Variables

Geometric Stiffness Matrix

Definition : FEBioSource2.9/FEBioMech/FEElasticSolidDomain.cpp, line 316

```
//! calculates element's geometrical stiffness component for integration point n
void FEElasticSolidDomain::ElementGeometricalStiffness(FESolidElement &el, matrix &ke)
```

Comments The geometric stiffness matrix is one of the two terms that is used to linearize the continuum formulation of the weak form of problem. This term is presented in Wriggers, 2008, pg. 137, eq 4.107.

$$\bar{g}_{IK} = \left(\nabla_{\bar{x}} N_I\right)^T \bar{\boldsymbol{\sigma}} \nabla_{\bar{x}} N_K$$

Notice that the terms in the above equation are derived with respect to the deformed configuration (denoted by the over-bar). This will have to be changed for the inverse formulation because otherwise, those terms will be derived with respect to the reference configuration.

Material Stiffness Matrix

Definition : FEBioSource2.9/FEBioMech/FEElasticSolidDomain.cpp, line 354

```
//! Calculates element material stiffness element matrix
void FEElasticSolidDomain::ElementMaterialStiffness(FESolidElement &el, matrix &ke)
```

Comments The material stiffness matrix is one of the two terms that is used to linearize the continuum formulation of the weak form of problem. This term is presented in Wriggers, 2008, pg. 138, eq 4.109.

$$\bar{\boldsymbol{B}}_{0I}^T \bar{\boldsymbol{D}}^M \bar{\boldsymbol{B}}_{0K}$$

Notice that the terms in the above equation are derived with respect to the deformed configuration (denoted by the over-bar). This will have to be changed for the inverse formulation because otherwise, those terms will be derived with respect to the reference configuration.

Left Cauchy-Green deformation tensor $b^{ij} = G^{AB} {F^i}_A {F^J}_B$

Definition : FEBioSource2.9/FEBioMech/FEElasticMaterial.cpp, line 113

This function defines the left Cauchy-Green deformation tensor using the deformation gradient (${F^i}_A = \frac{\partial \Phi^i}{\partial A^A}$).

$$b^{ij} = G^{AB} {F^i}_A {F^J}_B$$

Comments The inverse formulation uses the inverse deformation gradient ($f^A{}_i = \frac{\partial \varphi^A}{\partial x^i}$), therefore, the way that the left Cauchy-Green deformation tensor is calculated needs to be changed. Using the relation $(c^{-1})^{ij} = b^{ij}$, we can calculate the left Cauchy-Green deformation gradient using the inverse deformation gradient.

$$b^{ij} = (c^{-1})^{ij}$$

$$c_{ij} = G_{BA} f^A{}_i f^B{}_j$$

Calculation of b^{ij} will likely change in the code.

Deformation Gradient ($F^i{}_A$)

Definition : FEBioSource2.9/FECore/FESolidDomain.cpp, line 340

The original code defines the deformation gradient using the following equation:

$$F^i{}_A = x^i \frac{\partial\,^\alpha N}{\partial A^A}$$

Where x^i is the coordinates of an element node in the deformed configuration, and the derivative of the shape function (specified by α) is take with respect to the reference configuration (A^A).

The gradient of the shape function ($\frac{\partial\,^\alpha N}{\partial A^A}$) is calculated using this equation:

$$\frac{\partial\,^\alpha N}{\partial A^A} = \frac{\partial\,^\alpha N}{\partial \xi^\gamma} (J^{-1})^\gamma{}_A$$

Where ξ^γ indicates the basis of the isoparametric element, and $(J^{-1})^\gamma{}_A$ is the element inverse Jacobian for the element (section).

Comments There may not need to be changes made to the deformation gradient variable. There are two values that depend on the configuration, x^i and $(J^{-1})^\gamma{}_A$. In the code, the variable x^i can be taken as the node's coordinate in the reference configuration, and the code would not have to be changed. Despite causing some confusion in variable names, this variable may not need to be changed. Similarly, the calculation to define $(J^{-1})^\gamma{}_A$ may not need to be changed in the code (see section for more detail).

Element Inverse Jacobian

Declaration : FEBioSource2.9/FECore/FEElement.h, line 248,

```
vector<mat3d>        m_J0i;           //!< inverse of reference Jacobian
```

Definition : FEBioSource2.9/FECore/FESolidDomain.cpp, line 88

```
el.m_J0i[n] = mat3d(Ji);
```

Comments The element's inverse Jacobian appears to be defined when the element mesh is initialized, meaning this occurs once before the problem is started to solve.

Given that the given geometry is taken to be in the deformed configuration, the calculation of this value may not change. Technically the meaning of the value is different, so the variable name(s) may be misleading.

Code changes

References

[1] P. Wriggers. *Nonlinear finite element methods. [electronic resource]*. Springer, 2008. ISBN: 978-3-540-71001-1.