

# 1. Viscous Stress

## 1.1. Introduction

Viscous stress is a measure of how a fluid's internal frictional forces affect the flow of momentum through the fluid. The way in which the viscous stress depends on the rate of strain, that is the rate at which some deformation occurs, and the way in which this stress depends on other variables like temperature and magnetic field, is entirely governed by the specific fluid in question. Some fluids, water being a good example, behave in a *Newtonian* fashion, where the viscous stress is linearly proportional to the rate of strain. Other fluids, such as ketchup, are examples of *shear-thinning* fluids, where the higher the rate of strain, that is the faster a deformation occurs, the less viscous the fluid behaves. There are of course many other non-Newtonian materials each with interesting properties arising from the nature of the viscous stress and its relationship to the strain rate.

## 1.2. The Viscous Stress Tensor

The viscous stress tensor  $\sigma$  is introduced into the momentum balance as,

$$\rho \frac{Dv_i}{Dt} = \frac{\partial \sigma_{ij}}{\partial x_j}, \quad (1)$$

where  $\rho$  is the density and  $v_i$  is the fluid velocity in the  $i$ -th direction. Similarly the tensor contributes to the energy balance in the following way,

$$\rho \frac{D\epsilon}{Dt} = \varepsilon_{ij} \sigma_{ij}, \quad (2)$$

where  $\epsilon$  is the internal energy,  $\varepsilon$  is the rate of strain tensor, and we have made use of the Einstein summation convention that repeated indices in a term indicate summation,

$$x_i y_i \equiv \sum_{i=1}^3 x_i y_i.$$

## 1.3. Newtonian Fluids

For Newtonian fluids, viscous stress is directly proportional to the rate of strain so for a compressible fluid, the viscous stress tensor is given by

$$\sigma_{ij} = 2\nu(\varepsilon_{ij} - \frac{1}{3}\delta_{ij}\nabla \cdot \vec{v}),$$

where  $\nu$  is the dynamic viscosity, and  $\delta_{ij}$  is the Kronecker delta.

The strain rate tensor is given by

$$\varepsilon_{ij} = \frac{1}{2} \left( \frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right).$$

Sometimes bulk viscosity is included as an extra term  $\mu_v \nabla \cdot \vec{v}$  in the viscous stress tensor that influences uniform compression and dilation of a fluid, however it is usually an effect even smaller than conventional viscosity and so here it is neglected (TODO: CITE SOMETHING)

### 1.3.1. Viscous Forces

To calculate the force resulting from this form of viscosity, we calculate the  $i$ -th component of the divergence of the viscous stress tensor,

$$\frac{\partial \sigma_{ij}}{\partial x_j} = 2\nu \frac{\partial}{\partial x_j} \left( \varepsilon_{ij} - \frac{1}{3} \delta_{ij} \nabla \cdot \vec{v} \right). \quad (3)$$

Differentiating each term separately, it can be shown that

$$\frac{\partial \varepsilon_{ij}}{\partial x_j} = \frac{1}{2} \left( \nabla^2 v_i + \frac{\partial}{\partial x_i} (\nabla \cdot \vec{v}) \right),$$

and

$$\frac{\partial}{\partial x_j} \delta_{ij} \nabla \cdot \vec{v} = \frac{\partial}{\partial x_i} (\nabla \cdot \vec{v}).$$

Subbing these back into equation (3),

$$\frac{\partial \sigma_{ij}}{\partial x_j} = \nu \left( \frac{1}{3} \frac{\partial}{\partial x_i} (\nabla \cdot \vec{v}) + \nabla^2 v_i \right),$$

that is,

$$\nabla \cdot \sigma = \frac{\nu}{3} \nabla (\nabla \cdot \vec{v}) + \nu \nabla^2 \vec{v}.$$

Physically, these two terms represent different ways in which the internal frictional forces can manifest themselves. The first term involving  $\nabla(\nabla \cdot \vec{v})$  is a resistance to compression or expansion. In the incompressible limit, where  $\nabla \cdot \vec{v} \rightarrow 0$  this term would disappear as expected. The second term involving  $\nabla^2 \vec{v}$  looks very much like a diffusion term and in fact can represent the diffusion of momentum through the fluid through neighbouring layers of fluid.

### 1.3.2. Mathematical Properties

In a Newtonian fluid, since the viscous stress is linearly related to the strain rate, the viscous stress tensor inherits many of the properties of the strain rate tensor.

The strain rate tensor in this case is easily shown to be symmetric, that is  $\varepsilon_{ij} = \varepsilon_{ji}$ , the symmetry being a consequence of conservation of angular momentum and the isotropy of the fluid. As a result, the Newtonian viscous stress tensor is also symmetric, reducing the number of individually varying components from 9 down to 6.

The viscous stress tensor in the given form is also traceless, that is  $\sigma_{ii} = 0$ . This is to ensure uniform compression or dilation results in zero viscous behaviour.

## 1.4. Lare3D implementation

### 1.4.1. Calculation of the viscosity stress tensor

In order to try and elucidate what the Lare3D code is doing in implementing Newtonian viscosity I shall split the code up and explain the sections. All of this code occurs inside nested for loops that run over the indices representing the  $x$ ,  $y$  and  $z$  directions, as below. The loop also sets some useful variables for the previous and next indices.

```
D0 iz = 0, nz + 1
  izm = iz - 1
  izp = iz + 1
  D0 iy = 0, ny + 1
    iym = iy - 1
    iyp = iy + 1
    D0 ix = 0, nx + 1
      ixm = ix - 1
      ixp = ix + 1
```

The following code finds the velocity in the  $x$  direction,  $dvxdx$ , at the centre of a cell by extrapolating values at the boundaries.

```
vxb = (vx(ix ,iy ,iz ) + vx(ix ,iym,iz ) &
      + vx(ix ,iy ,izm) + vx(ix ,iym,izm)) * 0.25_num

vxbm = (vx(ixm,iy ,iz ) + vx(ixm,iym,iz ) &
      + vx(ixm,iy ,izm) + vx(ixm,iym,izm)) * 0.25_num

dvxdx = (vxb - vxbm) / dxb(ix)
```

Similar code is used to find the other derivatives of  $v_x$ ,  $v_y$  and  $v_z$  with respect to  $x$ ,  $y$  and  $z$ .

The values for the strain rate tensor, represented as  $s_{xy}$  for the  $xy$  component, are given by the following code. Note that there are only six components due to the symmetric nature of the tensor.

```
sxy = dvxy * 0.5_num
sxz = dvxz * 0.5_num
syz = dvyz * 0.5_num
sxx = (2.0_num * dvxdx - dvydy - dvzdz) * third
syy = (2.0_num * dvydy - dvxdx - dvzdz) * third
szz = (2.0_num * dvzdz - dvxdx - dvydy) * third
```

The strain rate tensor is incorporated into the total stress tensor in a Newtonian way, as can be seen in the following code for the  $xx$  component.

```
qxx(ix,iy,iz) = qxx(ix,iy,iz) + 2.0_num * sxx * rho(ix,iy,iz) * visc3
```

Similar calculations for the other components of  $q$  are performed. Note the multiplication by  $\rho$  here to account for the compressibility. In the event of

an incompressible fluid being simulated, this would cancel in the momentum equation.

#### 1.4.2. Use of viscosity stress tensor

After the calculation of the viscosity stress tensor, it is used to calculate the viscous heating term to be added to the energy using equation (2).

```
visc_heat(ix,iy,iz) = &
    qxy(ix,iy,iz) * dvxy + qxz(ix,iy,iz) * dvxz &
    + qyz(ix,iy,iz) * dvyz + qxx(ix,iy,iz) * dvxdx &
    + qyy(ix,iy,iz) * dvydy + qzz(ix,iy,iz) * dvzdz
```

Then, at a later stage, the two contributions from the viscous forces, the parallel and shear forces, are calculated as per equation (1). The parallel viscous force is calculated in the  $x$  direction as follows, with similar calculations done for  $y$  and  $z$ . Recall that the values of  $\mathbf{q}$  were calculated at the cell centres, thus in order to correctly calculate the force at the cell boundaries, we must interpolate, done here using a simple spatial average.

```
w1 = (qxx(ix ,iy ,iz ) + qxx(ix ,iyp,iz ) &
    + qxx(ix ,iy ,izp) + qxx(ix ,iyp,izp)) * 0.25_num
w2 = (qxx(ixp,iy ,iz ) + qxx(ixp,iyp,iz ) &
    + qxx(ixp,iy ,izp) + qxx(ixp,iyp,izp)) * 0.25_num
fx = fx + (w2 - w1) / dxc(ix)
```

The shear viscous forces are then calculated for each of the off-parallel directions. The contributions to the  $x$  direction are shown.

```
w1 = (qxy(ix ,iy ,iz ) + qxy(ixp,iy ,iz ) &
    + qxy(ix ,iy ,izp) + qxy(ixp,iy ,izp)) * 0.25_num
w2 = (qxy(ix ,iyp,iz ) + qxy(ixp,iyp,iz ) &
    + qxy(ix ,iyp,izp) + qxy(ixp,iyp,izp)) * 0.25_num
fx = fx + (w2 - w1) / dyc(iy)
```

```
w1 = (qxz(ix ,iy ,iz ) + qxz(ixp,iy ,iz ) &
    + qxz(ix ,iyp,iz ) + qxz(ixp,iyp,iz )) * 0.25_num
w2 = (qxz(ix ,iy ,izp) + qxz(ixp,iy ,izp) &
    + qxz(ix ,iyp,izp) + qxz(ixp,iyp,izp)) * 0.25_num
fx = fx + (w2 - w1) / dzc(iz)
```