MuMap_fwd

1.0

Generated by Doxygen 1.8.11

Contents

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all documented namespaces with brief descriptions:

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 3

Namespace Documentation

3.1 mumap_fwd Namespace Reference

Classes

- class EOS
- class Melt
- class Poroelasticity
- class Solid

Functions

- def Moduli_to_Velocity (G, K, rho)
- def Velocity_to_Moduli (vs, vp, rho)
- def PREM_Profile (depth)

3.1.1 Detailed Description

```
MuMaP Forwad
Copyright Saswata Hier-Majumder, 2017
Royal Holloway University of London

This module contains classes and functions relevant to calculate the
effect of melting on the elastic moduli and seismic wave velocities in
partially molten rocks. To run the demonstrations, a working version of
Python 2.7.12 or higher, numpy, and matplotlib libraries are required.

Detailed description for each class is provided within the classes.
This is an overview of the classes.

EOS: Contains functions for equations of state for solids and melts.
This class also contains the PREM model of Dziewonski and Anderson (1984).
To initiate this class for a given material known values of some paramters
need to be provided. In most cases, this class will be initiated from
either the Solid or the Melt class, which contain default values for the
parameters.

Solid: This class contains a number of variables and functions relating
to the properties of the solid. All of the parameters are provided with
a default value. See the docstring for more details. By default, this class
uses the PREM model to evaluate the elastic properties corresponding to the
depth parameter provided during instantiation.
```

```
Melt: This class contains physical paramters for calculating the EOS of the
melt. It also defaults to a dihedral angle of 15 degrees. The choice
of parameters for the EOS are set by the variable melt_comp. Please
see the docstring for __init__ for a full description of the currently
available choices.

Poroelasticity:  This class contains functions for calculating effective
elastic properties for a given melt fraction and physical properties
of the solid and the melt, contained in those classes. This class contains
three choices of contiguity models, see the docstrings for more information.
The default model is von Bargen and Waff, which allows for variation
in the idhedral angle. Do not use this model for higher melt fractions.

Example1:
Plot Vinet equation of state of MORB using parameters
from Guillot and Sator(2007), run demo2.py for more plots
    >>> from mumap import *
    >>> import matplotlib.pylab as plt
    >>> rho1=np.linspace(2800.0,4000.0)
    >>> melt1=Melt(melt_comp=2,rho=rho1)
    >>> melt1.Melt_EOS.Vinet()
    >>> plt.plot(melt1.Melt_EOS.P/1.0e9,melt1.Melt_EOS.rho)
    >>> plt.show()

Example 2:
Plot the Vs/V0 as a function of melt fraction. In this case
all the melt resides in films of aspect ratio 0.01. This uses the formulation
of Walsh, 1969
    >>> from mumap import*
    >>> import matplotlib.pylab as plt
    >>> phi=np.linspace(1.0e-3,0.15)
    >>> rock=Poroelasticity(phi=phi)
    >>> rock.Melt=Melt(rho=3200.0)
    >>> rock.film(aspect=0.01)
    >>> plt.plot(rock.meltfrac,rock.vs_over_v0)
    >>> plt.show()

Example 3:
Plot Vs/V0 as a function of melt fraction. In this case, the melt geometry
is calculated from von Bargen and Waff (1986) and the effective elastic
moduli are calculated following Hier-Majumder et al. (2014). Run demo4.py
for more information.
    >>> from mumap import *
    >>> import matplotlib.pylab as plt
    >>> phi1=np.linspace(1.0e-3,0.15)
    >>> rock=Poroelasticity(phi=phi1)
    >>> rock.tube()
    >>> plt.show()
```

3.1.2 Function Documentation

3.1.2.1 def mumap_fwd.Moduli_to_Velocity (*G, K, rho*)

```
This function calculates Vs and Vp from elastic modulie.
Please use SI units
Args:
     G :  Shear modulus in Pa
     K :  Bulk modulus in Pa
     rho: Density in kg/m^3
Returns:
     vs : Shear wave speed in m/s
     vp : P wave speed in m/s
```

3.1.2.2 def mumap_fwd.PREM_Profile (*depth*)

```
This function is used as the default EOS for solids,
unless otherwise chosen. The values of vs and vp are calculated
from the PREM model.
See Table 1 of Dziewonski and Anderson, 1981 for reference
Input:
      depth:  Desired depth ARRAY for calculation, in m
Returns:
      rho  :  Density of the solid
      vs   :  Shear wave speed of the solid (m/s)
      vp   :  P wave velocity of the solid  (m/s)
      G    :  Shear modulus of the solid  (Pa)
      K    :  Bulk modulus of the solid (Pa)
      nu   :  Poisson's ratio of the solid
The returned variables are stored as class attributes
```

3.1.2.3 def mumap_fwd.Velocity_to_Moduli (*vs, vp, rho*)

```
This utility subroutine calculates shear modulus G,
bulk modulus K, and Poisson's ratio nu, from known
shear and P wave velocities. Please use SI units.
Args:
     vs : Shear wave speed in m/s
     vp : P wave speed in m/s
     rho: Density in kg/m^3
Returns:
     G :  Shear modulus in Pa
     K :  Bulk modulus in Pa
     nu:  Poisson's ratio
```

Chapter 4

Class Documentation

4.1 mumap_fwd.EOS Class Reference

Public Member Functions

- def __init__ (self, K0, rho, Kp, rho0)
- def Vinet (self)
- def BM3 (self)
- def PREM (self, depth=60.0e3)

Public Attributes

- **K0**
- **rho**
- **Kp**
- **rho0**
- **P**
- **K**
- **vs**
- **vp**

4.1.1 Detailed Description

```
This class contains equations of state
to calculate density and elastic
moduli of solids and melts

To initiate this class, the following parameters are needed
Parameters:
     K0   : Bulk modulus of the material at surface, in Pa
     rho  : Density (kg/m^3) at the desired condition can be an array
     Kp   : Pressure derivative of the bulk modulus
     rho0 : Density (kg/m^3) at surface
These parameters are used by Vinet and BM3 (Third order Birch-Murnaghan
EOS), can be used for both solid and melt. The EOS PREM is for solids
only. This class is instantiated within classes Melt and Solid
```

4.1.2 Constructor & Destructor Documentation

4.1.2.1 def mumap_fwd.EOS.__init__ (*self, K0, rho, Kp, rho0*)

```
Initiate the class with known values of density, rho, surface
bulk modulus, K0, and the pressure derivative
of the bulk modulus, Kp.
```

4.1.3 Member Function Documentation

4.1.3.1 def mumap_fwd.EOS.BM3 (*self*)

```
This function uses the third order Birch-Murnaghan EOS
to calculate the bulk modulus and pressure for a given density.
Inputs:
     rho  : Density
     rho0 : Surface density
     Kp   : Pressure derivative of bulk modulus
     K0   : Bulk modulus at surface
These inputs are created during instantiation of the class
Returns:
     P    : Pressure in Pa
     K    : Bulk modulus in Pa
Both of these are stored as public attributes
```

4.1.3.2 def mumap_fwd.EOS.PREM (*self, depth =* 60.0e3)

```
This function is used as the default EOS for solids,
unless otherwise chosen. The values of vs and vp are calculated
from the PREM model.
See Table 1 of Dziewonski and Anderson, 1981 for reference
Input:
      depth:  Desired depth for calculation, in m
Returns:
      rho  :  Density of the solid
      vs   :  Shear wave speed of the solid (m/s)
      vp   :  P wave velocity of the solid  (m/s)
      G    :  Shear modulus of the solid  (Pa)
      K    :  Bulk modulus of the solid (Pa)
      nu   :  Poisson's ratio of the solid
The returned variables are stored as class attributes
```

4.1.3.3 def mumap_fwd.EOS.Vinet (*self*)

```
This function uses the Vinet EOS to calculate
the bulk modulus and pressure for a given density.
Inputs:
     rho  : Density
     rho0 : Surface density
     Kp   : Pressure derivative of bulk modulus
     K0   : Bulk modulus at surface
These inputs are created during instantiation of the class
Returns:
     P    : Pressure in Pa
     K    : Bulk modulus in Pa
Both of these are stored as public attributes
```

The documentation for this class was generated from the following file:

- /media/sash/23CD-8107/codes/sashcodes/MuMAP/python/mumap_fwd.py

4.2 mumap_fwd.Melt Class Reference

Public Member Functions

- def __init__ (self, theta=20.0, rho=3000.0, melt_comp=1)

Public Attributes

- **theta**
- **K0**
- **Kp**
- **rho0**
- **description**
- **rho**
- **Melt_EOS**
- **P**

4.2.1 Detailed Description

```
This class contains functions and propeties
of the melt. To initiate the class, the following paramters
are needed
Parameters:
    theta     : degrees, solid-melt dihedral angle
    rho       : Melt density (kg/m^3)
    melt_comp : Melt composition for the EOS
Pressure of the melt is calculated by using the EOS
of choice using the melt_comp variable.
```

4.2.2 Constructor & Destructor Documentation

4.2.2.1 def mumap_fwd.Melt.__init__ (*self*, *theta* = 20.0, *rho* = 3000.0, *melt_comp* = 1)

```
This function sets the properties of the melt
 See Table 1 of Wimert and Hier-Majumder(2012) for details
The choices are
    1 = peridotite melt, (2273 K) Guillot and Sator (2007)
    2 = MORB,(2073 K) Guillot and Sator  (2007)
    3 = peridotite melt, Ohtani and Maeda (2001)
    4 = MORB, Ohtani and Maeda (2001)
    5= peridotite+5%CO2 from Ghosh etal (2007)
Any other value defaults to 1
The optional value of dihedral angle
is the second input.
```

The documentation for this class was generated from the following file:

- /media/sash/23CD-8107/codes/sashcodes/MuMAP/python/mumap_fwd.py

4.3 mumap_fwd.Poroelasticity Class Reference

Public Member Functions

- def __init__ (self, theta=20.0, phi=0.15)
- def WHM12 (self)
- def HMRB06 (self)
- def VBW86 (self)
- def set_contiguity (self, contiguity_model=1)
- def film (self, aspect=0.01)
- def tube (self)
- def HH2000_Film (self)
- def HH2000_combined (self)

Public Attributes

- **theta**
- **meltfrac**
- **Contiguity**
- **Melt**
- **Solid**
- **vs_over_v0**
- **vp_over_v0**
- **K_over_K0**
- **G_over_G0**
- **aspect**

4.3.1 Detailed Description

```
This class contains functions and variables
relevant to calculating the effective properties
for different melt geometries. There are 3
different functions for calculating contiguity,
the ratio between the area of grain-grain contact and
the surface area of a grain. The result is saved
in a variable, self.Contiguity.

On initiation, the following variables are created
Parameters
    theta      : dihedral angle
    melfrac    : Melt volume fraction (between 0 and 1)
    Contiguity : Area fraction of grain-grain contact
    Melt       : Object melt is created with theta
    Solid      : Object solid is created
```

4.3.2 Constructor & Destructor Documentation

4.3.2.1 def mumap_fwd.Poroelasticity.__init__ (*self*, *theta* = 20.0, *phi* = 0.15)

```
Initiates the melt geometry class
theta is the dihedral angle
```

4.3.3 Member Function Documentation

4.3.3.1 def mumap_fwd.Poroelasticity.film (*self*, *aspect* = 0.01)

```
This function calculates Vs/Vs0,Vp/VP0, K/K0, and G/G0
Based on the model of Walsh, 1969. On input, the aspect
ratio is the ratio between the minor and major axis
of the elliptical melt inclusion.
Subscript 1 is the stronger phase.
On return, the four members in the array
elastic_melt are respectively
Vs/Vs0,Vp/VP0, K/K0, and G/G0. Please only enter SI units.
Input:
    aspect         : Aspect ratio of melt film
Parameters
    self.Solid.K   : Solid bulk modulus
    self.Solid.G   : Solid shear modulus
    self.Solid.rho : Density of the solid
    self.meltfrac  : Melt volume fraction
Returns:
    vs_over_v0     : Normalized shear velocity
    vp_over_v0     : Normalized P wave velocity
    K_over_K0      : Normalized bulk modulus
    G_over_G0      : Normalized shear modulus
```

4.3.3.2 def mumap_fwd.Poroelasticity.HH2000_combined (*self*)

```
This function calculates the S and P wave velocity reduction
following the model of Paradigm 2 of Hammonds and Humphreys (2000)
It assumes relaxed, random cuspate melt geometry. See Table 2
of Hammond and Humphreys (2000) for details.
Inputs:
       meltfrac   : Melt fraction in the rock
Returns:
       vs_over_v0 : Dimnesionless normalized Vs
       vp_over_v0 : Dimensionless normalized vp

All variables are stored as public attributes to the class
```

4.3.3.3 def mumap_fwd.Poroelasticity.HH2000_Film (*self*)

```
This function calculates the S and P wave velocity reduction
following the model of Paradigm 1 of Hammonds and Humphreys (2000)
It assumes relaxed, random cuspate melt geometry. See Table 2
of Hammond and Humphreys (2000) for details.
Inputs:
meltfrac   : Melt fraction in the rock
Returns:
vs_over_v0 : Dimnesionless normalized Vs
vp_over_v0 : Dimensionless normalized vp

All variables are stored as public attributes to the class
```

4.3.3.4 def mumap_fwd.Poroelasticity.HMRB06 (*self*)

```
This function returns the two dimensional
measurement of contiguity from Hier-Majumder
et al (2006). This is not recommended as it typically
returns contiguity values higher than the 3D models.
Parameters:
    meltfrac    : melt volume fraction
    theta       : semidihedral angle
Returns:
    Contiguity  : Fractional area of grain-grain contact
```

4.3.3.5 def mumap_fwd.Poroelasticity.set_contiguity (*self, contiguity_model* = 1)

```
Calculates contiguity, default is VBW86
```

4.3.3.6 def mumap_fwd.Poroelasticity.tube (*self*)

```
This function calculates the S and P wave velocity reduction
following the model of Hier-Majumder et al. (2014)
Inputs:
      meltfrac            : Melt fraction in the rock
      contiguity          : contiguity of the rock
      Solid.K             : Bulk modulus of solid
      Solid.G             : Shear modulus of solid
      Solid.rho           : Density of the solid
      Solid.nu            : Poisson's ratio of the Solid
      Melt.Melt_EOS.K     : Bulk modulus of the melt
      Melt.Melt_EOS.rho   : Density of the melt


Returns:
      vs_over_v0          : Dimensionless normalized Vs
      vp_over_v0          : Dimensionless normalized vp
      K_over_K0           : Dimensionless normalized bulk modulus
      G_over_G0           : Dimensionless normalized shear modulus
All variables are stored as public attributes to the class.
```

4.3.3.7 def mumap_fwd.Poroelasticity.VBW86 (*self*)

```
This function returns the contiguity
of a partially molten unit cell as a function
of melt volume fraction and dihedral angle,
using the parametrization of von Bargen and
Waff (1986). Notice that the parameter Agg
shouldn't become zero at zero melt fraction,
as erroneously indicated in their article.
it is fixed by subtracting it from pi to match their
Figure 10. Doesn't work
beyond melt volume fraction fo 0.18.
Parameters:
    meltfrac    : melt volume fraction
    theta       : semidihedral angle
Returns:
    Contiguity  : Fractional area of grain-grain contact
```

4.3.3.8 def mumap_fwd.Poroelasticity.WHM12 (*self*)

```
This function returns the contiguity as a function of
melt fraction. The dihedral angle, even taken as an input
is currently not used, as the model of Wimert and
Hier-Majumder is valid for a constant dihedral angle of
approximately 30 degreees. Doesn't work well beyond melt
volume fraction of 0.25.
Parameters:
    meltfrac    : melt volume fraction
Returns:
    Contiguity  : Fractional area of grain-grain contact
```

The documentation for this class was generated from the following file:

- /media/sash/23CD-8107/codes/sashcodes/MuMAP/python/mumap_fwd.py

4.4 mumap_fwd.Solid Class Reference

Public Member Functions

- def __init__ (self, K0=126.3e9, Kp=4.28, G0=78.0e9, Gp=1.71, rho0=2600.0, depth=60.0e3)

Public Attributes

- **K0**
- **K**
- **Kp**
- **G0**
- **G**
- **rho0**
- **rho**
- **depth**
- **Solid_EOS**
- **nu**

4.4.1 Detailed Description

```
This class contains functions and properties of the solid.
To initiate this class, the following parameters are needed
Parameters:
    K0    : Bulk modulus of the material at surface, in Pa
    Kp    : Pressure derivative of the bulk modulus
    G0    : Shear modulus of the material at surface, in Pa
    Gp    : Pressure derivative of the shear modulus
    rho0  : Density (kg/m^3) at surface
    depth : m, below the surface
By default, solid parameters are evaluated using the PREM
model.
```

4.4.2 Constructor & Destructor Documentation

4.4.2.1 def mumap_fwd.Solid.__init__ (*self*, *K0* = 126.3e9, *Kp* = 4.28, *G0* = 78.0e9, *Gp* = 1.71, *rho0* = 2600.0, *depth* = 60.0e3)

```
initiates the class, assigns the values of K0,Kp,G0, and Gp
from Abramson et al (1997), Table 4.1 of Poirier's Introduction
to the physics of the Earth's interior 2nd ed.
rho0 is from Prem model. Depth is given 60 km by default
On initiation, K,G, and nu are given surface values
```

The documentation for this class was generated from the following file:

- /media/sash/23CD-8107/codes/sashcodes/MuMAP/python/mumap_fwd.py

Index