

OBERON - OBliquity and Energy balance Run On Nbody systems

Duncan Forgan ([github/dh4gan](https://github.com/dh4gan))

August 26, 2016

Contents

1 Getting OBERON

OBERON is hosted on github at +++

2 Compiling OBERON

The source code is in the /src folder, which contains a Makefile. With the appropriate software installed, type 'make' to compile an executable.

3 Code Design

This code utilises object oriented programming, with two main classes.

The `Body` class encapsulates the properties of each body, including methods for calculating gravitational forces. The derived `Star`, `Planet` and `World` classes contain methods for calculating luminosity and climate/spin evolution.

The `System` class takes a vector of `Body` pointers, and controls method calls for the evolution of the collection of `Body` objects under gravity, and `World` objects' climate evolution.

When the code is run, any `World` object in the simulation will have its climate computed under the assumption that it is Earthlike, with each `World` receiving its own set of output files.

4 Input Options

OBERON requires a single input parameter file which contains the essential information required to run the simulation. The parameter file system is quite flexible - the ordering of the options in the file is relatively unimportant, provided that all the global options are at the top of the file before individual body data.

Each variable is read in by checking the first string on each line. Please ensure that the keywords as given below are the first strings on each line, with no whitespace at the beginning, otherwise they will be ignored.

The read in system is robust to '=' and '#' symbols, so do feel free to use these as spacers between data points, or indeed to "comment out" a row and replace it with another input.

4.1 Global Options

- `ParType` 'Positional/Orbital' The format of data for entry (either Cartesian co-ordinates for position and velocity or orbital elements)
- `NBodyOutput` (string) The filename for the N Body data file
- `SnapshotTime` (double) The time interval between data dumps and snapshots
- `NGridPoints:` (integer) The number of grid points used by the latitudinal energy balance model
- `MaximumTime:` (double) The maximum runtime of the simulation
- `SystemName:` (string) A descriptive string for the simulation
- `Number Bodies:` (integer) The number of bodies in the system
- `Restart:` 'T/F' is the simulation a restart? True or False (careful with this - not stable currently) (Default: False)
- `FullOutput` 'T/F' True: Output full snapshots as well as log files with surface averaged values for each body, False: log files only (Default: False)
- `TidalHeating` 'T/F' Compute the tidal heating of the Worlds by the object they orbit? (Default:False)

- PlanetaryIllumination 'T/F' Are 'Planet' objects also luminosity sources? (Default: False)
- ObliquityEvolution 'T/F' Allow the spin parameters of 'World' objects to change? (Default: False)
- CarbonateSilicateCycle 'T/F' Model the effects of the carbonate-silicate cycle? (Default: False)

4.2 Body Options (must be specified for each body)

- BodyName: (string)
- BodyType: 'Star/Planet/World'
- Mass: (double)
- Radius: (double)

For Positional Files:

- Position: (double) (double) (double) Cartesian position vector (astronomical units)
- Velocity: (double) (double) (double) Cartesian velocity vector (2π *AU/yr)

For Orbital Files:

- SemiMajorAxis: (double) Semimajor Axis (AU)
- Eccentricity: (double) Eccentricity
- Inclination: (double) Inclination (+++units?+++)
- LongAscend: (double) Longitude of the Ascending Node
- Periapsis: (double) Argument of Periapsis
- MeanAnomaly: (double) Mean Anomaly
- OrbitCentre: (integer) Where is the initial orbit focus? -1 = (0,0,0), 0=system centre of mass, 1,2,3... = Body 1,2,3)

4.2.1 Star Options

- Luminosity (double) Bolometric Luminosity (solar luminosity)

4.2.2 World Options

- RotationPeriod (double) World rotation period in days
- Obliquity (double) initial obliquity in degrees
- WinterSolstice (double) orbital longitude of the winter solstice (degrees)
- OceanFraction (double, [0.0 → 1.0]) Fraction of the world's surface that is ocean
- InitialTemperature (double) Initial surface temperature of the world (at all latitudes)
- IceMeltingOn 'T/F' Is latent heat of melting for ice accounted for in climate calculation? True or False (Default: False)

5 Outputs

:

- `<WorldName>.<number>` - a snapshot of `<WorldName>`'s latitudinal climate properties
- `<WorldName>.<log>` - a log file for `<WorldName>` tracking globally averaged climate properties and position/orbital properties

An N Body file is also produced (name specified by the user), which is more suited to plotting the entire simulation's evolution (Star and Planet objects included).