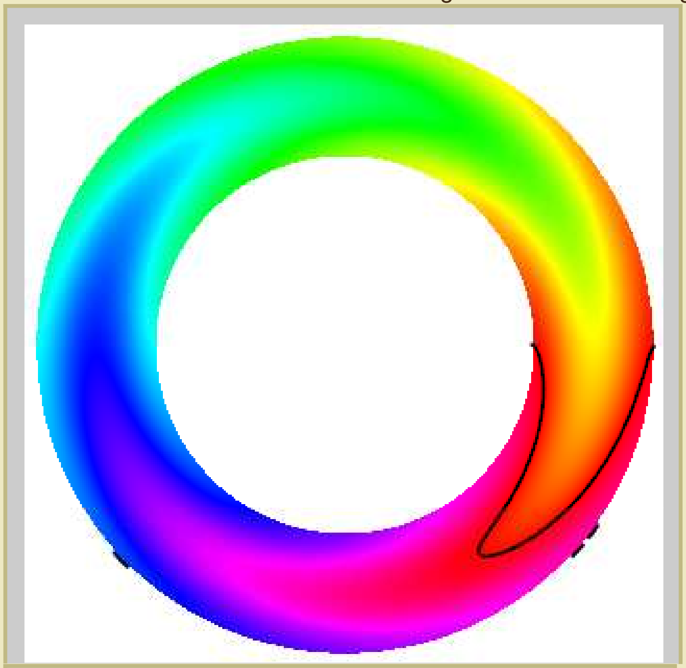
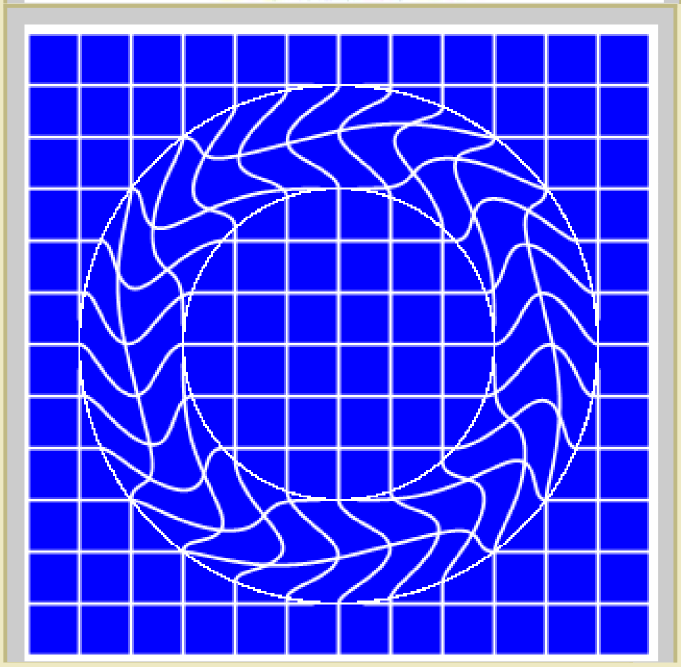
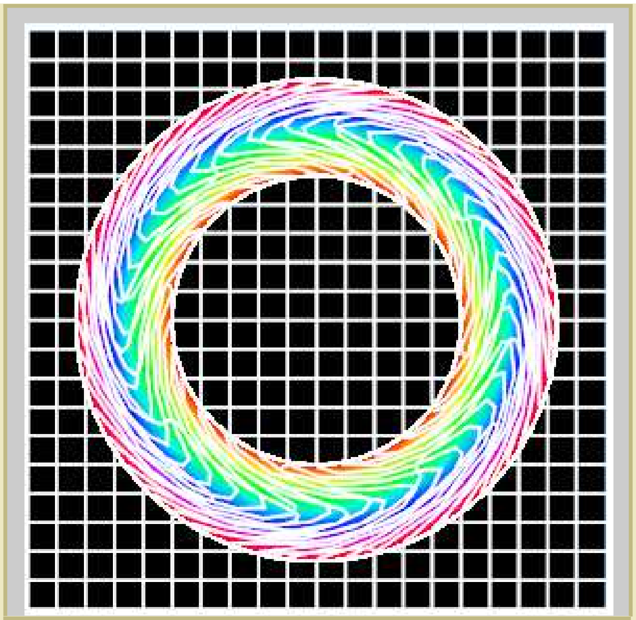
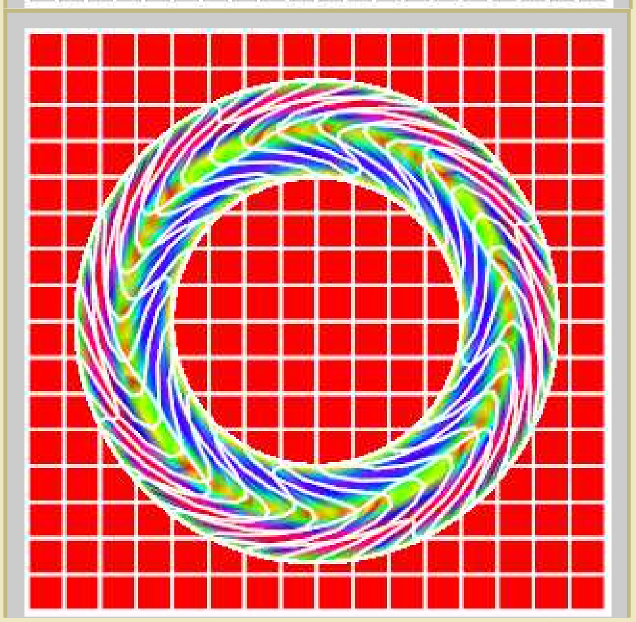
This notebook provides step-by-step derivation of the Method of Manufactured Solutions (MMS) for an extremely high-shear prescribed deformation.
This is the working notebook that was ultimately summarized in Appendix 1 of reference [1].

This notebook includes commands to generate the following pictures of a special deformation:









As seen, this deformation is designed so that all points in the ENTIRE domain undergo a state of simple shear along with superimposed rotation. As such, this motion is an excellent choice to test large-deformation numerical codes for robustness and and accurate in their solution to the equations of motion as well as testing the constitutive model (and/or its implementation) for proper behavior under extraordinarily large shears (like what might be expected near a penetrator) and under superimposed rotation (as required from material frame indifference yet shockingly often missed in constitutive model testing).

The basic idea behind MMS is to dream up an interesting deformation like the one in this notebook, from which you can then derive corresponding kinematics quantities, such as

displacement
velocity
accelerations
deformation gradient tensor

strain and strain rate
Jacobian of the deformation (from which evolving mass density is then known)

Knowing these known at all locations, and also knowing your choice of an elastic constitutive model, you should have enough information to determine the corresponding stress tensors required to induce the motion. In this notebook, we avoid introducing any particular constitutive model until the very last step. This way, the bulk of the work can be applied to any arbitrary choice of isotropic elastic constitutive model (or any inelastic one if unloading is disallowed). Because this notebook designs a very special material motion in which all points are undergoing a state of varying degrees of simple shear with varying degrees of superimposed rotation, we can write the stress state generically as a function of the shear strain at any given location. The detail of the function would come from considering your choice of constitutive model in the special case of simple shear without superimposed rotation, and (as further clarified below) it is therefore treated as a known generic function. Simple shear is simple enough that this function can be found analytically for most material models, but even that isn't needed -- if you can't get it analytically, then you can generate it numerically and represent it via a lookup table (although such an approach has the disadvantage of not properly testing your constitutive model for things like the host code failing to send the correct strain definition as input).

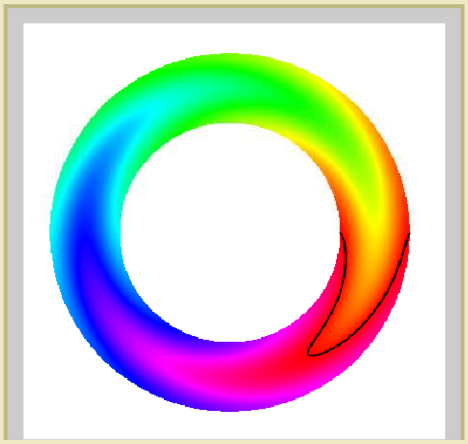
Once the local material motion is applied along with the elastic constitutive model to find the local stress at each location in the domain of the manufactured material motion, the next step is to then evaluate the divergence of stress. Recalling that the acceleration was a previously computed kinematic property of the MMS, and now knowing the stress divergence, the corresponding body force may be determined from the equations of motion! That's what this notebook does.

An MMS verification test is performed in a host (such as an FEM code) as follows:

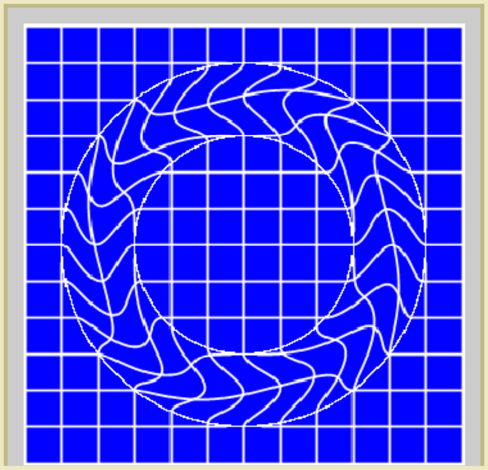
1. use the analytically determined initial particle velocities (from this notebook) to initialize the state
(the MMS in this notebook is designed to have a zero initial velocity, alleviating the burden of this step)
2. use the analytically determined body force (from this notebook) to drive material motion
3. use the analytically determined surface tractions (from this notebook) as part of what drives the material motion
(the MMS in this notebook is designed to have surface tractions all zero, alleviating the burden of this step)
4. Run the host code to extract displacement as a function of time and position
5. Compare the computed displacement to the MMS analytical displacement (from this notebook) to evaluate
simulation errors, which (after resolving constitutive model and algorithm bugs that
are often revealed by doing an MMS simulation in the first place) are ultimately the result of
mesh refinement errors.
6. Redo the computations on increasingly refined meshes to obtain a rational assessment
of the solution algorithm's rate of convergence in non-trivial heterogenous material motion.

Note: because this particular MMS will involve motion inside a ring such that material strain is identically zero at the boundary of that ring, a host code has two options:

Option A. Run a simulation of just the ring itself using zero traction at the boundary:



Option B. Run a simulation of a domain in which there is material both inside and outside the ring:



This is a convenient option for Eulerian codes that always run all simulations on rectangular domains.

This option is also a nice way to clearly see simulation errors, because any grid motion outside the ring is a visible indicator of error -- the exact solution has no motion whatsoever!

Now let's get started working out the kinematics for the "Generalized Vortex" rotation of material points around the ring illustrated above...

For this MMS, we consider a mapping from initial position $\underset{\sim}{X}$ to current position $\underset{\sim}{x}$ given by

$$\underset{\sim}{x} = \underset{\approx}{Q}.\underset{\sim}{X}$$

where $\underset{\approx}{Q}$ is an orthogonal tensor with components

$$[\underset{\approx}{Q}] = \begin{pmatrix} \text{Cos}[\alpha] & -\text{Sin}[\alpha] & 0 \\ \text{Sin}[\alpha] & \text{Cos}[\alpha] & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

This notebook allows the rotation angle to vary with radial coordinate R, but not with angular coordinate. The rotation angle is also allowed to vary with time. Specifically,

α[R,t]=g[t]h[R]

where the functions g and h are given (specific examples are selected below for the visualizations, but they are kept generic for now in order to give as much generality as possible to the MMS formulas while still keeping them easy to work with). The function h[R] controls the shape of the bump in motion, illustrated by the black line in the first image above. The g[t] function controls the magnitude.

If the g[t] function has the property that g[0] = 0, then it follows that there is no deformation in the initial state. If it has a zero slope at time zero so that g'[0] = 0, then the material has no initial velocity, thus making it easier to test a code by not requiring it to set up a complicated initial velocity field.

If the h[R] function has the property that its value and slope are both zero at the inner and outer radii of the ring, this MMS will entail zero traction at the ring boundary, thus simplifying the testing of the code. The h[R] function can, of course, later be changed to have a nonzero slope at the boundaries to test the code's ability to correctly impose shear tractions. In that case, the appropriate shear traction would be inferred from the constitutive model in simple shear (beyond scope of this notebook).

Keeping the α[R,t] function generic, the goal of this notebook is to find that body force field corresponding to this motion.

Note that

$$\frac{d\underset{\approx}{Q}}{d\alpha} = \underset{\approx}{A}.\,\underset{\approx}{Q},$$

where $\underset{\approx}{A}$ is the axial tensor associated with the axis of rotation. Namely,

$$\underset{\approx}{A} = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

This is simply a counter-clockwise rotation of 90° in the plane of motion.

Thus, for any scalar *s*,

$$\frac{\partial \underset{\approx}{Q}}{\partial s} = \frac{d\underset{\approx}{Q}}{d\alpha}\frac{\partial \alpha}{\partial s} = \underset{\approx}{A}.\,\underset{\approx}{Q}\frac{\partial \alpha}{\partial s}$$

This lemma will be used in upcoming formulas whenever any derivative of the $\underset{\approx}{Q}$ tensor is needed.

Letting $\underset{\sim}{\mathbf{e}}_r$, $\underset{\sim}{\mathbf{e}}_\theta$, and $\underset{\sim}{\mathbf{e}}_z$ denote the spatial cylindrical basis vectors, notice (for future reference):

$$\underset{\approx}{A} \cdot \underset{\sim}{\mathbf{e}}_r = \underset{\sim}{\mathbf{e}}_\theta$$

$$\underset{\approx}{A} \cdot \underset{\sim}{\mathbf{e}}_\theta = -\underset{\sim}{\mathbf{e}}_r$$

$$\underset{\approx}{A} \cdot \underset{\sim}{\mathbf{e}}_z = \underset{\sim}{\mathbf{0}}$$

Velocity and acceleration

Recall that

$$\underset{\sim}{x} = \underset{\approx}{Q}.\underset{\sim}{X}$$

The velocity is the time derivative holding $\underset{\sim}{X}$ constant:

$$\underset{\sim}{v} = \left(\frac{\partial \underset{\sim}{x}}{\partial t}\right)_{\underset{\sim}{X}} = \frac{\partial \underset{\approx}{Q}}{\partial t}.\underset{\sim}{X} = \underset{\approx}{A}.\underset{\approx}{Q}\frac{\partial \alpha}{\partial t}.\underset{\sim}{X} = g'[t]\, h[R]\, \underset{\approx}{A}.\underset{\approx}{Q}.\underset{\sim}{X}$$

or

$$\underset{\sim}{v} = g'[t]\, h[R]\, \underset{\approx}{A}.\underset{\sim}{x}$$

The acceleration is

$$\underset{\sim}{a} = g''[t]\, h[R]\, \underset{\approx}{A}.\underset{\sim}{x} + g'[t]\, h[R]\, \underset{\approx}{A}.\underset{\sim}{v}$$

or

$$\underset{\sim}{a} = g''[t]\, h[R]\, \underset{\approx}{A}.\underset{\sim}{x} - (g'[t]\, h[R])^2\, \underset{\sim}{x}$$

The spatial components of acceleration are

$$a_r = g''[t]\, h[R]\, \underset{\approx}{A}.\underset{\sim}{x} - (g'[t]\, h[R])^2\, \underset{\sim}{x}$$

Noting that $\underset{\sim}{x} = r\, \underset{\sim}{\mathbf{e}}_r = R\, \underset{\sim}{\mathbf{e}}_r$, we get familiar equations for circular motion:

$$\underset{\sim}{v} = R\, \omega\, \underset{\sim}{\mathbf{e}}_{\vartheta},$$

$$\underset{\sim}{a} = R\, \dot{\omega}\, \underset{\sim}{\mathbf{e}}_{\vartheta} - R\, \omega^2\, \underset{\sim}{\mathbf{e}}_r$$

where

ω=g'[t] h[R]

and

$\dot{\omega}$=g''[t] h[R]

Deformation gradient

Let $\underset{\sim}{\boldsymbol{E}}_R$, $\underset{\sim}{\boldsymbol{E}}_\Theta$, and $\underset{\sim}{\boldsymbol{E}}_z$ denote the reference cylindrical basis vectors (i.e., those corresponding to a particle's initial position $\underset{\sim}{X}$ on the path instead of its current position $\underset{\sim}{x}$).

The reference gradient of the rotation angle is given by

$$\frac{d\alpha}{d\underset{\sim}{X}} = g[t]\, \frac{dh}{dR}\, \frac{dR}{d\underset{\sim}{X}} = g[t]\, h'[R]\, \underset{\sim}{\boldsymbol{E}}_R$$

Recall that

$$\frac{d\underset{\approx}{Q}}{d\alpha} = \underset{\approx}{A}.\underset{\approx}{Q}$$

The deformation gradient is

$$\underset{\approx}{F} = \frac{d\underset{\sim}{x}}{d\underset{\sim}{X}} = \frac{d\left(\underset{\approx}{Q}.\underset{\sim}{X}\right)}{d\underset{\sim}{X}}$$

$$= \underset{\approx}{Q} + \underset{\sim}{X}.\frac{d\underset{\approx}{Q}^T}{d\underset{\sim}{X}}$$

$$= \underset{\approx}{Q} + \underset{\sim}{X}.\frac{d\underset{\approx}{Q}^T}{d\alpha}\frac{d\alpha}{d\underset{\sim}{X}}$$

$$= \underset{\approx}{Q} + \underset{\sim}{X}.\frac{d\underset{\approx}{Q}^T}{d\alpha}\, g[t]\, h'[R]\ \underset{\sim}{\boldsymbol{E}}_R$$

$$= \underset{\approx}{Q} + g[t]\, h'[R]\left(\frac{d\underset{\approx}{Q}}{d\alpha}.\underset{\sim}{X}\right)\ \underset{\sim}{\boldsymbol{E}}_R$$

$$= \underset{\approx}{Q} + g[t]\, h'[R]\left(\underset{\approx}{A}.\,\underset{\approx}{Q}.\underset{\sim}{X}\right)\ \underset{\sim}{\boldsymbol{E}}_R$$

$$= \underset{\approx}{Q}.\left(\underset{\approx}{I} + g[t]\, h'[R]\left(\underset{\approx}{A}.\underset{\sim}{X}\right)\ \underset{\sim}{\boldsymbol{E}}_R\right)$$

$$= \underset{\approx}{Q}.\left(\underset{\approx}{I} + g[t]\, R\, h'[R]\ \underset{\sim}{\boldsymbol{E}}_\Theta\ \underset{\sim}{\boldsymbol{E}}_R\right)$$

In the penultimate step, we used the easily confirmed fact that $\underset{\approx}{Q}^T.\underset{\approx}{A}.\,\underset{\approx}{Q} = \underset{\approx}{A}$.

In the ultimate step, it was observed that $\underset{\approx}{A}.\underset{\sim}{X} = \underset{\approx}{A}.\left(R\ \underset{\sim}{\boldsymbol{E}}_R\right) = R\,\underset{\approx}{A}.\,\underset{\sim}{\boldsymbol{E}}_R = R\ \underset{\sim}{\boldsymbol{E}}_\Theta$

The result

$$\underset{\approx}{F} = \underset{\approx}{Q}.\left(\underset{\approx}{I} + g[t]\, R\, h'[R]\ \underset{\sim}{\boldsymbol{E}}_\Theta\ \underset{\sim}{\boldsymbol{E}}_R\right)$$

is a state of simple shear in the Θ direction with shear plane tangent to the circumference, and with superimposed rotation.

Define

$$\xi[R] = \frac{1}{2}\mathrm{R\ h'[R]}$$

Then the large-deformation shear strain is

$$\epsilon[t,\ R] = g[t]\ \xi[R]$$

The deformation gradient may be written as

$$\underset{\approx}{F} = \underset{\approx}{Q}.\underset{\approx}{q}.\underset{\approx}{\mathcal{F}}.\underset{\approx}{q}^T$$

where

$$\underset{\approx}{\mathcal{F}} = \underset{\approx}{I} + 2\, g[t]\ \xi[R]\ \underset{\sim}{\boldsymbol{E}}_2\ \underset{\sim}{\boldsymbol{E}}_1$$

$$\underset{\approx}{q} = \begin{pmatrix} \mathrm{Cos}[\Theta] & -\mathrm{Sin}[\Theta] & 0 \\ \mathrm{Sin}[\Theta] & \mathrm{Cos}[\Theta] & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Note that

$$\underset{\approx}{F} = \underset{\approx}{r}.\underset{\approx}{\mathcal{F}}.\underset{\approx}{q}^T$$

where

$$\underset{\approx}{r} = \begin{pmatrix} \mathrm{Cos}[\vartheta] & -\mathrm{Sin}[\vartheta] & 0 \\ \mathrm{Sin}[\vartheta] & \mathrm{Cos}[\vartheta] & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Note that $\underset{\approx}{\mathcal{F}}$ is independent of the angular coordinate, and $\frac{d\underset{\approx}{\mathcal{F}}}{dR} = 2\, g[t]\ \xi'[R]\ \underset{\sim}{\boldsymbol{E}}_2\ \underset{\sim}{\boldsymbol{E}}_1$

Note that $\underset{\approx}{q}$ is independent of the radial coordinate, and $\frac{d\underset{\approx}{q}}{d\Theta} = \underset{\approx}{A}.\underset{\approx}{q}$

Note that $\underset{\approx}{r}$ depends on both angular and radial coordinates. Noting that $\vartheta = \Theta+\alpha = \Theta+g[t]h[R]$, we have

$$\frac{d\underset{\approx}{r}}{d\vartheta} = \underset{\approx}{A}.\underset{\approx}{r}$$

$$\left(\frac{\partial \underset{\approx}{r}}{\partial\Theta}\right)_R = \frac{d\underset{\approx}{r}}{d\vartheta}\frac{\partial\vartheta}{\partial\Theta} = \underset{\approx}{A}.\underset{\approx}{r}$$

$$\left(\frac{\partial \underset{\approx}{r}}{\partial R}\right)_\Theta = \frac{d\underset{\approx}{r}}{d\vartheta}\frac{\partial\vartheta}{\partial R} = g[t]\, h'[R]\, \underset{\approx}{A}.\underset{\approx}{r}$$

Using all of these things, the gradient of the deformation gradient (which is here for completeness to help any future efforts to include gradient plasticity, but isn't used again here in this notebook) is

$$\frac{d\underset{\approx}{F}}{d\underset{\sim}{X}} = \left(\frac{\partial \underset{\approx}{F}}{\partial R}\right)_\Theta \underset{\sim}{\mathbf{E}}_R + \frac{1}{R}\left(\frac{\partial \underset{\approx}{F}}{\partial \Theta}\right)_R \underset{\sim}{\mathbf{E}}_\Theta$$

$$= \left(g[t]\, h'[R]\, \underset{\approx}{A}.\underset{\approx}{F} + \underset{\approx}{r}\,.2\, g[t]\, \xi'[R]\;\; \underset{\sim}{E}_2\;\; \underset{\sim}{E}_1.\underset{\approx}{q}^T\right)\underset{\sim}{\mathbf{E}}_R + \frac{1}{R}\left(\underset{\approx}{A}.\underset{\approx}{F} + \underset{\approx}{F}.\underset{\approx}{A}^T\right)\underset{\sim}{\mathbf{E}}_\Theta$$

$$= \left(g[t]\, h'[R]\, \underset{\approx}{A}.\underset{\approx}{F} + 2\, g[t]\, \xi'[R]\;\; \underset{\sim}{\mathbf{e}}_\vartheta\, \underset{\sim}{\mathbf{E}}_R\right)\underset{\sim}{\mathbf{E}}_R + \frac{1}{R}\left(\underset{\approx}{A}.\underset{\approx}{F} + \underset{\approx}{F}.\underset{\approx}{A}^T\right)\underset{\sim}{\mathbf{E}}_\Theta$$

```
{R Cos[α] Cos[Θ] - R Sin[α] Sin[Θ], R Cos[Θ] Sin[α] + R Cos[α] Sin[Θ], 0}
```

Recall that

$$\underset{\approx}{\mathcal{F}} = \underset{\approx}{I} + 2\,\epsilon[t,\, R]\;\; \underset{\sim}{E}_2\;\; \underset{\sim}{E}_1$$

This is a pure shear. Suppose that the material is homogeneous and isotropic. Let $\underset{\approx}{\mathcal{S}}$ denote the second-Piola Kirchhoff (PK2) stress associated with the deformation $\underset{\approx}{\mathcal{F}}$. Then, for an isotropic material, it follows that the PK2 stress associated with $\underset{\approx}{F}$ must be

$$\underset{\approx}{S} = \underset{\approx}{q}.\underset{\approx}{\mathcal{S}}.\underset{\approx}{q}^T$$

The first Piola-Kirchhoff (PK1) stress is then

$$\underset{\approx}{T} = \underset{\approx}{F}.\underset{\approx}{S} = \left(\underset{\approx}{Q}.\underset{\approx}{q}.\underset{\approx}{\mathcal{F}}.\underset{\approx}{q}^T\right).\left(\underset{\approx}{q}.\underset{\approx}{\mathcal{S}}.\underset{\approx}{q}^T\right) = \left(\underset{\approx}{Q}.\underset{\approx}{q}.\underset{\approx}{\mathcal{F}}.\underset{\approx}{\mathcal{S}}.\underset{\approx}{q}^T\right) = \underset{\approx}{Q}.\underset{\approx}{q}.\underset{\approx}{\mathcal{T}}.\underset{\approx}{q}^T$$

or

$$\underset{\approx}{T} = \underset{\approx}{r}.\underset{\approx}{\mathcal{T}}.\underset{\approx}{q}^T$$

where

$$\underset{\approx}{\mathcal{T}} = \underset{\approx}{\mathcal{F}} \cdot \underset{\approx}{\mathcal{S}}$$

is the PK1 stress associated with deformation $\underset{\approx}{\mathcal{F}}$, which depends on R indirectly through dependence of the shear strain on R, but this reference PK1 stress is not dependent on the angular coordinate. Thus

$$\frac{d\underset{\approx}{\mathcal{T}}}{dR} = \frac{d\underset{\approx}{\mathcal{T}}}{d\epsilon}\frac{d\epsilon}{dR} = \frac{d\underset{\approx}{\mathcal{T}}}{d\epsilon}\, g[t]\, \xi'[R]$$

Divergence of PK1

The gradient of PK1 stress is

$$\frac{d\underset{\sim}{T}}{d\underset{\sim}{X}} = \left(\frac{\partial \underset{\sim}{T}}{\partial R}\right)_\Theta \underset{\sim}{\mathbf{E}}_R + \frac{1}{R}\left(\frac{\partial \underset{\sim}{T}}{\partial \Theta}\right)_R \underset{\sim}{\mathbf{E}}_\Theta$$

$$= \left(\frac{\partial\, \underset{\sim}{r}.\underset{\sim}{\mathcal{T}}.\underset{\sim}{q}^T}{\partial R}\right)_\Theta \underset{\sim}{\mathbf{E}}_R + \frac{1}{R}\left(\frac{\partial\, \underset{\sim}{r}.\underset{\sim}{\mathcal{T}}.\underset{\sim}{q}^T}{\partial \Theta}\right)_R \underset{\sim}{\mathbf{E}}_\Theta$$

$$= \left(g[t]\, h'[R]\, \underset{\sim}{A}.\underset{\sim}{r}.\underset{\sim}{\mathcal{T}}.\underset{\sim}{q}^T + g[t]\, \xi'[R]\ \ \underset{\sim}{r}.\frac{d\underset{\sim}{\mathcal{T}}}{d\epsilon}.\underset{\sim}{q}^T\right)\underset{\sim}{\mathbf{E}}_R + \frac{1}{R}\left(\underset{\sim}{A}.\underset{\sim}{r}.\underset{\sim}{\mathcal{T}}.\underset{\sim}{q}^T + \underset{\sim}{r}.\underset{\sim}{\mathcal{T}}.\underset{\sim}{q}^T.\underset{\sim}{A}^T\right)\underset{\sim}{\mathbf{E}}_\Theta$$

$$= \left(g[t]\, h'[R]\, \underset{\sim}{A}.\underset{\sim}{T} + g[t]\, \xi'[R]\ \ \underset{\sim}{r}.\frac{d\underset{\sim}{\mathcal{T}}}{d\epsilon}.\underset{\sim}{q}^T\right)\underset{\sim}{\mathbf{E}}_R + \frac{1}{R}\left(\underset{\sim}{A}.\underset{\sim}{T} + \underset{\sim}{T}.\underset{\sim}{A}^T\right)\underset{\sim}{\mathbf{E}}_\Theta$$

The reference divergence of PK1 stress is then

$$\rho_0(\underset{\sim}{a} - \underset{\sim}{b}) = \mathrm{DIV}(\underset{\sim}{T}) = \left(g[t]\, h'[R]\, \underset{\sim}{A}.\underset{\sim}{T}.\underset{\sim}{\mathbf{E}}_R + g[t]\, \xi'[R]\ \ \underset{\sim}{r}.\frac{d\underset{\sim}{\mathcal{T}}}{d\epsilon}.\underset{\sim}{\mathbf{E}}_1\right) + \frac{1}{R}\left(\underset{\sim}{A}.\underset{\sim}{T}.\underset{\sim}{\mathbf{E}}_\Theta + \underset{\sim}{T}.\underset{\sim}{\mathbf{E}}_R\right)$$

The spatial radial component is

$$\rho_0(a_r - b_r) = \left(-g[t]\, h'[R]\, \underset{\sim}{\mathbf{e}}_\vartheta.\underset{\sim}{T}.\underset{\sim}{\mathbf{E}}_R + g[t]\, \xi'[R]\ \underset{\sim}{\mathbf{E}}_1.\frac{d\underset{\sim}{\mathcal{T}}}{d\epsilon}.\underset{\sim}{\mathbf{E}}_1\right) + \frac{1}{R}\left(-\underset{\sim}{\mathbf{e}}_\vartheta.\underset{\sim}{T}.\underset{\sim}{\mathbf{E}}_\Theta + \underset{\sim}{\mathbf{e}}_r.\underset{\sim}{T}.\underset{\sim}{\mathbf{E}}_R\right)$$

$$= \left(-g[t]\, h'[R]\, \mathcal{T}_{21} + g[t]\, \xi'[R]\ \frac{d\mathcal{T}_{11}}{d\epsilon}\right) + \frac{1}{R}(\mathcal{T}_{11} - \mathcal{T}_{22})$$

$$= \left(\xi'[R]\ \frac{d\mathcal{T}_{11}}{d\epsilon} - h'[R]\, \mathcal{T}_{21}\right) g[t] + \frac{1}{R}(\mathcal{T}_{11} - \mathcal{T}_{22})$$

The spatial tangential component is

$$\rho_0(a_\vartheta - b_\vartheta) = \left(g[t]\, h'[R]\, \underset{\sim}{\mathbf{e}}_r.\underset{\sim}{T}.\underset{\sim}{\mathbf{E}}_R + g[t]\, \xi'[R]\ \underset{\sim}{\mathbf{E}}_2.\frac{d\underset{\sim}{\mathcal{T}}}{d\epsilon}.\underset{\sim}{\mathbf{E}}_1\right) + \frac{1}{R}\left(\underset{\sim}{\mathbf{e}}_r.\underset{\sim}{T}.\underset{\sim}{\mathbf{E}}_\Theta + \underset{\sim}{\mathbf{e}}_\vartheta.\underset{\sim}{T}.\underset{\sim}{\mathbf{E}}_R\right)$$

$$= \left(g[t]\, h'[R]\, \mathcal{T}_{11} + g[t]\, \xi'[R]\ \frac{d\mathcal{T}_{21}}{d\epsilon}\right) + \frac{1}{R}(\mathcal{T}_{12} + \mathcal{T}_{21})$$

$$= \left(\xi'[R]\ \frac{d\mathcal{T}_{21}}{d\epsilon} + h'[R]\, \mathcal{T}_{11}\right) g[t] + \frac{1}{R}(\mathcal{T}_{12} + \mathcal{T}_{21})$$

NOTE KEY ADVANTAGE: This result is expressed in terms of the CARTESIAN components of the PK1 stress corresponding to a homogeneous pure shear. That means you only need to evaluate the constitutive model for that special case.

Finding body force

Recall that

$$\underset{\sim}{v} = R\,\omega\, \underset{\sim}{\mathbf{e}}_\vartheta,$$

$$\underset{\sim}{a} = R\,\dot{\omega}\, \underset{\sim}{\mathbf{e}}_\vartheta - R\,\omega^2\, \underset{\sim}{\mathbf{e}}_r$$

where

where ω =g'[t] h[R]

$\dot{\omega}$ =g''[t] h[R]

Thus

$$b_r = -R\,(g'[t]\, h[R])^2 - \frac{1}{\rho_0}\left\{\left(\xi'[R]\ \frac{d\mathcal{T}_{11}}{d\epsilon} - h'[R]\, \mathcal{T}_{21}\right) g[t] + \frac{1}{R}(\mathcal{T}_{11} - \mathcal{T}_{22})\right\}$$

$$b_\vartheta = R\, g''[t]\, h[R] - \frac{1}{\rho_0}\left\{\left(\xi'[R]\ \frac{d\mathcal{T}_{21}}{d\epsilon} + h'[R]\,\mathcal{T}_{11}\right) g[t] + \frac{1}{R}\left(\mathcal{T}_{12} + \mathcal{T}_{21}\right)\right\}$$

Algorithm

Procedure:

1. Select g[t] and h[R] functions

2. Select an isotropic elastic constitutive model

   Work out the analytical solution for the Cartesian components of the PK1 stress,

$$\underset{\sim}{\mathcal{T}} = \begin{pmatrix} \mathcal{T}_{11} & \mathcal{T}_{12} & \mathcal{T}_{13} \\ \mathcal{T}_{21} & \mathcal{T}_{22} & \mathcal{T}_{23} \\ \mathcal{T}_{31} & \mathcal{T}_{32} & \mathcal{T}_{33} \end{pmatrix}$$

   corresponding to a homogeneous deformation with deformation gradient

$$\underset{\sim}{\mathcal{F}} = \begin{pmatrix} 1 & 0 & 0 \\ 2\epsilon & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

3. Substitute the result into the above formulas for b_r and b_ϑ and evaluate the result using

$$\epsilon = \frac{1}{2}\, g[t]\, R\, h'[R] = g[t]\, \xi[R]$$

   This will give for b_r and b_ϑ as functions of time and radius.

4. Evaluate the body force vector by

$$\underset{\sim}{\mathbf{b}} = b_r\, \underset{\sim}{\mathbf{e}}_r + b_\vartheta\, \underset{\sim}{\mathbf{e}}_\vartheta$$

   If Cartesian component are desired, evaluate this body force vector using

$$\underset{\sim}{\mathbf{e}}_r = \mathrm{Cos}[\vartheta]\, \underset{\sim}{\mathbf{E}}_1 + \mathrm{Sin}[\vartheta]\, \underset{\sim}{\mathbf{E}}_2$$

$$\underset{\sim}{\mathbf{e}}_\vartheta = -\mathrm{Sin}[\vartheta]\, \underset{\sim}{\mathbf{E}}_1 + \mathrm{Cos}[\vartheta]\, \underset{\sim}{\mathbf{E}}_2$$

   where

$$\vartheta = \Theta + g[t]\ h[R]$$

Specific study

The above sections provide all of the analytical work phrased in terms of generic functions g[t], h[R], and whatever the isotropic elastic constitutive model predicts for the $\mathcal{T}_{11}(\epsilon)$, $\mathcal{T}_{22}(\epsilon)$, and $\mathcal{T}_{21}(\epsilon)$ PK1 stress components in response to a deformation gradient $\begin{pmatrix} 1 & 0 & 0 \\ 2\epsilon & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$. In other words, the body force is determined at this point entirely in terms of these five user-controllable functions. This “specific study” section now makes specific choices for the g[t] and h[R] functions in order to generate graphics of the MMS deformation.

In[1]:=
```
ClearAll["Global`*"]
```

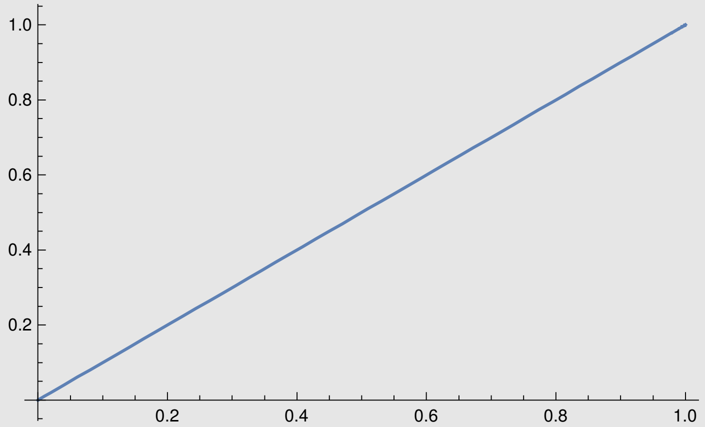
Step 1: select formula for time and radial dependence of the rotation angle

In[2]:=
```
g =.
h =.
```

Set g[t] such that g=1 when t=tref.

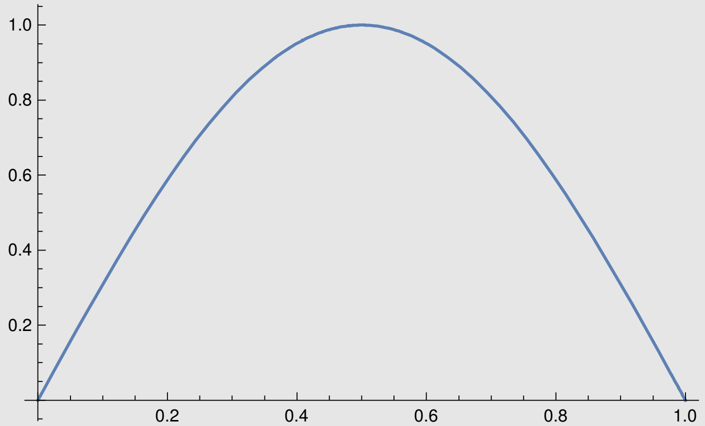
In[4]:=
```
tref = 1;
g[t_] := t/tref
Plot[g[t], {t, 0, tref}]
```

Out[6]=



In[7]:=
```
tref = 1;
g[t_] := Sin[π t/tref]
Plot[g[t], {t, 0, tref}]
```

Out[9]=



In[10]:=
```
α = g[t] h[R]
```

Out[10]=
```
h[R] Sin[π t]
```

Define the radial value at the center of the ring:

$$c = \frac{1}{2}(a + b)$$

define a normalized radial location variable,

$$\overline{R} = \frac{R-c}{b-c}$$

Note that

$\overline{R} = -1$ at R=a

$\overline{R} = +1$ at R=b

define

$$\eta = c_0 + c_2 \overline{R}^2 + c_4 \overline{R}^4$$

$$\frac{d\eta}{d\overline{R}} = 2\, c_2 \overline{R} + 4\, c_4 \overline{R}^3$$

Demand that

$\eta[0]=1$	$\Longrightarrow$	$c_0=1$
$\eta[1]=0$	$\Longrightarrow$	$c_0 + c_2 + c_4=0$
$\eta'[1]=0$	$\Longrightarrow$	$2\, c_2 + 4\, c_4=0$

Thus

$c_0=1$

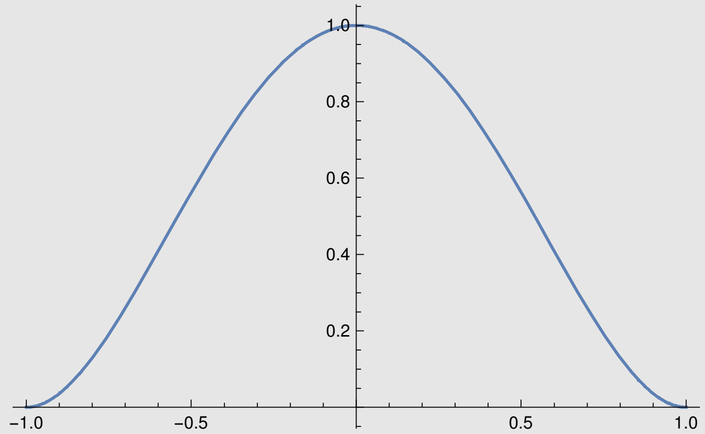
$c_4=1$

$c_2=-2$

The following plot confirms that this function is zero and has zero slope at its endpoints:

In[110]:=
```
η[Rbar_] := 1 - 2 Rbar^2 + Rbar^4
Plot[η[Rbar], {Rbar, -1, 1}]
```

Out[111]=

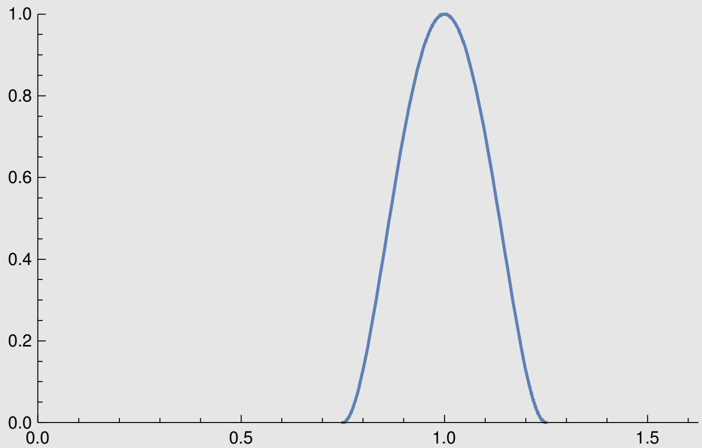


Perform a change of variables to get a similar bump from radial locations a to b.
This defines the shape for the ring deformation, h[R]:

In[13]:=

```
rref = 1;
a = rref (1 - 1 / 4);
b = rref (1 + 1 / 4);
c = 1/2 (b + a);
h[R_] := Evaluate[η[(R - c)/(b - c)]]
Plot[h[R], {R, a, b}, PlotRange → {{0, b + a / 2}, {0, 1}}]
h[R]
```

Out[18]=



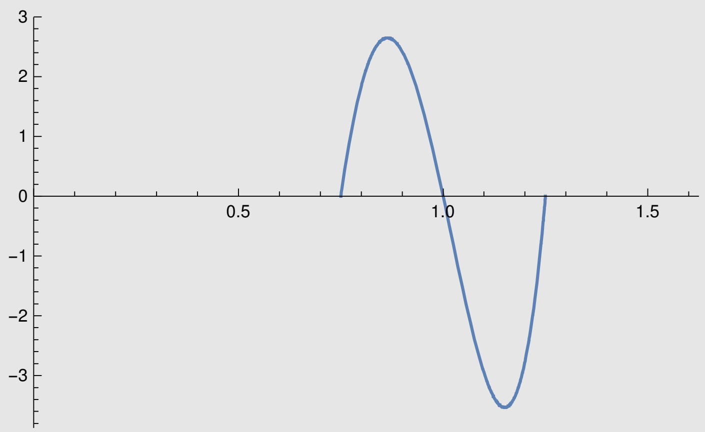
Out[19]=

$$1 - 32 \, (-1 + R)^2 + 256 \, (-1 + R)^4$$

It is understood that the h[R] function must evaluate to zero for R<a and for R>b.

Reference shear strain within the zone a<R<b:

In[20]:=
```
ξ[R_] := 1/2 R h'[R]
Plot[ξ[R], {R, a, b}, PlotRange → {{0, b + a / 2}, All}]
ξ[R]
```

Out[21]=


Out[22]= $\frac{1}{2}\left(-64\ (-1+R)+1024\ (-1+R)^3\right)\ R$

This is the shear strain in the material when the amplitude function g[t] is unity.

Step 2:

In[23]:=
$$\mathcal{F} = \begin{pmatrix} 1 & 0 & 0 \\ 2\epsilon & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix};$$
```
J = Det[ℱ]
```

Out[24]= 1

Consider, as an example, simple neo-Hookian constitutive model.
Under conditions of simple shear, with the deformation gradient tensor given by the formula in the boxed cell above.
Below is the calculation of Cauchy stress $\underset{\approx}{\sigma}$ from this deformation.

Keep in mind that our MMS calculation of the body force requires components of the PK1 stress, not the Cauchy stress, so an appropriate conversion is performed using standard transformations in Continuum Mechanics:

In[25]:=
```
τ = 2 G ϵ;
𝕀 = IdentityMatrix[3];
σ = (λ Log[J])/J 𝕀 + μ/J (ℱ.Transpose[ℱ] - 𝕀);
Print["Cauchy stress is ", σ // MatrixForm];
Print["But we need the PK1 stress, which is, for this model,..."];
PK2 = Inverse[ℱ].(J σ).Transpose[Inverse[ℱ]];
𝒯 = PK1 = ℱ.PK2 // Simplify;
𝒯 // MatrixForm
```

Cauchy stress is $\begin{pmatrix} 0 & 2\,\epsilon\,\mu & 0 \\ 2\,\epsilon\,\mu & 4\,\epsilon^2\,\mu & 0 \\ 0 & 0 & 0 \end{pmatrix}$

But we need the PK1 stress, which is, for this model,...

Out[32]//MatrixForm=

$$\begin{pmatrix} 0 & 2\,\epsilon\,\mu & 0 \\ 2\,\epsilon\,\mu & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Incidentally, we note that this simple shear motion induces a nonzero mechanical pressure:

In[33]:=
```
pressure = 1/3 Tr[σ]
```

Out[33]=

$$\frac{4\,\epsilon^2\,\mu}{3}$$

Step 3:
Apply formula for evaluating the body forces components:

$$b_r = -R\,(g'[t]\,h[R])^2 - \frac{1}{\rho_0}\left\{\left(\xi'[R]\,\frac{d\mathcal{T}_{11}}{d\epsilon} - h'[R]\,\mathcal{T}_{21}\right)g[t] + \frac{1}{R}\,(\mathcal{T}_{11} - \mathcal{T}_{22})\right\}$$

$$b_\vartheta = R\,g''[t]\,h[R] - \frac{1}{\rho_0}\left\{\left(\xi'[R]\,\frac{d\mathcal{T}_{21}}{d\epsilon} + h'[R]\,\mathcal{T}_{11}\right)g[t] + \frac{1}{R}\,(\mathcal{T}_{12} + \mathcal{T}_{21})\right\}$$

In[34]:=
```
br = -R (g'[t] h[R])^2 -
    1/ρ0 ((ξ'[R] D[𝒯[[1, 1]], ϵ] - h'[R] 𝒯[[2, 1]]) g[t] + 1/R (𝒯[[1, 1]] - 𝒯[[2, 2]])) /.
   ϵ → g[t] ξ[R] // Simplify
```

Out[34]=

$$-\pi^2\,R\,\left(15 - 32\,R + 16\,R^2\right)^4\,\mathrm{Cos}[\pi\,t]^2 + \frac{\left(-64\,(-1+R) + 1024\,(-1+R)^3\right)^2\,R\,\mu\,\mathrm{Sin}[\pi\,t]^2}{\rho 0}$$

In[35]:=
```
bϑ = R g''[t] h[R] -
    1/ρ0 ((ξ'[R] D[𝒯[[2, 1]], ϵ] + h'[R] 𝒯[[1, 1]]) g[t] + 1/R (𝒯[[1, 2]] + 𝒯[[2, 1]])) /.
   ϵ → g[t] ξ[R] // Simplify
```

Out[35]=
$$-\frac{1}{\rho 0}\left(64\left(-45+188\,R-240\,R^2+96\,R^3\right)\mu+\pi^2 R\left(15-32\,R+16\,R^2\right)^2\rho 0\right)\mathrm{Sin}[\pi\,t]$$

Step 4:
body force vector relative to spatial cylindrical basis

In[36]:=
```
bcylindrical = {br, bϑ, 0};
bcylindrical // MatrixForm
```

Out[37]//MatrixForm=
$$\begin{pmatrix} -\pi^2 R\left(15-32\,R+16\,R^2\right)^4 \mathrm{Cos}[\pi\,t]^2+\frac{\left(-64\,(-1+R)+1024\,(-1+R)^3\right)^2 R\,\mu\,\mathrm{Sin}[\pi\,t]^2}{\rho 0} \\ -\frac{\left(64\left(-45+188\,R-240\,R^2+96\,R^3\right)\mu+\pi^2 R\left(15-32\,R+16\,R^2\right)^2\rho 0\right)\mathrm{Sin}[\pi\,t]}{\rho 0} \\ 0 \end{pmatrix}$$

body force vector in Cartesian basis, recognizing that the current angular coordinate is the initial coordinate Θ plus the material rotation angle α:

In[38]:=
```
ϑ = Θ + α
```

Out[38]=
$$\Theta+\left(1-32\,(-1+R)^2+256\,(-1+R)^4\right)\mathrm{Sin}[\pi\,t]$$

In[39]:=
$$\mathrm{dircos}=\begin{pmatrix}\mathrm{Cos}[\vartheta] & \mathrm{Sin}[\vartheta] & 0\\ -\mathrm{Sin}[\vartheta] & \mathrm{Cos}[\vartheta] & 0\\ 0 & 0 & 1\end{pmatrix};$$

In[40]:=
```
bCartesian = Transpose[dircos].bcylindrical // Simplify;
bCartesian // MatrixForm
```

Out[41]//MatrixForm=
$$\left(\begin{array}{l} \mathrm{Cos}\left[\Theta+\left(15-32\,R+16\,R^2\right)^2\mathrm{Sin}[\pi\,t]\right]\left(-\pi^2 R\left(15-32\,R+16\,R^2\right)^4\mathrm{Cos}[\pi\,t]^2+\frac{\left(-64\,(-1+R)+1024\,(-1+R)^3\right)^2 R,}{\rho 0}\right. \\ -\frac{\left(64\left(-45+188\,R-240\,R^2+96\,R^3\right)\mu+\pi^2 R\left(15-32\,R+16\,R^2\right)^2\rho 0\right)\mathrm{Cos}\left[\Theta+\left(15-32\,R+16\,R^2\right)^2\mathrm{Sin}[\pi\,t]\right]\mathrm{Sin}[\pi\,t]}{\rho 0}+\left(-\pi^2 R\left(15-32\,R+16\,R^2\right.\right. \\ 0 \end{array}\right.$$

In[42]:=
```
bcylindrical /. {R → 3 / 4}
```

Out[42]=
$$\left\{0, -\frac{96\,\mu\,\mathrm{Sin}[\pi\,t]}{\rho 0}, 0\right\}$$

Verification that the spatial eqs of motion are satisfied by this MMS

Define a notationally appealing macro that lets us access vector components as subscripts instead of the ugly two-bracket [[...]] syntax that is the default in Mathematica:

In[43]:=
```
u_j__ := u[[j]]
```

Let's remind ourselves of the Cauchy stress tensor in the case of simple shear without superimposed rotation:

In[44]:=
```
σ // MatrixForm
```

Out[44]//MatrixForm=
$$\begin{pmatrix} 0 & 2\,\epsilon\,\mu & 0 \\ 2\,\epsilon\,\mu & 4\,\epsilon^2\,\mu & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Based on our analysis of this MMS, these Cartesian components for the simple shear problem without superimposed rotation are the same as the spatial cylindrical components in the actual MMS problem.

Define a substitution pattern that replaces the generic strain with its actual value as a function of position and time. Also include a substitution that specifies the current radial coordinate to be equal to the initial one:

In[45]:=
```
sub = {ϵ → g[t] ξ[R], R → r}
```

Out[45]=
$$\left\{\epsilon \to \frac{1}{2}\left(-64\,(-1+R)+1024\,(-1+R)^3\right) R\,\mathrm{Sin}[\pi\,t],\ R \to r\right\}$$

Obtain the cylindrical components of Cauchy stress as functions of position and time:

In[46]:=
```
σrr = σ_{1,1}
```

Out[46]=
0

In[47]:=
```
σrϑ = σ_{1,2} //. sub
```

Out[47]=
$$\left(-64\,(-1+r)+1024\,(-1+r)^3\right) r\,\mu\,\mathrm{Sin}[\pi\,t]$$

In[48]:=
```
σϑr = σ2,1 //. sub
```
Out[48]= $\left(-64\,(-1+r)+1024\,(-1+r)^3\right)\, r\,\mu\, \mathrm{Sin}[\pi\, t]$

In[49]:=
```
σϑϑ = σ2,2 //. sub
```
Out[49]= $\left(-64\,(-1+r)+1024\,(-1+r)^3\right)^2 r^2\,\mu\, \mathrm{Sin}[\pi\, t]^2$

Now compute the radial and tangential components of the stress divergence:

In[50]:=
$$\texttt{divSigr} = \frac{1}{r}\,\texttt{D[r σrr, r]} - \frac{σϑϑ}{r}\ \texttt{// Simplify}$$
$$\texttt{divSigϑ} = \frac{1}{r^2}\,\texttt{D}\left[r^2\ σϑr,\ r\right] + \frac{1}{r}\,(σϑr - σrϑ)\ \texttt{// Simplify}$$

Out[50]= $-\left(-64\,(-1+r)+1024\,(-1+r)^3\right)^2 r\,\mu\, \mathrm{Sin}[\pi\, t]^2$

Out[51]= $64\,\left(-45+188\, r-240\, r^2+96\, r^3\right)\,\mu\, \mathrm{Sin}[\pi\, t]$

Recall that that $\underset{\sim}{x} = r\,\underset{\sim}{\mathbf{e}}_r = R\,\underset{\sim}{\mathbf{e}}_r$

$\underset{\sim}{v} = R\,\omega\,\underset{\sim}{\mathbf{e}}_\vartheta$,

$\underset{\sim}{a} = R\,\dot{\omega}\,\underset{\sim}{\mathbf{e}}_\vartheta - R\,\omega^2\,\underset{\sim}{\mathbf{e}}_r$

where

ω=g'[t] h[R]

and

$\dot{\omega}$=g''[t] h[R]

In[52]:=
```
ω = g'[t] h[R] //. sub
ωdot = g''[t] h[R] //. sub
```
Out[52]= $\pi\,\left(1-32\,(-1+r)^2+256\,(-1+r)^4\right)\, \mathrm{Cos}[\pi\, t]$

Out[53]= $-\pi^2\,\left(1-32\,(-1+r)^2+256\,(-1+r)^4\right)\, \mathrm{Sin}[\pi\, t]$

Noting that the current mass density equals initial mass density divided by the Jacobian, the components of density times acceleration as well as density times body force are given by

In[54]:=
$$\rho ar = \frac{\rho 0}{J}\left(-r\,\omega^2\right)$$
$$\rho br = \frac{\rho 0}{J}\,\text{bcylindrical[[1]] //. sub}$$

Out[54]=
$$-\pi^2\left(1-32\,(-1+r)^2+256\,(-1+r)^4\right)^2 r\,\rho 0\,\text{Cos}[\pi\,t]^2$$

Out[55]=
$$\rho 0\left(-\pi^2 r\left(15-32\,r+16\,r^2\right)^4\text{Cos}[\pi\,t]^2+\frac{\left(-64\,(-1+r)+1024\,(-1+r)^3\right)^2 r\,\mu\,\text{Sin}[\pi\,t]^2}{\rho 0}\right)$$

In[56]:=
$$\rho a\vartheta = \frac{\rho 0}{J}\left(r\,\omega\text{dot}\right)$$
$$\rho b\vartheta = \frac{\rho 0}{J}\,\text{bcylindrical[[2]] //. sub}$$

Out[56]=
$$-\pi^2\left(1-32\,(-1+r)^2+256\,(-1+r)^4\right) r\,\rho 0\,\text{Sin}[\pi\,t]$$

Out[57]=
$$-\left(64\left(-45+188\,r-240\,r^2+96\,r^3\right)\mu+\pi^2 r\left(15-32\,r+16\,r^2\right)^2\rho 0\right)\text{Sin}[\pi\,t]$$

The following now confirms that the derived body force does indeed satisfy the equations of motion:

In[58]:=
```
Simplify[divSigr + ρbr - ρar]
Simplify[divSigϑ + ρbϑ - ρaϑ]
```

Out[58]= 0

Out[59]= 0

Visualization

In[60]:=
```
SetDirectory[NotebookDirectory[]]
```

Out[60]= C:\ccccccccccToss\MPM_MATLAB_Code_Sadeghirad2018\BrannonMathematicaMMSfiles

Multiplier on the time increment between frames in movies (set =1 for high fidelity, set =20 for just a few frames for testing).

In[61]:=
```
fac = 1
```

Out[61]= 1

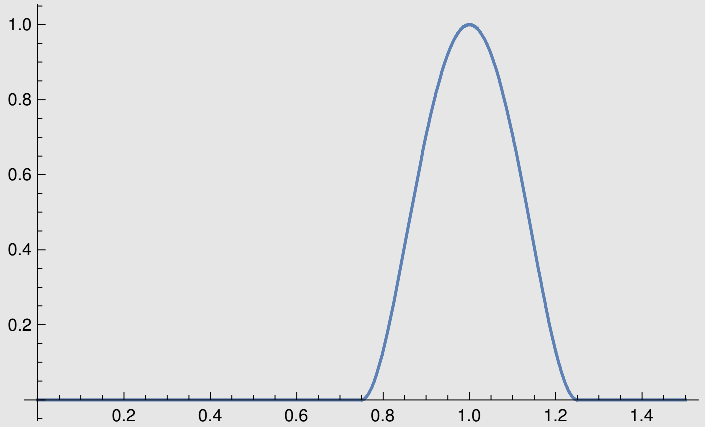
In[62]:=
```
Rbar =.;
η[Rbar_] := 1 - 2 Rbar^2 + Rbar^4
```

In[64]:=
```
η'[Rbar]
```

Out[64]= $-4\,\text{Rbar} + 4\,\text{Rbar}^3$

In[65]:=
```
rref = 1;
a = rref (1 - 1 / 4);
b = rref (1 + 1 / 4);
c = 1/2 (b + a);
h[R_] := If[a < R < b, Evaluate[η[(R - c)/(b - c)]], 0];
Plot[h[R], {R, 0, 1.2 b}, PlotRange → All]
h[R]
```

Out[70]=



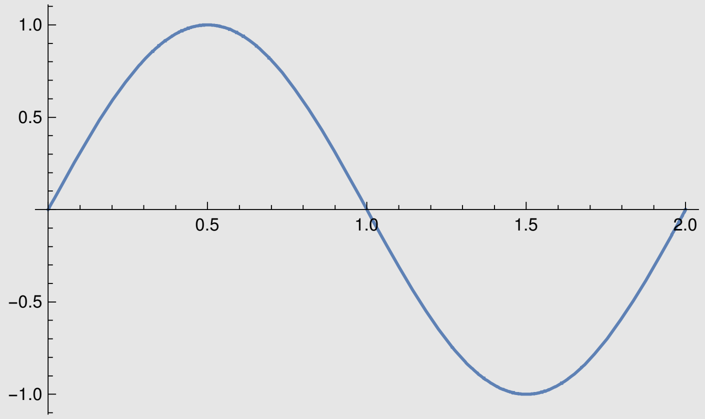
Out[71]= $\text{If}\left[\frac{3}{4} < R < \frac{5}{4},\ 1 - 32\,(-1 + R)^2 + 256\,(-1 + R)^4,\ 0\right]$

In[72]:=
```
hprime[R_] := If[-a < R < b, Evaluate[η'[(R - c)/(b - c)]], 0];
hprime[R]
```

Out[73]= $\text{If}\left[-\frac{3}{4} < R < \frac{5}{4},\ -16\,(-1 + R) + 256\,(-1 + R)^3,\ 0\right]$

In[74]:=
```
tref = 1;
g[t_] := Sin[π t/tref]
Plot[g[t], {t, 0, 2 tref}]
```

Out[76]=



Define a function to evaluate the continuum mapping,

$\underset{\sim}{x} = \underset{\approx}{Q} . \underset{\sim}{X}$

In[77]:=
```
ppts = 30;
```

In[78]:=
```
q[X1_, X2_, t_] := Evaluate[
  (({{Cos[α], -Sin[α]}, {Sin[α], Cos[α]}}) /. {α → g[t] h[R]}) /. {R → Sqrt[{X1, X2}.{X1, X2}]}
 ]
```

In[79]:=
```
map[X1_, X2_, t_] := q[X1, X2, t].{X1, X2}
```

In[80]:=
```
Xval = 1.2 b;
dt = fac * .02 tref;
frames = Table[
   ParametricPlot[map[X1, X2, tval], {X1, -Xval, Xval}, {X2, -Xval, Xval},
    MeshStyle → {{Thick, Blue}}, PlotPoints → ppts,
    Axes → False, Frame → False]
   , {tval, 0, 2 tref - dt, dt}];
Export["GeneralizedVortexRing.gif", frames]
```

Out[83]=
```
GeneralizedVortexRing.gif
```

The following exports single frames of the animation.

To make an animated gif from the exported frames, execute the following on a Linux Workstation:

convert -verbose -delay 30 -loop 0 -density 100 gvs*.gif GeneralizedVortexSquareSmall.gif

This notebook currently does not export the images of the individual frames. If you want them, uncomment the “do loop” below.

In[84]:=

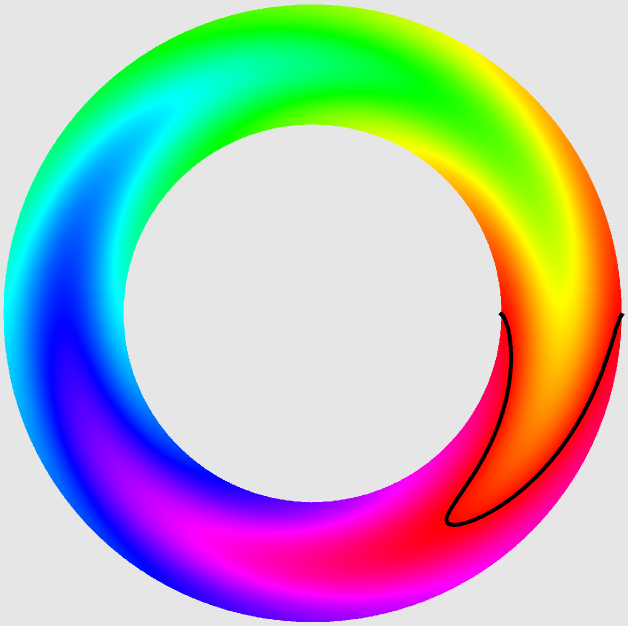
```
nfrms = 40;
inc = Max[2, Round[Length[frames]/nfrms]];
(*Do[Export["gvs"<>IntegerString[i,10,3]<>".gif",frames[[i]]],
 {i,1,Length[frames],inc}]*)
```

Test frame to check if picture is as desired:

In[86]:=

```
tval = 1.5 tref;
pic = ParametricPlot[map[R Cos[θ], R Sin[θ], tval], {R, a, b}, {θ, 0, 2 π},
  MeshStyle → {None, {Thick, Black}}, PlotPoints → ppts, BoundaryStyle → {Thick, Black},
  ColorFunction → Function[{x, y, r, t}, Hue[t / (2 Pi)]], ColorFunctionScaling → False,
   Axes → False, Frame → False]
(*Export["GenVort.png",pic]*)
```

Out[87]=



WARNING, BE PREPARED TO WAIT...
The following command takes about 15 minutes to finish.

In[88]:=
```
dt = fac * .02 tref;
frames = Table[
   ParametricPlot[map[R Cos[θ], R Sin[θ], tval], {R, a, b}, {θ, 0, 2 π},
    MeshStyle → {None, {Thick, Black}}, PlotPoints → ppts, BoundaryStyle → {Thick, Black},
    ColorFunction → Function[{x, y, r, t}, Hue[t / (2 Pi)]],
    ColorFunctionScaling → False,
     Axes → False, Frame → False]
   , {tval, 0, 2 tref - dt, dt}];
Export["GeneralizedVortexRing.gif", frames]
```

Out[90]=
```
GeneralizedVortexRing.gif
```

The following exports single frames of the animation.
To make an animated gif from the exported frames, execute the following on a Linux Workstation:

convert -verbose -delay 30 -loop 0 -density 100 gvr*.gif GeneralizedVortexRingSmall.gif

This command is currently commented out because we don’t currently need single frames.

In[91]:=
```
(*nfrms=40;
inc=Max[2,Round[Length[frames]/nfrms]];
Do[Export["gvr"<>IntegerString[i,10,3]<>".gif",frames[[i]]],
 {i,1,Length[frames],inc}]*)
```

Here is a hint of some things that can be done to alter the appearance

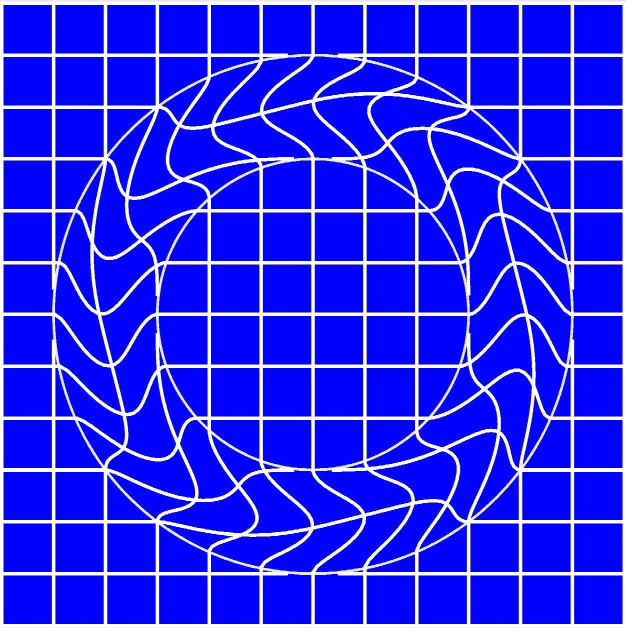
WARNING, BE PREPARED TO WAIT...
The following command takes about five minutes to finish.

In[92]:=

```
mesh1 = Table[{X, {Thickness[0.005], White, Opacity[1]}}, {X, -Xval, Xval, Xval / 6}];
mesh2 = Table[{X, {Thickness[0.005], White, Opacity[1]}}, {X, -Xval, Xval, Xval / 6}];

Xval = 1.2 b;
dt = fac * 0.02 tref;
frames = Table[
   ParametricPlot[map[X1, X2, tval], {X1, -Xval, Xval}, {X2, -Xval, Xval},
    PlotStyle → Directive[Opacity[0.999], Blue],
    Mesh → {mesh1, mesh2}, PlotPoints → ppts,
    Axes → False, Frame → False]
   , {tval, 0, 2 tref - dt, dt}];
frames[[5]]
Export["GeneralizedVortexSquareBlue.gif", frames]
```

Out[97]=



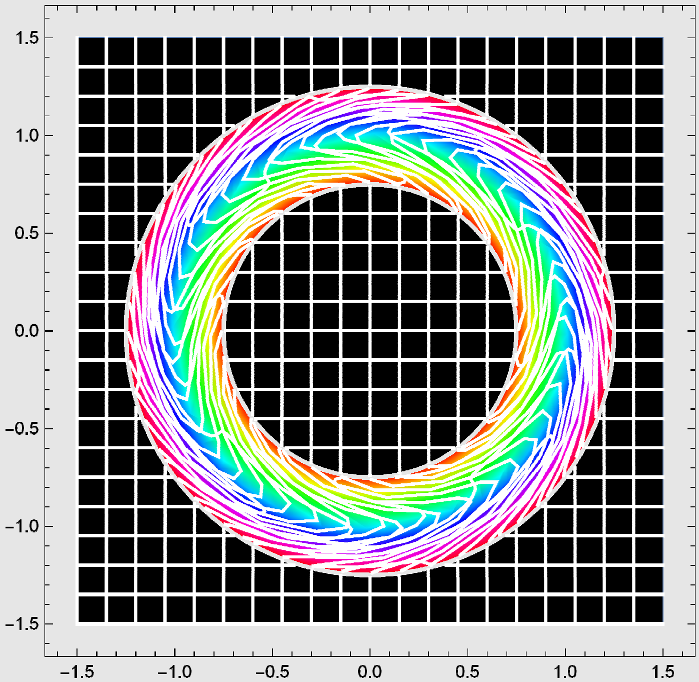
Out[98]=

```
GeneralizedVortexSquareBlue.gif
```

In[99]:=

```
mesh1 = Table[{X, {Thickness[0.005], White, Opacity[1]}}, {X, -Xval, Xval, Xval / 10}];
mesh2 = Table[{X, {Thickness[0.005], White, Opacity[1]}}, {X, -Xval, Xval, Xval / 10}];
ParametricPlot[map[X1, X2, .5], {X1, -Xval, Xval}, {X2, -Xval, Xval},
 MeshFunctions → {#3 &, #4 &},
 ColorFunction → Function[{X1, X2, x1, x2}, rdum = (Sqrt[X1^2 + X2^2] - a)/(b - a);
   If[0 < rdum < 1, Hue[rdum], Black]], ColorFunctionScaling → False,
 Mesh → {mesh1, mesh2},
 Axes → False]
```

Out[101]=

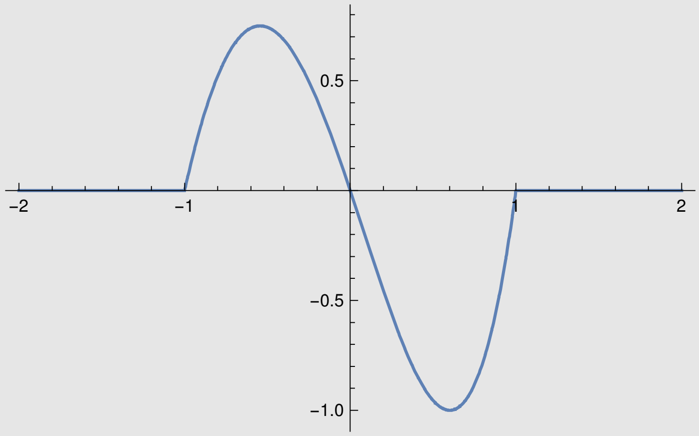


In[102]:=

```
strain[Rbar_] := 1/2 (Rbar + 1/(b / c - 1)) η'[Rbar];
peakstrain = Abs[N[Minimize[{strain[Rbar], -1 < Rbar < 1}, Rbar]][[1]]]
intensity[Rbar_] := Evaluate[If[-1 < Rbar < 1, strain[Rbar]/peakstrain, 0]]
Plot[intensity[Rbar], {Rbar, -2, 2}]
```

Out[103]= 3.53282

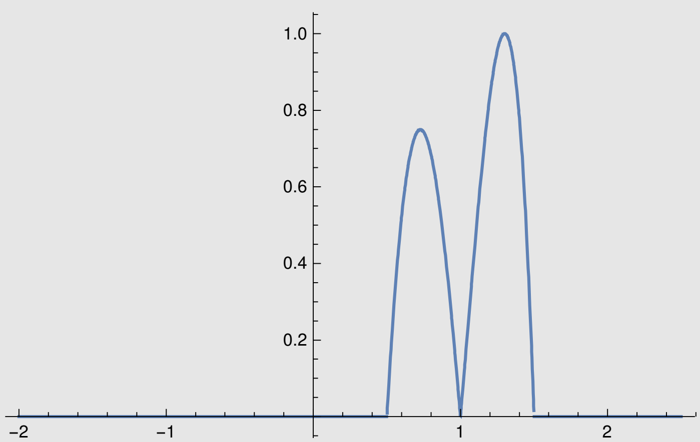
Out[105]=



In[106]:=

```
Plot[Abs[intensity[(x - c)/(b - a)]], {x, -c / (b - a), 2 b}]
```

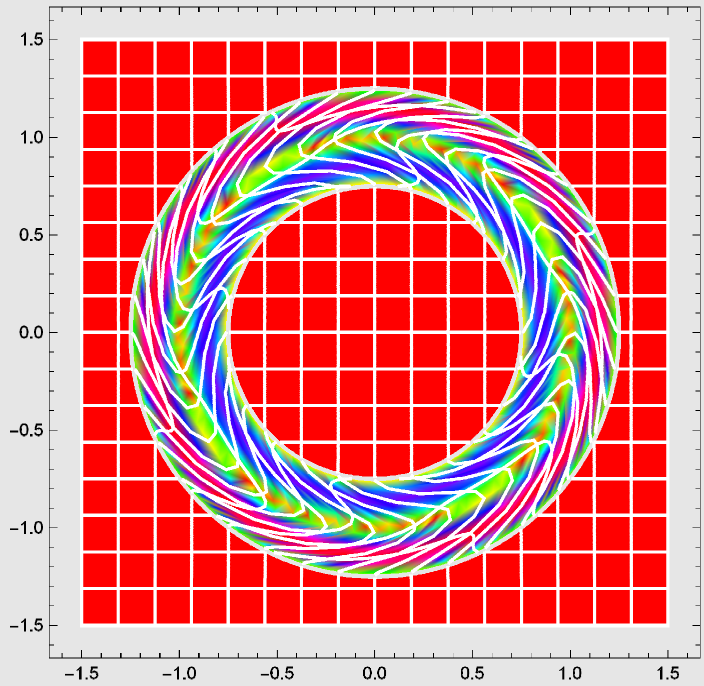
Out[106]=



In[107]:=

```
mesh1 = Table[{X, {Thickness[0.005], White, Opacity[1]}}, {X, -Xval, Xval, Xval / 8}];
mesh2 = Table[{X, {Thickness[0.005], White, Opacity[1]}}, {X, -Xval, Xval, Xval / 8}];
ParametricPlot[map[X1, X2, .2], {X1, -Xval, Xval}, {X2, -Xval, Xval},
 MeshFunctions → {#3 &, #4 &},
 ColorFunction → Function[{X1, X2, x1, x2}, rdum = (Sqrt[X1^2 + X2^2] - c)/((b-a)/2);
   Hue[Abs[intensity[rdum]]]], ColorFunctionScaling → False,
 Mesh → {mesh1, mesh2},
 Axes → False]
```

Out[109]=



Enjoy playing around with this stuff!

REFERENCES:
[1] Kamojjala, K.C., R. Brannon, A. Sadeghirad, and J. Guilkey (2015) Verification tests in solid mechanics, Engineering with
Computers, v. 2, pp. 193-213
[2] Sadeghirad, A., R. M. Brannon, and J. Burghardt (2011) A convected particle domain interpolation technique to extend
applicability of the material point method for problems involving massive deformations, Int. J. Numer. Meth. in Engr.,
86(12), pp. 1435-1456.