

Underworld3 Documentation

Front page

Underworld Team

2024-06-03

Table of contents

1 Underworld 3

Front page

Underworld Team

LANDING Page

Not much content in the intro at this point

Update the `_toc.yml` first, then fix the rest

Part I

Introduction

2 Underworld 3

Underworld Team

+++

Not much content in the intro at this point

Update the `_toc.yml` first, then fix the rest

Part II

Constitutive Models

3 Underworld 3: Constitutive Models

Underworld Team

4 Introduction to constitutive relationships in Underworld3

Constitutive models relate fluxes of a quantity to the gradients of the unknowns. For example, in thermal diffusion, there is a relationship between heat flux and temperature gradients which is known as *Fourier's law* and which involves an empirically determined thermal conductivity.

$$q_i = k \frac{\partial T}{\partial x_i}$$

and in elasticity or fluid flow, we have a relationship between stresses and strain (gradients of the displacements) or strain-rate (gradients of velocity) respectively

$$\tau_{ij} = \mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right)$$

where η , here, is a material property (elastic shear modulus, viscosity) that we need to determine experimentally.

These material properties can be non-linear (they depend upon the quantities that they relate), and they can also be anisotropic (they vary with the orientation of the material). In the latter case, the material property is described through a *constitutive tensor*. For example:

$$q_i = k_{ij} \frac{\partial T}{\partial x_j}$$

or

$$\tau_{ij} = \mu_{ijkl} \left(\frac{\partial u_k}{\partial x_l} + \frac{\partial u_l}{\partial x_k} \right)$$

+++ {"magic_args": "[markdown]"}

4.1 Constitutive tensors

In Underworld, the underlying implementation of any constitutive relationship is through the appropriate tensor representation, allowing for a very general formulation for most problems. The constitutive tensors are described symbolically using **sympy** expressions. For scalar equations, the rank 2 constitutive tensor (k_{ij}) can be expressed as a matrix without any loss of generality.

For vector equations, the constitutive tensor is of rank 4, but it is common in the engineering / finite element literature to transform the rank 4 tensor to a matrix representation. This has some benefits in terms of legibility and, perhaps, results in common code patterns across the formulation for scalar equations.

We can write the stress and strain rate tensors (τ_{ij} and $\dot{\epsilon}_{ij}$ respectively) as

$$\tau_I = (\tau_{00}, \tau_{11}, \tau_{22}, \tau_{12}, \tau_{02}, \tau_{01})$$

$$\dot{\epsilon}_I = (\dot{\epsilon}_{00}, \dot{\epsilon}_{11}, \dot{\epsilon}_{22}, \dot{\epsilon}_{12}, \dot{\epsilon}_{02}, \dot{\epsilon}_{01})$$

where I indexes $\{i, j\}$ in a specified order and symmetry of the tensor implies only six (in 3D) unique entries in the nine entries of a general tensor. With these definitions, the constitutive tensor can be written as C_{IJ} where I , and J range over the independent degrees of freedom in stress and strain (rate). This is a re-writing of the full tensor, with the necessary symmetry required to respect the symmetry of τ and $\dot{\epsilon}$

The simplified notation comes at a cost. For example, $\tau_I \dot{\epsilon}_I$ is not equivalent to the tensor inner product $\tau_{ij} \dot{\epsilon}_{ij}$, and C_{IJ} does not transform correctly for a rotation of coordinates.

However, a modest transformation of the matrix equations is helpful:

$$\tau_I^* = C_{IJ}^* \epsilon_J^*$$

Where

$$\tau_I^* = P_{IJ} \tau_J^*$$

$$\epsilon_I^* = P_{IJ} \epsilon_J^*$$

$$C_{IJ}^* = P_{IK} C_{KL} P_{LJ}$$