rsfmodel Documentation

Release 0.0.1

John R. Leeman

August 02, 2016

Contents

Here we document the API and installation of rsfmodel.

Contents:

CHAPTER 1

Installation Guide

1.1 Requirements

rsfmodel requires the following packages:

- NumPy >= 1.8.0
- Matplotlib >= 1.4.0

Installation Instructions for NumPy and SciPy can be found at: http://www.scipy.org/scipylib/download.html

Installation Instructions for Matplotlib can be found at: http://matplotlib.org/downloads.html

1.2 Installation

The source code can be downloaded from GitHub. In the base directory of the download, run:

```
python setup.py install
```

This will install rsfmodel into your current Python installation.

CHAPTER 2

The rsfmodel API

Release 0.0.1

Date August 02, 2016

2.1 Plotting

2.1.1 `rsfmodel.plot`

2.2 Rate and State Framework

2.2.1 `rsfmodel.rsf`

exception `rsfmodel.rsf.`**`IncompleteModelError`**

Bases: `exceptions.Exception`

Special error case for trying to run the model with inadequate information.

class `rsfmodel.rsf.`**`LoadingSystem`**

Bases: `object`

Contains attributes relating to the external loading system.

Attributes

k	(float) System stiffness in units of fricton/displacement.
time	(list) List of time values at which the system will be solved.
load-point_velocity	(list) List of the imposed loadpoint velocities at the corresponding time value. Must be same length as time.
v	(float) Slider velocity
mu	(float) The current friciton value of the system.

Methods

`friction_evolution`(*loadpoint_vel*)

`velocity_evolution`()

class `rsfmodel.rsf.`**`Model`**

Bases: *`rsfmodel.rsf.LoadingSystem`*

Houses the model coefficients and does the integration.

Attributes

mu0	(float) Reference friction value at vref.
a	(float) Rate and state constitutive parameter.
vref	(float) System reference velocity at which the reference friction is measured.
state_relations	(list) List of state relations to be used when calculating the model.
results	(namedtuple) Stores all model outputs.

Methods

`readyCheck`()

Determines if all necessary parameters are set to run the model. Will raise appropriate error as necessary.

`solve`(*threshold=2*, ***kwargs*)

Runs the integrator to actually solve the model and returns a named tuple of results.

Parameters threshold : float

Threshold used to determine when integration care should be taken. This threshold is in terms of maximum load-point acceleration before time step is marked.

Returns results : named tuple

Results of the model

2.3 State Relations

2.3.1 `rsfmodel.staterelations`

class `rsfmodel.staterelations.`**`DieterichState`**

Bases: *`rsfmodel.staterelations.StateRelation`*

The slowness or Dieterich state relation as proposed by [1].

Notes

$$\frac{d\theta}{dt} = 1 - \frac{V_{\text{slider}}\theta}{D_c}$$

[1] Dieterich, J. "Modeling of rock friction: 1. Experimental results and constitutive equations." Journal of Geophysical Research: Solid Earth (19782012) 84.B5 (1979): 2161-2168.

Methods

evolve_state (*system*)

set_steady_state (*system*)

class `rsfmodel.staterelations.`**`NagataState`**

Bases: *rsfmodel.staterelations.StateRelation*

The Nagata state relation as proposed by [2]:

Notes

$$\frac{d\theta}{dt} = 1 - \frac{V_{\text{slider}}\theta}{D_c} - \frac{c}{b}\theta\frac{d\mu}{dt}$$

Methods

evolve_state (*system*)

set_steady_state (*system*)

class `rsfmodel.staterelations.`**`PrzState`**

Bases: *rsfmodel.staterelations.StateRelation*

The PRZ state relation as proposed by [3]:

Notes

$$\frac{d\theta}{dt} = 1 - \left(\frac{V_{\text{slider}}\theta}{2D_c}\right)^2$$

Methods

evolve_state (*system*)

set_steady_state (*system*)

velocity_component (*system*)

Perrin-Rice velocity contribution

$$V_{\text{contribution}} = b\ln\left(V_{\text{prz0}}\theta\right)$$

class `rsfmodel.staterelations.`**`RuinaState`**

Bases: *rsfmodel.staterelations.StateRelation*

The slip or Ruina state relation as proposed by [4].

[2] Nagata, K., Nakatani, M., Yoshida, S., "A revised rate-and-state -dependent friction law obtained by constraining constitutive and evolution laws separately with laboratory data," Journal of Geophysical Research: Solid Earth, vol 117, 2012.

[3] Perrin, G., Rice, J., and Zheng, G. "Self-healing slip pulse on a frictional surface." Journal of the Mechanics and Physics of Solids 43.9 (1995): 1461-1495.

[4] Ruina, Andy. "Slip instability and state variable friction laws." J. geophys. Res 88.10 (1983): 359-10.

Notes

$$\frac{dheta}{dt} = -\frac{V_{\text{slider}}\theta}{D_c}\ln\left(\frac{V_{\text{slider}}\theta}{D_c}\right)$$

Methods

evolve_state (*system*)

set_steady_state (*system*)

class rsfmodel.staterelations.**StateRelation**

Bases: object

Abstract state relation object that contains the generally used atributes in state relations (b,Dc).

Attributes

b	(float) Rate and state empirical parameter b.
Dc	(float) Critical slip distance.
state	(float) State variable.

Methods

velocity_component (*system*)

General velocity contribution from a given state variable

Notes

$$V_{\text{contribution}} = b\ln\left(\frac{V_0\theta}{D_c}\right)$$

CHAPTER 3

Indices and tables

- genindex
- modindex
- search

Python Module Index

r

Index