Neural Network Trained Controller For Atmospheric Entry in Mars Environment

Matthew Keosayian*, Hao Wang†, and Tarek Elgohary‡

Abstract

We submit a new method using neural networks for a Mars capsule controller during atmospheric entry. This method utilizes the controller from the Apollo missions as a reference for data when training the neural network. The simulation with the trained network can replicate the results from Apollo with variations in the initial conditions, e.g. position. Using the real data from Apollo the controller's performance is evaluated. The controller is then adapted to the Mars environment where it will satisfy the requirement for landing accuracy. The use of machine learning has great potential for future Mars missions.

Introduction

Atmospheric entry has always posed a difficult challenge in space environments. The goal is to land safely and accurately, with very little error in how much a spacecraft misses the target. However, the priority is not always precision as there are other requirements that must be satisfied.[1] Those requirements include maximum G-force, heat thresholds, and deceleration limits.[1] Recently, earth missions have become easier as there have been many more launches and landings. Due to this increase, scientists can study the launches and re-entries in order to increase the efficiency and durability of rockets and capsules. However, this is all just within the earth environment. The obstacle now is trying to land on other planets, like Mars. The recent interest in Mars makes landing on Mars safely and with precision a priority. Thus, advances must be made to improve the reliability of landing on Mars. If the idea of manned missions is to become reality, the error in target miss distance is going to have to be very small, as the astronauts may not have the resources to survive larger errors. Therefore, it is of the utmost importance to create a reliable landing algorithm that will be able to account for the Mars environment with accuracy and be able to handle changes in conditions. The implementation of machine learning might prove to be a great asset in improving the Mars landing controller.

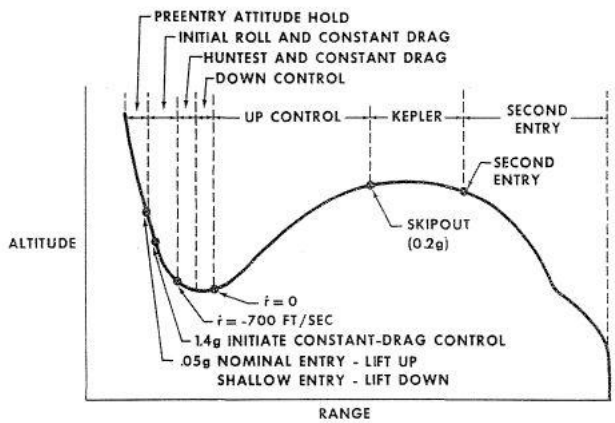
Background

The original entry configuration of Mars landers utilized a ballistic trajectory that rendered the spacecraft uncontrollable with a very large landing error.[6] However, the idea of ballistic lifting became the new configuration that utilized bank angle to change the direction of the lift vector.[5] The result of this configuration was improved accuracy as the spacecraft can adjust itself before deploying the parachute. The MSL/Curiosity was the first to fully utilize the configuration and adjust the bank angle, which in turn proved the idea correct in that it would improve accuracy.[5] The ballistic-lifting configuration is based on the same configuration that was used to land the Apollo spacecraft.[5] The details of the Apollo re-entry are in Re-

*Undergraduate student, Department of Mechanical Engineering, Manhattan College, The Bronx, NY 10471
†PhD student, Department of Mechanical and Aerospace Engineering, University of Central Florida, Orlando, FL 32826
‡Assistant Professor/Advisor, Department of Mechanical and Aerospace Engineering, University of Central Florida, Orlando, FL 32826

Entry Guidance by Raymond Morth.[6] Morth explains that within the controller there is the Initial roll, Huntest, Down-control, Up-control, Kepler, and Second entry phase during atmospheric entry.[6]

Figure 1: Apollo Entry Guidance Phases[3]

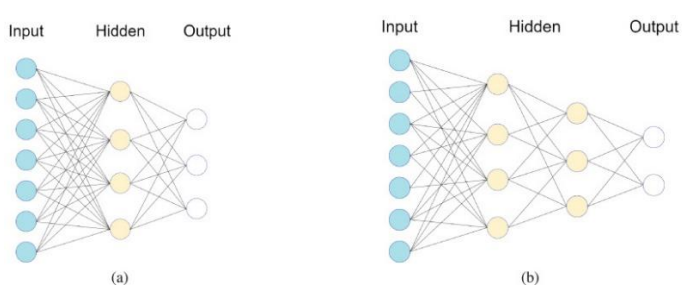


Using these flight stages, the Apollo algorithm was created and was able to successfully land spacecraft on earth and was extrapolated into the Mars environment for the MSL capsule.[5] However, the landing error was still around 100 kilometers, which must decrease significantly if there is ever going to be a manned mission.[5] As technology progresses into the future, machine learning has become a more viable option in helping increase the landing accuracy. There have been multiple attempts recently to utilize this technology.[4] One idea was using reinforcement learning and the gauss pseudospectral method to be used during powered Mars entry.[4] Reinforcement learning uses a system of either positive or negative reinforcement over many trials. Positive reinforcement rewards good results and penalizes bad results. Negative reinforcement is when an attribute is strengthened when a bad behavior has stopped. The pseudospectral method is used to increase accuracy by enlarging the order of the polynomials when the problem is converted in a series of meshes.[4] This method, while effective, used more nominal initial conditions which is not really going to cut it for the true Mars environment. Reinforcement learning is more complex in nature which is why deep neural networks with their simpler systems may prove to be more ideal.

Neural Networks

Neural networks are based on cognitive and information theories. They seek to replicate the learning process of human neurons to create complex interconnected neural structures.[2]

Figure 2: Neural Network Schematic[2]



They are composed of three categories: Inputs, Hidden Layers, and Outputs. The inputs are the data that the network reads and will run through black box of hidden layers. The hidden layers are where the main processing goes on. Each line has a weight that varies with each input and respective node in the hidden layer. The nodes act as activation functions to process the data, e.g. hyperbolic tangent sigmoid, linear, etc. The number of nodes per layer within the black box can vary but the user must be careful not to overfit the network. There needs to be the right balance in order to get the best performance, and that just takes a lot of testing. The outputs are the final simulated results after processing the inputs through the network.

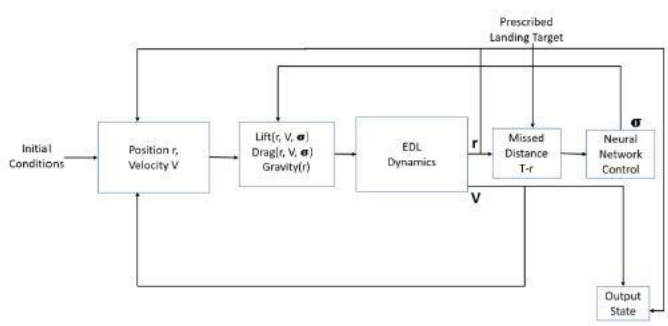
Research Approach

The first problem that needs to be tackled is producing a dataset to train the neural network in an earth environment. Using MATLAB, a program is made to replicate the Apollo landing using data given in Re-entry guidance by Morth.[6]

Figure 3: Initial Conditions and Target for Earth[6,7]

Height, *h*	121920 meters
Velocity, V	11033.76 m/s
Flight Path Angle, γ	-6 degrees
Range	3704000 meters
Latitude, ϕ	-12.7 degrees
Longitude, θ	122.9 degrees
Heading Angle, ψ	61 degrees
Terminated Height	7620 meters
Target in Earth Fixed Frame	(-4754400.51, 3771311.48, 1964589.29) meters

Figure 4: Neural Network in Control System[7]



Using these initial conditions, the initial position can be calculated in component form and then the Apollo landing can be simulated. Once that one test is completed successfully, then variations to the initial components of position can be varied to create a larger dataset. The inputs would be position and velocity, and the output would be the bank angle. More inputs could be added as needed but the less components that the system works with then the less complex the system. Once the dataset is acquired, MATLAB Neural Network Toolbox can be utilized to train a neural network using the dataset. However, it is going to take multiple attempts to figure out how many layers are needed to make the network as accurate as possible. After each network is trained, it will be ran using Simulink functions to process the initial conditions

through the neural network, with the one output being the bank angle. This control system is shown in Figure 4 and demonstrates that in this method the neural network is involved throughout nearly the entire control system. Once the normalized landing error is within the tolerable range of about 27 kilometers, then the network can be tested using available Mars conditions.[7] There is not that much data that was released from the Mars missions but there is a decent amount released from the MSL mission that can be used. Again, the simulations will be conducted but this time using Mars parameters and will attempt to meet the 27-kilometer threshold. Monte Carlo simulations will then be run for both environments to test their viability over variation in initial position. This variation will come from a normalized random distribution curve with a standard deviation of 500/3 in the x, y, and z directions.[7] Therefore the Monte Carlo simulation will represent a variation of 99.7% of points within 500 meters, due to the three standard deviations. Upon succession in the linear format of the controller, the controller will then be adapted into a non-linear format using differential forms of the atmospheric dynamics equations.[3] This new format will then be tested similar to the linear form, first getting under 27-kilometers error, then running Monte Carlo simulations to test viability.

Results and Discussion

After many days of training networks and testing, it was found that roughly 8+ hidden layers were required for processing the data to get close to the desired error. The number of nodes increase exponentially from 2, e.g. 4,8,16,32, and so on as the number of hidden layers increases. The best results were from a network trained with these parameters:

Figure 5: Neural Network Training Parameters

Iterations:	1000
Hidden Layers:	10
Learning Rate:	0.01
Inputs:	Current Position from Target
Output:	Bank Angle

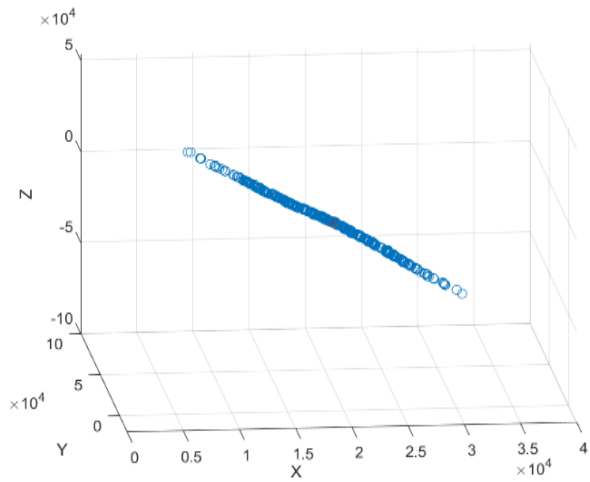
The preliminary tests came out very satisfactory:

Figure 6: Initial Test in Earth Environment

Normalized Miss Error:	23547.49 meters
Time Step:	2.5 seconds

The objective is to get the time step as low as possible (0.1 is ideal), as the less time it takes to process the faster the lander can adjust in a real scenario. The results show that the network was able to satisfy the 27-kilometer requirement, which was then followed with a Monte Carlo simulation with variation within about 500 meters in initial position:

Figure 7: Missed Distance with Initial Position Variation in Earth Environment



The slightly hidden red dot in the center represents the nominal trajectory with no variation in initial position. The plot signifies that even with variation, the neural network is still able to effectively operate and achieve a satisfactory miss error. The research was then able to move into the Mars environment.

Figure 8: Initial Conditions and Target for Mars[7]

Height, *h*	135400 meters
Velocity, V	6000 m/s
Flight Path Angle, γ	-11.5 degrees
Latitude, ϕ	0 degrees
Longitude, θ	0 degrees
Heading Angle, ψ	90 degrees
Terminated Height	6000 meters by estimation
Target in Earth Fixed Frame	(3237225.27, 51680.98, 1024523.02) meters

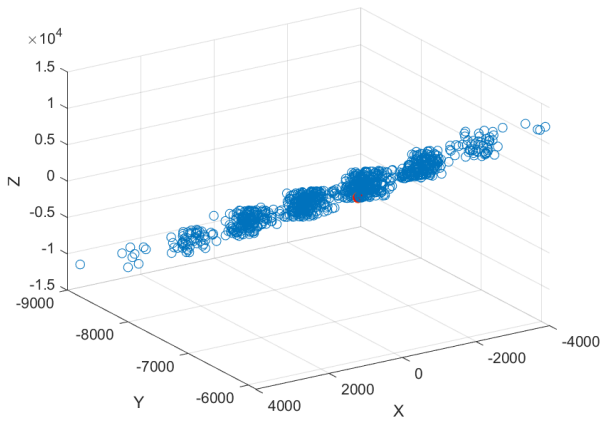
When the same network was tested using Mars parameters, it was able to produce these results:

Figure 8: Initial Test in Mars Environment

Normalized Miss Error:	7168.63 meters
Time Step:	1 second

Not only did the landing error improve but it also did it with a 1.5 second decrease in the time step. A Monte Carlo simulation was then run to test viability:

Figure 9: Missed Distance with Initial Position Variation in Mars Environment



Again, the red dot represents a nominal trajectory with no variation in initial position. While there was a larger spread of landing positions, all the landing points still satisfied the 27-kilometer requirement. Therefore, the linear form of the neural network controller proved to be viable.

The next step was to adapt the controller to a non-linear form. The 10-layer network would be applied again in the new model:

Figure 10: Initial Test in Earth Environment with Non-Linear Controller

Normalized Miss Error:	37902.54 meters
Time Step:	About 2.5 seconds

The results deemed that the 10-layer network was nearly suitable with the same nominal initial conditions. However, the earth model still needs to be worked on further to allow Monte Carlo simulations to be run. Just to experiment, the 10-layer network was then tested in the Mars environment and produced a surprisingly satisfactory result:

Figure 11: Initial Test in Mars Environment with Non-Linear Controller

Normalized Miss Error:	10604.61 meters
Time Step:	5 seconds

This result demonstrates that the neural network can still work in a Mars model but there still needs to be more work done to support this outcome. Monte Carlo simulations also still need to be run to see how the network handles with variation in initial position.

While there is still work to be done in the non-linear model, the results show that a neural network has great potential for future controller configurations given that more evidence is provided to support the non-linear data.

Future Work

While there are signs that neural networks can be viable candidates in the non-linear model, further experimentation needs to be conducted. It is still under question as to why the Mars model achieved a satisfactory error, but the Earth model did not in the non-linear controller. A new method of training networks could be conducted to possibly improve accuracy. This could be done by picking an initial point on earth and creating a hypothetical "box" around that point. Then trajectories can be simulated from different points to generate a dataset. Using a newly trained network from that dataset, accuracy can hopefully be increased, building a stronger foundation for future Mars controllers.

Acknowledgements

I would like to thank Dr. Elgohary and Hao Wang for their guidance through this research. Ms. Wang has been a great mentor and colleague and has helped me through all the questions I have had. I would also like to thank Dr. Gordon and Dr. Kauffman for running this research program and keeping it organized for us. Most of all, I would like to thank Robert Burke for leading this pack of students and always looking out for us and answering our countless questions.

References

[1] D. Hicks, Kerry. *Introduction to Astrodynamic Reentry*. 2009.

[2] Hatcher, W. G., and W. Yu. "A Survey of Deep Learning: Platforms, Applications and Emerging Research Trends." *IEEE Access*, vol. 6, 2018, pp. 24411–32, doi:10.1109/ACCESS.2018.2830661.

[3] Ives, Dallas, et al. "Apollo-Derived Mars Precision Lander Guidance." *23rd Atmospheric Flight Mechanics Conference*, American Institute of Aeronautics and Astronautics, 2013, doi:10.2514/6.1998-4570.

[4] Jiang, Xiuqiang, et al. "Integrated Guidance for Mars Entry and Powered Descent Using Reinforcement Learning and Pseudospectral Method." *Acta Astronautica*, 2019, doi:10.1016/j.actaastro.2018.12.033.

[5] Li, Shuang, and Xiuqiang Jiang. "Review and Prospect of Guidance and Control for Mars Atmospheric Entry." *Progress in Aerospace Sciences*, 2014, doi:10.1016/j.paerosci.2014.04.001.

[6] Morth, Raymond Harry., et al. *Reentry Guidance for Apollo*. Massachusetts Institute of Technology, Instrumentation Laboratory, 1966.

[7] Wang, Hao, et al. *NEURAL NETWORK TRAINED CONTROLLER FOR ATMOSPHERIC ENTRY IN MARS MISSIONS*. 2019.