Stardust-Reloaded: The H2020 Space Debris and Asteroid research network

Event: Local Training Workshop – I

Location: Madrid, Spain (virtual event)

Date: May 21st, 2020

Project: Asteroid families in the proper elements space

Mentor: Bojan Novakovic

Co-mentors: Zoran Knezevic, Marco Fennuci, Victor Rodriguez

Purpose:

The ESRs should:

- be able to understand the main steps of the procedure to compute the proper orbital elements of asteroids
- be able to understand the main difference between the osculating and proper orbital elements
- learn basics about the codes used to compute the proper elements
- be able to identify the main problems and limitations in the computation and exploitation of the proper elements
- to familiarize themselves with the web-pages providing osculating and proper orbital elements of asteroids, and data on asteroid families
- to apply machine learning tool(s) to classify asteroids into families, using the proper elements data

Brief introduction:

Asteroid proper orbital elements are quasi-constants of motion, that remain practically unchanged over long time spans. These elements serve as parameters for classification of asteroids into families, and as a tool to study their long-term dynamics. The most accurate proper elements are obtained by applying the so-called synthetic theory, that consists of purely numerical procedures. It includes the numerical integration of the asteroid orbits carried out in the framework of a suitable dynamical model, on-line digital filtering of the short-periodic perturbations to produce mean elements, removal of the forced oscillations, and the Fourier analysis of secular time series to remove long-periodic perturbations and compute the proper orbital elements [1].

The first step, that is numerical orbit propagation, is performed starting from the instantaneous osculating orbital elements. The osculating elements are obtained from the observations, and are available at AstDyS and NEODyS web pages, maintained by University of Pisa[1]. Once the numerical integrations of orbits are completed, these data are then used to compute the proper elements. A public domain *OrbFit* package can be used for both, orbit propagation and proper elements computation. The obtained set of asteroid proper elements should then be analyzed in order to identify possibly statistically significant clusterings. In this last step, machine learning techniques should be involved, and the aim should be to test at least two different clustering algorithms.

1 The osculating orbital elements of asteroids are also provided by other web-services, such is the Jet Propulsion Laboratory (JPL), and the Minor Planet Center (MPC).

Technical description:

Here we describe the main steps that should be taken in order to complete this exercise. You will use a numerical integration program called ORBIT9 to perform your long-term integrations. ORBIT9 is one of the workhorses of the professional astronomical community to determine the behavior of small bodies in the Solar System. It has been used in large number of publications in the scientific literature. ORBIT9 is designed to integrate a set of mutually gravitationally interacting bodies (usually planets) together with a group of test particles (usually asteroids) that feel the gravitational influence of the massive bodies but do not affect each other or the massive bodies. ORBIT9[2] numerical integrator executable code and necessary input files are available at https://github.com/stardust-r/LTW-I/tree/master/Asteroid%20Families/Orbit9 To make the code executable on an UNIX machine you may use e.g. `chmod +x orbit9` from the command line. In addition, you will need a small code named PROPSYNTH to compute proper elements from the integrations that you will perform, and should also use PYTHON or MATLAB to visualize some data.

The last task will be to implement unsupervised machine learning based clustering algorithms (in PYTHON or MATLAB) to identify asteroid clusters[3]. This identification should be performed in 3D proper elements space (proper semi-major axis, proper eccentricity and sine of proper inclination). The standard metrics to measure distances in this space should be used, as defined in [2].

Literature:

Main references:

1. Knežević Z., Milani A., 2003, **Proper element catalogs and asteroid families**. Astronomy and Astrophysics 403, 1165
2. Zappala, V., Cellino, A., Farinella, P., Knežević, Z. 1990. **Asteroid Families. I. Identification by Hierarchical Clustering and Reliability Assessment.** The Astronomical Journal 100, 2030

Additional references:

3. Carruba, V., Aljbaae, S., Lucchini, A. 2019. **Machine-learning identification of asteroid groups.** Monthly Notices of the Royal Astronomical Society 488, 1377
4. Milani, A., Cellino, A., Knežević, Z., Novaković, B., Spoto, F., Paolicchi, P. 2014. **Asteroid families classification: Exploiting very large datasets.** Icarus 239, 46
5. Nesvorny, D., Brož, M., Carruba, V. 2015. **Identification and Dynamical Properties of Asteroid Families.** Asteroids IV 297, University of Arizona Press
6. Knežević, Z. 2017. **Computation of Asteroid Proper Elements: Recent Advances**. Serbian Astronomical Journal 194, 1

European Commission

Horizon 2020
European Union funding
for Research & Innovation

---

2 Orbit9 integrator is a part of OrbFit software package. For more information please visit http://adams.dm.unipi.it/~orbmaint/orbfit/

3 https://hdbscan.readthedocs.io/en/latest/how_hdbscan_works.html

APPENDIX - A

Basic user guide for ORBIT9 code

Setup: ORBIT9 takes four input files: orb9.opt, barmerxx.inc, pvenexx.inc and input.cat. Input file orb9.opt tells the code how to perform the calculation, pvenexx.inc supplies the initial conditions of the planets, barmerxx.inc provides data for barycentric corrections (if applied), and input.cat gives the initial positions of the small bodies. You will need these files before you run the code. In addition, you should copy these three files in the working directory of ORBIT9: filter.100, random.dat, and rk.coe.

- orb9.opt file is used to set the integrations performed by ORBIT9
- barmerxx.inc - provides barycentric correction data for a given epoch
- planets.inc - provides initial conditions for planets, for a given epoch
- input.cat - provides initial conditions for planets, for a given epoch

To run intergrations with ORBIT9, only orb9.opt file should be properly set. An example of the orb9.opt file is shown below, and options that need to be set are highlighted in bold.

****

;input control and options file for orbit8
; 1. input file names
inplan='planets.inc'; input file name for initial conditions of the planets
ibar= 1; barycentric correction 1=yes 0=no
inbar='barmerxx.inc'; file containing the barycentric correction
inast='input.cat'; input file name for initial conditions of the asteroids
; 2. job description: ast. are ordered in two list, with and without LCE
nvz= 5 ; no. Lyapounov exponents
4
10
24
490
11805

; 3. options: output $ = option not yet implemented
dt= -100.d0 ; time between two outputs
nout= 20000; output number at job termination
idump= 500; number of outputs between dump/renormalisation
nsamp= 100; sampling ratio of the filter; choose 5, 20,50 or 100
iprqua= 1; how much output: 1 elem. filtr; 2 also sampl.;
sysz= 'HEL' ; output system: BAR, HEL, JAC, HEC
refz= 'INVL1B' ; output reference system: INVL1B, ECLM50, EQUM00, etc
; 4. options: variational equation
v1= 1.00d+06; variation vector norm forcing renormalisation
semim= 3.0; approx. semimajor axis for norm of var.vect.
; 5. options: propagator
h= 0.2; stepsize for multistep (maximum stepsize if automatic)
iauto= 1; automatic stepsize control 1=on 0=off
error= 1.d-13; target integration error, radiants/revolution^2
iord= 12; order of multistep predictor
epms= 0.0d-12; convergence control for corrector $
iork= 12; order of starter
eprk= 1.00d-10; convergence control for implicit Runge-Kutta
lit1= 10; iterations for first step
lit2= 4; iterations for following steps
imet= 1; main integration method: 1 multistep 2 symplectic
iusci=10; output of numerical convergence controls 0 no
icha=1; what to do if non convergent 0 interactive 1 batch, change step $

ll=12; control for Everhart (>0 automatic stepsize control)
; 6. control of close approach output
dmint=0.01; close appr. distance for terrestrial planets
dminj=0.1; close approach distance for giant planets
npoint=10; minimum number of data points for a deep close appr.
; 7. optional perturbations
irelj2=0; relativistic and J2 perturbations 0=no 1=yes
iyark=0; Yarkovsky effect as secular drift in a 0=no 3=yes 1,2=not in use
; END OF INPUT
****

Explanations for the lines that may need to be edited:

- **inplan='planets.inc';** this is the name of the input file containing initial conditions of the planets
- **inbar='barmerxx.inc';** this is the name of the file containing the barycentric correction. Usually a file called barmerxx.inc contains the data for seven planets, and a file named barsunxx.inc provides the barycentric correction for four planets
- **inast='input.cat';** the name of the input file with initial conditions for the asteroids. You may need to change it to allnum.cat, in order to use catalog downloaded from AstDys
- **nvz= 5 ;** number of asteroids to be integrated
- Below the "nvz", please provide a list of all asteroids whose orbits you would like to propagate. *Note that one asteroid per line should be provided!*

Explanations for some important settings that you do not need to modify:

- **dt= -100.d0** ; time between two outputs
- **nout= 20000**; output number at job termination
- "dt" is a time between two outputs, while the "nout" is the number of outputs. Therefore, a total integration time may be obtained by multiplying "dt" and "nout". Note that a negative value of "dt" indicates backward orbit propagation
- **iprqua= 1;** how much output: 1 elem. filtr; 2 also sampl. ORBIT9 provides two types of outputs: osculating elements and mean (filtered) elements. The mean elements are obtained by "on-line" removing short-periodic oscillations from the osculating elements. Option "**iprqua**" sets whether to print both type of elements (option 2), or only mean elements (option 1). *Note that for computation of the proper elements, the mean elements are used!*
- **h= 0.2;** stepsize for multistep (maximum stepsize if automatic) The step size is usually determined automatically. In case you want to enter it manually, you should ensure it is at least about 1/25 of the smallest orbital period in the problem, preferably about 1/50.
- **iauto= 1;** automatic stepsize control 1=on 0=off. Here you chose between the automatic and manual integrator step control. If you turn off automatic control, please see the previous point.

Outputs: ORBIT9 provides the following outputfiles files: orb9.clo, orb9.dma, orb9.dmp, orb9.num, orb9.out, vast.fil, vpla.fil and (optionally) vast.dat and vpla.dat.

- orb9.clo - contains data on close approaches between the small bodies and the planets
- orb9.dma and orb9.dmp - dump files used to extend integrations
- orb9.num - provides some data on the process of numerical integration
- orb9.out - data on the integration initial conditions
- vast.fil, vpla.fil - time series of mean orbital elements for asteroids and planets, respectively
- vast.dat, vpla.dat - time series of osculating orbital elements for asteroids and planets, respectively

Note: for the computation of proper elements, only the vast.fil file is needed!

Important note: ORBIT9 IS UNIX executable files. They should work on MAC, but cannot be used on machines running WINDOWS operating systems!