Viscoelastic material deformation

Hasan Eruslu
University of Delaware

September 27, 2019

1 Fractional Zener model

Our goal is computing a FEM approximation of the viscoelastic wave propagation problem with Fractional Zener model: for $t \geqslant 0$

$$-\mathrm{div}\,\boldsymbol{\sigma}(t) + \rho\,\ddot{\mathbf{u}}(t) = \mathbf{f}(t) \qquad \text{in } \Omega, \tag{1.1a}$$

$$\mathbf{u}(t) = \mathbf{u}_D(t) \qquad \text{on } \Gamma_D, \tag{1.1b}$$

$$\boldsymbol{\sigma}(t)\,\boldsymbol{\nu} = \boldsymbol{\sigma}_N(t)\,\boldsymbol{\nu} \qquad \text{on } \Gamma_N, \tag{1.1c}$$

where the stress-strain relation can be written in Laplace domain as

$$\mathbf{S}(s) = \mu(s)\boldsymbol{\varepsilon}(\mathbf{U}(s)) + \lambda(s)(\nabla\cdot\mathbf{U}(s))\mathrm{I} \qquad \forall s \in \mathbb{C}_+ := \{s \in \mathbb{C} : \mathrm{Re}\, s > 0\}, \tag{1.1d}$$

with

$$\mu(s) = \frac{m_\mu + b_\mu s^{\nu_\mu}}{1 + a s^{\nu_\mu}}, \qquad \lambda(s) = \frac{m_\lambda + b_\lambda s^{\nu_\lambda}}{1 + a s^{\nu_\lambda}} \tag{1.1e}$$

for $m_\mu, b_\mu, \nu_\mu, m_\lambda, b_\lambda, \nu_\lambda \in L^\infty(\Omega)$ strictly positive satisfying

$$a m_\mu \leqslant b_\mu, \qquad a m_\lambda \leqslant b_\lambda.$$

Here we assume homogeneous initial conditions

$$\mathbf{u}(0) = \mathbf{0}, \qquad \dot{\mathbf{u}}(0) = \mathbf{0}. \tag{1.1f}$$

1.1 Parameter details

1. ρ: Mass density function can be heterogenous. Name in the simulation: `rho`
2. $m_\mu, b_\mu, \nu_\mu, m_\lambda, b_\lambda, \nu_\lambda$: Viscoelastic parameters can be heterogenous. Corresponding names in the simulation: `m_mu, b_mu, nu_mu, m_lam, b_lam, nu_lam`

2 Solver details

Viscoelasticity solver computes the FEM coefficient vector of the displacement and the stress at DOF. This is done via the following steps.

1. Sample the forcing term, Dirichlet and Neumann data at the time steps. This is done in a parallel loop using a helper function. Depending on the time stepping method, we sample a little differently:

   (a) **Sampling for the CQ method.** We create $3N_{\text{dof}} \times 1$ vectors

   $$\mathbf{f}^n, \quad \mathbf{u}^n_{h,D}, \quad \mathbf{t}^n, \qquad n = 1, \ldots, N_t$$

   corresponding to load, Dirichlet and Neumann approximations at time steps $t_n = n\Delta$ (uniform time steps). Then we combine all these samples and construct the $6N_{\text{dof}} \times N_t$ matrix X in the following way

   $$\mathrm{X} = \left[\begin{array}{c|c|c|c} X_1 & X_2 & \cdots & X_{N_t} \end{array}\right],$$

   where each column is

   $$X_n = \begin{bmatrix} \mathbf{u}^n_{h,D} \\ \mathbf{f}^n + \mathbf{t}^n \end{bmatrix} \qquad n = 1, \ldots, N_t.$$

   (b) **Sampling for the RKCQ method.** We run a parallel loop over time steps $n = 1, \ldots, N_t - 1$, and a serial loop over stages $\ell = 1, \ldots, S$ to create $3N_{\text{dof}} \times 1$ vectors

   $$\mathbf{f}^{n,\ell}, \quad \mathbf{u}^{n,\ell}_{h,D}, \quad \mathbf{t}^{n,\ell},$$

   corresponding to load, Dirichlet and Neumann approximations at time steps

   $$t_{n,\ell} = (n + c_\ell)\Delta, \qquad n = 1, \ldots, N_t, \quad \ell = 1, \ldots, S.$$

   Here the index ℓ corresponds to the variable `ns` in the code. The vector $\mathbf{c}$, on the other hand, is from the Butcher tableau and equal to the row sums of the $S \times S$ matrix $\mathbf{A}$. This matrix is an optional input if one wants to use Runge-Kutta as time-stepping method.
   Now we first create $6N_{\text{dof}} \times N_t$ matrices

   $$\mathrm{X}^\ell = \left[\begin{array}{c|c|c|c} X^\ell_1 & X^\ell_2 & \cdots & X^\ell_{N_t} \end{array}\right] \qquad \ell = 1, \ldots, S$$

   where each column is

   $$X^\ell_n = \begin{bmatrix} \mathbf{u}^{n,\ell}_{h,D} \\ \mathbf{f}^{n,\ell} + \mathbf{t}^{n,\ell} \end{bmatrix} \qquad n = 1, \ldots, N_t, \quad \ell = 1, \ldots, S.$$

Then we vertically stack matrices X^ℓ for $\ell = 1, \dots, S$

$$\mathrm{X} = \begin{bmatrix} \mathrm{X}^1 \\ \vdots \\ \mathrm{X}^S \end{bmatrix},$$

to obtain a single $6N_{\mathrm{dof}}S \times N_t$ matrix.

2. We construct the transfer function $\mathcal{F}_h$ in the Laplace domain such that

$$\mathcal{F}_h(s, \mathbf{u}_D, \mathbf{f}, \boldsymbol{\sigma}_N) = (\mathbf{u}^h, \boldsymbol{\sigma}^h)$$

is FEM approximation of

$$\begin{aligned} -\mathrm{div}\, \boldsymbol{\sigma} + s^2\, \rho\, \mathbf{u} &= \mathbf{f} && \text{in } \Omega, \\ \mathbf{u} &= \mathbf{u}_D && \text{on } \Gamma_D, \\ \boldsymbol{\sigma}\, \boldsymbol{\nu} &= \boldsymbol{\sigma}_N\, \boldsymbol{\nu} && \text{on } \Gamma_N. \end{aligned}$$

3. Using the transfer function $\mathcal{F}_h$ and sampled data X we perform an all-at-once CQ time stepping.

4. We finally output the FEM approximation of displacement and post-processed stress

$$\mathbf{u}_h^n, \quad \boldsymbol{\sigma}_h^n$$

at the time steps t_n for $n = 1, \cdots, N$.