



University of Colorado
Boulder

Deep learning atmospheric prediction algorithm for enhanced Mars EDL guidance

Daide Amato, Shayna Hume, Evan Roelke, Benjamin Grace, Jay W. McMahon
Smead Aerospace Engineering Sciences, University of Colorado, Boulder, CO, USA



Abstract

Uncertainty in atmospheric density and wind is a major cause of sub-optimal performance in the Entry, Descent, and Landing (EDL) guidance at Mars. We improve the robustness of current EDL guidance algorithms to uncertain dynamic environments by proposing a reliable on-board atmospheric estimation algorithm. The algorithm consists of a deep, recurrent neural network using an efficient architecture for time-series predictions, the Long Short-Term Memory (LSTM) cell. The LSTM network is trained on entry trajectories simulated with the Fully Numerical Predictor-corrector Guidance (FNPEG); in each trajectory the vehicle is subject to density and wind fields from instances of the Mars Global Reference Atmospheric Model (GRAM) 2010. Predictions of density and wind as a function of altitude expected along the trajectory are obtained from onboard acceleration measurements and state estimates. The algorithm achieves a RMS value over time for the relative density error in the order of 10% for samples in the validation dataset, and significantly improves performance with respect to an exponential fit to the density.

Introduction

Groundbreaking advances in EDL guidance are required to satisfy accuracy and reliability requirements of future Mars missions [1]. In this context, Numerical Predictor Correct (NPC) guidance algorithms have been shown to significantly improve performance with respect to analytical guidance laws. In the predictor step of a NPC, the current state is propagated forward under an assigned control law until meeting a prescribed final condition. Subsequently, a new control law is computed in the corrector step with the aim of minimizing an error metric (e.g. a range error), and the procedure is iterated until reaching a sufficiently small error tolerance.

Dynamical models in the predictor step often rely on simplified assumptions, such as the density being an exponential function of altitude exclusively. Several approaches have been developed to provide accurate atmospheric information to the predictor in order to improve the performance of aerodynamic entry. Most of these rely on a multiplicative scale factor either for the density or for the aerodynamic force, which is estimated through a low-pass or fading-memory filter. However, all of these approaches rely on the assumption that the density is well described by an exponential law and is exclusively a function of altitude, which do not generally apply due to the time and spatial variability of the Martian atmosphere. In addition, none of these methods explicitly considers the presence of winds, which can be a significant hazard during the final EDL stages [2].

In this work, we present a deep learning algorithm to **predict density and wind from onboard state estimates and acceleration measurements**. The algorithm is trained on entry trajectories simulated with the FNPEG guidance algorithm [3], assuming density and wind provided by the Mars GRAM 2010 atmospheric model [4].

FNPEG entry guidance

At each guidance cycle, FNPEG computes the bank angle law that minimizes the range error with respect to a prescribed range-to-go to be reached at the final value of the energy $-e = -\mu/r + v^2/2$.

The bank angle magnitude is parametrized as a linear function of the energy,

$$\sigma(e; \sigma_0) = \sigma_0 + \frac{e - e_f}{e_0 - e_f} (\sigma_f - \sigma_0), \quad \sigma_0 \geq 0, \sigma_f > 0. \quad (1)$$

The current bank angle σ_0 that minimizes the final range error $s(e_f) - s_{go}$ is found within a predictor-corrector scheme. In the predictor step, the final range s_f is obtained by numerically integrating the longitudinal equations of motion (i.e. $\dot{r}, \dot{v}, \dot{\gamma}, \dot{s}$), assuming the bank angle law in eq. (1). This is followed by a corrector step in which the value of σ_0 is updated through an adaptive Gauss-Newton method. The longitudinal and lateral equations of motion are decoupled under the assumption that the offset between the heading angle and the course to the target is small. In the lateral guidance, the sign of the bank angle is changed when the vehicle crosses a velocity-dependent deadband on the offset between the heading and the course to the target site. This ensures that the vehicle travels along the great circle connecting the current and target sites.

Since the current state is numerically integrated until reaching the final energy in the predictor step, uncertainty on density and wind will impact the accuracy and reliability of the FNPEG guidance.

Mars GRAM 2010 dataset

In order to build the dataset for the training of the neural networks, we simulate 10^4 entry trajectories in which the density and wind are computed through a MATLAB implementation of the Mars GRAM 2010 atmospheric model [4], and guidance is provided by the FNPEG algorithm. In each of the samples, the background dust optical depth and the mean value and scale parameter of the longitude-dependent density wave for the GRAM model are randomly sampled. The initial conditions and FNPEG target range and final energy are the same for all trajectories. Figure 1 shows the density and wind as a function of range (measured from the entry interface) for the trajectories in the dataset.

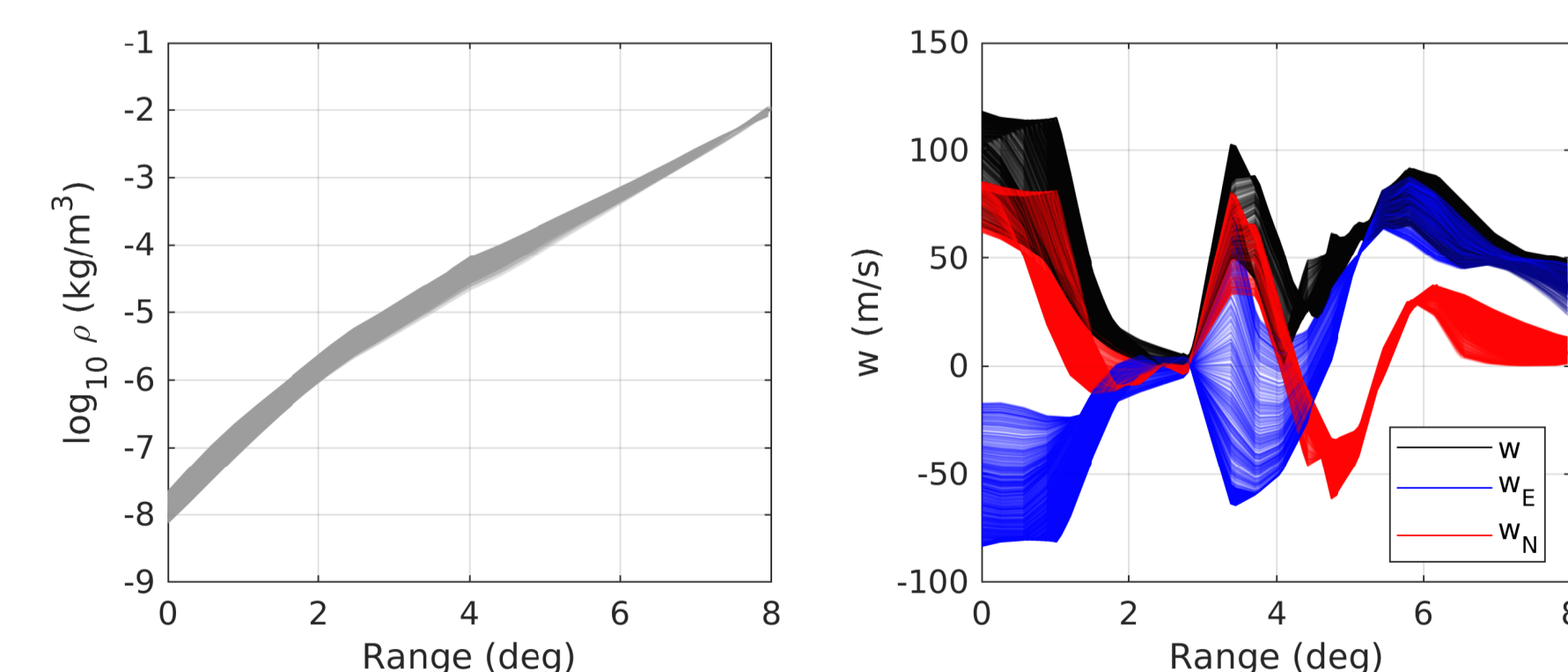


Figure 1: Density (left panel) and wind velocity (right panel) as a function of range from the entry interface for the trajectories in the GRAM dataset. In the right panel, w_E, w_N denote the East and North wind velocity components, and $w = \sqrt{w_E^2 + w_N^2}$ is the wind velocity magnitude.

Deep learning for atmospheric predictions

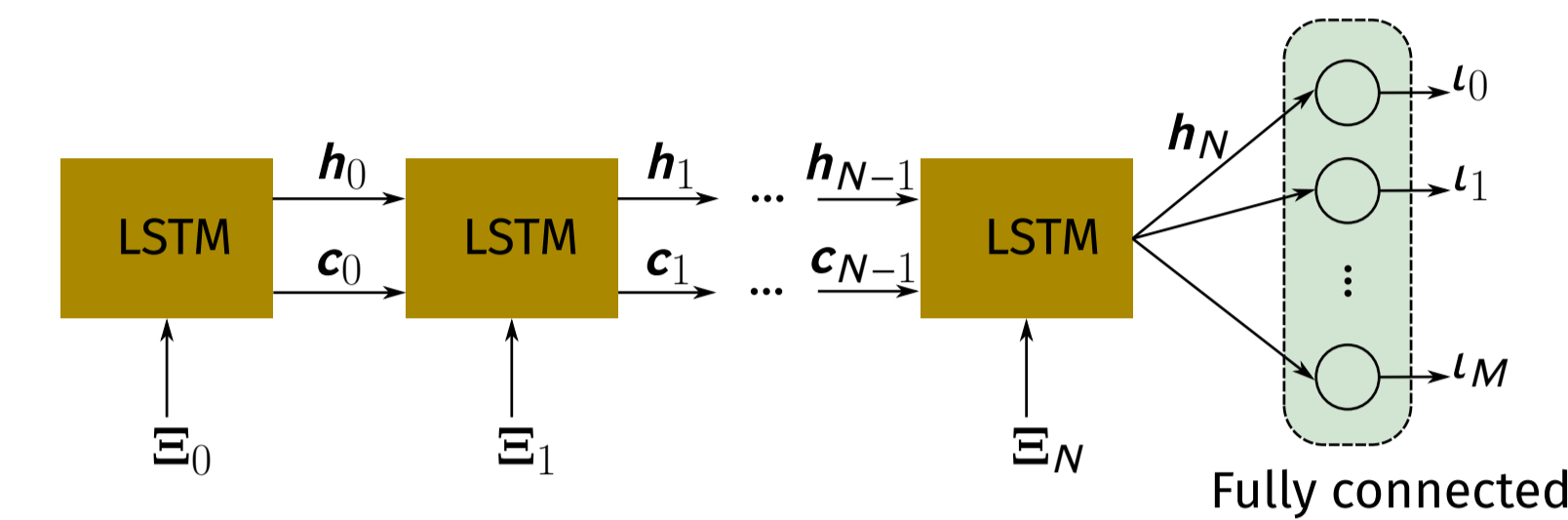


Figure 2: Computational graph of the LSTM deep learning architecture. The feature vector at time i and the j th target are Ξ_i, t_j , respectively. The targets are either normalized pseudo-density values or wind components, discretized over a set of altitudes. The hidden and cell states at time i are h_i, c_i .

Density and wind encountered in the FNPEG predictor step are predicted from state estimates and acceleration measurements through deep, recurrent neural networks (RNN). We choose the Long Short-Term Memory (LSTM) RNN architecture, which is particularly efficient for predictions from sequential data [5]. At time t_i during the entry phase, the set of features that are input to the LSTMs is

$$\Xi_i = \{x(t_i), a(t_i), \sigma(t_i)\}, \quad (2)$$

where x is the state in spherical coordinates, a is the acceleration measured in a Mars-centered frame, and σ is the current bank angle. We use separate networks to compute either density values $\rho(h_j)$ or wind velocity components $w(h_j)$ discretized over altitudes $h_j \in [h_{min}, h_{max}]$ from the feature vector Ξ_i and the hidden state h_i , which contains information about past elements of the feature sequence. The hidden state of the LSTM cell h_i is passed to a fully connected layer to obtain the targets; the algorithm is portrayed through the computational graph in fig. 2.

LSTM network training and performance

To improve the performance of the network and mitigate the impact of round-off error, features and targets are normalized by subtracting the mean and dividing by the standard deviation, which are computed element-wise over the entire dataset.

Training performance

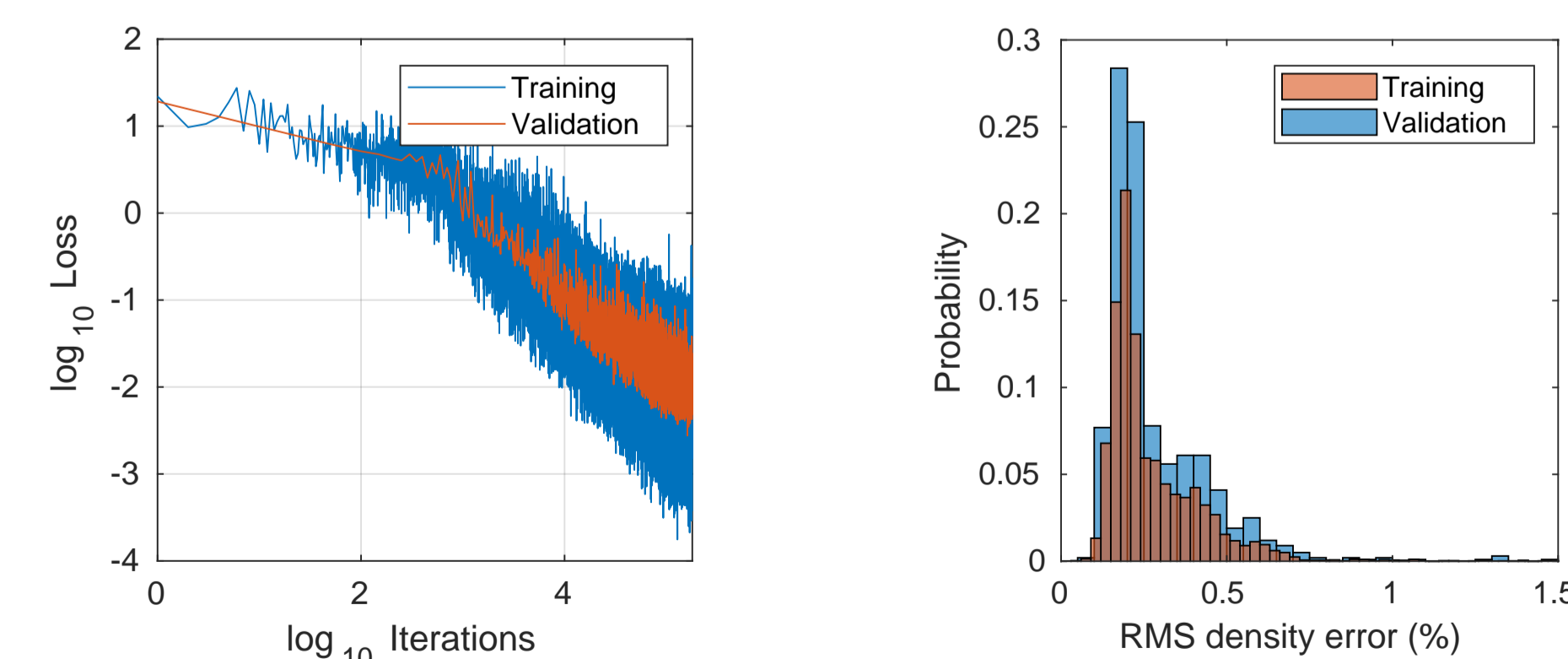


Figure 3: Training and validation losses as a function of iterations (left panel) and RMS of the relative density error over all altitudes at the final time (right panel).

We consider the training of the LSTM network portrayed in fig. 2 with 128 units for each of the LSTM gates. The network is trained for 200 epochs using the adam optimizer in the MATLAB Deep Learning toolbox, with a mini-batch size of 10. To assess

the generalization capability of the network, we reserve 10% of the total number of samples as a validation dataset. Figure 3 shows the mean squared error loss computed over the training and validation datasets as a function of the number of iterations. The training and validation losses are of the same order of magnitude on average, proving that the network generalizes well to samples not seen in the training dataset. At the final time, the RMS value of the relative density error is less than 0.5% on average in both the validation and training datasets.

To assess the performance of the network at intermediate times, we show the relative density error as a function of altitude and time for an individual sample in the validation dataset in fig. 4. The RMS of the relative density error over time is around 10% for all altitudes, and is always less than the error obtained when using an exponential fit to the density dataset.

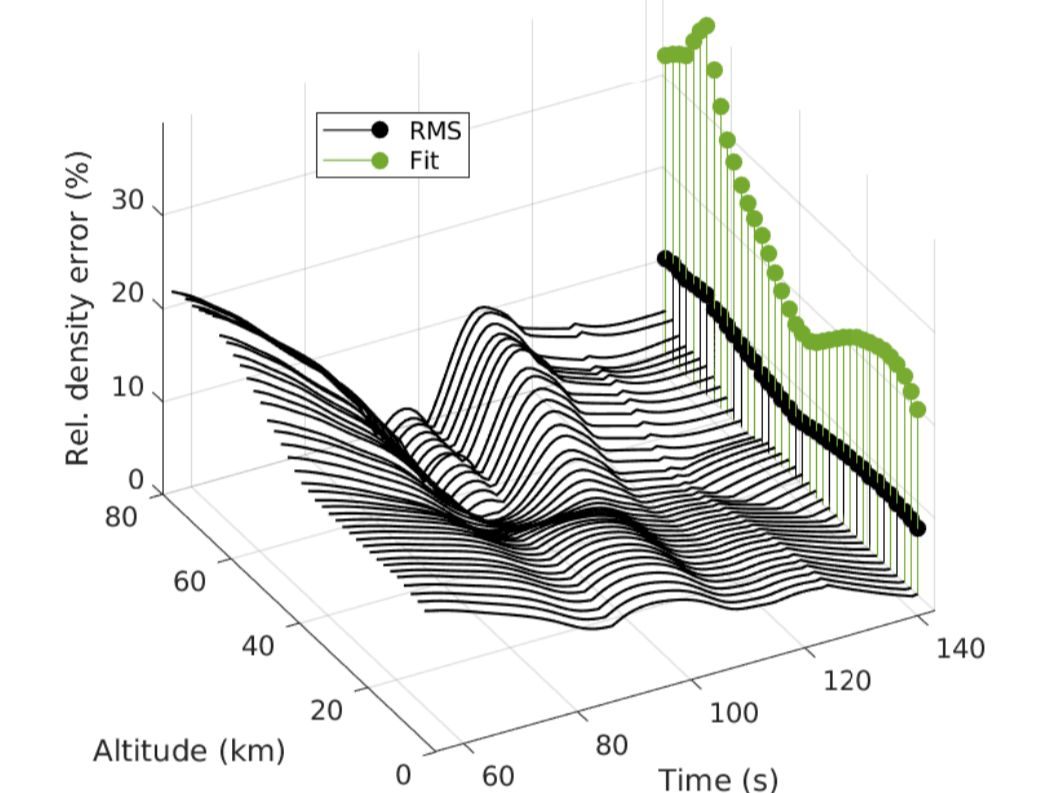


Figure 4: Relative density error as a function of time and altitude for a single sample from the validation dataset. Black stems correspond to the RMS value of the error over time. Green stems correspond to the relative density error obtained with an exponential fit to the dataset.

Conclusions

- We present a deep learning algorithm for the prediction of atmospheric density and wind from onboard measurements during the atmospheric entry of Mars vehicles.
- The algorithm achieves 10% RMS relative error in the prediction of density during entry, and is more accurate than a single exponential fit.
- Future work will examine the performance of the algorithm in predicting winds and its impact on the FNPEG guidance accuracy.

References

- [1] R. D. Braun and R. M. Manning. Mars exploration entry, descent and landing challenges. In *2006 IEEE Aerospace Conference*. IEEE, March 2006. doi: 10.1109/AERO.2006.1655790.
- [2] M. P. Golombek, J. A. Grant, T. J. Parker, D. M. Kass, et al. Selection of the Mars Exploration Rover landing sites. *Journal of Geophysical Research: Planets*, 108(E12), 2003. doi: 10.1029/2003JE002074.
- [3] P. Lu. Entry Guidance: A Unified Method. *Journal of Guidance, Control, and Dynamics*, 37(3), 2014. doi: 10.2514/1.62605.
- [4] H. L. Justh. Mars Global Reference Atmospheric Model 2010 Version: Users Guide. Technical Report TM-2014-217499, NASA Marshall Spaceflight Center, Huntsville, AL, USA, 2014.
- [5] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, Cambridge, Massachusetts, USA, 2016. ISBN 978-0-262-03561-3.