contact mechanics solver

Yulin, Gitsuzo, Bogdan

structure dynamics and vibration laboratory

yulin.shi@mail.mcgill.ca

April 6, 2015

1.1 Introduction - System of interesting

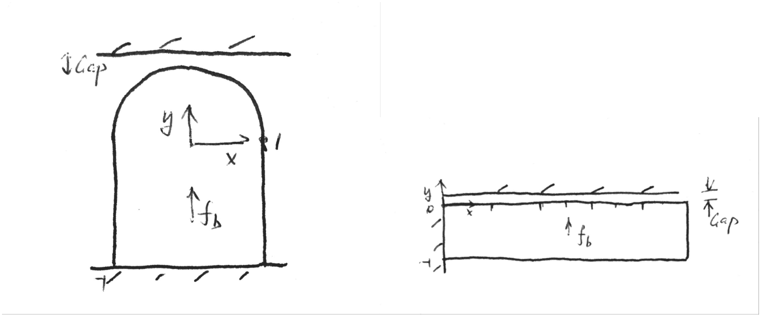


Figure: example of contact problem

1.2 Introduction - simplification assumptions

- small deformation - linear elasticity
- homogeneous, isotropic
- frictionless, no stick
- node-to-node contact, single body contact
- Optional: static

Recall the elasticity problem

$$\mathrm{div}\sigma = f_b, \quad + \text{boundary conditions} \tag{1}$$

in discrete FEM expression:

$$\mathbf{K}(\mathbf{u}) - \mathbf{f}^{ext} + \mathbf{G}^c(\mathbf{u}) = 0 \tag{2}$$

$\mathbf{G}^c(\mathbf{u})$ is the contact induced term. It varies according to contact search algorithm we choose. subject to contact constrains

$$\begin{aligned} \mathbf{u} - \mathbf{g} &\leq \mathbf{0} \\ \lambda &\leq \mathbf{0} \\ (\mathbf{u} - \mathbf{g})\lambda &= \mathbf{0} \end{aligned} \tag{3}$$

2.1 algorithm - Lagrange multiplier method

In Lagrange multiplier method, the contact induced term is

$$\mathbf{G}^c = -\mathbf{C}^c \lambda \tag{4}$$

Therefore, (2) becomes (5)

$$\mathbf{K}(\mathbf{u}) - \mathbf{f}^{ext} - \mathbf{C}^c \lambda = 0 \tag{5}$$

wherein $\mathbf{C}^c$ is the contact indicator matrix. It's a boolean diagonal matrix whose elements are defined as:

$$c_{ij}^c = \begin{cases} 1, & \text{if the ith DOF is in contact, and } i = j \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

2.1 algorithm - Lagrange multiplier method

Since in 3, $u_i - g_i = 0$ is satisfied only in contact condition, $\lambda = 0$ is satisfied only in free condition, the contact constraint condition 3 can be rewritten in one equation

$$\mathbf{0} = \mathbf{C}^c(\mathbf{u} - \mathbf{g}) + (\mathbf{I} - \mathbf{C}^c)\lambda \tag{7}$$

We also know from 3 that, for free DOFs, $u_i - g_i < 0$, and for contact DOFs, $\lambda < 0$. Written in one equation, we have:

$$\mathbf{C}^c = \mathrm{diag}\left[\mathrm{heaviside}\left((\mathbf{I} - \mathbf{C}^c)(\mathbf{u} - \mathbf{g}) - \mathbf{C}^c\lambda\right)\right] \tag{8}$$

In total, we have three equations (5), (7), (8) to solve three variables $\mathbf{u}, \lambda, \mathbf{C}^c$. It's a complete problem.

2.1 algorithm - Lagrange multiplier method

For linear problem $\mathbf{K}(\mathbf{u}) = \mathbf{K}\mathbf{u}$, the unknown variables $(\mathbf{u}, \lambda)$ can be solved in one step

$$\begin{bmatrix} \mathbf{u} \\ \lambda \end{bmatrix} = \begin{bmatrix} \mathbf{K} & -\mathbf{C}^c \\ \mathbf{C}^c & \mathbf{I} - \mathbf{C}^c \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{f} \\ \mathbf{C}^c \mathbf{g} \end{bmatrix} \tag{9}$$

and contact indicator matrix $\mathbf{C}^c$ can be solved iteratively (10)

$$\mathbf{C}^c \leftarrow \mathrm{diag}\left[\mathrm{heaviside}\left((\mathbf{I} - \mathbf{C}^c)(\mathbf{u} - \mathbf{g}) - \mathbf{C}^c \lambda\right)\right] \tag{10}$$

2.1 algorithm - Lagrange multiplier method

Algorithm 1 Lagrange multiplier algorithm

Initialize contact assumption: $\mathbf{C}^c \leftarrow \mathbf{0}$
for over $\mathbf{C}^c$ **do**
    $\mathbf{C}_0^c \leftarrow \mathbf{C}^c$
    Solve linear Lagrange Muliplier equation (9)
    Update contact/free set according to Equation (10)
    if $\mathbf{C}_0^c = \mathbf{C}^c$ **then**
        BREAK
    end if
end for

2.1 algorithm - penalty method

In penalty method, contact related term

$$\mathbf{G}^c = \epsilon \mathbf{C}^c(\mathbf{u} - \mathbf{g}) \tag{11}$$

again $\mathbf{C}^c$ is the contact indicator matrix. penalty term $\epsilon > 0$ is a constant scalar.
(2) becomes (5)

$$\mathbf{K}(\mathbf{u}) - \mathbf{f}^{ext} + \epsilon \mathbf{C}^c(\mathbf{u} - \mathbf{g}) = 0 \tag{12}$$

Contact indicator matrix is updated as

$$\mathbf{C}^c_{k+1} \leftarrow \mathrm{diag}\left[\mathrm{heaviside}(\mathbf{u} - \mathbf{g})\right] \tag{13}$$

Algorithm 2 penalty algorithm

Initialize penalty term $\epsilon \leftarrow \epsilon_0$

Initialize contact assumption: $\mathbf{C}_0^c \leftarrow \mathbf{I}, \quad \mathbf{C}^c \leftarrow \mathbf{0}$

for over ϵ **do**

  for over $\mathbf{C}^c$ **do**

    $\mathbf{C}_0^c \leftarrow \mathbf{C}^c$

    Solve dynamic penalty equation (12)

    Update contact indicator $\mathbf{C}^c$ according to Equation (13)

    if $\mathbf{C}_0^c = \mathbf{C}^c$ **then**

      BREAK

    end if

  end for

  if converged: $norm[\mathbf{C}^c(\mathbf{u} - \mathbf{g}] <$ threshold **then**

    BREAK

  end if

  $\epsilon \leftarrow 10 \times \epsilon$

end for

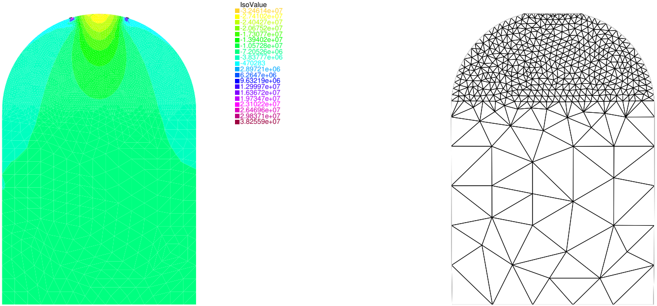


Figure: left: σ_2, right: moved mesh

3.1 search for contact segments - iteration 2

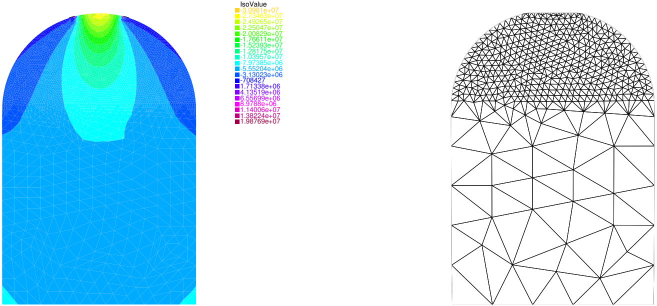


Figure: left: σ_2, right: moved mesh

3.1 search for contact segments - iteration 3

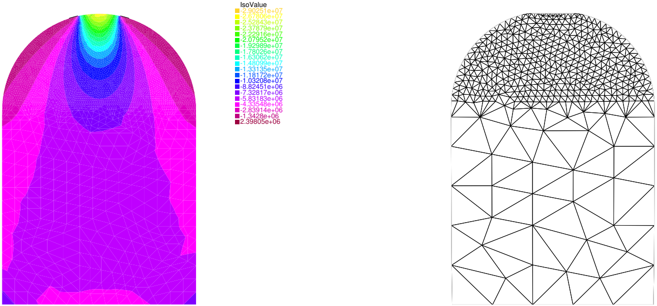


Figure: left: σ_2, right: moved mesh

The above 3 slides show how the iteration over $\mathbf{C}^c$ converges before reach the correct contact segment

The following will be the converged contact answers given different contact conditions

3.2 contact given $gap = -0.01$ m

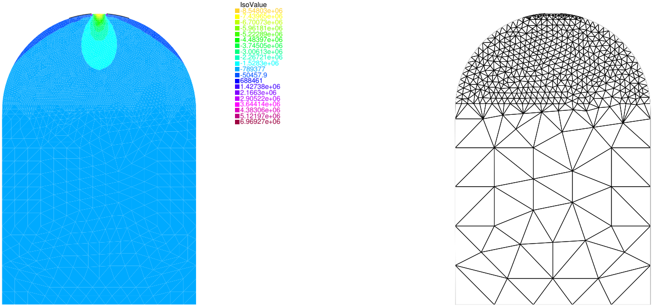


Figure: left: σ_2, right: moved mesh

3.2 contact given $gap = -0.02$ m

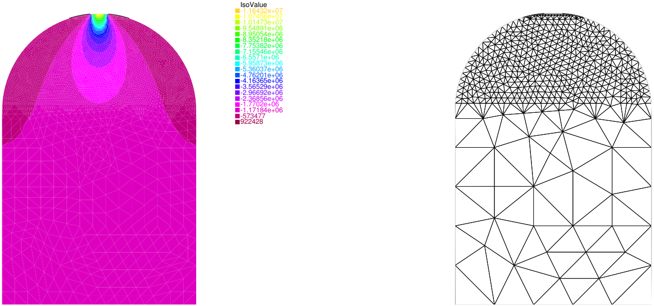


Figure: left: σ_2, right: moved mesh

3.2 contact given *gap* $= -0.03$ m

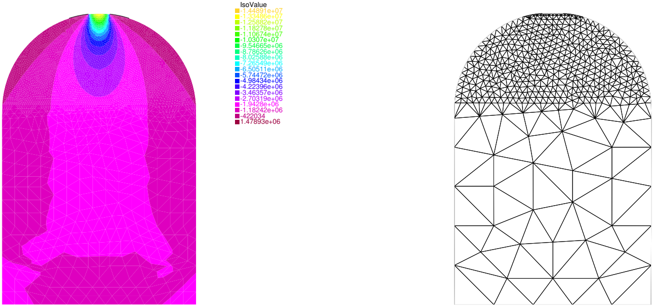


Figure: left: σ_2, right: moved mesh

3.2 contact given $gap = -0.04$ m

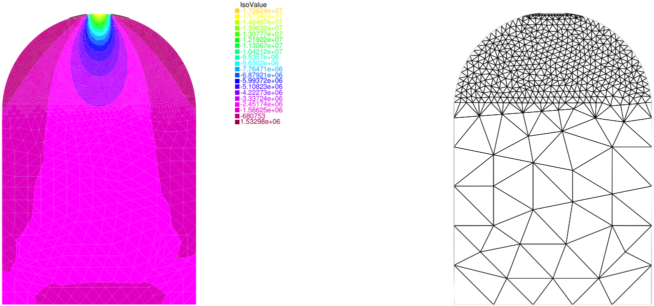


Figure: left: σ_2, right: moved mesh

3.2 contact given $gap = -0.05$ m

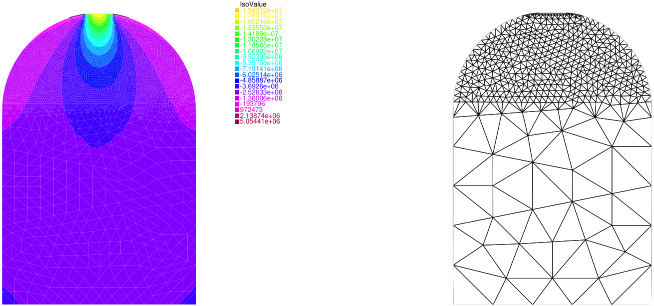


Figure: left: σ_2, right: moved mesh

3.2 contact given $gap = -0.06$ m

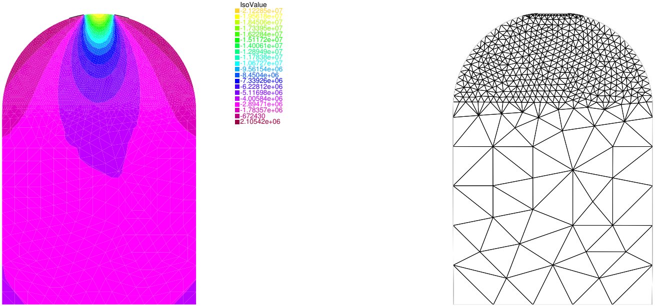


Figure: left: σ_2, right: moved mesh

3.2 contact given *gap* $= -0.07$ m

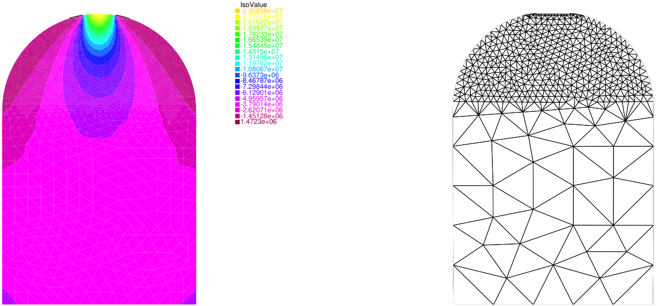


Figure: left: σ_2, right: moved mesh

3.2 contact given $gap = -0.08$ m

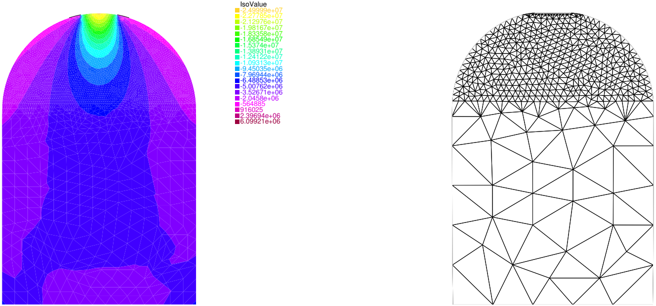
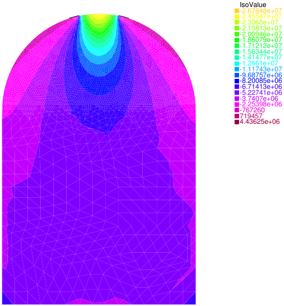


Figure: left: σ_2, right: moved mesh

3.2 contact given $gap = -0.09$ m



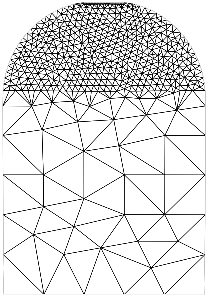


Figure: left: σ_2, right: moved mesh

3.2 contact given $gap = -0.1$ m

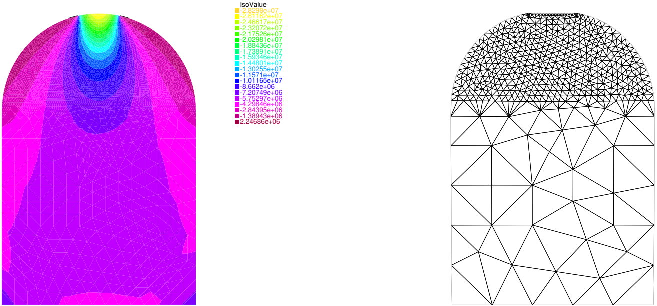


Figure: left: σ_2, right: moved mesh

Thank you!