OCTA

Integrated simulation system for soft materials

Multi-Phase Dynamics Program

Muffin

version 5.1

User's Manual

- Volume V -

Multi-Phase Elasticity Simulator

Elastica

OCTA User's Group

January 01 2016

Authors of the Manual

Makoto Sasaki and Tatsuya Yamaue

Programers

Makoto Sasaki and Tatsuya Yamaue

Version 5.0 & 5.1 release

Programer, Authors of the Manual   Tatsuya Yamaue, You Iida, Taku Ozawa

Acknowledgment

This work is supported by the national project, which has been entrusted to the Japan Chemical Innovation Institute (JCII) by the New Energy and Industrial Technology Development Organization (NEDO) under METI's Program for the Scientific Technology Development for Industries that Creates New Industries.

This work is also partially supported by CREST-JST (Japan Science and Technology Agency) from 2003FY.

Contents

List of Figures

Chapter 1

Theoretical background of Elastica

A multiphase linear elastic simulator "Elastica" has functions as follows.

- Static analysis of three dimensional linear elastic materials with multiphase structure under a stress and distortion.
- A setup of an isotropic elasticity and anisotropic elasticity (axisymmetric anisotropy) is possible for every component.

1.1 Basic equations of Elastica

The multiphase linear elasticity simulator Elastica has the functions as follows.

- Possible to treat a system in which two or more materials with different elastic properties (elastic parameter) are mixed.
- An anisotropic elastic material can be treated. The system in which the component of isotropic elasticity and anisotropic elasticity are mixed can also be dealt with.
- The deformation of the linear elastic material is calculated by the three-dimensional and two-dimensional finite element method.
- Possible to use arbitrary shape of tetrahedrons as a mesh. The mesh can be generated internally or can be inputted that created by a mesh generation tool of Muffin (Milk) as a Delaunay mesh, and you can input mesh data generated for other finite element method programs.
- As the method of the specification of a deformation, a volume force, a surface stress and a fixed displacement boundary conditions, and a volume expansion ratio can be used.
- Spatial distributions of the the displacement vector and distortion energy are obtained as calculation results.

1.1.1 Principle of the calculation

The deformation free energy of a linear elastic material can be written as a functional of the displacement vector distribution $\boldsymbol{u}(\boldsymbol{x})$ [1]

$$F[\boldsymbol{u}(\boldsymbol{x})] = \int_V d^d x \frac{1}{2} D_{ijkl}(\boldsymbol{x}) e_{ij} e_{kl} - \int_V d^d x \rho(\boldsymbol{x}) g_i u_i(\boldsymbol{x}) - \int_{S_t} d^{d-1} x T_i(\boldsymbol{x}) u_i(\boldsymbol{x}), \tag{1.1}$$

where $d^d x$ and $d^{d-1}x$ are the volume and surface element, respectively. The rule of taking the sum for the same subscript of a tensor or vector is used. e_{ij} is a strain tensor calculated from the displacement vector. The strain tensor is calculated in the range of a very small deformation as follows:

$$e_{ij} = \frac{1}{2}\left(\frac{\partial u_j}{\partial x_i} + \frac{\partial u_i}{\partial x_j}\right), \tag{1.2}$$

$\rho(\boldsymbol{x})$ is the mass density, $\boldsymbol{g}$ is the gravity acceleration. $\boldsymbol{T}(\boldsymbol{x})$ is a surface stress (load) applied per unit area on the surface. S_t is the surface where the surface load is given. d is the dimensionality of space. The elasticity-tensor D_{ijkl} is the 4th rank elastic modulus. Because the strain tensor e_{ij} is symmetric for i and j, it is useful to define D_{ijkl} to be symmetric for the exchange of subscripts (i, j) and (k, l). The number of the components which are not zero among D_{ijkl} changes according to the degree of the anisotropy of an elastic material. In the case of an isotropic elastic material, the number of non-zero elements of D_{ijkl} is two, and the equation of free energy becomes as

$$\begin{aligned} F[\boldsymbol{u}(\boldsymbol{x})] &= \int_V d^d x \left\{ G(\boldsymbol{x}) \left(e_{ij} - \frac{1}{d}\delta_{ij} e_{ll} \right)^2 + \frac{K(\boldsymbol{x})}{2}(e_{ll})^2 - K(\boldsymbol{x})\alpha(\boldsymbol{x}) e_{ll} \right\} \\ &\quad - \int_V d^d x \rho(\boldsymbol{x}) g_i u_i(\boldsymbol{x}) - \int_{S_t} d^{d-1} x T_i u_i(\boldsymbol{x}), \end{aligned} \tag{1.3}$$

where $K(\boldsymbol{x})$ is the bulk modulus, $G(\boldsymbol{x})$ is the shear modulus and $\alpha(\boldsymbol{x})$ is the volume expansion ratio.

In dealing with the system of a mixture of materials with different elastic parameters, D_{ijkl} is calculated from the elasticity-tensor D^{α}_{ijkl} and a volume fraction field $\Psi^{\alpha}(\boldsymbol{x})$ of component α which constitutes a material as

$$D_{ijkl}(\boldsymbol{x}) = \sum_{\alpha} D^{\alpha}_{ijkl} \Psi^{\alpha}(\boldsymbol{x}). \tag{1.4}$$

The deformation of an elastic material is determined by a displacement vector distribution $\boldsymbol{u}(\boldsymbol{x})$ which gives the minimum of the free energy F. The minimum of the free energy is defined as a stationary point for a virtual displacement vector distribution $\delta\boldsymbol{u}(\boldsymbol{x})$.

$$\begin{aligned} \delta F[\delta \boldsymbol{u}] &= \int_V d^d x D_{ijkl}(\boldsymbol{x}) e_{kl} \delta e_{ij} - \int_V d^d x \rho(\boldsymbol{x}) g_i \delta u_i - \int_{S_t} d^{d-1} x T_i(\boldsymbol{x}) \delta u_i \\ &= \int_V d^d x D_{ijkl}(\boldsymbol{x}) \frac{\partial u_k}{\partial x_l} \frac{\partial}{\partial x_j} \delta u_i - \int_V d^d x \rho(\boldsymbol{x}) g_i \delta u_i - \int_{S_t} d^{d-1} x T_i(\boldsymbol{x}) \delta u_i \\ &= 0, \end{aligned} \tag{1.5}$$

where $\delta F[\delta\boldsymbol{u}]$ is the difference of the free energy between the situations affecting a displacement vector distribution $\boldsymbol{u}(\boldsymbol{x})$ and affecting $\boldsymbol{u}(\boldsymbol{x}) + \delta\boldsymbol{u}(\boldsymbol{x})$.

In the case of an isotropic elastic material, the equation (1.5) is simplified as

$$\begin{aligned} \delta F[\delta \boldsymbol{u}] &= \int_V d^d x \left\{ 2G(\boldsymbol{x}) \left(\frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) - \frac{1}{d} \delta_{ij} \frac{\partial u_l}{\partial x_l} \right) + \frac{K(\boldsymbol{x})}{2} \delta_{ij} \frac{\partial u_l}{\partial x_l} - K(\boldsymbol{x})\alpha(\boldsymbol{x}) \right\} \frac{\partial \delta u_i}{\partial x_j} \\ &\quad - \int_V d^d x \rho(\boldsymbol{x}) g_i \delta u_i - \int_{S_t} d^{d-1} x T_i(\boldsymbol{x}) \delta u_i \\ &= 0 \end{aligned} \tag{1.6}$$

Here, the following relation is used:

$$\left(e_{ij} - \frac{1}{d} \delta_{ij} e_{ll} \right) \delta_{ij} = 0. \tag{1.7}$$

The stress tensor is defined as

$$\sigma_{ij} = \frac{\partial F}{\partial e_{ij}} = 2G(\boldsymbol{x})(e_{ij} - \frac{1}{d}\delta_{ij} e_{ll}) + K(\boldsymbol{x})\delta_{ij} e_{ll} - K(\boldsymbol{x})\alpha(\boldsymbol{x})\delta_{ij}. \tag{1.8}$$

1.1.2 Discretization by the finite element method

Using the finite element method, the force balance equation (1.5) or (1.6) can be solved. A component of the displacement vector in a finite element mesh is expressed by a linear combination of a linear interpolation function $L_I(\boldsymbol{x})$ which has the value 1 at node I which constitutes the element.

$$u_i(\boldsymbol{x}) = \sum_I L_I(\boldsymbol{x}) u_i^I \tag{1.9}$$

In the free energy equation expressed by this displacement vector, we can get a linear equations for an unknown variable u_i^I, by imposing the following condition

$$\frac{\delta F}{\delta u_i^I} = 0. \tag{1.10}$$

For an isotropic elastic material, the linear equation for an element e becomes as

$$\begin{aligned} V_e \sum_J \left[\sum_j G_e \left\{ (\nabla_i L_J \cdot \nabla_j L_I) u_j^I + (\nabla_j L_J \cdot \nabla_j L_I) u_j^J \right\} + \left(K_e - \frac{2}{d} G_e \right) \sum_k \left(\nabla_k L_J \cdot \nabla_i L_I \right) u_k^J \right] \\ = K_e \alpha_e (\nabla_i L_I) + \sum_J \rho_i^J g_i \left[\int_e d^d x L_I L_J \right] + \sum_J T_i^J \left[\int_e d^{d-1} x L_I L_J \right], \end{aligned} \tag{1.11}$$

where V_e is the volume of the element, K_e, G_e and α_e are the element averages of the bulk modulus, the shear modulus and the volume expansion ratio, respectively, and J is the index of nodes which constitutes the element. Quantities calculated in parenthesis [...] on the left hand side are components of a matrix for the linear equation. The actual equations are obtained by addition of both sides of this equation over all the elements to which node I belongs. A displacement vector is calculated by solving the obtained linear equations using the conjugate gradient method.

1.1.3 Treatment of anisotropic elastic material

The anisotropy of an elastic material appears according to the type of a crystal. For example,

- axisymmetric:

  It has an anisotropy in a specific spatial axis direction, besides isotropic for perpendicular directions.

- orthorhombic:

  It has a different elastic modulus for each of three orthogonal spatial axis directions.

Elastica can treat materials with axisymmetric anisotropy. The direction of a main axis can be given arbitrarily. The elastic coefficient with an orthotropy or a higher asymmetry is expressed as the coefficient matrix C which generally connects stress-tensor σ_{ij} and strain-tensor e_{ij} as follows.

$$\begin{bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{zz} \\ \sigma_{yz} \\ \sigma_{zx} \\ \sigma_{xy} \end{bmatrix} = \begin{bmatrix} C_{11} & C_{12} & C_{13} & 0 & 0 & 0 \\ C_{12} & C_{22} & C_{23} & 0 & 0 & 0 \\ C_{13} & C_{23} & C_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & C_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & C_{66} \end{bmatrix} \begin{bmatrix} e_{xx} \\ e_{yy} \\ e_{zz} \\ e_{yz} \\ e_{zx} \\ e_{xy} \end{bmatrix} \tag{1.12}$$

In the case of an axisymmetric elastic material, the C matrix can be expressed using five independent parameters n, l, k, m, and $\mu(s)$ as follows (when the axis of anisotropy is set as x-axis).

$$\begin{bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{zz} \\ \sigma_{yz} \\ \sigma_{zx} \\ \sigma_{xy} \end{bmatrix} = \begin{bmatrix} n & l & l & 0 & 0 & 0 \\ l & k+m & k-m & 0 & 0 & 0 \\ l & k-m & k+m & 0 & 0 & 0 \\ 0 & 0 & 0 & m & 0 & 0 \\ 0 & 0 & 0 & 0 & \mu & 0 \\ 0 & 0 & 0 & 0 & 0 & \mu \end{bmatrix} \begin{bmatrix} e_{xx} \\ e_{yy} \\ e_{zz} \\ e_{yz} \\ e_{zx} \\ e_{xy} \end{bmatrix} \tag{1.13}$$

Although the equation (1.13) serves as an expression in a case that the principal axis of an anisotropy is in agreement with a specific axis of coordinates. Generally the axis of coordinates at a time of a simulation is not necessarily in agreement with a principal axis. Moreover, we may encounter a material which is a mixture of components having different axes of an anisotropy. So, in Elastica, directions of axis of an anisotropy can be taken arbitrarily. Making n_i the unit vector along a principal axis of axisymmetric anisotropic material, the elastic energy density f can be expressed as follows.

$$
\begin{aligned}
f &= D_1(e_{ii})^2 + D_2(n_i n_j e_{ij})^2 + D_3(e_{ll} \cdot n_i n_j e_{ij}) + D_4 n_l e_{il} \cdot n_k e_{ik} + D_5 e_{ij} e_{ij} \\
&= \frac{1}{2}\left[2D_1\delta_{ij}\delta_{kl} + 2D_2 n_i n_j n_k n_l + D_3(\delta_{ij} n_k n_l + \delta_{kl} n_i n_j) + D_4(\delta_{ik} n_j n_l + \delta_{jl} n_i n_k) + D_5(\delta_{ik}\delta_{jl} + \delta_{jk}\delta_{il})\right] \\
&\quad e_{ij} e_{kl} \\
&\equiv \frac{1}{2} D_{ijkl} e_{ij} e_{kl} \qquad (1.14)
\end{aligned}
$$

Coefficients D_1, D_2, D_3, D_4, and D_5 can be connected with five elastic parameters in the equation (1.13). Using this expression, we can calculate a deformation of the linear elastic material having arbitrary anisotropy axis directions. The following relations are obtained by correspondence of the coefficient of the equation (1.13) and expressions $\sigma_{ij} = \frac{\delta f}{\delta e_{ij}}$ under $n_x = 1, n_y = n_z = 0$.

$$
\begin{aligned}
D_1 &= (k - m)/2 \qquad (1.15) \\
D_2 &= (n + k - m)/2 - l - \mu \qquad (1.16) \\
D_3 &= l - k + m \qquad (1.17) \\
D_4 &= \mu - m \qquad (1.18) \\
D_5 &= m \qquad (1.19)
\end{aligned}
$$

A finite element discretization equation for u_i^I using the elastic energy can be written in the following form as a generalized form of the equation (1.11). This equation can be applied to both isotropic and anisotropic elastic materials.

$$
\begin{aligned}
& V_e \sum_J \left[\sum_j \sum_l D_{ijkl} \nabla_j L_I \nabla_l L_J \right] u_k^J \\
&= \sum_J \rho_i^J g_i \left[\int_e d^d x L_I L_J \right] + \sum_J T_i^J \left[\int_e d^{d-1} x L_I L_J \right] \qquad (1.20)
\end{aligned}
$$

1.1.4 Treatment of mixed system of components with different symmetry

The Elastica can treat a mixed system of isotropic elastic materials and anisotropic ones. In the Elastica, the elasticity tensor is treated as a weight averaged elasticity tensor of each component by the volume fraction of each component as the equation (1.4).

$$
f = \frac{1}{2} \sum_\alpha \left(D_{ijkl}^\alpha \Psi^\alpha \right) e_{ij} e_{kl} \qquad (1.21)
$$

Therefore, in generation of a finite-element matrix, the matrix for the elasticity tensor $D_{ijkl}^\alpha \Psi_\alpha$ is generated for every component, and the matrix for the mixture is generated by taking the sum of those matrices. So Elastica can treat any mixture of the materials with any anisotropy.

1.1.5 Spatial distribution of anisotropy axis

The elasticity tensor in an element is obtained from the anisotropic elastic tensor $D_{ijkl}(\boldsymbol{n}^K)$ expressed as a function of the anisotropy axis vector $\boldsymbol{n}^K$ at the node K and volume fraction Ψ_K at the node K by the interpolation as

$$
D_{ijkl}(\boldsymbol{x}) = \sum_K L_K(\boldsymbol{x}) D_{ijkl}(\boldsymbol{n}^K) \Psi_K. \qquad (1.22)
$$

The left-hand side of the equation (1.20) is rewritten as

$$
\begin{aligned}
& \sum_K \left\{ \sum_J \left[\sum_j \sum_l \left(D_{ijkl}(\boldsymbol{n}^K) \Psi_K \int_e d^d x L_K(\boldsymbol{x}) \right) \nabla_j L_I \nabla_l L_J \right] \right\} u_k^J \\
&= V_e \sum_K \left\{ \sum_J \left[\sum_j \sum_l \left(\frac{1}{p_e} D_{ijkl}(\boldsymbol{n}^K) \Psi_K \right) \nabla_j L_I \nabla_l L_J \right] \right\} u_k^J \qquad (1.23)
\end{aligned}
$$

where p_e is the number of the nodes which constitute an element. This equation means that we can make matrix for a mixture by taking a summation over node K for matrices created by using $\frac{1}{p_e}D_{ijkl}(\boldsymbol{n}^K)\Psi_K$ as an anisotropic elasticity tensor at each node K which may have a different anisotropy axis.

1.1.6 Boundary condition

The boundary condition which can be dealt with by Elastica is a "fixed displacement boundary condition" and a "surface load condition". When the surface of the system is composed as $S = S_u + S_t$,

- on fixed displacement boundary S_u :
$$u_i = \bar{u}_i \tag{1.24}$$
- on surface load boundary S_t :
$$\sigma_{ij} n_j = T_i \tag{1.25}$$

For a free surface, the surface load is zero. In the finite element method, such a free surface can be treated naturally by doing nothing on the surface.

1.1.7 Processing of fixed displacement boundary condition

In the Elastica, the fixed boundary condition is realized by the penalty number method (Yagawa and Yoshimura [2], p.50). In this method, even for the nodes which constitutes fixed displacement boundary condition, a matrix composition process is performed according to the equation (1.11) as if the displacements on the nodes are unknown. Next, taking a big number α as a penalty number, α is added to the diagonal elements of the matrix for displacements on the fixed displacement boundary u_i^I, and $\alpha \bar{u}_i^I$ is added to the corresponding components of a right-hand-side vector. The fixed displacement boundary condition can be taken in by solving the obtained linear equation.

Chapter 2

Sample problems of Elastica

2.1 Sample problems of Elastica

In this section, we show 11 samples of Elastica. Input UDF files and output files corresponding to these applications are prepared in a directory of the Muffin distribution as a sub-directory according to the problem.

2.1.1 Application 01 : Simple shear deformation

- 3D
  1) Shear deformation of an isolated cube. (non-periodic) (EX01/EX01_cube_in.udf)
  Usage : muffin5e_elastica -I EX01_cube_in.udf -O EX01_cube_ou.udf
  2) Simple shear deformation. (periodic) (EX01/EX01_bulk_in.udf)
  Usage : muffin5e_elastica -I EX01_bulk_in.udf -O EX01_bulk_ou.udf

- 2D
  1) Shear deformation of an isolated square. (non-periodic) (EX01/2D/EX01_square2d_in.udf)
  Usage : muffin5e_elastica -I EX01_square2d_in.udf -O EX01_square2d_ou.udf
  2) Simple shear deformation. (periodic) (EX01/2D/EX01_bulk2d_in.udf)
  Usage : muffin5e_elastica -I EX01_bulk2d_in.udf -O EX01_bulk2d_ou.udf

2.1.2 Application 02 : Load to a long rod

- 3D : Load to a long rod (EX02/EX02_in.udf)
  Usage : muffin5e_elastica -I EX02_in.udf -O EX02_ou.udf

- 2D : Load to a long square (EX02/2D/EX02_2d_in.udf)
  Usage : muffin5e_elastica -I EX02_2d_in.udf -O EX02_2d_ou.udf

2.1.3 Application 03 : Shear Deformation of two components phase separation system

Shear Deformation of two components phase separation system, which are calculated by PhaseSeparation_FEM using Flory Huggins model without flow. (EXAMPLE 03 of PhaseSeparation_FEM)

1. Open EX03_in.udf, and using the action "import_fields", import EX03/EX03_ou.udf of PhaseSeparation_FEM.

2. Calculation
   Usage : muffin5e_elastica -I EX03_in.udf -O EX03_ou.udf

2.1.4 Application 04 : Compress two components lamellar phase calculated by SUSHI

Compress two components lamellar phase, which are imported from 1D calculation of blend_uot.udf of SUSHI.

1. Open EX04_in.udf, and using the action "import_fields_1d", import blend_uot.udf of SUSHI.
2. Calculation
   Usage : muffin5e_elastica -I EX04_in.udf -O EX04_ou.udf

2.1.5 Application 05 : Load to two component cylinder phase calculated by SUSHI

Load to upper surface (Z-max surface) of two component cylinder like phase separation body, which are imported from susi3_cylinder3D_uot.udf of SUSI3.

1. Open EX05_in.udf, and using the action "import_fields", import susi3_cylinder3D_uot.udf of SUSHI.
2. Calculation
   Usage : muffin5e_elastica -I EX05_in.udf -O EX05_ou.udf

2.1.6 Application 06 : Shear deformation of a cube including sphere like rubber. I

Shear deformation of a material which includes one sphere like rubber. Phase separation structure is generated by initial procedure, named "ONE_SPHERE".
Usage : muffin5e_elastica -I EX06_in.udf -O EX06_ou.udf

2.1.7 Application 07 : Shear deformation of a cube including sphere like rubber. II

Shear deformation of a material which includes one sphere like rubber. Grid is generated using Milk and phase separation structure is applied by PartialRegionCondition (region_condition[]).

1. Generate a mesh by Milk.
   Usage : milk5_3d -I EX07_milk_onesphere_in.udf -O EX07_milk_onesphere_ou.udf
2. Open EX07_in.udf, and using the action "import_mesh", import EX07_milk_onesphere_ou.udf.
3. Calculation
   Usage : muffin5e_elastica -I EX07_in.udf -O EX07_ou.udf

2.1.8 Application 08 : Surface modulation of a thin film lamellar phase

Surface modulation by volume expansion and shrink of a thin film of periodic two components lamellar phase, which are imported from blend_uot.udf of SUSHI.

1. Open EX08_in.udf, and using the action "import_fields_1d", import blend_uot.udf of SUSHI.
2. Calculation
   Usage : muffin5e_elastica -I EX08_in.udf -O EX08_ou.udf

2.1.9 Application 09 : Bending of bi-metal

Bending of a bilayer film by volume expansion and shrink of each layer. Mesh is prepared using only the mesh generation of Elastica.

1. Generate a mesh :
   Usage : muffin5e_elastica -I EX09_meshgenerate_in.udf -O EX09_meshgenerate_ou.udf
2. Open EX09_in.udf, and using the action "import_mesh", import EX09_meshgenerate_ou.udf.
3. Reload the UDF, and load and run the python command "make_bilayer.py" to generate bilayer structure.
4. Calculation
   Usage : muffin5e_elastica -I EX09_in.udf -O EX09_ou.udf

2.1.10 Application 10 : Shrinking of a thin film on hard substrate with notch

Shrinking of a film (K=G=1) contacted with hard substrate (K=G=1000) with a hard rectangular notch. Mesh is prepared using only the mesh generation of Elastica.

1. Generate a mesh :
   Usage : muffin5e_elastica -I EX10_meshgenerate_in.udf -O EX10_meshgenerate_ou.udf
2. Open EX10_in.udf, and using the action "import_mesh", import EX10_meshgenerate_ou.udf.
3. Reload the UDF, and load and run the python command "make_bilayer_with_notch.py" to generate bilayer structure with notch.
4. Calculation
   Usage : muffin5e_elastica -I EX10_in.udf -O EX10_ou.udf
5. Analyze the stress distribution along a axis by the python command "stress_analysis.py".

2.1.11 Application 11 : Deformation of a half spherical shell under the negative pressure

Deformation of a half spherical membrane (K=G=1) under the negative pressure. Bottom line (surface) is clumped on the substrate. Mesh is prepared using Milk (EX12 of MILK).

1. Generate a mesh :
   Usage : milk5_3d -I EX11_milk_in.udf -O EX11_milk_ou.udf
2. Open EX11_in.udf, and using the action "import_mesh", import EX11_milk_ou.udf.
3. Calculation
   Usage : muffin5e_elastica -I EX11_in.udf -O EX11_ou.udf

2.2 Detail procedures of analysis using Elastica

Detail procedures of analysis are explained for application 3 and application 5.

2.2.1 Application 3: Isotropic linear elastic analysis by a morphology from PhaseSeparation simulator

[Problem setting]

1. Input a morphology of the phase separation structure (with no flow) by the Flory-Huggins free energy of two component system (50:50) calculated by the PhaseSeparation_FEM simulator as shown in Fig.2.1.
2. Both components have an isotropic elasticity.
3. Set a bulk modulus and a shear modulus of the first component (color red) to 10, and set those of the second component (color blue) to 40.
4. Apply a shear by putting displacement on the upper boundary perpendicular to the Y-axis into +X direction and the lower boundary into -X, and simulate deformation.
5. The color contour of the state of a deformation and a free energy are drawn.

[Sample UDF files]

- Output UDF file from PhaseSeparation_FEM:
  `MUFFIN5/sample/muffin5ebeta/PhaseSeparation_FEM/EX03/EX03_out.udf`
- Input UDF file: `MUFFIN5/sample/muffin5ebeta/Elastica/EX03/EX03_in.udf`
- Output UDF file: `MUFFIN5/sample/muffin5ebeta/Elastica/EX03/EX03_ou.udf`

[Making input UDF file]

1. Open an input UDF of Elastica `MUFFIN5/sample/muffin5ebeta/Elastica/EX03/EX03_in.udf` on GOURMET.

2. Pop-up another Editor window by selecting GOURMET's "NEW WINDOW" menu item, and open an output UDF of the example 3 of PhaseSeparation_FEM simulator `MUFFIN5/sample/muffin5ebeta/PhaseSeparation_FEM/EX03/EX03_out.udf`. You can view a phase separation structure as shown in Fig.2.1 by drawing the volume fraction field data of the last record (No.5). To draw the figure, right-click on "EX03_out.udf" in the left window of GOURMET, and select and execute "show_field" from the pop-up menu (Select z-sections as region).

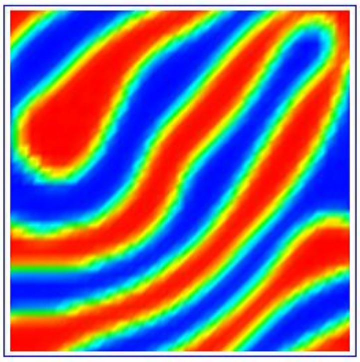

Figure 2.1: Elastica (linear elasticity) example 3: Morphology as an input (from example 3 of PhaseSeparation_FEM)

3. We convert the morphology data.
   In the editor window for `MUFFIN5/sample/muffin5ebeta/Elastica/EX03/EX03_in.udf`, right-click on "EX03_in.udf" in the left window of GOURMET, and select and execute "import_fields" from the pop-up menu.

4. Then set parameters for conversion in the dialog. Set "import_udf_path" to the output UDF of PhaseSeparation_FEM simulator, `MUFFIN5/sample/muffin5ebeta/PhaseSeparation_FEM/EX03/EX03_ou.udf`, and set "import_record_no" to 5 as shown in Fig.2.2. You can select the file by right-click on the text box. Set "save_as" to `EX03_2_in.udf`, because you already opened `EX03_in.udf`.

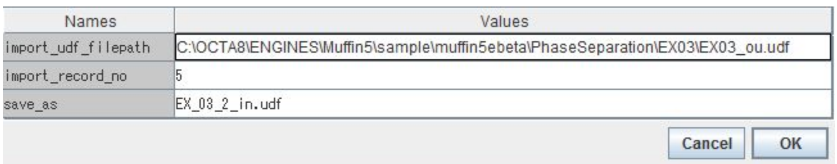

Figure 2.2: Elastica (linear elasticity) example 3: Zooming parameters input dialog

5. After finishing conversion, file `EX03_2_in.udf` will be created but we still use `EX03_in.udf` because this file already has converted data.

6. Next, set parameters. You need to change "mesh parameters"(parameter.mesh_parameter) as shown in Fig.2.3.This is because the mesh of Phaseseparation_FEM uses periodic boundary condition but following calculation of Elastica does not use it.

7. Set those in "physical parameters"(parameter.physical_parameter[]).Input as the following table.

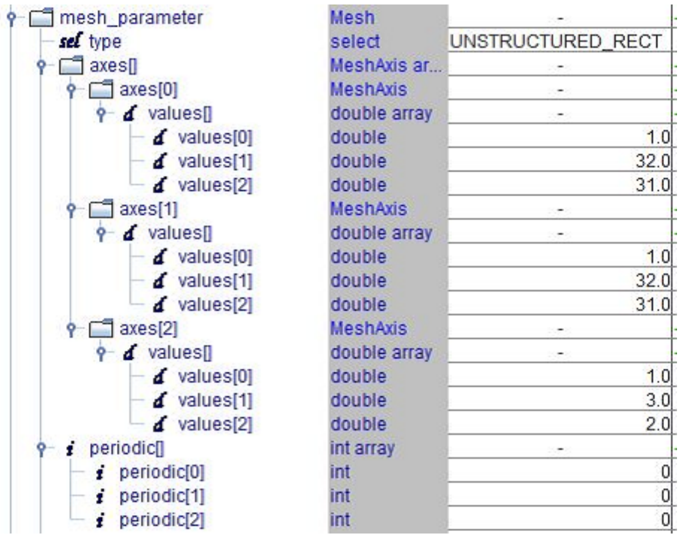

Figure 2.3: Elastica (linear elasticity) example 3: Mesh parameter

Parameter	Value	Meanings
NUMBER_OF_COMPONENTS	2	2 component system
MODULUS_ANISOTROPY_0	Isotropic	1st component has isotropic elasticity
MODULUS_ANISOTROPY_1	Isotropic	2nd component has isotropic elasticity
BULK_MODULUS	$[10, 40]$(array)	Bulk modulus. 10 for 1st component, 40 for 2nd
SHEAR_MODULUS	$[10, 40]$(array)	Sear modulus. 10 for 1st component, 40 for 2nd

8. Next set boundary conditions. The partial region (boundary) condition part (region_condition.condition[]) should be specified as follows.

Condition	Partial region	field	condition name	value
condition1	BOUNDARY_VERTEX_YMIN	Displacement	D_VEC	$[-5.0, 0.0, 0.0]$
condition2	BOUNDARY_VERTEX_YMAX	Displacement	D_VEC	$[+5.0, 0.0, 0.0]$

These conditions mean applying $+5.0$ displacement in the X-direction on the upper surface perpendicular to the Y-axis (BOUNDARY_VERTEX_YMAX). -5.0 displacement on the bottom X-direction on the upper surface perpendicular to the Y-axis (BOUNDARY_VERTEX_YMIN). It is giving a shear deformation. (D_VEC is the meaning of the Dirichlet conditions of displacement)

[Running the Simulator]

Execute Elastica by the command
muffin5e_elastica -I EX03_in.udf -O EX03_ou.udf

[Drawing of the result]

1. Open the resultant output UDF file on GOURMET, move to a record after deformation (record No.1).

2. Let's try viewing a color contour output of the volume fraction field. Right-click on "EX03_ou.udf" in the left window of GOURMET, and select and execute "show_field" from the pop-up menu. You will get a contour figure like Fig.2.4 (the left part). The red part is the softer first component and the blue part is the harder second component. Comparing with the morphology view of the record 0 before deformation, we can see that the hard blue part retains its shape, but the soft red part has larger deformation.

3. Next, let's try viewing a color contour output of the free energy. Right-click on "EX03_ou.udf" in the left window of GOURMET, execute "show_field" from the pop-up menu and select "FreeEnergy" field. You will get a contour figure like Fig.2.4 (the right part). It can be observed that the free energy (strain) has concentrated on the interface between the two phases.

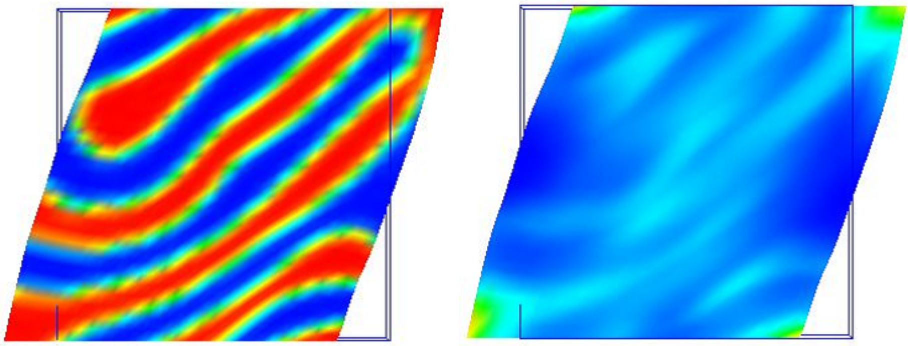

Figure 2.4: Elastica (linear elasticity) example 3: Morphology after displacement and free energy.

2.2.2 Application 5: Input of a morphology from SUSHI and anisotropic linear elasticity analysis

[Problem setting]

1. Input a morphology of the cylinder structure of phase separation of the di-block copolymer system (the ratio of length 5:15) calculated by the SUSHI simulator. Cylinder structure is parallel to the Z-axis.
2. Both components are axisymmetric anisotropic elastic materials, and make the axis of an anisotropy parallel to the Z-axis (axis of the cylinder morphology structure).
3. The first component (component which compose red cylinders) are hard in the Z-direction and soft in X- and Y-direction. Specifically, five elastic parameters (see theory section 1.1.3) are set into $(n, l, k, m, \mu) = (100, 20, 10, 5, 5)$. On the other hand, the second component (blue sea) is soft in Z-direction, and hard in X- and Y-direction. Five elastic parameters are set into $(n, l, k, m, \mu) = (20, 20, 100, 5, 5)$.
4. Apply a load on the upper face perpendicular to the Z-axis in +Z direction. The bottom boundary perpendicular the Z-axis is fixed without displacement, so the material is stretched in Z direction.
5. The color contour of the state of a deformation and a strain energy is drawn.

[Sample UDF files]

- An output UDF file from SUSHI: `MUFFIN5/sample/muffin5ebeta/Elastica/EX05/sushi3_cylinder3D_uot.udf`
- Input UDF file: `MUFFIN5/sample/muffin5ebeta/Elastica/EX05/EX05_in.udf`
- Output UDF file: `MUFFIN5/sample/muffin5ebeta/Elastica/EX05/EX05_out.udf`

[Making input UDF file]

1. Open an input UDF of Elastica `MUFFIN5/sample/muffin5ebeta/Elastica/EX05/EX05_in.udf` on GOURMET.
2. We convert the morphology data.
In the editor window for `MUFFIN5/sample/muffin5ebeta/Elastica/EX05/EX05_in.udf`, right-click on "EX05_in.udf" in the left window of GOURMET, and select and execute "import_fields" from the pop-up menu.
3. Then set parameters for conversion in the dialog. Set "import_udf_path" to the output UDF of SUSHI `MUFFIN5/sample/muffin5ebeta/Elastica/EX05/susi3_cylinder3D_uot.udf`, and set "import_record_no" to 1. You can select the file by right-click on the text box. Set "save_as" to `EX05_2_in.udf`, because you already opened `EX05_in.udf`.
4. After finishing conversion, file `EX05_2_in.udf` will be created but we still use `EX05_in.udf` because this file already has converted data.

5. Next, set parameters. You need to set those in "physical parameters" (parameter.physical_parameter[]). Input as the following table.

Parameter	Value	Meanings
NUMBER_OF_COMPONENTS	2	2 component system
MODULUS_ANISOTROPY_0	Axisymmetric	1st component has axisymmetric anisotropy
MODULUS_ANISOTROPY_1	Axisymmetric	2nd component has axisymmetric anisotropy
MODULUS_0	$[100, 20, 10, 5, 5]$(array)	moduli of the 1st component
MODULUS_1	$[20, 20, 100, 5, 5]$(array)	moduli of the 1st component
MODULUS_AXIS_0_0	$[0.0, 0.0, 1.0]$(array)	anisotropy axis of the 1st component(Z-axis)
MODULUS_AXIS_1_0	$[0.0, 0.0, 1.0]$(array)	anisotropy axis of the 2nd component(Z-axis)
AXISYMMETRIC_MODULUS _EVALUATION_AXES	$[1.0, 0.0, 0.0,$ $0.0, 1.0, 0.0,$ $0.0, 0.0, 1.0]$(array)	calculate free energy quadratures for averaged axisymmetric moduli evaluation for X,Y and Z direction here.

6. Next set boundary conditions. The partial region (boundary) condition part (region_condition.condition[]) should be specified as follows.

Condition	Partial region	field	condition name	value
condition1	BOUNDARY_VERTEX_ZMIN	Displacement	D_VEC	$[0.0, 0.0, 0.0]$
condition2	BOUNDARY_FACE_ZMAX	Displacement	N_LOAD	$[0.0, 0.0, 10.0]$

   These conditions mean applying load of 10.0 in the Z-direction on the upper surface perpendicular to the Z-axis (BOUNDARY_FACE_ZMAX), and fix the displacement of the lower surface perpendicular to the Z-axis (BOUNDARY_FACE_ZMIN). The string D_VEC means the Dirichlet conditions of the displacement, and N_LOAD means a Neumann condition for the displacement.

[Running the Simulator]

Execute Elastica by the command
muffin5e_elastica -I EX05_in.udf -O EX05_ou.udf

You will get numeric values like the following lines on the standard output by running the input UDF on Elastica.

```
===========================================================
====== total free energy     : 4956.54
====== strain multiple for K : 82.8955
====== strain multiple for G : 263.137
====== energy term for anisotropic modulus evaluation ===
  == axis ( 1, 0, 0 )
   coeff 0 = 165.791
   coeff 1 = 5.51128
   coeff 2 = -26.5938
   coeff 3 = 0.402436
   coeff 4 = 318.401
  == axis ( 0, 1, 0 )
   coeff 0 = 165.791
   coeff 1 = 5.71867
   coeff 2 = -26.919
   coeff 3 = 0.831829
   coeff 4 = 318.401
  == axis ( 0, 0, 1 )
   coeff 0 = 165.791
   coeff 1 = 294.217
   coeff 2 = 219.304
   coeff 3 = 28.1567
   coeff 4 = 318.401
====== free energy maximum = 2.96863 at ( 0, 15, 0 )
```

```
       displacement ( 1.26046e-13, -1.25515e-13, 3.17924e-13 )
====== free energy minimum = 0.817829 at ( 6, 7, 0 )
       displacement ( 1.17777e-14, 1.76429e-14, 6.33716e-13 )
==========================================================
```

These are free energy coefficient described in (1.14) for each axes given by the parameter AXISYMMETRIC_MODULUS_EVALUATION_AXES. These values are included in output UDF.

[Drawing the result]

1. Open the result output UDF file on GOURMET, move to a record after deformation(record No.1).
2. Let's try viewing a color contour output of the volume fraction field. right-click on "EX05_ou.udf" in the left window of GOURMET and execute "show_field" from the pop-up menu. You will get a contour figure like Fig.2.5 (the left part). The red part is first component and the blue part is the second component. Comparing with the morphology view of the record 0 before the deformation, we can see that the red cylinder part, which is hard in stretching direction but soft in the perpendicular directions, reduces its radius, and the blue part, which is soft in stretching direction but hard in the perpendicular directions, has more displacement than cylinder parts and shrinking in sides.

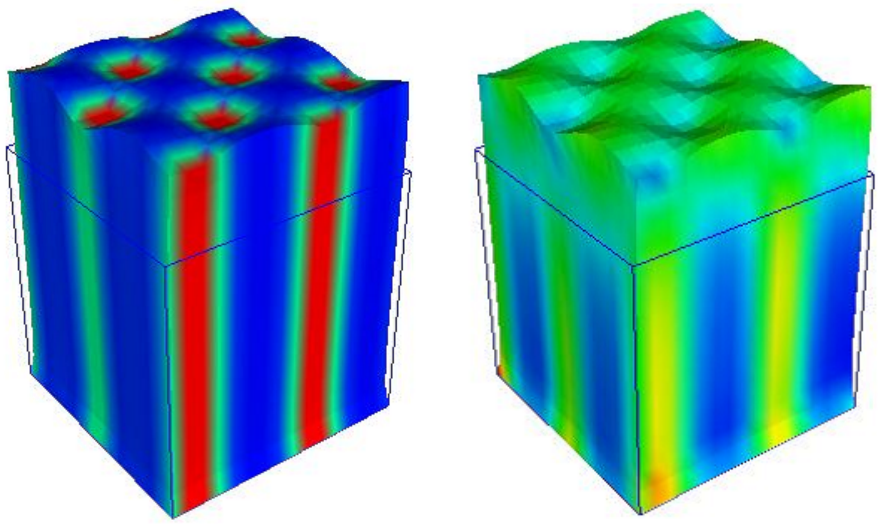

Figure 2.5: Elastica (linear elasticity) example 5: Morphology after displacement and free energy.

3. Next, let's try viewing a color contour output of the free energy. Right-click on "EX05_ou.udf" in the left window of GOURMET, and execute "show_field" from the pop-up menu and select "FreeEnergy" field. You will get a contour figure like Fig.2.5 (the right part). It can be observed that the strain has concentrated in the harder red parts near the fixed displacement face.
4. Figure 2.6 shows the free energy ditribution on the interface after the deformation, and the parameter set for the visualization. By setting isosurface_cap to off, deformed interfacacial structure is shown.

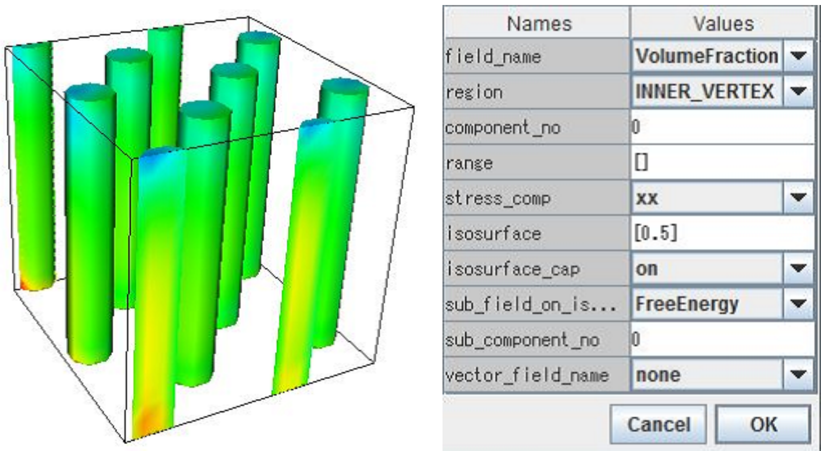

Figure 2.6: Elastica (linear elasticity) example 5: Free energy on the interface after the deformation.

Chapter 3

Operation guide of Elastica

3.1 Commands and parameters for fields of Elastica

3.1.1 List of fields for Elastica

Name of Parameters	Meanings and notations in theory
Displacement	Displacement vector $\boldsymbol{u}$
FreeEnergy	Free energy(deformation free energy) f
VolumeFraction	Volume fraction ψ_α
Stress	Stress tensor σ

3.1.2 Input parameters of Elastica

Solver parameter

Name of Parameters	Meanings and notations in theory
CONVERGENCE_CRITERION_FOR_CG_1	Convergence criterion for CG solver of linear equation. When the norm of residue vector is less than this criterion, calculation is judged to have converged. The default value is 0.5×10^{-6}
CONVERGENCE_CRITERION_FOR_CG_2	Another convergence criterion for CG solver of linear equation. When the ratio of norm of residue vector and right hand side vector is less than this criterion, calculation is judged to have converged. The default value is zero, and it means that this criterion is not applied. If fixed displacement condition, which is treated by the penalty method, is applied, this criterion should be set to zero.
MATRIX_SOLVER	Linear equation (matrix equation) solver name to be used. Either "ICCG" or "CG" Default is "ICCG".
PENALTY_NUMBER_FOR_DIRICHLET_BC	A penalty number to handle Dirichlet condition (a very large number). The default value is 10^{13}
ELEMENTS_PER_MATRIX_MERGE	In composition procedure of a matrix (stiffness matrix) for displacement calculation, the matrix may not be composed for all elements at once, but can be composed incrementally for groups of elements. The number of elements of the groups is specified by this parameter. The default is 5000. The size of memory for matrix composition can be reduced if number of elements is larger than value of this parameter.

Physical parameter

Name of Parameters	Meanings and notations in theory
NUMBER_OF_COMPONENTS	number of components
AVERAGED_VOLUME_FRACTION	averaged volume fraction as initial value $\psi_{\alpha 0}$. Value for each component is given as an parameter array element.
GRAVITY_X	X component of gravitational acceleration vector.
GRAVITY_Y	Y component of gravitational acceleration vector.
GRAVITY_Z	Z component of gravitational acceleration vector.
GRAVITY	gravity acceleration vector given as an array. ie.) [g_X, g_Y, g_Z]
MASS_DENSITY	mass density of each component (mass density when volume fraction is 1.0).
AXISYMMETRIC_MODULUS _EVALUATION_AXES	arbitrary number of 3-dimensional axis vector. Each vector is used as "imaginary" axis vector for which averaged parameters (quadratic form of strain tensor elements which appear with D_k in expression of deformation free energy), used to calculate anisotropic moduli data, are calculated assuming that calculation system have anisotropy axis into the direction of the vector.
MODULUS_ANISOTROPY_α	anisotropy of component α. A strings either "Isotropic" (isotropic elastic material) or "Axisymmetric" (axissymetric anisotropy) can be specified. If no string is given for component α, the component is treated as an isotropic material.
BULK_MODULUS	isotropic bulk modulus of of each component. Value for each component is given as an parameter array element.
SHEAR_MODULUS	isotropic shear modulus of of each component. Value for each component is given as an parameter array element.
MODULUS_α	moduli of component α. For isotropic components, 2 values are given as bulk modulus and shear modulus in this order. For anisotropic components, 5 values are given as (n, l, k, m, μ) in this order.
MODULUS_AXIS_α_i	When component α is anisotropic, give an axis vectors for the component. The index i is an integer starts from zero, if n axis vectors are necessary, you must give vectors from $i = 0$ to $i = n - 1$ as parameters. Currently, axissymetry is the only supported anisotropy, so only parameter with $i = 0$ is necessary.

3.1.3 Boundary conditions (partial region conditions) of Elastica

The following partial region conditions are all for the displacement vector field.

Partial region condition	treatment
D_VEC	Set displacement on vertices in specified partial region(fixed displacement condition). Give a 3-dimensional vector. Conditions "D_VX","D_VY" and "D_VZ" are prepared for cases in which not all displacement vector components should be fixed.
D_VX	X component of fixed displacement vector.
D_VY	X component of fixed displacement vector.
D_VZ	X component of fixed displacement vector.
N_LOAD	Set load on vertices in specified partial region(fixed load condition). Give a 3-dimensional vector.

3.1.4 Commands and parameters for fields of Elastica

Displacement : displacement field - commands

Displacement	Name
Initialization	"INITIALIZE:TO_ZERO"
Time evolution	"SOLVE:LINEAR_ELASTICITY:ISOTROPIC"
Time evolution	"SOLVE:LINEAR_ELASTICITY:ANISOTROPIC"
Time evolution	"MOVE:POSITION_OF_VERTEX"
Analysis	"OUTPUT:AVS"
Evaluation	"EVALUATE:TRUE"

1. **Displacement - initialization commands**

Name	**"INITIALIZE:TO_ZERO"**
Function	Set field values to zero.

2. **Displacement - time evolution commands**

Name	**"SOLVE:LINEAR_ELASTICITY:ISOTROPIC"**
Function	Calculate displacement of isotropic elastic material.
Dependent field	Displacement
Dependent field	VolumeFraction
Dependent parameter	NUMBER_OF_COMPONENTS
Dependent parameter	CONVERGENCE_CRITERION_FOR_CG_1
Dependent parameter	CONVERGENCE_CRITERION_FOR_CG_2
Dependent parameter	DIMENSION_OF_SPACE
Dependent parameter	MATRIX_SOLVER
Dependent parameter	ELEMENTS_PER_MATRIX_MERGE
Dependent parameter	GRAVITY
Dependent parameter	GRAVITY_X
Dependent parameter	GRAVITY_Y
Dependent parameter	GRAVITY_Z
Dependent parameter	MASS_DENSITY
Dependent parameter	PENALTY_NUMBER_FOR_DIRICHLET_BC
Dependent parameter	MODULUS_ANISOTROPY_α
Dependent parameter	BULK_MODULUS
Dependent parameter	SHEAR_MODULUS
Dependent parameter	MODULUS_α
Dependent parameter	MODULUS_AXIS_α_i

Name	**"SOLVE:LINEAR_ELASTICITY:ANISOTROPIC"**
Function	Calculate displacement of anisotropic elastic material.
Dependent field	Displacement
Dependent field	VolumeFraction
Dependent parameter	NUMBER_OF_COMPONENTS
Dependent parameter	CONVERGENCE_CRITERION_FOR_CG_1
Dependent parameter	CONVERGENCE_CRITERION_FOR_CG_2
Dependent parameter	DIMENSION_OF_SPACE
Dependent parameter	MATRIX_SOLVER
Dependent parameter	ELEMENTS_PER_MATRIX_MERGE
Dependent parameter	GRAVITY
Dependent parameter	GRAVITY_X
Dependent parameter	GRAVITY_Y
Dependent parameter	GRAVITY_Z
Dependent parameter	MASS_DENSITY
Dependent parameter	PENALTY_NUMBER_FOR_DIRICHLET_BC
Dependent parameter	MODULUS_ANISOTROPY_α
Dependent parameter	BULK_MODULUS
Dependent parameter	SHEAR_MODULUS
Dependent parameter	MODULUS_α
Dependent parameter	MODULUS_AXIS_α_i

Name	**"MOVE:POSITION_OF_VERTEX"**
Function	Move mesh points applying displacement field vector.
Dependent field	Displacement

3. **Displacement - analysis commands**

Name	**"OUTPUT:AVS"**
Function	Output calculation results on an AVS format file(field-data).
Dependent field	Displacement
Dependent parameter	DIMENSION_OF_SPACE

4. **Displacement - evaluation commands**

Name	**"EVALUATE:TRUE"**
Function	Always return "true" flag. This function is used to perform an analysis command with a constant time step interval.

FreeEnergy : free energy field - commands

FreeEnergy	Name
Time evolution	"SOLVE:LINEAR_ELASTICITY"
Analysis	"OUTPUT:AVS"
Evaluation	"EVALUATE:TRUE"

1. **FreeEnergy - time evolution commands**

Name	**"SOLVE:LINEAR_ELASTICITY"**
Function	Calculate free energy.
Dependent field	Displacement
Dependent field	VolumeFraction
Dependent parameter	NUMBER_OF_COMPONENTS
Dependent parameter	AXISYMMETRIC_MODULUS_EVALUATION_AXES
Dependent parameter	MODULUS_ANISOTROPY_α
Dependent parameter	BULK_MODULUS
Dependent parameter	SHEAR_MODULUS
Dependent parameter	MODULUS_α
Dependent parameter	MODULUS_AXIS_α_i

2. **FreeEnergy - analysis commands**

Name	**"OUTPUT:AVS"**
Function	Output calculation results on an AVS format file(field-data).

3. **FreeEnergy - evaluation commands**

Name	**"EVALUATE:TRUE"**
Function	Always return "true" flag. This function is used to perform an analysis command with a constant time step interval.

VolumeFraction : volume fraction field - commands

VolumeFraction	Name
Initialization	"INITIALIZE:ONE_COMPONENT"
Initialization	"INITIALIZE:TWO_COMPONENT"
Initialization	"INITIALIZE:UNIFORM"
Analysis	"OUTPUT:AVS"
Evaluation	"EVALUATE:TRUE"

1. **VolumeFraction - initialization commands**

Name	**"INITIALIZE:ONE_COMPONENT"**
Function	Initialize as one component system ($\psi_0 = 1$)
Dependent parameter	NUMBER_OF_COMPONENTS

Name	**"INITIALIZE:TWO_COMPONENT"**
Function	Initialize as two component system ($\psi_1 = 1 - \psi_0$)
Dependent parameter	NUMBER_OF_COMPONENTS

Name	**"INITIALIZE:UNIFORM"**
Function	Initialize as a multi-component uniform mixture system.
Dependent parameter	NUMBER_OF_COMPONENTS
Dependent parameter	AVERAGED_VOLUME_FRACTION

2. **VolumeFraction - analysis commands**

Name	**"OUTPUT:AVS"**
Function	Output calculation results on an AVS format file(field-data).

3. **VolumeFraction - evaluation commands**

Name	**"EVALUATE:TRUE"**
Function	Always return "true" flag. This function is used to perform an analysis command with a constant time step interval.

Stress : stress tensor field - commands

Stress	Name
Time evolution	"SOLVE:STRESS"
Evaluation	"EVALUATE:TRUE"

1. **Stress - time evolution commands**

Name	**"SOLVE:STRESS"**
Function	Calculate stress tensor.
Dependent field	Displacement
Dependent field	VolumeFraction
Dependent parameter	NUMBER_OF_COMPONENTS
Dependent parameter	AXISYMMETRIC_MODULUS_EVALUATION_AXES
Dependent parameter	MODULUS_ANISOTROPY_α
Dependent parameter	BULK_MODULUS
Dependent parameter	SHEAR_MODULUS
Dependent parameter	MODULUS_α
Dependent parameter	MODULUS_AXIS_α_i

2. **Stress - evaluation commands**

Name	**"EVALUATE:TRUE"**
Function	Always return "true" flag. This function is used to perform an analysis command with a constant time step interval.

References

1) L.D.Landau, and E.M.Lifshitz, eds.: *Theory of Elasticity - 3rd Edition*, Butterworth-Heinemann (1986).

2) M.Yagawa, and S.Yoshimura, : *Finite Element Method (Computer Physics and CAE - series 1)*, Baifukan (1991).