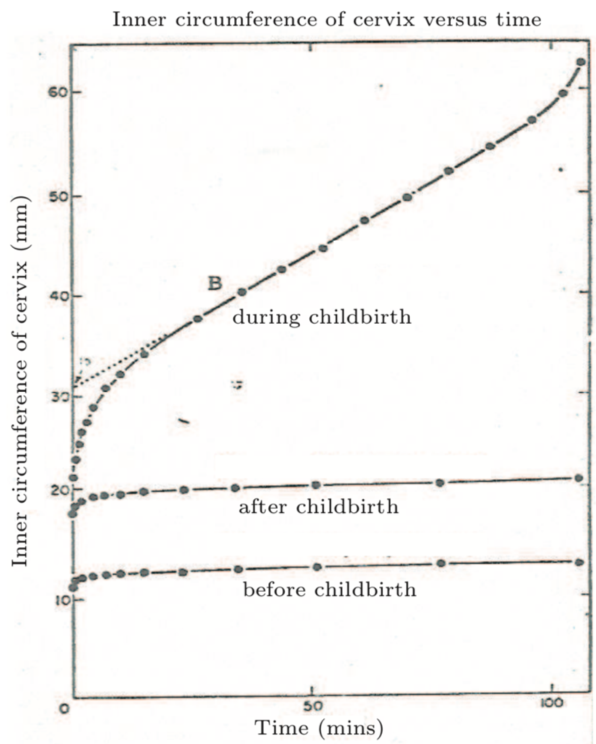
ENGN 2340 Final Project: Viscoelasticity in soft biological tissues

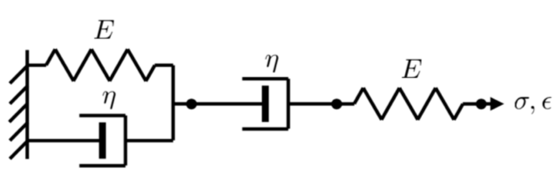
Xuliang Qian

Many soft biological tissues have viscoelastic effects.
In 1959, Harkness and Harkness measured creep response of rat cervical tissue subjected to a constant dilating force experimentally. Below is a modified figure borrowed from ENGN 2220.



We can see that the properties of rat cervical tissue experience a dramatic transformation during pregnancy, in particular, the creep goes almost linear with time during childbirth. This is in accordance to its function – to help the passage of the child.

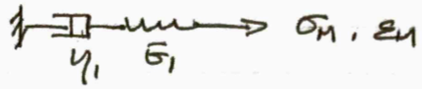
In ENGN 2220, we modeled this viscoelastic behavior using the following rheological model (the Burgers model), which consists of a Maxwell element in series with a Kelvin-Voigt element:



In fact, the Burgers model could also capture the creep for a viscoelastic liquid. For simplicity, we just took the stiffness of both springs to be E and the viscosity of both dashpots to be η.

The derivation for 1D governing equation is:

1) For Maxwell element:

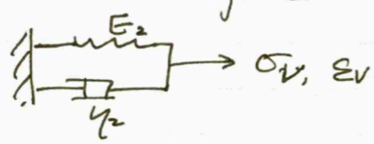


We have: $\sigma_M = E_1 \varepsilon_M^e = \eta_1 \dot{\varepsilon}_M^v$

$\varepsilon_M = \varepsilon_M^e + \varepsilon_M^v$

$$\Rightarrow \dot{\varepsilon}_M = \dot{\varepsilon}_M^e + \dot{\varepsilon}_M^v = \frac{\dot{\sigma}_M}{E_1} + \frac{\sigma_M}{\eta_1}$$

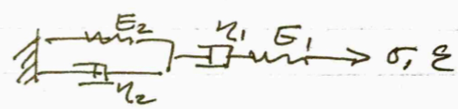
2) for Kelvin-Voigt element:



We have: $\sigma_V = E_2 \varepsilon_V + \eta_2 \dot{\varepsilon}_V$

3) For Burgers model: Maxwell in series with Kelvin-Voigt:

$$\begin{cases} \sigma = \sigma_M = \sigma_V \\ \varepsilon = \varepsilon_M + \varepsilon_V \end{cases}$$



combine: $\dot{\sigma} = E_2 (\dot{\varepsilon} - \dot{\varepsilon}_M) + \eta_2 (\ddot{\varepsilon} - \ddot{\varepsilon}_M)$

$$= E_2 \left(\dot{\varepsilon} - \frac{\dot{\sigma}}{E_1} - \frac{\sigma}{\eta_1} \right) + \eta_2 \left(\ddot{\varepsilon} - \frac{\ddot{\sigma}}{E_1} - \frac{\dot{\sigma}}{\eta_1} \right).$$

$$\Rightarrow \sigma + \left(\frac{\eta_1}{E_2} + \frac{\eta_1}{E_1} + \frac{\eta_2}{E_2} \right) \dot{\sigma} + \frac{\eta_1 \eta_2}{E_1 E_2} \ddot{\sigma} = \eta_1 \dot{\varepsilon} + \frac{\eta_1 \eta_2}{E_2} \ddot{\varepsilon}$$

or:

$$\left[1 + \left(\frac{\eta_1}{E_2} + \frac{\eta_1}{E_1} + \frac{\eta_2}{E_2} \right) \frac{d}{dt} + \frac{\eta_1 \eta_2}{E_1 E_2} \frac{d^2}{dt^2} \right] \sigma(t) = \left(\eta_1 \frac{d}{dt} + \frac{\eta_1 \eta_2}{E_2} \frac{d^2}{dt^2} \right) \varepsilon(t)$$

Governing Equation for 1D.

if assume $E_1 = E_2$, $\eta_1 = \eta_2$

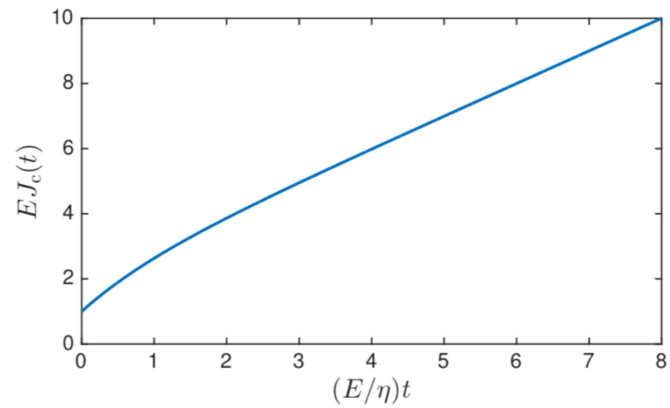
$$\Rightarrow \left[1 + \frac{3\eta}{E} \frac{d}{dt} + \left(\frac{\eta}{E} \right)^2 \frac{d^2}{dt^2} \right] \sigma(t) = \left(\eta \frac{d}{dt} + \frac{\eta^2}{E} \frac{d^2}{dt^2} \right) \varepsilon(t).$$

For simplified 1D equation (E1=E2, η1=η2), we could show that the strain and creep for a stress step input is:

$$\epsilon(t) = \frac{1}{E}\left[1-\exp\left(-\frac{E}{\eta}t\right)\right]\sigma_0 + \left(\frac{1}{E}+\frac{t}{\eta}\right)\sigma_0$$

$$J_c(t) = \frac{\epsilon(t)}{\sigma_0} = \frac{1}{E}\left[1-\exp\left(-\frac{E}{\eta}t\right)\right] + \left(\frac{1}{E}+\frac{t}{\eta}\right).$$

By plotting E Jc(t) versus (E/η)t, we could see the behavior describe earlier:



Next step is to generalize 1D to 3D. The derivation for 3D governing equation is shown below:

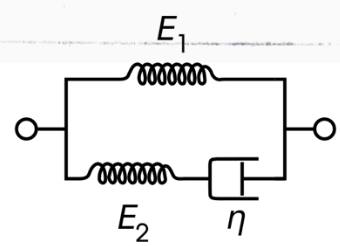
For 3D: $E \rightarrow C^{E}_{ijkl}$ $\quad \eta \rightarrow C^{\nu}_{ijkl}$.

$$\left[\delta_{pq} + 3C^{\nu}_{ijkl}(C^{E})^{-1}_{klpq}\frac{d}{dt} + C^{\nu}_{ijmn}C^{\nu}_{mnrs}(C^{E})^{-1}_{rsxy}(C^{E})^{-1}_{xypq}\frac{d^2}{dt^2}\right]\sigma_{pq}(t)$$

$$= \left[C^{\nu}_{ijpq}\frac{d}{dt} + C^{\nu}_{ijmn}C^{\nu}_{mnrs}(C^{E})^{-1}_{rspq}\frac{d^2}{dt^2}\right]\varepsilon_{pq}(t).$$

Governing Equation for 3D

However, it could be rather hard to implement.



So I switch to a simpler but well-known model, the standard linear solid (SLS) model, see above, and successfully implement the SLS model in ABAQUS UMAT.

The pages below show the derivation of the theoretical side of FEM implementation.

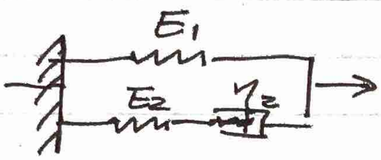
ENGN 2340

Final Project: Theoretical Part:

by Xuliang Qian

We are studying FEM implement of SLS model. (instead of Burgers model) for simplicity.

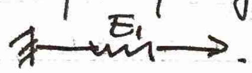
From undergraduate textbook, the SLS model is defined as:



E_1, E_2 : springs with stiffness E_1 & E_2

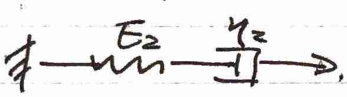
η_2 : dashpot with viscosity η_2

① Simple spring element:



$\sigma_{spring} = E_1 \varepsilon_{spring}$

② Maxwell element:



$$\dot{\varepsilon}_{maxwell} = \frac{\dot{\sigma}_{Maxwell}}{E_2} + \frac{\sigma_{Maxwell}}{\eta_2}$$

Combine ① & ②
$$\begin{cases} \sigma = \sigma_{spring} + \sigma_{maxwell} \\ \varepsilon = \varepsilon_{spring} = \varepsilon_{Maxwell} \end{cases}$$

$$\Rightarrow \dot{\varepsilon} = \frac{\dot{\sigma} - \dot{\sigma}_{spring}}{E_2} + \frac{\sigma - \sigma_{spring}}{\eta_2}$$

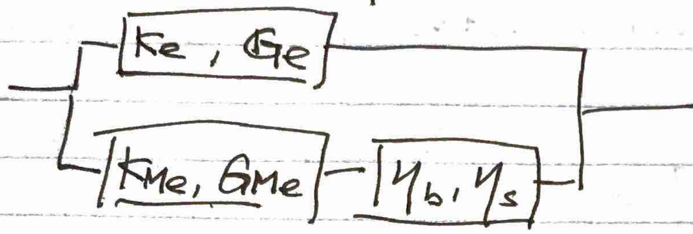
$$\Rightarrow \dot{\varepsilon} = \left(\frac{\dot{\sigma}}{E_2} - \frac{E_1}{E_2}\dot{\varepsilon}\right) + \left(\frac{\sigma}{\eta_2} - \frac{E_1}{\eta_2}\varepsilon\right)$$

where "$\cdot$" stand for d/dt.

$$\Rightarrow \left(1 + \frac{E_1}{E_2}\right)\dot{\varepsilon} + \frac{E_1}{\eta_2}\varepsilon = \frac{\dot{\sigma}}{E_2} + \frac{\sigma}{\eta_2}. \quad \text{(For 1D)}$$

LHS have strain terms,
RHS have stress terms,
they are separated, which is always good!

Next, is to try something dangerous (but fun):
To generalized this 1D governing equation into 3D!
For 3D, we redefine the ~~SLS~~ SLS model as follows:



In short, what I did here is:

$E_1 \rightarrow K_e, G_e$

$E_2 \rightarrow K_{Me}, G_{Me}$

$\eta_2 \rightarrow \eta_b, \eta_s.$

where: K_e, K_{Me} are bulk modulus
G_e, G_{Me} are shear modulus.
η_b, η_s are volume viscosity and shear viscosity respectively.

Rewrite our governing equation in 3D form.
First, look at LHS.

$$LHS = \left(1+\frac{G_e}{G_{Me}}\right)\left[\dot{\varepsilon}_{ij} - \frac{1}{3}\delta_{ij}\,tr(\dot{\underline{\varepsilon}})\right] + \frac{1}{3}\left(1+\frac{K_e}{K_{Me}}\right)\delta_{ij}\,tr(\dot{\underline{\varepsilon}})$$
$$+\frac{G_e}{\eta_s}\left[\varepsilon_{ij} - \frac{1}{3}\delta_{ij}\,tr(\underline{\varepsilon})\right] + \frac{K_e}{\eta_s}\times\frac{1}{3}\delta_{ij}\,tr(\underline{\varepsilon}).$$

Here I intentionally separate the volumetric strain and shear strain. This would make latter minupalation somewhat simpler.

Then we take a look at RHS, it has the same dimension as $\left(\frac{\text{strain}}{\text{time}}\right)$, but contains all the stress terms.

Recall in ENGN 2210,

$$\underline{\varepsilon} = \frac{1}{2G}\left[\underline{\sigma} - \frac{\lambda}{2G+3\lambda}(tr\,\underline{\sigma})\,\underline{I}\right].$$

plug in: $\lambda = K - \frac{2}{3}G$.

$$\Rightarrow \underline{\underline{\varepsilon}} = \frac{1}{2G}\underline{\sigma} + \frac{1}{3}\left(\frac{1}{3K} - \frac{1}{2G}\right)tr(\underline{\sigma})\underline{I}.$$

$$\Rightarrow RHS = \frac{1}{2G_{Mo}}\dot{\sigma}_{ij} + \frac{1}{3}\left(\frac{1}{3K_{Me}} - \frac{1}{2G_{Me}}\right)tr(\dot{\underset{\sim}{\sigma}})\delta_{ij}$$

$$+ \frac{1}{2\eta_s}\sigma_{ij} + \frac{1}{3}\left(\frac{1}{3\eta_b} - \frac{1}{2\eta_s}\right)tr(\underline{\sigma})\delta_{ij}$$

Combine & simplify:

StrainTerm = StressTerm

where:

StrainTerm

$$= \left(1 + \frac{G_e}{G_{Me}}\right)\dot{\varepsilon}_{ij} + \frac{1}{3}\left(\frac{K_e}{K_{Me}} - \frac{G_e}{G_{Me}}\right)\delta_{ij}\,tr(\dot{\underline{\sigma}}).$$

$$+ \frac{G_e}{\eta_s}\varepsilon_{ij} + \frac{1}{3}\left(\frac{K_e}{\eta_b} - \frac{G_e}{\eta_s}\right)\delta_{ij}\,tr(\underline{\varepsilon}).$$

Stress Term

$$= \frac{1}{2G_{Mo}}\dot{\sigma}_{ij} + \frac{1}{3}\left(\frac{1}{3K_{Me}} - \frac{1}{2G_{Me}}\right)tr(\dot{\underline{\sigma}})\delta_{ij}$$

$$+ \frac{1}{2\eta_s}\sigma_{ij} + \frac{1}{3}\left(\frac{1}{3\eta_b} - \frac{1}{2\eta_s}\right)tr(\underline{\sigma})\delta_{ij}$$

So how to implement above into code (ABAQUS UMAT)?

First we take good care of all the annoying parameters.

Define: $\frac{1}{3}\left(\frac{K_e}{\eta_b} - \frac{G_e}{\eta_s}\right) = \lambda_1$ $\quad \frac{1}{3}\left(\frac{K_e}{K_{Me}} - \frac{G_e}{G_{Me}}\right) = \lambda_2$

$\frac{G_e}{\eta_s} = 2\mu_1$, $\left(1 + \frac{G_e}{G_{Me}}\right) = 2\mu_2$

$$\Rightarrow \text{strainterm} = 2\mu_1\varepsilon_{ij} + \lambda_1 tr(\underline{\varepsilon})\delta_{ij}$$
$$+ 2\mu_2\dot{\varepsilon}_{ij} + \lambda_2 tr(\dot{\underline{\varepsilon}})\delta_{ij}$$

Also define:

$$\frac{1}{3}\left(\frac{1}{3\eta_b}-\frac{1}{2\eta_s}\right)=\varphi_1, \quad \frac{1}{3}\left(\frac{1}{3K_{Me}}-\frac{1}{2G_{Me}}\right)=\varphi_2.$$

$$\frac{1}{2\eta_s}=2\psi_1, \quad \frac{1}{2G_{Me}}=2\psi_2$$

$$\Rightarrow \text{StressTerm} = \varphi_1 \, tr(\underline{\sigma})\,\delta_{ij} + 2\psi_1 \sigma_{ij}$$
$$+ \varphi_2 \, tr(\dot{\underline{\sigma}})\,\delta_{ij} + 2\psi_2 \dot{\sigma}_{ij}$$

Next, let's consider time discretization:

Similar to what we did in class

$$\varepsilon_{ij} \rightarrow \varepsilon_{ij} + \Delta\varepsilon_{ij} \qquad \sigma_{ij} \rightarrow \sigma_{ij} + \Delta\sigma_{ij}$$

$$tr(\underline{\varepsilon}) \rightarrow tr(\underline{\varepsilon}) + \Delta tr(\underline{\varepsilon}) \qquad tr(\underline{\sigma}) \rightarrow tr(\underline{\sigma}) + \Delta tr(\underline{\sigma})$$

$$\dot{\varepsilon}_{ij} \rightarrow \frac{\Delta\varepsilon_{ij}}{\Delta t} \qquad \dot{\sigma}_{ij} \rightarrow \frac{\Delta\sigma_{ij}}{\Delta t}$$

$$tr(\dot{\underline{\varepsilon}}) \rightarrow \frac{\Delta tr(\underline{\varepsilon})}{\Delta t} \qquad tr(\dot{\underline{\sigma}}) \rightarrow \frac{\Delta tr(\underline{\sigma})}{\Delta t}$$

as $t \rightarrow t+\Delta t$.

Plug in and Simplify:

I double checked this part, it's correct!

StrainTerm $\cdot \Delta t$

$$= 2\mu_1(\varepsilon_{ij}+\Delta\varepsilon_{ij})\Delta t + \lambda_1 \delta_{ij}[tr(\underline{\varepsilon}) + \Delta tr(\underline{\varepsilon})]\Delta t$$
$$+ 2\mu_2 \Delta\varepsilon_{ij} + \lambda_2 \delta_{ij} \Delta tr(\underline{\varepsilon}).$$

$$= 2\Delta t\, \varepsilon_{ij}\cdot\mu_1 + 2\Delta\varepsilon_{ij}(\Delta t\mu_1 + \mu_2)$$
$$+ \Delta t\, \delta_{ij}\, tr(\underline{\varepsilon})\cdot\lambda_1 + \Delta tr(\underline{\varepsilon})\,\delta_{ij}\,(\Delta t\lambda_1 + \lambda_2).$$

Define: $\Delta t \lambda_1 + \lambda_2 = A$
$\Delta t \mu_1 + \mu_2 = B$

$\Rightarrow$ strainTerm $\cdot \Delta t = A \Delta tr(\varepsilon)\delta_{ij} + B \Delta \varepsilon_{ij}$
$+ \Delta t\, \delta_{ij} tr(\varepsilon) \cdot \lambda_1 + 2\Delta t\, \varepsilon_{ij} \cdot \mu_1$

StressTerm $\cdot \Delta t$

$$= \Delta t\, \varphi_1 \delta_{ij}[tr(\sigma) + \Delta tr(\sigma)] + 2\Delta t\, \psi_1 (\sigma_{ij} + \Delta\sigma_{ij})$$
$$+ \varphi_2 \delta_{ij} \Delta tr(\sigma) \qquad + 2\psi_2 \Delta\sigma_{ij}$$

$$= \Delta t[\varphi_1 tr(\sigma)\delta_{ij} + 2\psi_1 \sigma_{ij}] + (\varphi_1 \Delta t + \varphi_2)\Delta tr(\sigma)\delta_{ij}$$
$$+ 2(\psi_1 \Delta t + \psi_2)\Delta\sigma_{ij}$$

Define: $\Delta t \varphi_1 + \varphi_2 = D$
$\Delta t\, \psi_1 + \psi_2 = F$

$\Rightarrow$ StressTerm $\cdot \Delta t = D\delta_{ij}\Delta tr(\sigma) + F\Delta\sigma_{ij}$
$+ \Delta t[\varphi_1 tr(\sigma)\delta_{ij} + 2\psi_1 \sigma_{ij}]$.

Next, we separate normal stress and shear stress.

To obtain normal components, just let $i=j$ and permute:

3 StressTerm $\cdot \Delta t = 3D\Delta tr(\sigma) + F\Delta tr(\sigma)$
$+ \Delta t[3\varphi_1 tr(\sigma) + 2\psi_1 tr(\sigma)]$.

3 strainTerm $\cdot \Delta t = 3A\Delta tr(\varepsilon) + B\Delta tr(\varepsilon)$
$+ \Delta t[3\lambda_1 tr(\varepsilon) + 2\mu_1 tr(\varepsilon)]$

Simplify:

$\Rightarrow$ 3 · StressTerm · Δt

$= (3D+F)\Delta tr(\underline{\sigma}) + \Delta t(3\varphi_1+2\psi_1)tr(\underline{\sigma}).$

$= [(3\varphi_1+2\psi_1)\Delta t + (3\varphi_2+2\psi_2)]\,\Delta tr(\underline{\sigma})$

$\quad + \Delta t(3\varphi_1+2\psi_1)\,tr(\underline{\sigma}).$

Define: $G = \varsigma_1\Delta t + \varsigma_2$ $\quad \begin{cases} \varsigma_1 = 3\varphi_1+2\psi_1 = \frac{1}{3\eta_b} \\ \varsigma_2 = 3\varphi_2+2\psi_2 = \frac{1}{3K_{ne}} \end{cases}$

(I) --- $\therefore$ 3 · Stress Term · $\Delta t = G\,\Delta tr(\underline{\sigma}) + \Delta t\,\varsigma_1 \cdot tr(\underline{\sigma}).$

Similarly:

3 · Strain Term · Δt

$= (3A+B)\Delta tr(\underline{\varepsilon}) + \Delta t(3\lambda_1+2\mu_1)tr(\underline{\varepsilon}).$

$= [(3\lambda_1+2\mu_1)\Delta t + (3\lambda_2+2\mu_2)]\,\Delta tr(\underline{\varepsilon})$

$\quad + \Delta t(3\lambda_1+2\mu_1)\,tr(\underline{\varepsilon})$

Define: $H = \omega_1\Delta t + \omega_2$ $\quad \begin{cases} \omega_1 = 3\lambda_1+2\mu_1 = \frac{K_e}{\eta_b} \\ \omega_2 = 3\lambda_2+2\mu_2 = 1 + \frac{K_e}{K_{ne}} \end{cases}$

(II) --- $\therefore$ 3 strainTerm · $\Delta t = H\Delta tr(\underline{\varepsilon}) + \Delta t\,\omega_1 tr(\underline{\varepsilon}).$

Combine equation (I) and (II), we have the relation:

$$G\,\Delta tr(\underline{\sigma}) + \Delta t\cdot\varsigma_1\cdot tr(\underline{\sigma}) = H\,\Delta tr(\underline{\varepsilon}) + \Delta t\cdot\omega_1 tr(\underline{\varepsilon}).$$

(III) --- $\Rightarrow \Delta tr(\underline{\sigma}) = \frac{1}{G}\left[H\cdot tr(\underline{\varepsilon}) + \Delta t\left(\omega_1 tr(\underline{\varepsilon}) - \varsigma_1\cdot tr(\underline{\sigma})\right)\right]$

With the obtain equation (III), we are now able to decouple σ_{ij} from stressTerm.

Let's take a look at xx component.

$i=j=x, \quad \delta_{ij}=1, \quad \varepsilon_{ij}=\varepsilon_{xx}, \quad \sigma_{ij}=\sigma_{xx}$

$tr(\sigma)=\sigma_{xx}+\sigma_{yy}+\sigma_{zz}, \quad tr(\varepsilon)=\varepsilon_{xx}+\varepsilon_{yy}+\varepsilon_{zz}.$

plug in eq.(III)

StressTerm$\cdot\Delta t$:

$$= F\Delta\sigma_{ij}+\frac{D}{G}[H\cdot\Delta tr(\varepsilon)+\Delta t(\omega_1 tr(\varepsilon)-\varsigma_1 tr(\sigma))]\delta_{ij}$$

$$+\Delta t[\varphi_1 tr(\sigma)\delta_{ij}+2\psi_1\sigma_{ij}]$$

$$= F\Delta\sigma_{ij}+\frac{DH}{G}\Delta tr(\varepsilon)+\Delta t(\varphi_1-\frac{D\varsigma_1}{G})tr(\sigma)\delta_{ij}$$

$$+\Delta t\cdot 2\psi_1\sigma_{ij}+\Delta t\frac{D\omega_1}{G}tr(\varepsilon)\delta_{ij}$$

plug in $i=j=x$

$$= F\Delta\sigma_{xx}+\frac{DH}{G}[\Delta\varepsilon_{xx}+\Delta\varepsilon_{yy}+\Delta\varepsilon_{zz}]+\Delta t(\varphi_1-\frac{D\varsigma_1}{G})(\sigma_{xx}+\sigma_{yy}+\sigma_{zz})$$

$$+\Delta t\cdot 2\psi_1\sigma_{xx}+\Delta t\frac{D\omega_1}{G}(\varepsilon_{xx}+\varepsilon_{yy}+\varepsilon_{zz}) \quad \cdots\cdots (IV)$$

StrainTerm$\cdot\Delta t$

plugin $i=j=x$

$$= A\Delta tr(\varepsilon)\delta_{ij}+B\Delta\varepsilon_{ij}+\Delta t\,\delta_{ij}\cdot tr(\varepsilon)\lambda_1+2\Delta t\,\varepsilon_{ij}\mu_1$$

$$= A(\Delta\varepsilon_{xx}+\Delta\varepsilon_{yy}+\Delta\varepsilon_{zz})+B\Delta\varepsilon_{xx}+\Delta t\,\lambda_1(\varepsilon_{xx}+\varepsilon_{yy}+\varepsilon_{zz})$$

$$+2\Delta t\,\varepsilon_{xx}\mu_1. \quad \cdots\cdots (V)$$

Combine (IV) and (V)

$$F\Delta\sigma_{xx}=(\Delta\varepsilon_{xx}+\Delta\varepsilon_{yy}+\Delta\varepsilon_{zz})(A-\frac{DH}{G})+\Delta\varepsilon_{xx}\cdot B$$

$$+(\varepsilon_{xx}+\varepsilon_{yy}+\varepsilon_{zz})(\lambda_1-\frac{D\omega_1}{G})\Delta t+\varepsilon_{xx}\cdot 2\mu_1\Delta t$$

$$-(\sigma_{xx}+\sigma_{yy}+\sigma_{zz})(\varphi_1-\frac{D\varsigma_1}{G})\Delta t-\sigma_{xx}\cdot 2\psi_1\Delta t$$

$$\Rightarrow \Delta\sigma_{xx} = (\Delta\varepsilon_{xx}+\Delta\varepsilon_{yy}+\Delta\varepsilon_{zz}) \times \frac{A - \frac{DH}{G}}{F}$$

$$+ \Delta\varepsilon_{xx} \times \frac{B}{F}$$

$$+ (\varepsilon_{xx}+\varepsilon_{yy}+\varepsilon_{zz}) \times \frac{\Delta t}{F} \times (\lambda_1 - \frac{D\omega_1}{G})$$

$$+ \varepsilon_{xx} \times \frac{\Delta t}{F} \times 2\mu_1$$

$$+ (\sigma_{xx}+\sigma_{yy}+\sigma_{zz}) \times \frac{\Delta t}{F} \times (-\varphi_1 + \frac{DS_1}{G})$$

$$+ \sigma_{xx} \times \frac{\Delta t}{F} \times (-2\psi_1). \quad \text{--- (VI)}$$

For other normal components, just change $i=j=\begin{cases} y \\ z. \end{cases}$

For shear components, it's much easier!

$i \neq j, \quad \delta_{ij} = 0.$

Stress Term $\cdot \Delta t = F\Delta\sigma_{ij} + 2\Delta t \cdot \psi_1 \sigma_{ij}$

Strain Term $\cdot \Delta t = B\Delta\varepsilon_{ij} + 2\Delta t \cdot \mu_1 \varepsilon_{ij}$

combine: $\Delta\sigma_{ij} = \frac{1}{F}[B\Delta\varepsilon_{ij} + 2\Delta t(\mu_1 \varepsilon_{ij} - \psi_1 \sigma_{ij})]$.

in ABAQUS UMAT, we use engineering shear strain γ_{ij}

$$\varepsilon_{ij} = \frac{\gamma_{ij}}{2}, \quad \Delta\varepsilon_{ij} = \frac{\Delta\gamma_{ij}}{2}$$

$$\Rightarrow \Delta\sigma_{ij} = \frac{B}{2F}\Delta\gamma_{ij} + \frac{\Delta t}{F}\mu_1 \cdot \gamma_{ij} - \frac{2\Delta t}{F}\psi_1 \cdot \sigma_{ij} \quad \text{--- (VII).}$$

write (VI) and (VII) in matrix form:
(so that we could write this into UMAT).

normal components: Define: $\text{Stress Inc}^{(N)} = \begin{bmatrix} \Delta\sigma_{xx} \\ \Delta\sigma_{yy} \\ \Delta\sigma_{zz} \end{bmatrix}$ $\quad \text{Stress}^{(N)} = \begin{bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{zz} \end{bmatrix}$

$\text{Strain Inc}^{(N)} = \begin{bmatrix} \Delta\varepsilon_{xx} \\ \Delta\varepsilon_{yy} \\ \Delta\varepsilon_{zz} \end{bmatrix}$ $\quad \text{Strain}^{(N)} = \begin{bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \varepsilon_{zz} \end{bmatrix}$.

Define $[C] = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$ $\quad [I] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$.

$$\Rightarrow \text{Stress Inc}^{(N)} = \text{DDSDDE}^{(N)} \cdot \text{Strain Inc}^{(N)} + \text{DDSDE}^{(N)} \cdot \text{Strain}^{(N)} + \text{DDSDS}^{(N)} \cdot \text{Stress}^{(N)}$$

where $\text{DDSDDE}^{(N)} = \dfrac{A - \frac{DH}{G}}{F}[C] + \dfrac{B}{F}[I]$

$$\text{DDSDE}^{(N)} = \frac{\Delta t}{F}\left(\lambda_1 - \frac{D\omega_1}{G}\right)[C] + \frac{2\Delta t}{F}\mu_1[I]$$

$$\text{DDSDS}^{(N)} = -\frac{\Delta t}{F}\left(\varphi_1 - \frac{DS_1}{G}\right)[C] - \frac{2\Delta t}{F}\psi_1[I]$$

For shear components, we perform similar tricks:

Define: $\text{Stress Inc}^{(S)} = \begin{bmatrix} \Delta\sigma_{ij} \\ \Delta\sigma_{ik} \\ \Delta\sigma_{jk} \end{bmatrix}$ $\quad \text{Stress}^{(S)} = \begin{bmatrix} \sigma_{ij} \\ \sigma_{ik} \\ \sigma_{jk} \end{bmatrix}$

$\text{Strain Inc}^{(S)} = \begin{bmatrix} \Delta\varepsilon_{ij} \\ \Delta\varepsilon_{ik} \\ \Delta\varepsilon_{jk} \end{bmatrix}$ $\quad \text{Strain}^{(S)} = \begin{bmatrix} \varepsilon_{ij} \\ \varepsilon_{ik} \\ \varepsilon_{jk} \end{bmatrix}$

$$\Rightarrow \text{Stress Inc}^{(S)} = \text{DDSDDE}^{(S)} \cdot \text{Strain Inc}^{(S)} + \text{DDSDE}^{(S)} \cdot \text{Strain}^{(S)} + \text{DDSDS}^{(S)} \cdot \text{Stress}^{(S)}.$$

where: $DDSDDE^{(s)} = \frac{B}{2F} [I]$

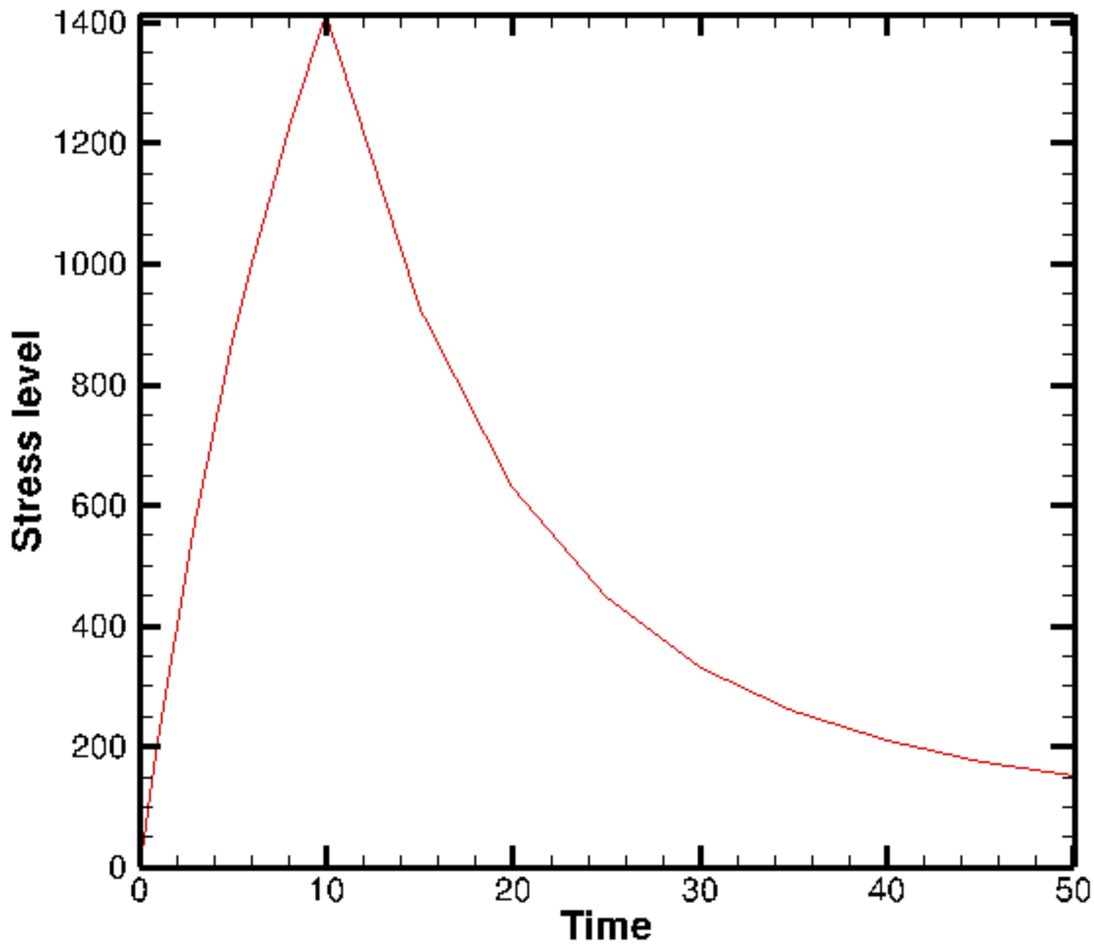
$$DDSDT^{(s)} = \frac{\Delta t}{F} \mu_1 [I]$$

$$DDSDS^{(s)} = -\frac{2\Delta t}{F} \mu_1 [I],$$

Now that we solve the time evolution bruteforcedly, we implement this in code (ABAQUS UMAT) to see what happens.

Results:

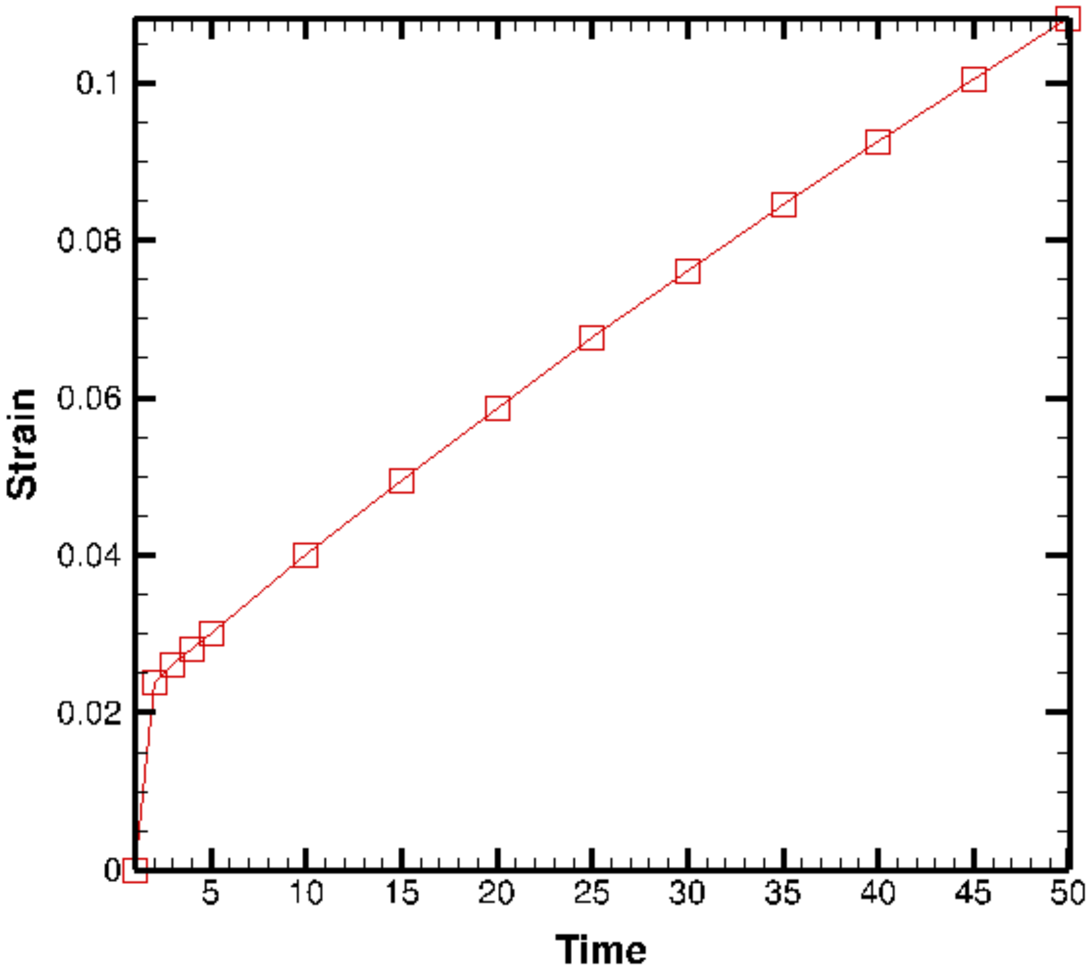
First, I test a simple 2-element model with prescribed displacement (time-dependant) at one end, fix the other end and let the rest faces be free. The prescribed displacement would increase to 0.1 from t=0 to t=10. After that, it will stay fixed.



The loading curve as well as the relaxation curve is in good agreement pure theoretical calculation (in 1D). This shows that the way I implement SLS model into FEM is correct.

Then I test the same 2-element sample with force loading (also time-dependent) in one end, fix the other end while letting the rest faces be free. The prescribed loading would increase to 500 from t=0 to t=1. After that, it will stay fixed. This setting is to mimic the loading condition for rat cervical tissue during childbirth.

The result is:



Here, we could see after a sudden increase (which corresponds to the sudden stress increment), the strain level will increase almost linearly with time (meanwhile, the stress is constant).
This strain-time relation is in good agreement with the experimental result.

Therefore, I conclude that SLS model is able to capture the linear creep response observed in rat cervical tissue during childbirth.

Link to my Github repo:
https://github.com/monyxl/EN234_FEA

Here below is my UMAT code:

```fortran
!
!    ABAQUS format user material subroutine for small strain hypoelastic material
!
!

      SUBROUTINE UMAT(STRESS,STATEV,DDSDDE,SSE,SPD,SCD,
     1 RPL,DDSDDT,DRPLDE,DRPLDT,
     2 STRAN,DSTRAN,TIME,DTIME,TEMP,DTEMP,PREDEF,DPRED,CMNAME,
     3 NDI,NSHR,NTENS,NSTATV,PROPS,NPROPS,COORDS,DROT,PNEWDT,
     4 CELENT,DFGRD0,DFGRD1,NOEL,NPT,LAYER,KSPT,KSTEP,KINC)
!
      INCLUDE 'ABA_PARAM.INC'
!
      CHARACTER*80 CMNAME
      DIMENSION STRESS(NTENS),STATEV(NSTATV),
     1 DDSDDE(NTENS,NTENS),DDSDDT(NTENS),DRPLDE(NTENS),
     2 STRAN(NTENS),DSTRAN(NTENS),TIME(2),PREDEF(1),DPRED(1),
     3 PROPS(NPROPS),COORDS(3),DROT(3,3),DFGRD0(3,3),DFGRD1(3,3)
      double precision :: KE, GE, KMe, GMe, GV, KV
      double precision :: lambda1, lambda2, mu1, mu2
      double precision :: phi1, phi2, psi1, psi2
      double precision :: omega1, omega2, zeta1, zeta2
      double precision :: A, B, C, D, F, G, H
      double precision, dimension(NTENS,NTENS) :: ones, diag
      double precision, dimension(NTENS,NTENS) :: DDSDE
      double precision, dimension(NTENS,NTENS) :: DDSDS
      double precision, dimension(NTENS) :: SIGold


      integer :: j

       KE = PROPS(1)
       GE = PROPS(2)
       KMe = PROPS(3)
       GMe = PROPS(4)
       GV = PROPS(5)
       !KV = PROPS(6) infinity

       !lambda1 = (-GE/GV+KE/KV)/3.d0
       lambda1 = (-GE/GV+0.d0)/3.d0
       lambda2 = (-GE/GMe+KE/KMe)/3.d0
       mu1 = GE/(2.d0*GV)
       mu2 = (1.d0+GE/GMe)/2.d0
       phi1 = -1.d0/(6.d0*GV)!+1.d0/(9.d0*KV)
       phi2 = -1.d0/(6.d0*GMe) + 1.d0/(9.d0*KMe)
       psi1 = 1.d0/(4.d0*GV)
       psi2 = 1.d0/(4.d0*GMe)
       !omega1 = KE/KV
       omega1 = 0.d0
       omega2 = 1.d0 + KE/KMe
       !zeta1 = 1.d0/(3.d0*KV)
       zeta1 = 0.d0
       zeta2 = 1.d0/(3.d0*KMe)

       !ABCDFGH
```

```
  A = lambda1*DTIME + lambda2
  B = 2.d0*(mu1*DTIME + mu2)

  D = phi1*DTIME + phi2
  F = 2.d0*(psi1*DTIME + psi2)
  G = zeta1*DTIME + zeta2
  H = omega1*DTIME + omega2

  !Initialize
  DDSDDE = 0.d0
  DDSDE = 0.d0
  DDSDS = 0.d0
  SIGold = 0.d0
  ones = 0.d0
  diag = 0.d0

  !Define Tensor coefficients
  ones(1:NTENS,1:NTENS) = 1.d0
  diag(1:NTENS,1:NTENS) = 0.d0
  forall(K1=1:NTENS) diag(K1,K1) = 1.d0

  !Define DDSDDE, DDSDE, DDSDS related to Normal Stress
  DDSDDE(1:NDI,1:NDI) = (A-D*H/G)/F*ones(1:NDI,1:NDI)
!                            + B/F*diag(1:NDI,1:NDI)

  DDSDE(1:NDI,1:NDI) = DTIME/F*(lambda1-omega1*D/G)*
!          ones(1:NDI,1:NDI) + 2.d0*DTIME*mu1/F*diag(1:NDI,1:NDI)

  DDSDS(1:NDI,1:NDI) = -DTIME*(-phi1+D*zeta1/G)/F*
!          ones(1:NDI,1:NDI) - 2.d0*DTIME*psi1/F*diag(1:NDI,1:NDI)

  !Define DDSDDE, DDSDE, DDSDS related to Shear Stress
  DDSDDE(NDI+1:NTENS,NDI+1:NTENS) = B/2.d0/F
!                               *diag(NDI+1:NTENS,NDI+1:NTENS)
  DDSDE(NDI+1:NTENS,NDI+1:NTENS) = DTIME*mu1/F
!                               *diag(NDI+1:NTENS,NDI+1:NTENS)
  DDSDS(NDI+1:NTENS,NDI+1:NTENS) = -2.d0*DTIME*psi1/F
!                               *diag(NDI+1:NTENS,NDI+1:NTENS)

  !Store present stress state ub array SIGold
  do i = 1,ntens
      SIGold(i) = stress(i)
  end do

  !Update stresses
  do i = 1,ntens
  do j = 1,ntens
     stress(i) = stress(i) + DDSDDE(i,j)*(DSTRAN(j))
!                          + DDSDE(i,j)*(STRAN(j))
!                          + DDSDS(i,j)*(SIGold(j))

  end do
  end do
  return

 RETURN
 END SUBROUTINE UMAT
```

For input files, I edited the one we used in HW6 for porous elastic material, and play with the loading conditions:

1. For displacement loading, I used:

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%% BOUNDARY CONDITIONS %%%%%%%%%%%%%%%%%%%%%%%%%%%%

%      The BOUNDARY conditions key starts definition of BCs
       BOUNDARY CONDITIONS

%       The HISTORY key defines a time history that can be applied to DOFs or distributed
loads
        HISTORY, dof_history
          0.d0, 0.d0                  % Each line gives a time value and then a function
value
         10.d0, 0.1d0
        END HISTORY

        HISTORY, dload_history
          0.d0, 0.d0
         1.d0, 500.d0
        END HISTORY

%       The NODESET key defines a list of nodes
        NODESET, node1
           1
         END NODESET
        NODESET, left
           1, 4, 5, 8
        END NODESET
        NODESET, right
           9, 10, 12, 11
        END NODESET
        NODESET, side
           1, 2, 5, 6, 11, 9
        END NODESET

%      The ELEMENTSET key defines a list of elements
       ELEMENTSET, end_element
           2
       END ELEMENTSET

%       The DEGREE OF FREEDOM key assigns values to nodal DOFs
%       The syntax is node set name, DOF number, and either a value or a history name.
%
        DEGREES OF FREEDOM
           1, 3, VALUE, 0.d0
           side, 2, VALUE, 0.d0
           left, 1, VALUE, 0.d0
           right, 1, HISTORY, dof_history
        END DEGREES OF FREEDOM
```

……

2. For force loading, I used:

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%% BOUNDARY CONDITIONS %%%%%%%%%%%%%%%%%%%%%%%%%%%

%     The BOUNDARY conditions key starts definition of BCs
      BOUNDARY CONDITIONS

%       The HISTORY key defines a time history that can be applied to DOFs or distributed
loads
        HISTORY, dof_history
          0.d0, 0.d0                  % Each line gives a time value and then a function
value
         10.d0, 0.1d0
        END HISTORY

        HISTORY, dload_history
          0.d0, 0.d0
         1.d0, 500.d0
        END HISTORY

%       The NODESET key defines a list of nodes
        NODESET, node1
           1
         END NODESET
        NODESET, left
           1, 4, 5, 8
        END NODESET
        NODESET, right
           9, 10, 12, 11
        END NODESET
        NODESET, side
           1, 2, 5, 6, 11, 9
        END NODESET

%      The ELEMENTSET key defines a list of elements
       ELEMENTSET, end_element
           2
       END ELEMENTSET

%       The DEGREE OF FREEDOM key assigns values to nodal DOFs
%       The syntax is node set name, DOF number, and either a value or a history name.
%
        DEGREES OF FREEDOM
           1, 3, VALUE, 0.d0
           side, 2, VALUE, 0.d0
           left, 1, VALUE, 0.d0
        END DEGREES OF FREEDOM


        DISTRIBUTED LOADS
          end_element, 4, NORMAL,dload_history
        END DISTRIBUTED LOADS
```

……