Package ‘earthtide’

June 18, 2019

Type Package

Title Parallel Implementation of ‘ETERNA 3.40’ for Prediction and Analysis of Earth Tides

Version 0.0.9

Date 2019-05-17

Maintainer Jonathan Kennel <j kennel@uoguelph.ca>

Description This is a port of ‘Fortran ETERNA 3.4’
  <http://igets.u-strasbg.fr/soft_and_tool.php> by H.G. Wenzel
  for calculating synthetic Earth tides using the
  Hartmann and Wenzel (1994) <doi:10.1029/95GL03324> or

BugReports https://github.com/jkennel/earthtide/issues

URL https://github.com/jkennel/earthtide

License GPL-3

Depends R (>= 3.4.0)

Imports Rcpp (>= 1.0.0), RcppParallel (>= 4.4.2), R6 (>= 2.3.0)

LinkingTo Rcpp (>= 1.0.0), RcppParallel (>= 4.4.2), RcppArmadillo (>= 0.9.200.7.0), BH (>= 1.69.0-1)

Suggests testthat (>= 2.1.0), knitr, rmarkdown

RoxygenNote 6.1.1

VignetteBuilder knitr

Encoding UTF-8

LazyData TRUE

NeedsCompilation yes

SystemRequirements C++11

Author Jonathan Kennel [aut, cre, trl],
  Beth Parker [ths],
  Wenzel Hans-Georg [ctb]

Repository CRAN

Date/Publication 2019-06-17 22:00:07 UTC
**Description**

The goal of this package is to generate synthetic earth tides for use in the R programming language and in particular environmental models. Code was parallelized and refactored to minimize duplication, and to allow for future improvements.

**Details**

You can learn about the earthtide package in the vignettes: browseVignettes(package = "earthtide")

**Author(s)**

**Maintainer:** Jonathan Kennel <jkennel@uoguelph.ca> [translator]

Other contributors:

- Beth Parker <bparker@uoguelph.ca> [thesis advisor]
- Wenzel Hans-Georg [contributor]

**References**


**See Also**

Useful links:

- [https://github.com/jkennel/earthtide](https://github.com/jkennel/earthtide)
- Report bugs at [https://github.com/jkennel/earthtide/issues](https://github.com/jkennel/earthtide/issues)
calc_earthtide

**earthtide** This is a wrapper to the Earthtide R6 class for the prediction of Earth tides. This function is provided for users who would prefer a more typical R function.

### Usage

```r
calc_earthtide(utc, do_predict = TRUE, method = "gravity",
                 astro_update = 1, latitude = 0, longitude = 0, elevation = 0,
                 azimuth = 0, gravity = 0, earth_radius = 6378136.3,
                 earth_eccen = 0.006694379514, cutoff = 1e-06, wave_groups = NULL,
                 catalog = "ksm04", eop = NULL, return_matrix = FALSE,
                 scale = TRUE, ...)
```

### Arguments

- **utc** The date-time in UTC (POSIXct vector).
- **do_predict** run in predict or analyze mode
- **method** One or more of "gravity", "tidal_potential", "tidal_tilt", "vertical_displacement",
  "horizontal_displacement", "n_s_displacement", "e_w_displacement", "vertical_strain",
  "areal_strain", "volume_strain", "horizontal_strain", or "ocean_tides", "pole_tide",
  "lod_tide". The pole tide and lod_tide are used in predict mode even if do_predict
  is FALSE. More than one value can only be used if do_predict == TRUE.
- **astro_update** Integer that determines how often to phases are updated in number of samples.
  Defaults to 1 (every sample), but speed gains are realized with larger values. Typically updating every
  hour will have speed gains and keep precision (ie 3600 for one second data, 60 for minute data, 1 for hourly data).
- **latitude** The station latitude (numeric) defaults to 0.
- **longitude** The station longitude (numeric) defaults to 0.
- **elevation** The station elevation (m) (numeric) defaults to 0.
- **azimuth** Earth azimuth (numeric) defaults to 0.
- **gravity** Gravity at the station (m/s^2) (numeric) 0 to estimate gravity from elevation and latitude.
- **earth_radius** Radius of earth (m) (numeric) defaults to 6378136.3
- **earth_eccen** Eccentricity of earth (numeric) defaults to 6.6943979514e-3
- **cutoff** Cutoff amplitude for constituents (numeric) defaults to 1e-6.
wave_groups Two column data.frame having start and end of frequency groups (data.frame). This data.frame must have two columns with the names 'start', and 'end' signifying the start and end of the wave groupings. An optional third column 'multiplier' can be provided to scale the particular wave group. If column names do no match, the inferred column positions are start, end, multiplier.
catalog Use the "hw95s" catalog or "ksm04" catalog (character).
eop User defined Earth Orientation Parameter (EOP) data.frame with the following columns: datetime, ddt, ut1_utc, lod, x, y, dx, dy
return_matrix Return a matrix of tidal values instead of data.frame. The datetime column will not be present in this case (logical).
scale Scale results when do_predict is FALSE

Value
data.frame of tidal results

Examples
tms <- as.POSIXct('1990-01-01', tz = 'UTC') + c(0, 3600)
wave_groups = data.frame(start = 0, end = 8, multiplier = 1.5)
et <- calc_earthtide(utc = tms,
do_predict = TRUE,
method = c('tidal_potential', 'lod_tide', 'pole_tide'),
astro_update = 1,
latitude = 52.3868,
longitude = 9.7144,
elevation = 110,
gravity = 9.8127,
cutoff = 1.0e-5,
catalog = 'ksm04',
wave_groups = wave_groups)

Earthtide Earthtide class Class to generate synthetic earthtide signals.

Description
Earthtide class
Class to generate synthetic earthtide signals.

Format
An R6Class generator object
Usage

```r
et <- Earthtide$new(
  utc = as.POSIXct("2017-01-01", tz = "UTC") + 0:(24 * 7) * 3600,
  latitude = 52.3868,
  longitude = 9.7144,
  catalog = "ksm04",
  wave_groups = data.frame(start = 0.0, end = 6.0))
```

```r
et$predict(method = "gravity", astro_update = 1)
et$analyze(method = "gravity", astro_update = 1)
et$lod_tide()
et$pole_tide()
et$tide()
et$print()
```

Arguments

Earthtide$new

- `et`: An Earthtide object.
- `utc`: The date-time in UTC (POSIXct vector).
- `latitude`: The station latitude (numeric) defaults to 0.
- `longitude`: The station longitude (numeric) defaults to 0.
- `elevation`: The station elevation (m) (numeric) defaults to 0.
- `azimuth`: Earth azimuth (numeric) defaults to 0 (degrees)
- `gravity`: Gravity at the station (m/s^2) (numeric) 0 to estimate gravity from elevation and latitude.
- `earth_radius`: Radius of earth (m) (numeric) defaults to 6378136.3
- `earth_eccen`: Eccentricity of earth (numeric) defaults to 6.69439795140e-3
- `cutoff`: Cutoff amplitude for constituents (numeric) defaults to 1e-6
- `wave_groups`: Two column data.frame having start and end of frequency groups (data.frame). This data.frame must have two columns with the names 'start', and 'end' signifying the start and end of the wave groupings. An optional third column 'multiplier' can be provided to scale the particular wave group. If column names do no match, the inferred column positions are start, end, multiplier.
- `catalog`: Use the "hw95s" catalog or "ksm04" catalog (character).
- `eop`: User defined Earth Orientation Parameter (EOP) data.frame with the following columns: datetime, ddt, ut1_utc, lod, x, y, dx, dy
- `...`: Currently not used.

Earthtide$predict, Earthtide$analyze

- `method`: For predict and analyze. One of "gravity", "tidal_potential", "tidal_tilt", "vertical_displacement", "horizontal_displacement", "n_s_displacement", "e_w_displacement", "vertical_strain", "areal_strain", "volume_strain", "horizontal_strain" or "ocean_tides".
**Details**

$\texttt{new(utc, latitude, longitude, elevation, azimuth, gravity, earth\_radius, earth\_eccen, cutoff, wave\_groups, catalog, ...)}$
create a new Earthtide object and initialize catalog, station and times.

$\texttt{predict(method, astro\_argument, return\_matrix)}$ generate a combined synthetic Earth tide.

$\texttt{analyze(method, astro\_argument, return\_matrix, scale)}$ generate components of the Earth tide for analysis.

$\texttt{lod\_tide()}$ generate components of the LOD (Length Of Day) tide.

$\texttt{pole\_tide()}$ generate components of the pole tide.

$\texttt{tide()}$ get the tide data.frame.

$\texttt{print()}$ print the Earthtide object.

**References**


**Examples**

```r
et <- Earthtide$new(
  utc = as.POSIXct("2017-01-01", tz = "UTC") + 0:(24 * 7) * 3600, 
  latitude = 52.3868, 
  longitude = 9.7144, 
  catalog = "ksm04", 
  wave_groups = data.frame(start = 0.0, end = 6.0))

et predict(method = "gravity", astro_update = 1)

plot(gravity~datetime, et$tide(), type='l')
```
eterna_wavegroups

**Description**

This data.frame contains wavegroups for different data time spans. The wavegroups should be subset prior to use and the 'time' column provides guidelines based on your input time span.

**Usage**

```r
eterna_wavegroups
```

**Format**

A data.frame The columns are:

- **name**: wave group name
- **start**: lowest frequency of the wave group
- **end**: highest frequency of the wave group
- **time**: applicable to data of what length

**Examples**

```r
utils::data( eterna_wavegroups )
```

---

get_iers

**Description**

get_iers returns a data.frame of earth orientation parameters from (1962-present). This function requires an active internet connection. Bulletins A and B are combined giving precedence to B. The following datasets are downloaded (~ 7 MB):

- `ftp://cddis.gsfc.nasa.gov/pub/products/iers/tai-utc.dat`

**Usage**

```r
get_iers()
```
get_main_frequency

Value
data.frame of earth orientation parameters with the following columns: datetime, ddt, ut1_utc, lod, x, y, dx, dy.

Examples

```r
## Not run:
eop <- get_iers()
```

### End(Not run)

Description

Get the frequency of the wave with the maximum amplitude in a range.

Usage

```r
get_main_frequency(start, end)
```

Arguments

- `start`: the starting frequency in cycles per day (numeric)
- `end`: the ending frequency in cycles per day (numeric)

Value

the main frequency between start and end
Index

*Topic datasets
  eterna_wavegroups, 7
_PACKAGE (earthtide-package), 2

calc_earthtide, 3

Earthtide, 4
Earthtide-class (Earthtide), 4
earthtide-package, 2
eterna_wavegroups, 7

get_iers, 7
get_main_frequency, 8

R6Class, 4