Perform a CME Arrival Time Prediction with CAT-PUMA

Jiajia Liu (jj.liu@sheffield.ac.uk)
The University of Sheffield
25 Jan. 2018

CAT-PUMA is designed to have a very easy user-friendly approach. Users can download the CAT-PUMA engine ("engine.obj"), the source code ("cat_puma.py") of an example demonstrating how we perform the prediction, and the source code ("cat_puma_qt.py"') of a well-designed User Interface (UI) from the following link: https://github.com/PyDL/cat-puma. All codes are written in Python, and have been tested with Python 2.7 on two *Debian*-based x86-64 Linux systems (*Ubuntu* and *Deepin*) and the x86-64 *Windows* 10 system. Modifications of the code will be needed if one prefers to run CAT-PUMA with Python 3. Python libraries, including *datetime*, *numpy, pandas*, *pickle* and *scikit-learn* (v0.19.1), are needed for a proper run of "cat_puma.py". In the following, we first explain the example code "cat-puma.py" in details.

The first 134 lines in the code import necessary libraries and define functions that will be used in the main program. Lines 138 to 152 define that features we are going to use, value of *m* (see Liu et al., ApJ, 2018) and the location of the engine file. Users are not suggested to revise these lines. Lines 155 to 163 are as following:

```
# CME Parameters
time = '2015-12-28T12:12:00'    # CME Onset time in LASCO C2
width = 360.    # angular width, degree, set as 360 if it is halo
speed = 1212.    # linear speed in LASCO FOV, km/s
final_speed = 1243.    # second order final speed leaving LASCO FOV, km/s
mass = 1.9e16    # estimated mass using `cme_mass.pro' in SSWIDL or
                 # obtained from the SOHO LASCO CME Catalog
mpa = 163.    # degree, position angle corresponding to the fasted front
actual = '2015-12-31T00:02:00'    # Actual arrival time, set to None if unknown
```

The above lines define the onset time, angular width, average speed, final speed, estimated mass and MPA of the target CME. These parameters can easily be obtained from the *SOHO LASCO CME Catalog* (https://cdaw.gsfc.nasa.gov/CME_list/) if available or via analyzing LASCO fits files otherwise. Here, we employ a fast halo CME that erupted at 2015-12-28T12:12 UT as the first example. This event was not included in our input dataset when constructing CAT-PUMA. Line 166 defines whether a user prefers to obtain the solar wind parameters automatically. If yes, the code will download solar wind parameters for the specified CME automatically from the *OMNIWeb Plus* website (https://omniweb.gsfc.nasa.gov/).

Next, one can then run the code, typically via typing in the command *python2 cat_puma.py*, after following the above instructions to setup the user's own target CME. The prediction will be given within minutes. The prediction result for the above CME is as following (information in the last two lines will not be given if one has not specified the actual arrival time):

```
CME with onset time 2015-12-28T12:12:00 UT
will hit the Earth at 2015-12-30T18:29:33 UT
with a transit time of    54.3 hours
```

The actual arrival time is 2015-12-31T00:02:00 UT
The prediction error is   -5.5 hours

Alternatively, one can use the well-designed UI via running the command *python2 cat_puma_qt.py*. A proper run of it needs additional Python library *PyQt5* installed. Let us illustrate how this UI can be used with another example CME that erupted at 2016-04-10T11:12 UT. Again, this event was not included in our input dataset when constructing CAT-PUMA either. Figure 1a shows the UI and corresponding CME parameters for this event. Average speed (543 km/s, final speed (547 km/s), angular width (136°) and the MPA (25°) were obtained from the *SOHO LASCO CME Catalog*. The mass of the CME was estimated by the built-in function "cme_mass.pro" in the *SolarSoft IDL*. It turns out to be ~4.6×10^{15} g. By checking the option "Automatically Obtain Solar Wind Parameters", solar wind parameters are obtained automatically from the *OMNIWeb Plus* website (https://omniweb.gsfc.nasa.gov/) after clicking the "Submit" button. Then, actual values of the solar wind parameters is shown. Parameters that are not available from the *OMNIWeb Plus* website are set to 0.00001 (manually input of these parameters are then needed in this case, near real-time solar wind data can be download from the *CDAWeb* website (https://cdaweb.sci.gsfc.nasa.gov/istp_public/). Figure1b shows the prediction result for the above CME, revealing an error of 5.2 hours.

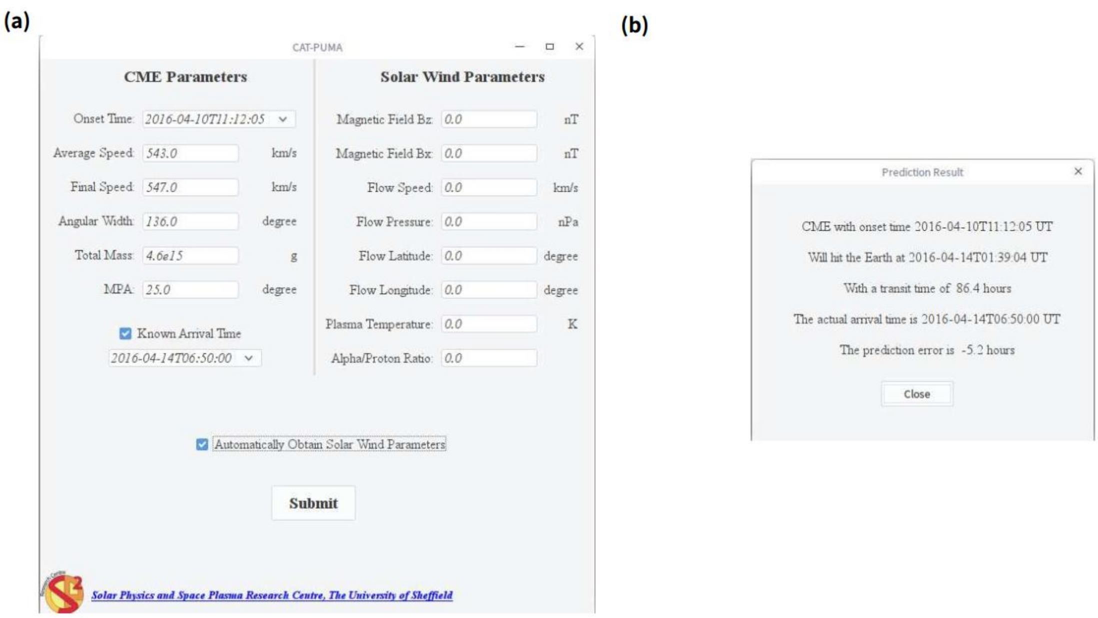

Figure 1. The User Interface of CAT-PUMA

Credit:

Please cite the following paper, if your work (including paper, code, presentation...) is based on the CAT-PUMA or (part of) its original code:

Liu, J., Ye, Y., Shen, C., Wang, Y., Erdélyi, R., A New Tool for CME Arrival Time Prediction Using Machine Learning Algorithms: CAT-PUMA, Astrophys. J., 2018.