ANALYSIS OF LANDING TRAJECTORY USING BACKWARD PROPAGATION

Melissa M. Onishi
Department of Mechanical Engineering
University of Hawai‘i at Mānoa
Honolulu, HI 96822

ABSTRACT

The past NASA landing missions such as Apollo, Vikings, Mars Pathfinder, the Mars Exploration Rovers and the Mars Science Laboratory projects have led to the necessity of designing the next generation of landers that will achieve safe and precision landing. According to the table Capabilities Roadmap: EDL (Entry, Descent and Landing), safe and precision landing is one of many challenges of future missions [1, 2]. In August 6, 2012, NASA Jet Propulsion Laboratory (JPL) led Mars Science Laboratory (MSL) mission successfully landed a robotic rover, known as Curiosity. However pursuing accurate or precision landing was not a priority of this mission. As directed by the NASA EDL Roadmap and inspired by this mission, further research has led to continuation of accomplishing this task of precision landing. The main goal of this research is to study and investigate the possibilities of achieving safe and precision landing with the awareness of hazard avoidance and with efficient usage of fuel [3]. It is expected that the results of this study will lead to improved simulations of landing trajectories with the creation of manifolds of initial and final points (conditions) for powered descent and landing. These manifolds can be created by a construction of envelopes of the landing trajectories using various ranges for trajectory and lander’s parameters.

INTRODUCTION

Throughout the semester I studied and worked on creating different envelope constructions using the concept of backward propagation. Some interesting results regarding the precision landing have been obtained in Ref [4].

Previous studies of NASA related issues in precision landing have slowly decreased the distance between the designated landing site to the actual landing site [3]. It is necessary to consider algorithms which identify and locate landing hazards, react to failed systems and low powered components to support terrain tracking [1]. Hopefully, with time I will be granted the opportunity to accomplish these challenges in achieving these tasks.

In early May, I was able to obtain Matlab software and was given a brief tutorial on basic command functions from my mentor. Within a short period of time, I was given a crash course on learning the systems of equations of motion and how they pertained to precision landing. I learned how to integrate the differential equations using an ODE call on Matlab which will be further discussed in the method section of this report.
I created two separate codes, a main file and a function file using Matlab software. The main file contains initial conditions and landing selections, while the function file consists of systems of equations that were integrated. The thrust direction is opposite to the velocity direction making thrust magnitude constant for this approach. The assumption about this direction means that \( \cos \alpha = \frac{v_x}{V} \), \( \cos \beta = \frac{v_y}{V} \), and \( \cos \gamma = \frac{v_z}{V} \), where \( \cos \alpha \), \( \cos \beta \), and \( \cos \gamma \) are the thrust direction cosines, \( v_x \), \( v_y \), and \( v_z \) are the velocity components, and \( V \) is the magnitude of the velocity vector. To make this application more realistic, actual Mars Newtonian gravity was used (\( \mu \)). The landing selection that was used in spherical coordinates contains \( \theta_{LS} = 30^\circ \) and \( \rho_{LS} = 60^\circ \), where \( \theta_{LS} = \left( \frac{\theta_{LS}^\circ}{180^\circ} \right) \pi \) and \( \rho_{LS} = \left( \frac{\rho_{LS}^\circ}{180^\circ} \right) \pi \). The landing coordinates used are stated as

\[
\begin{align*}
    x_{LS} &= (r_{LS})(\cos \rho_{LS})(\cos \theta_{LS}) \\
    y_{LS} &= (r_{LS})(\cos \rho_{LS})(\sin \theta_{LS}) \\
    z_{LS} &= (r_{LS})(\sin \rho_{LS})
\end{align*}
\]

where \( r_{LS} \) is constant and defined as Mars equatorial radius.

Systems of equations of motion used for integration:

\[
\begin{align*}
    \dot{x} &= V_x \\
    \dot{v}_x &= -\frac{\mu x}{r} + \left( \frac{c \beta}{m} \right) \cos \alpha \\
    \dot{y} &= V_y \\
    \dot{v}_y &= -\frac{\mu y}{r} + \left( \frac{c \beta}{m} \right) \cos \beta \\
    \dot{z} &= V_z \\
    \dot{v}_z &= -\frac{\mu z}{r} + \left( \frac{c \beta}{m} \right) \cos \gamma \\
    m &= m_0 - \beta t
\end{align*}
\]

Variables defined:

\( x, y, z, m \) are the position vector components of the lander’s center of mass, \( \mu \) and \( \beta \) are the constants. \(-\frac{\mu x}{r}, -\frac{\mu y}{r}, \) and \(-\frac{\mu z}{r}\) are the gravitational acceleration components, \( \left( \frac{c \beta}{m} \right) \) is the thrust acceleration magnitude.

Initial Conditions:

Initial time \((t_0) = 0\) \quad Specific Impulse \((Isp) = 220\) s

Gravity \((g_0) = 0.00981\ km/s^2\) \quad Thrust to Weight Ratio \(\frac{T}{W} = 3\)

Initial mass \((m_0) = 2835\ kg\) \quad Gravity on Mars \((g_m) = 3.7\ km/s^2\)

To obtain massflow rate and final time I did some minor algebra to obtain the equations shown above. Below is the calculation I found for massflow rate and final time.

Calculations:

\[ c = Isp \times g_0, T = (c)(\beta) \]

We know that: \( W = (m_0)(g), \left( \frac{T}{W} \right) = 3 \)

\[ T = 3W = 3(m_0)(g) = (c)(\beta) = (Isp)(g_0)(\beta) \]

\[ \beta = 3(m_0)(g) = \left( \frac{T}{W}m_0(g) \right) = \left( \frac{T}{W}m_0(g_0) \right) \]

Where \( g \) is the gravitational acceleration on the surface of Mars. Given in the equations of motion we find that \( m = m_0 - \beta t > 0 \). To obtain final time we can use \( m_1 = m_0 - \beta t_1 > 0 \) where \( m_0 > m_1 \). Solve for final time: \( t_1 = \frac{m_0 - m_1}{\beta} \)
DATA AND RESULTS

In these studies, the realistic results that could be potentially used for future missions have been used. Below are the figures with different independent and dependent variables. One out of the five figures contains six subplots with six different states. Each plot in figures 2-6 is a representation of a trajectory envelope that covers acceptable ranges of landing trajectory and its parameters. Notice that the mass vs. time plot in Fig. 5 looks different from the rest of the other figures. This particular plot is linear due to the liner dependence of mass on time ($m_t = m_0 - \beta t$). Fig. 2 displays envelope constructions of six subplots of the first six states stated in the function file. There is a seven kilometers difference found for these plots which is in a suitable range for achieving altitude between $5 \text{ km} \leq h \leq 10 \text{ km}$ where $h$ is altitude. Fig. 3 displays the envelope construction of position vs. time. I found that there was a seventeen kilometers distance between the highest and lowest trajectory for position. Fig. 4 displays the envelope construction of velocity vs. time. For velocity vs. time, there was about a $1 \text{ km/s}$ difference between the highest and lowest trajectory. Fig. 6 displays the envelope construction of altitude vs. position. Interestingly, this particular plot stops at a certain position once it achieves a certain altitude. I found that the difference between the highest and lowest trajectory was approximately $1 \text{ km}$.

![Figure 2 Envelope construction containing six subplots time vs. state](image)

Figure 2 Envelope construction containing six subplots time vs. state
DISCUSSION AND CONCLUSIONS

The simulations that were found from MATLAB software justified the purpose of using backward propagation for this application. By starting from the designated landing site and creating multiple trajectories from that point, we were able to create a variety of different envelopes. With the help of a working code, we saw that each envelope construction had different distances between the highest to lowest point of trajectory.

Initially, the proposed research started off as creating trajectory simulations and creating an envelope construction. There was no specific way of how I was going to create trajectory simulations. During mid-semester, I decided to use backward propagation in order to achieve precision landing. There are three tasks mentioned in the proposal for future studies. The next step is to achieve forward propagation using the results that were found in backward propagation using the envelopes of the trajectories and manifolds of the initial and final points of powered descent and landing trajectories. By conducting studies for forward propagation, we can increase the accuracy of envelopes that could be potentially used for future missions. In the future, I hope that my results will be useful for the next mission stated in the table year 2020 Capabilities Roadmap: EDL [1,3].
Throughout the semester, there were many challenges and strides made to accomplish the task proposed. The main component of this research was to understand the physics behind landing trajectories. It was not simple considering I had no prior knowledge or experience in understanding EDL trajectories. Secondly, getting accustomed to MATLAB was another challenge in itself such as learning new command functions. Overall we found that backward propagation allow us to review the behavior state of vector components, and to create landing trajectory envelopes. By creating different variations of landing site conditions we are able to create manifolds of landing points and initial points of powered descent trajectories. The manifolds consist of initial and final points that will be used for the future studies to increase landing accuracy [5]. It is expected that optimization of landing trajectory allows us to minimize the fuel consumption while providing precision landing.

REFERENCES


