SIMON FRASER UNIVERSITY
DEPARTMENT OF BIOMEDICAL PHYSIOLOGY AND KINESIOLOGY

NEUROMUSCULAR MECHANICS LABORATORY

TECHNICAL REPORT TR-02-2024

FLEXODEAL-LITE V0.1: SOLVER PERFORMANCE FOR DYNAMIC PASSIVE NEO-HOOKEAN DEFORMATION

AUTHOR(S): Javier Almonacid	**REVISED BY**:	**APPROVED BY:**
LAST UPDATED: February 12, 2024	**LAST REVISION:**	**DATE APPROVED:**

TECHNICAL REPORT TR-02-2024	Total Pages
FLEXODEAL-LITE V0.1: SOLVER PERFORMANCE FOR DYNAMIC PASSIVE NEO-HOOKEAN DEFORMATION	10

1. Executive Summary

TBD

Contents

2. Overall description of v0.1

The code begins from a dynamic extension of step-44 available at

https://github.com/javieralmonacid/dynamic-step-44.

First, the geometry is changed from the cube considered in step-44 to a more general hyper-rectangle (in deal.II's terminology) of length, width, and height provided in the parameters file. Then we implement a simple boundary condition for pushing/pulling which we describe below. Finally, in addition to the CG and Direct (UMFPACK) solvers implemented by default in step-44, we implement the GMRES solver.

2.1 Linear pull/push profile

First, we assume the standard colouring of faces for the hyper-rectangle, i.e. -x face = boundary ID 0, +x face = 1, -y face = 2, +y face = 3, -z face = 4, +z face = 5. Then, on the +x face of the block, given a start time t_{start}, an end time t_{end}, a maximum strain λ_0, a pulling/pushing strain rate ϵ_0, and a muscle length L_0, we prescribe the following length profile:

$$L(t) = \begin{cases} L_0 & t_{start} \leq t \leq t_0, \\ \mathrm{sign}(\lambda_0)\epsilon_0(t - t_*) + L_* & t_0 < t < t_1, \\ L_1 & t_1 \leq t \leq t_{end}. \end{cases} \tag{2.1}$$

Here,

$$t_* = \frac{t_{start} + t_{end}}{2}, \quad L_* = \frac{L_0 + L_1}{2}, \quad L_1 = \lambda_0(1 + L_0).$$

Moreover, if $L_1 > L_0$ (i.e. when pulling),

$$t_0 = \frac{L_0 - L_*}{\epsilon_0} + t_*, \quad t_1 = \frac{L_1 - L_*}{\epsilon_0} + t_*.$$

In turn, if $L_0 > L_1$ (i.e. when pushing),

$$t_0 = \frac{L_1 - L_*}{\epsilon_0} + t_*, \quad t_1 = \frac{L_0 - L_*}{\epsilon_0} + t_*.$$

The idea is that the line defining the length for times $t_0 < t < t_1$ never surpasses L_0 or L_1. This means that $t_0 \geq t_{start}$ and $t_1 \leq t_{end}$, which can be enforced by asking that

$$\epsilon_0 \geq \left| \frac{\lambda_0 L_0}{t_{end} - t_{start}} \right|.$$

The displacement boundary condition is then simply $\mathbf{u}_D = (u_D, 0, 0)$, with $u_D(t) = L(t) - L_0$. The remaining boundary conditions are:

- -x face: clamped in all three directions.
- -y, +y, -z, +z faces: traction free.

3. Numerical results (passive experiments)

We consider the following parameters :

- Geometry $[0, 3] \times [0, 1] \times [0, 1]$ with a scaling parameter of 1e-03[1]
- **set Type of simulation = dynamic**
- set Polynomial degree = 2
- set Quadrature order = 5
- Global refinement level = 3
- set Pressure ratio p/p0 = 0
- set Max iteration multiplier = 2
- set Residual = 1e-6 (linear solver residual)
- **set Use static condensation = false**
- set Preconditioner type = ssor
- set Preconditioner relaxation = 0.65
- **set Solver type = Direct**
- set Poisson's ratio = 0.4999
- set Shear modulus = 80.194e6
- set Max iterations Newton-Raphson = 20
- set Tolerance displacement = 1.0e-6
- set Tolerance force = 1.0e-9
- set End time = 0.525
- set Time step size = 0.025
- set Pulling face ID = 1
- set Pull time start = 0.0
- set Pull time end = 0.5
- set Pull strain = -0.15
- set Pull strain rate = 0.00090

The solver computes results in 272.8 seconds. We portray them next.

[1]The scaling parameter only affects the geometry of the block and not its material properties.

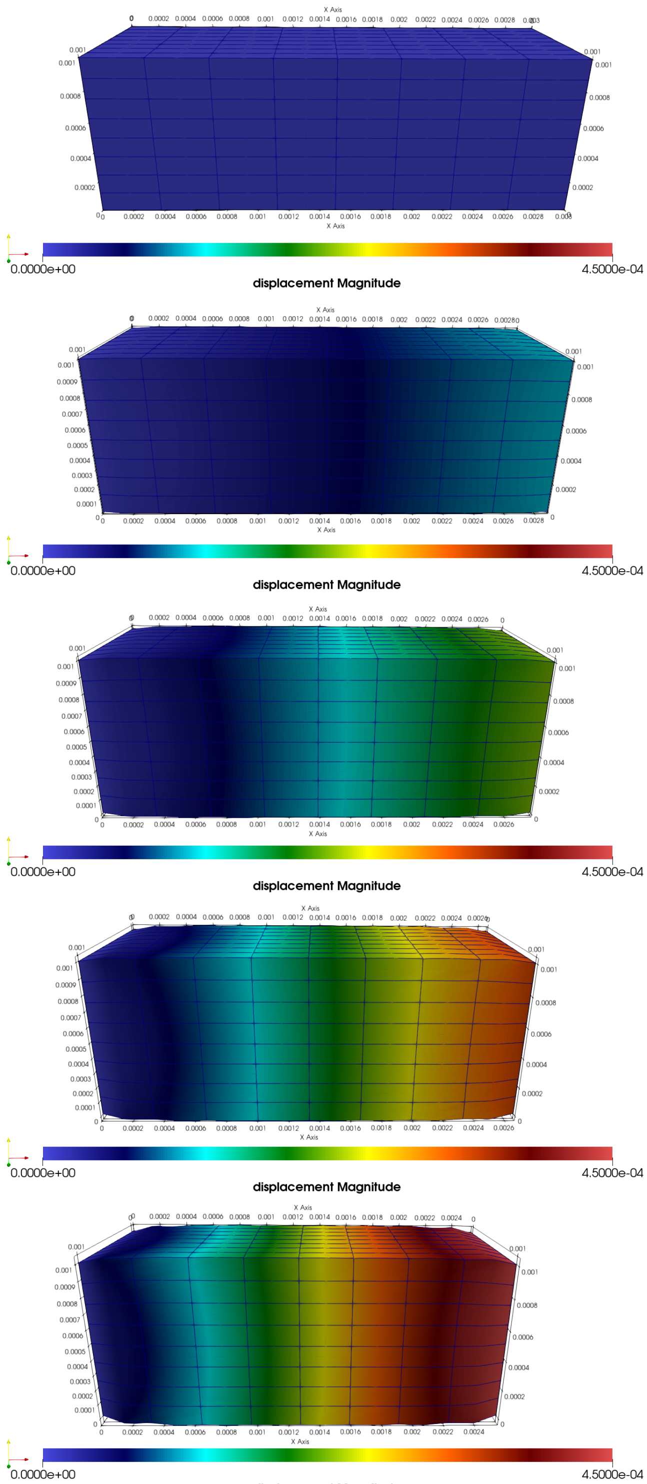

Figure 3.1: Screenshots of displacement at timesteps 0 of 20, 5 of 20, 10 of 20, 15 of 20, and 20 of 20.

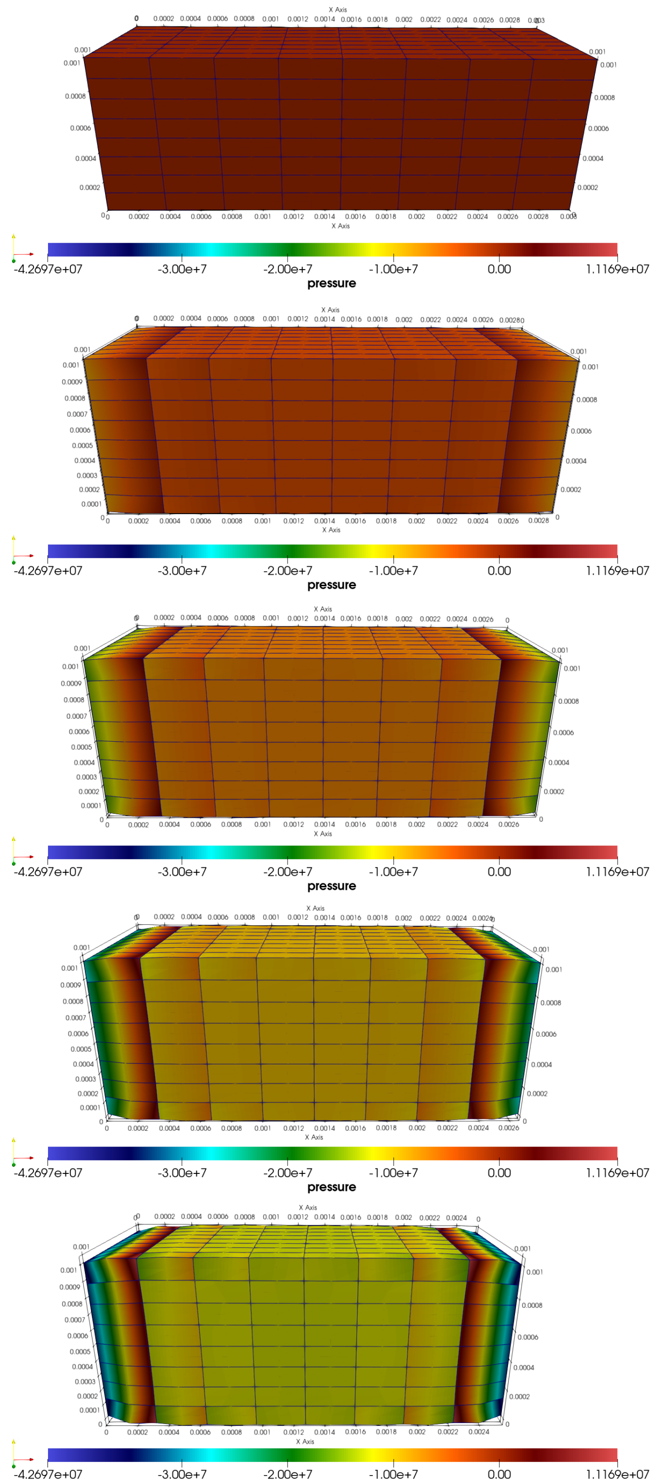

Figure 3.2: Screenshots of pressure at timesteps 0 of 20, 5 of 20, 10 of 20, 15 of 20, and 20 of 20.

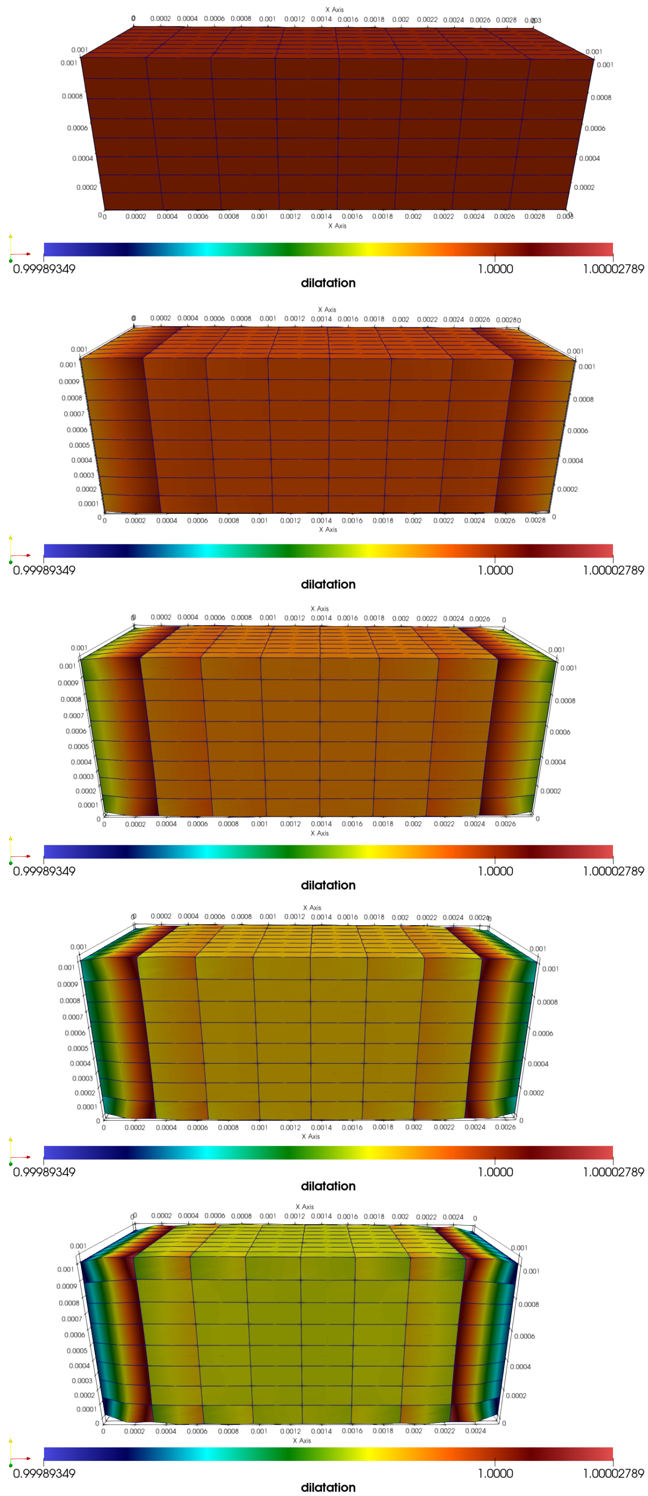

Figure 3.3: Screenshots of pressure at timesteps 0 of 20, 5 of 20, 10 of 20, 15 of 20, and 20 of 20.

Table 3.1: Comparative performance of implemented solvers for *quasi-static* deformation.

SC	Solver	Preconditioner	CPU Time (s)	Linear iterations
False	Direct	-	62.6	1,1,1
False	CG	ssor	254.1	565, 1039, 1855
False	CG	jacobi	78.3	1083, 952, 1627
False	GMRES	ssor	-	Too slow
False	GMRES	jacobi	-	Too slow
True	Direct	-	54.8	1,1,1
True	CG	ssor	1031.0	948, 3748, 5076, 5847
True	CG	jacobi	208.5	1875, 3311, 9679
True	GMRES	ssor	-	Failed at iteration #1
True	GMRES	jacobi	-	Failed at iteration #1

Table 3.2: Comparative performance of implemented solvers for *dynamic* deformation.

SC	Solver	Preconditioner	CPU Time (s)	Linear iterations
False	Direct	-	144.7	1,1,1
False	CG	ssor	338.1	568, 1022, 1885
False	CG	jacobi	163.3	1100, 879, 1562
False	GMRES	ssor	-	Too slow
False	GMRES	jacobi	-	Too slow
True	Direct	-	138.3	1,1,1
True	CG	ssor	1181.0	944, 3757, 5065, 5811
True	CG	jacobi	285.8	1875, 3339, 9675
True	GMRES	ssor	-	Failed at iteration #1
True	GMRES	jacobi	-	Failed at iteration #1

3.1 Comparative performance of solvers

The main goal of this section is to document the performance of the different solvers implemented in the code. Recall that, at this juncture, the material is purely Neo-Hookean with no fibre component. This means that the elasticity tensor $\mathbb{C}$ takes a simpler structure given that $\bar{\mathbb{C}}$ is identically zero. This type of experiment could also be thought as passively pulling a block of muscle in which the fibre component has been deactivated.

We repeat the experiment from previous section but set the end time to 0.275. This corresponds to the middle of the linear ramp. Here, report the number of linear iterations, as well as the number of nonlinear iterations. We see from Tables 3.1 and 3.2 that only when using CG + ssor the number of nonlinear iterations is 4. All other combinations of solvers complete a time step in 3 nonlinear iterations.

Bibliography

[1] G. A. HOLZAPFEL. Nonlinear solid mechanics: a continuum approach for engineering science. John Wiley & Sons, Ltd., Chichester, 2000.