Verification Tests

We've already seen that elastic materials are required to satisfy certain general principles. One important way to verify that a computer implementation of a material law is correct (that is to say, that you actually coded what you think you coded) is to test it according to these principles. Here we consider three such verification tests.

Test 1: Consistency

This test verifies the variational consistency of your energy, stress, and modulus implementation. Recall that the 1st P-K stress and Lagrangian tangent moduli are defined as

$$P_{iJ} = \frac{\partial w(\mathbf{F})}{\partial F_{iJ}} \tag{1}$$

$$C_{iJkL} = \frac{\partial P_{iJ}(\mathbf{F})}{\partial F_{kL}} = \frac{\partial^2 w(\mathbf{F})}{\partial F_{iJ}\partial F_{kL}}. \tag{2}$$

These definitions hold for *every valid* deformation gradient $\mathbf{F}$. Recall that $\mathbf{F}$ is in general an *arbitrary* second-order tensor, with the caveat that in order for the deformation to be physical, it must not exhibit any inversion of material, that is

$$J = \det \mathbf{F} > 0.$$

The set of such tensors is denoted

$$\mathrm{GL}^+(3,\mathbb{R}) = \{\mathbf{F} \in \mathrm{L}(\mathbb{R}^3,\mathbb{R}^3), \det \mathbf{F} > 0\}.$$

A typical computer implementation of a material law will involve one or more functions that calculate the values of w, P_{iJ}, and C_{iJkL} for input values of F_{iJ}. We can test that the calculated values are consistent with eons. (1) and (2) by also computing derivatives of w and P_{iJ} using *numerical differentiation*. The idea is to use divided differences to approximate the derivatives, and then compare those approximations to the values calculated from the analytical expressions (1) and (2).

Numerical Differentiation Formulae

Given: $f : \mathbb{R} \to \mathbb{R}, a \in \mathbb{R}$.

2-point formula:

$$f'(a) \approx \frac{f(a+h) - f(a)}{h} \equiv f'_h(a)$$

$$\text{Error} = f'_h(a) - f'(a) = -\frac{h}{2} f''(\eta), \quad \eta \in [a, a+h]$$

3-point formula:

$$f'(a) \approx \frac{f(a+h) - f(a-h)}{2h} \approx f'_h(a)$$

$$\text{Error} = f'_h(a) - f'(a) = -\frac{h^2}{6} f''(\eta), \quad \eta \in [a-h, a+h]$$

Implementation

(i) Generate random $\mathbf{F} \in \mathrm{GL}^+(3, \mathbb{R})$.

(ii) Compute $w(\mathbf{F})$, $P_{iJ}(\mathbf{F})$, and $C_{iJkL}(\mathbf{F})$.

(iii) Compute *approximations* $(P_h)_{iJ}(\mathbf{F})$, and $(C_h)_{iJkL}(\mathbf{F})$ by numerical differentiation and compare, e.g.,

$$(P_h)_{iJ} = \frac{w(F_{iJ} + h) - w(F_{iJ} - h)}{2h}$$

(Use implementation of material model to compute $w(F_{iJ} \pm h)$.)

Test:

$$\max_{i,J} |(P_h)_{iJ} - P_{iJ}| < \mathrm{TOL}|\mathbf{P}|?$$

$$(h\ 10^{-6} \quad \Rightarrow \quad \mathrm{TOL}\ h^2 10^{-12})$$

$$(C_h)_{iJkL} = \frac{P_{iJ}(F_{kL} + h) - P_{iJ}(F_{kL} - h)}{2h}$$

Test:

$$\max_{i,J,k,L} |(C_h)_{iJkL} - C_{iJkL}| < \mathrm{TOL}|\mathbf{C}|?$$

YES $\rightarrow$ Continue.
NO $\rightarrow$ Issue Warning; continue.

Test 2: M.F.I.

The point here is to verify the conditions for material frame indifference:

$$\left.\begin{aligned} w(\mathbf{QF}) &= w(\mathbf{F}) \\ P_{iJ}(\mathbf{QF}) &= Q_{ik} P_{kJ}(\mathbf{F}) \\ C_{iJkL}(\mathbf{QF}) &= Q_{ij} Q_{k\ell} C_{jJ\ell L}(\mathbf{F}) \end{aligned}\right\} \forall \mathbf{F} \in \mathrm{GL}^+(3, \mathbb{R}), \quad \forall \mathbf{Q} \in \mathrm{SO}(3)$$

Implementation

(i) Generate random $\mathbf{F} \in \mathrm{GL}^+(3, \mathbb{R})$.

(ii) Compute $w(\mathbf{F})$, $P_{iJ}(\mathbf{F})$, and $C_{iJkL}(\mathbf{F})$ directly from implementation of the material model.

(iii) Generate random $\mathbf{Q} \in \mathrm{SO}(3)$.

(iv) Compute $w(\mathbf{QF})$, $P_{iJ}(\mathbf{QF})$, and $C_{iJkL}(\mathbf{QF})$ (again using material model implementation).

(v) Test:

$$|w(\mathbf{QF}) - w(\mathbf{F})| < \mathrm{TOL}|w(\mathbf{F})|?$$

$$\max_{i,J} |P_{iJ}(\mathbf{QF}) - Q_{ik} P_{kJ}(\mathbf{F})| < \mathrm{TOL}|\mathbf{P}(\mathbf{F})|?$$

$$\max_{i,J,k,L} |C_{iJkL}(\mathbf{QF}) - Q_{ij} Q_{k\ell} C_{jJ\ell L}(\mathbf{F})| < \mathrm{TOL}|\mathbf{C}(\mathbf{F})|?$$

TRUE $\rightarrow$ Exit.
FALSE $\rightarrow$ Issue warning; exit

Test 3: Symmetry

Denote the group of orthogonal 2-tensors defining *symmetris* of a material with energy density w as

$$\mathrm{S}_w \equiv \{\mathbf{Q} \in \mathrm{O}(3) \quad | \quad w(\mathbf{FQ}) = w(\mathbf{F}), \quad \forall \mathbf{F} \in \mathrm{GL}^+(3, \mathbb{R})\}.$$

By definition then,

$$w(\mathbf{FQ}) = w(\mathbf{F}), \quad \forall \mathbf{F} \in \mathrm{GL}^+(3, \mathbb{R}), \forall \mathbf{Q} \in \mathrm{S}_w.$$

A straightforward calculation shows that this also requires the following for the stress and the moduli:

$$P_{iJ}(\mathbf{FQ}) = Q_{KJ} P_{iK}(\mathbf{F})$$

$$C_{iJkL}(\mathbf{FQ}) = Q_{MJ} Q_{NL} C_{iMkN}(\mathbf{F})$$

We can verify a computer implementation then by choosing random symmetries of a material, and testing to see that they leave the energy invariant, and cause the stress and moduli to transform appropriately.

Implementation

(i) Generate random $\mathbf{F} \in \mathrm{GL}^+(3, \mathbb{R})$.

(ii) Compute $w(\mathbf{F})$, $P_{iJ}(\mathbf{F})$, and $C_{iJkL}(\mathbf{F})$ directly from implementation of the material model.

(iii) Generate random $\mathbf{Q} \in \mathrm{S}_w$.

(iv) Compute $w(\mathbf{FQ})$, $P_{iJ}(\mathbf{FQ})$, and $C_{iJkL}(\mathbf{FQ})$ (again using material model implementation).

(v) Test:

$$|w(\mathbf{FQ}) - w(\mathbf{F})| < \text{TOL}|w(\mathbf{F})|?$$

$$\max_{i,J} |P_{iJ}(\mathbf{FQ}) - Q_{KJ} P_{iK}(\mathbf{F})| < \text{TOL}|\mathbf{P}(\mathbf{F})|?$$

$$\max_{i,J,k,L} |C_{iJkL}(\mathbf{FQ}) - Q_{MJ} Q_{NL} C_{iMkN}(\mathbf{F})| < \text{TOL}|\mathbf{C}(\mathbf{F})|?$$

TRUE $\rightarrow$ Exit.
FALSE $\rightarrow$ Issue warning; exit

A Note on Random Rotations

Choosing random symmetries and rotations can involve some subtlety. For instance, in the case of an isotropic material or in the M.F.I. test, we would need to consider how to draw a random rotation from $\mathrm{SO}(3)$. If we want the distribution of our random rotations to be uniform (i.e., all rotations are equally likely to be chosen) then it's even trickier. One strategy which doesn't give uniformly distributed rotations, but is still pretty good, is to choose a random vector $\boldsymbol{v} \in \mathbb{R}^3$, then normalize it $\boldsymbol{n} = \boldsymbol{v}/|\boldsymbol{v}|$, and then perform a 2-D rotation by random angle θ uniformly distributed in $[0, \pi]$ about the axis $\boldsymbol{n}$. The resulting 3-D rotation can be computed by Rodrigues' formula (which we will derive later),

$$\mathbf{Q} = \mathbf{I} + \sin\theta \hat{\boldsymbol{n}} + (1 - \cos\theta)(\boldsymbol{n} \otimes \boldsymbol{n} - \mathbf{I}),$$

where $\hat{\boldsymbol{n}}$ is the skew symmetrix tensor referred to as the *dual* of $\boldsymbol{n}$, having components

$$[\hat{n}_{ij}] = \begin{bmatrix} 0 & -n_3 & n_2 \\ n_3 & 0 & -n_1 \\ -n_2 & n_1 & 0 \end{bmatrix}.$$