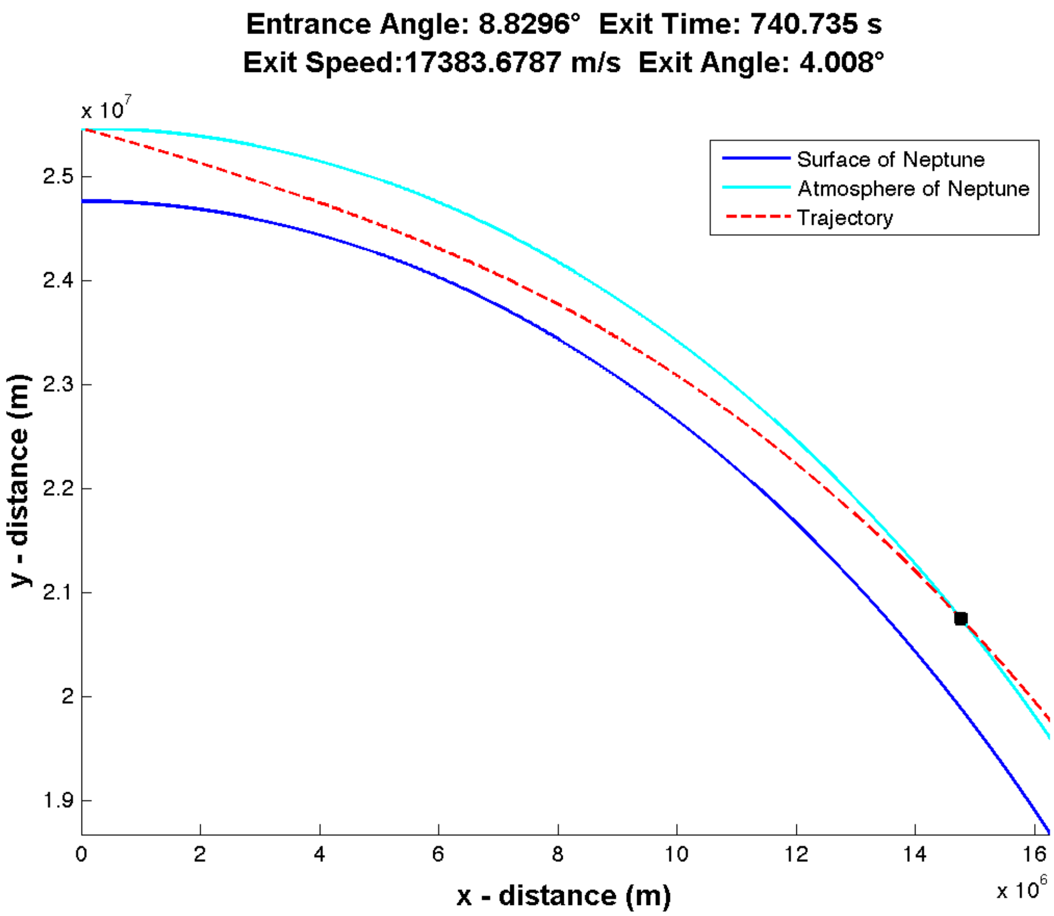
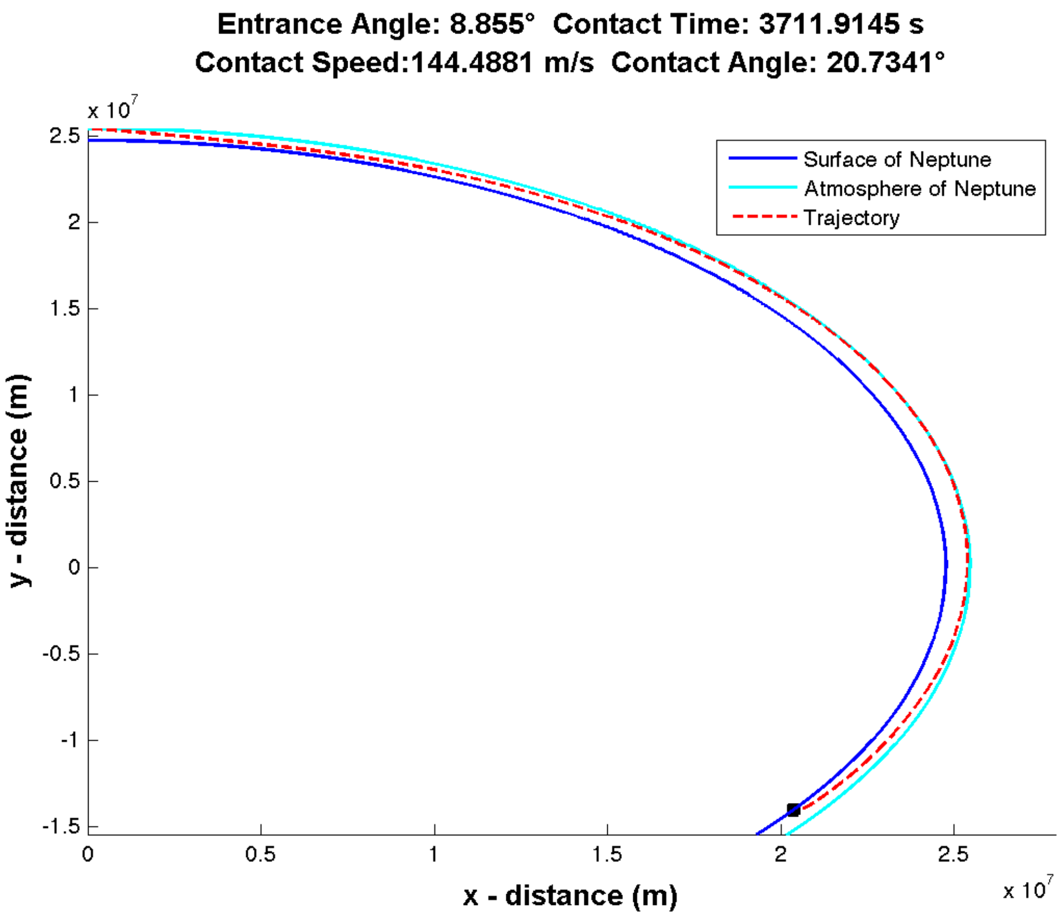
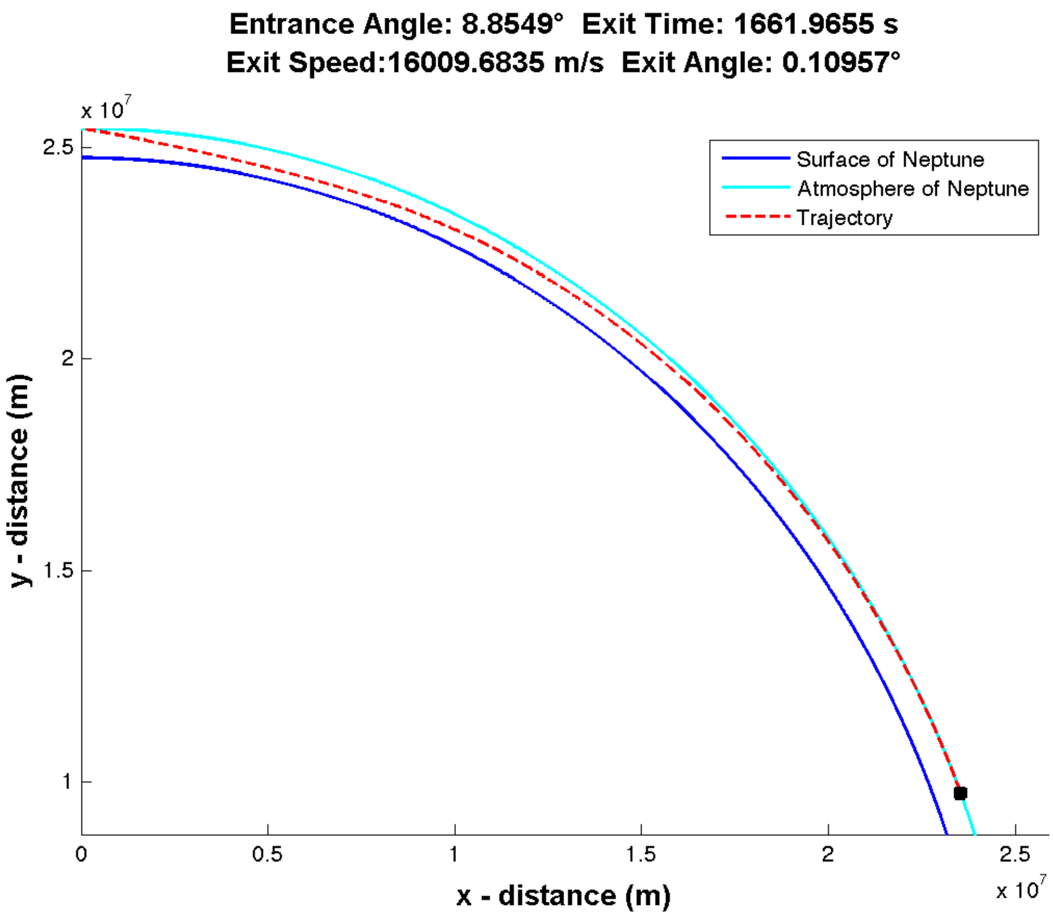

Entrance Angle: 8.8296° Exit Time: 740.735 s
Exit Speed:17383.6787 m/s Exit Angle: 4.008°
x 10^7
2.5
2.4
2.3
2.2
2.1
2
1.9
y - distance (m)
0
2
4
6
8
10
12
14
16
x 10^6
x - distance (m)
Surface of Neptune
Atmosphere of Neptune
Trajectory

Figure 2



Entrance Angle: 8.855° Contact Time: 3711.9145 s
Contact Speed:144.4881 m/s Contact Angle: 20.7341°
x 10^7
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
y - distance (m)
0
0.5
1
1.5
2
2.5
x - distance (m)
x 10^7
Surface of Neptune
Atmosphere of Neptune
Trajectory

Figure 3



Entrance Angle: 8.8549° Exit Time: 1661.9655 s
Exit Speed:16009.6835 m/s Exit Angle: 0.10957°
x 10^7
2.5
2
1.5
1
y - distance (m)
Surface of Neptune
Atmosphere of Neptune
Trajectory
0
0.5
1
1.5
2
2.5
x 10^7
x - distance (m)

Figure 4

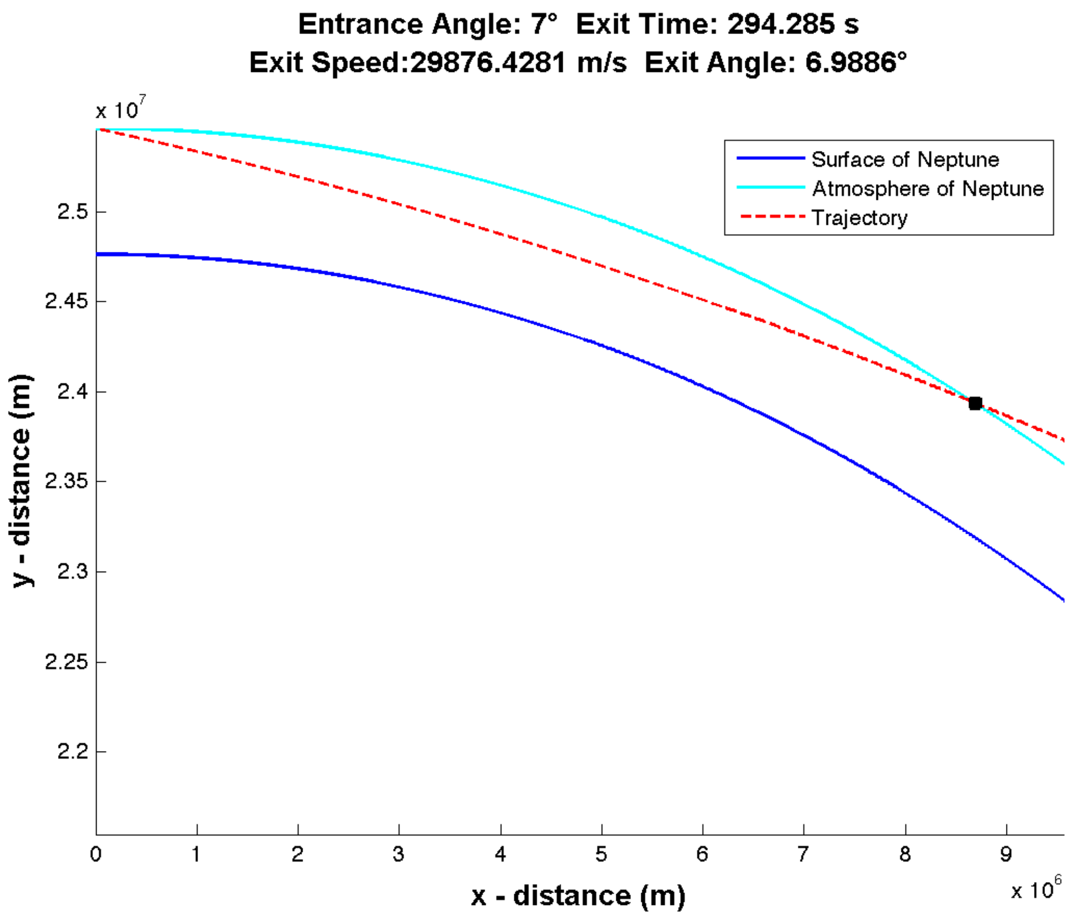
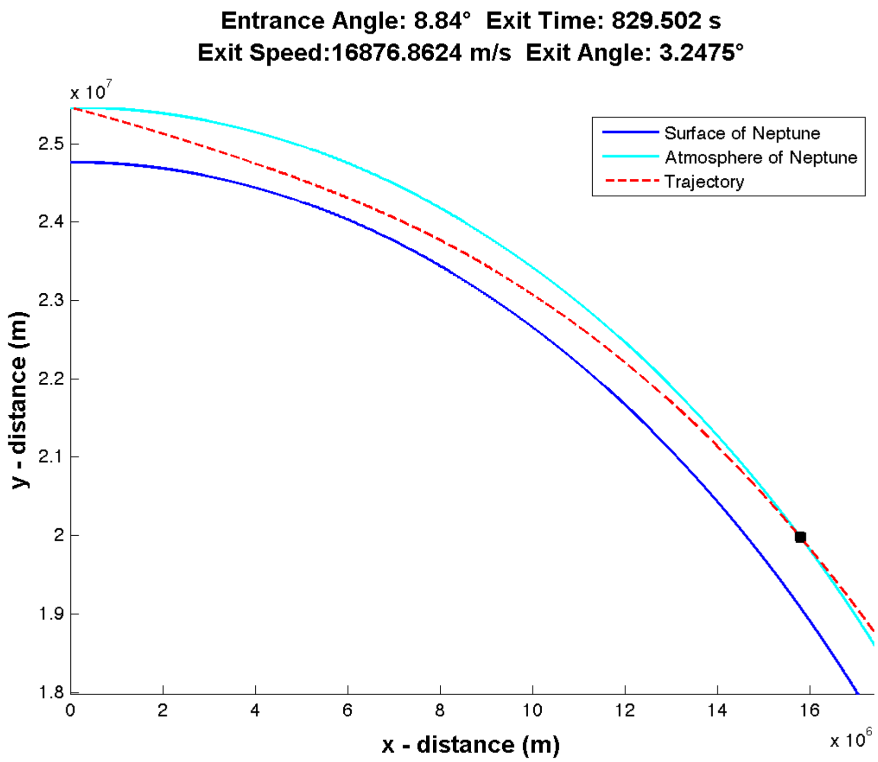

Entrance Angle: 7° Exit Time: 294.285 s
Exit Speed:29876.4281 m/s Exit Angle: 6.9886°
x 10^7
2.5
2.45
2.4
2.35
2.3
2.25
2.2
y - distance (m)
Surface of Neptune
Atmosphere of Neptune
Trajectory
0
1
2
3
4
5
6
7
8
9
x - distance (m)
x 10^6

Figure 5



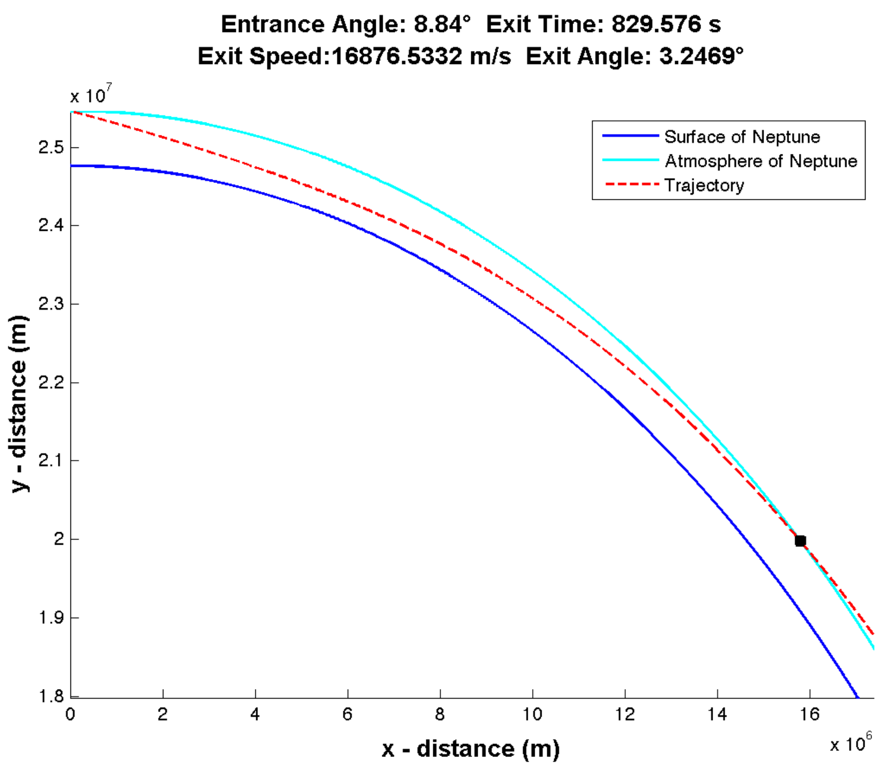


Figure 6: Comparison between time step = 0.001 s and time step = 0.0005 s