Stardust-R LTW-I - Re-entry of space objects under uncertainty

Date: 18/05/2020
Stardust-R WPs involved: WP2, WP1
Mentors:

Edmondo Misici (SU)
Victor Rodriguez (UPM)

Programming languages: MATLAB/Python

Abstract

As the Low Earth orbit (LEO) debris and spacecraft population that have exceeded their operational lifetime rises each year, the rate at which objects re-enter the Earth's atmosphere rises too. Most of these objects will probably not reach the ground for impact; however, large and medium objects have a high probability of surviving the harsh re-entry environment, potentially causing damage within a populated area. The impact location of an object re-entering the atmosphere is influenced by uncertainties in initial conditions, atmospheric characteristics, and object properties, as well as break-up/fragmentation events. Therefore, it is important to have an accurate estimate of the statistic ground footprint due to the involved uncertainties.

In this training exercise, machine learning approaches will be used to quantify the effect of uncertainties in atmospheric properties (density, temperature, composition, and free-stream air heat capacity), in initial conditions (re-entry flight path angle, speed, and direction angle), as well as object properties (mass), on the re-entry trajectory and ground impact location.

Description

The training exercise has been extracted mainly from [[1]] (here attached).

The case study is about gaining qualitative and quantitative understanding of the effects of uncertainties in atmospheric properties such as density, temperature, composition, and free-stream air heat capacity; uncertainties in initial conditions such as re-entry flight path angle, speed, and direction angle as well as object properties such as mass on the re-entry trajectory and ground impact location.

The case considers a spherical object undergoing a shallow, uncontrolled, re-entry with initial conditions corresponding to that of a re-entry from a circular orbit under only the effects of gravity and drag.

The uncertainty analysis will be performed by non-intrusive approaches based on machine learning techniques. [This is the core of the work]

The steps to perform are:

a) Implement a machine learning (ML) based approach able to handle continuous high-dimensional cases.
b) Interface the ML based mapping to propagate the provided input uncertainties through the dynamical model (DynMod).
c) Use the integrated system (ML+DynMod) to characterise the impact location.
d) Compare the obtained characterisation with the one obtained via pure Monte Carlo (with proper confidence intervals)
e) Submit, at the end of the allowed timeslot, figures and tables detailing the comparison, including details on computational costs.

---

[1] Mehta, P., Kubicek, M., Minisci, E., Vasile, M., Debris Re-Entry Modeling Using High Dimensional Derivative Based Uncertainty Quantification, AAS/AIAA Astrodynamics Specialists Conference, Vail, Colorado, USA, August 9--13, 2015.

Support

i) Models implementing the dynamics of the object (DynMod) will be provided in MATLAB.

Backgrounds for attendees

ML section

Attendees should be familiar with basic concepts around deep learning and neural networks, such as:

- Supervised vs unsupervised learning. In this we'll do a regression task, which is a type of supervised learning.
- Training and validation data
- Overfitting.
- Performance of a regression model (mean squared error)
- Parameters (weights and biases) & activations in a neural network
- Batch learning, learning rate

A good resource to review all of these concepts can be found in the talk "*Deep Learning for Space Guidance*", given by Roberto Furfaro & Richard Linares during the Stardust-R Training School II [2].

In practical terms, we will make use of the Python deep learning library fastai2 [3]. A good introduction to this library, and to practical deep learning in general, can be found in the first chapter of the fastai book, which is freely accessible here. In this exercise, we will work with tabular data, so attendees should make special focus on reading about that type of learning. Apart from fastai, data loading and preprocessing will be done using the pandas Python library. A useful resource for getting started with the basic functionality of this library can be found in this cheatsheet. FInally, in order to produce results, attendees need to know the basics for creating plots either in Python.

As a development environment, any Python 3 setup is suitable for this exercise. The use of Jupyter Notebooks is recommended due to its exploratory nature for drafting code and results. In this sense, we encourage you to use services like Google Colaboratory, a free Jupyter notebook environment that runs in the cloud and stores its notebooks on Google Drive or Github. It has everything pre-installed and ready-to-use, and provides easy sharing & collaborative capabilities for working in teams. A short tutorial on how to use Colab, and enable the GPU, can be found here.

[2] This material has restricted access for the Stardust-R network. If needed, contact the administrators of the workshop for getting the access code.

[3] fastai: A Layered API for Deep Learning: https://dev.fast.ai/