OCTA

Integrated simulation system for soft materials

Multi-Phase Dynamics Program

Muffin

version 4.1

User's Manual

- Volume VI -

Gel Dynamics Simulator

Geldyn

OCTA User's Group

March 03 2005

Authors of the Manual

Tatsuya Yamaue

Programers

Tatsuya Yamaue

Version 4.1 release

Programer, Authors of the Manual Tatsuya Yamaue

Acknowledgment

The first version of 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

1 What is GelDyn? 1

2 Theoretical Background of GelDyn 3
2.1 Dynamics of Gels . . . 3
2.1.1 Overview . . . 3
2.1.2 Notation of parameters and definitioin of variables . . . 3
2.1.3 Dimensionless expression for parameters and variables . . . 4
2.1.4 Collective diffusion model of gel networks . . . 4
2.1.5 Stress-diffusion coupling model of gels . . . 5
2.2 Free Energy and Stress Tensor of Gels . . . 5
2.3 Boundary Conditions of Geldyn . . . 6
2.3.1 Overview of boundary conditions . . . 6
2.3.2 Boundary conditions of Pressure p . . . 7
2.3.3 Boundary conditions of Displacement $\boldsymbol{u}$ (Polymer Velocity $\boldsymbol{v}_p$) . . . 7
2.4 Simulation Scheme for Dynamics of Gels . . . 8
2.4.1 Explicit scheme for collective diffusion model of gel networks . . . 8
2.4.2 Implicit scheme for collective diffusion model of gel networks . . . 9
2.4.3 Explicit scheme for stress-diffusion coupling model of gels . . . 9
2.4.4 Implicit scheme for stress-diffusion coupling model of gels . . . 10

3 Sample problems of GelDyn 13
3.1 Sample Problems for Swelling Dynamics of Geldyn . . . 13
3.1.1 Application 01: Swelling of 2-dimensional slab gels and surface folding . . . 13
3.1.2 Application 02: Swelling of 3D plate gels and pattern formation comparison with implicit solver . . . 14
3.1.3 Application 03: Free swelling of 2D slab gels comparison of two dynamics . . . 16
3.1.4 Application 04: Free Swelling of 3D long rod gels comparison of two dynamics . . . 17
3.1.5 Application 05: Free Swelling of 3D large disk gels comparison of two dynamics . . . 19

4 Operation Guide of GelDyn 21
4.1 Input Parameters of Geldyn . . . 21
4.1.1 Solver control parameters of Geldyn . . . 21
4.1.2 Physical parameters of Geldyn . . . 22
4.1.3 Analysis results of Geldyn . . . 23
4.2 Fields and Commands for Fields of Geldyn . . . 23
4.2.1 List of Fields for Geldyn . . . 23
4.2.2 List of Commands for Geldyn . . . 23

References 33

List of Figures

3.1 Application 01 of Geldyn : Swelling of 2-dimensional slab gels and folding pattern formations on surface 14
3.2 Application 02 of Geldyn : Swelling of 3D plate gels and Pattern formation 16
3.3 Application 03 of Geldyn : Free swelling of 2D slab gels. 17
3.4 Application 04 of Geldyn : Free swelling of 3D long rod gels. 19
3.5 Application 05 of Geldyn : Free swelling of 3D large disk gels. 20

Chapter 1

What is GelDyn?

The Gel Dynamics Simulator named **GelDyn** is the finite-element-method (FEM) simulator treating the large deformation dynamics of gels, accompanying the change of external stimuli, such as, temperature, pressure, e.t.c., and the load of body forces, such as gravity, and surface forces.
The main features are listed below.

1. Target
   - Swelling, deswelling and deformation of polymeric gels.
   - Solvent flow (diffusion) in polymeric gels with both permeable and impermeable surfaces.
   - Both 2D and 3D shapes are supported.
2. Dynamics
   - The stress-diffusion coupling model of polymer gels.
   - The collective diffusion model of gel networks. (for comparison)
3. Simulation Scheme
   The finite element method using the linear triangle (2D) and/or tetrahedron (3D) interpolation.
   - The explicit time evolution using non-linear elasticity minimizer. (explicit solvers)
   - The implicit time evolution for linearized gel models using matrix solver. (implicit solvers)
4. Solvers
   - The explicit stress-diffusion coupling solver. (for both 2D and 3D system)
   - The explicit collective diffusion solver. (for both 2D and 3D system)
   - The implicit stress-diffusion coupling solver. (for 3D system)
   - The implicit collective diffusion solver. (for 3D system)
   - The elasticity dynamics solver. (for 3D system)
5. Design of gels
   - Design of shapes.
   - Design of morphology. ... concentration of polymer, χ-parameter, moduli (the crosslinking number density), concentration of counter ions
6. Stimuli for the volume phase transition and deformation
   - Change of temperature
   - Change of χ-parameter
   - Change of solvent (water) pressure on permeable surfaces
   - Change of volume forces (gravity) and surface forces

7. Boundary Conditions
    - Pressure of pure solvent (water)
        - Permeable surfaces (dirichlet)
        - Impermeable surfaces (neumann)
    - Displacement (velocity) of polymer network
        - Fixed surfaces or surfaces moving with constant velocity (dirichlet)
        - Load of surface forces
8. Applications
    - Disposable diapers and sanitary napkins (super-water absorbents)
    - Temperature sensitive gels
    - Drug delivery systems (DDS)
    - Actuators, sensors and switching devices

Chapter 2

Theoretical Background of GelDyn

2.1 Dynamics of Gels

2.1.1 Overview

Gels are cross-linked networks of polymers swollen with a liquid and have both the liquid-like and solid-like properties, because a gel has a shear modulus though the major constituent is a liquid. The dynamics of gels has been described by the collective diffusion model of gel networks, [1, 2] which well reproduces the swelling phenomena in one axis, such as spherical gels. But the collective diffusion model of gel networks can't reproduce the swelling phenomena in two or three axes and the non-liner deswelling phenomena, such as surface skin formation and stagnation of deswelling.

Recently, several models, which are based on the two fluids model, [3] have been proposed as dynamics of gels. [3, 4] Here we have constructed the simulation scheme for large deformation of gels and developed the gel dynamics simulator using the stress-diffusion coupling model of gels, [5] in which the general deformation of polymer networks, the continuity of solvent and the coupling between solvent diffusion and network stress are considered. We have also developed the gel dynamics simulator using the collective diffusion model of gel networks for comparison.

2.1.2 Notation of parameters and definitioin of variables

Notation of symbols for Geldyn simulator are as follows:

Notation of parameters	Meaning of parameters
d	Dimension of system (2 or 3)
v_1	Volume of a monomar unit (ex. $v_1 = 3.0 \times 10^{-29} m^3$ for standard NIPA gels)
$k_B T$	Thermal energy per 1 particle. ($1 k_B T = 4.12 \times 10^{-21}$ J at $T = 298$K(25^oC))
η_s	Viscosity of solvent ($= 0.89 \times 10^{-3}$ Pa $\cdot$ sec $= 0.89 \times 10^{-2}$ Poise for water)
ν	Exponent (1/2 for *theta*-solvent and 3/5 for good-solvent)
ξ_b	Blob size of gels
ρ_s	Mass density of pure solvent
ρ_p	Mass density of pure polymer
$\boldsymbol{g}$	Gravity acceleration
B	Magnitude of elastic logarithmic term
Notation of variables	**Meaning of variables**
$\boldsymbol{x}$	Coordinate of material point of polymer
$\boldsymbol{x}_0$	Coordinate of material point of polymer in the reference state
$\boldsymbol{u}(\boldsymbol{x})$	Displacement of polymer
$\phi(\boldsymbol{x})$	Volume fraction of polymer
$\phi_0(\boldsymbol{x})$	Volume fraction of polymer in the reference state (0.07 for std NIPA)
$\boldsymbol{v}_s(\boldsymbol{x})$	Velocity of solvent
$\boldsymbol{v}_p(\boldsymbol{x})$	Velocity of polymer ($\equiv= \dot{\boldsymbol{u}}$)
$p(\boldsymbol{x})$	Pressure of solvent
$F\{\boldsymbol{x}\}$	Free energy of gels
$\sigma_{ij}(\boldsymbol{x})$	Cauchy stress tensor of polymer network
$W_{ij}(\boldsymbol{x})$	Finger strain tensor of polymer network
$\rho(\boldsymbol{x})$	Mass density of gels ($\equiv \rho_p \phi + \rho_s (1-\phi)$)
$\boldsymbol{J}(\boldsymbol{x})$	Flux of solvent
$\chi(\boldsymbol{x})$	Polymer-solvent interaction parameter
$\nu_0(\boldsymbol{x})$	Crosslinking number density in the reference state
$\nu_{i0}(\boldsymbol{x})$	Counter ion number density in the reference state
$\zeta(\phi)$	Friction coefficient between polymer and solvent
$K(\boldsymbol{x})$	Isotropic bulk modulus of gels
$\mu(\boldsymbol{x})$	Isotropic shear modulus of gels

2.1.3 Dimensionless expression for parameters and variables

In this section, we describe units for Geldyn and dimensionless expressions by the units. We will use the MKSA unit system throughout this section.

Units for dimensionless physical quantities

Physical quantity	unit	meaning
length x	l	system size (ex. $= 1.0mm = 1.0 \times 10^{-3} m$)
crosslinking number density ν_0	v_1	0.036 (ex. $\nu_0 = 1.2 \times 10^{27} m^{-3}$ for std NIPA)
ion number density ν_{i0}	v_1	0.072 (ex. $\nu_{i0} = 2.4 \times 10^{27} m^{-3}$ for std NIPA)
energy density and stress	$l^d k_B T / v_1$	
friction coefficient $\zeta(\phi)$	$\zeta_0 \equiv \zeta(\phi_0)$	$\zeta/\zeta_0 \approx (\phi/\phi_0)^{2\nu/(3\nu-1)}$, $(\zeta \approx 6\pi\eta_s \xi_b^{-2} \propto \phi^{2\nu/(3\nu-1)})$
time t	τ	$\tau = \zeta_0 l^(2-d) v_1 / k_B T$
velocity $\boldsymbol{v}$	v^*	$v^* = l/\tau$
pressure p	p^*	$p^* = \eta_w / \tau$
mass density ρ	ρ^*	(ex. mass density of water $\rho = 1.0 g/cm^3 = 1.0 \times 10^3 kg/m^3$)

2.1.4 Collective diffusion model of gel networks

The dynamics of polymer gels should be discussed from the stand point of the dynamics of polymer solutions. However, since the polymer gels have elastic properties due to crosslinking by chemical bonds (chemical gels)

and/or some kinds of interaction (physical gels), such as hydrogen bonding or hydrophobic interaction, the dynamics of polymer gels is well described in terms of the theory of elasticity. [6] T.Tanaka, et. al. first developed a theory of the collective diffusion dynamics of gel networks, named "Tanaka-Fillmore's theory". [2]

The equation of motion of elastic gel networks is described as

$$\rho\frac{\partial^2 \boldsymbol{u}}{\partial t} = \boldsymbol{\nabla}\cdot\boldsymbol{\sigma} + \rho\boldsymbol{g} - \zeta\frac{\partial \boldsymbol{u}}{\partial t} \tag{2.1}$$

Here, the first term of the right-hand side represent the elastic term and the last term of the right-hand side the contribution of the friction between the network and solvent molecules. In most cases, the acceleration term is much smaller than the other terms and the equation of the collective diffusion dynamics of gel networks is rewritten as follows.

$$\frac{\partial \boldsymbol{u}}{\partial t} = \zeta^{-1}\boldsymbol{\nabla}\cdot\boldsymbol{\sigma} + \rho\boldsymbol{g} \tag{2.2}$$

2.1.5 Stress-diffusion coupling model of gels

The dynamics of polymer gels has to be discussed from the stand point of the dynamics of polymer solutions. Recently, several models, which are based on the two fluids model and consider the dynamic coupling between polymer network stress and solvent diffusion, [3] have been proposed as dynamics of gels. [3, 4]

Here, we explain the stress-diffusion coupling dynamics of gels based on two fluids model. The equations of motion of gels are described as follows.

$$\zeta(\boldsymbol{v}_p - \boldsymbol{v}_s) = -\phi\boldsymbol{\nabla}p + \boldsymbol{\nabla}\cdot\boldsymbol{\sigma} + \rho_p\phi\boldsymbol{g} \tag{2.3}$$

$$\zeta(\boldsymbol{v}_s - \boldsymbol{v}_p) = -(1-\phi)\boldsymbol{\nabla}p + \rho_s(1-\phi)\boldsymbol{g} \tag{2.4}$$

The incompressibility of gels are written as

$$\boldsymbol{\nabla}\cdot(\phi\boldsymbol{v}_p + (1-\phi)\boldsymbol{v}_s) = 0. \tag{2.5}$$

eq.(2.3) and eq.(2.4) makes the mechanical balance equation between the pressure of solvent and the stress of polymer network.

$$\boldsymbol{\nabla}p = \boldsymbol{\nabla}\cdot\boldsymbol{\sigma} + \rho\boldsymbol{g} \tag{2.6}$$

Here, ρ is the average mass dencity of gels and defined by

$$\rho \equiv \rho_p\phi + \rho_s(1-\phi)$$

.

2.2 Free Energy and Stress Tensor of Gels

The general free energy of gels includes three terms as follows.

$$F = F_{mix} + F_{ion} + F_{el} \tag{2.7}$$

Here, F_{mix} is the mixing free energy of solvent and polymer networks.

$$F_{mix} = \int_V d^d x f_m(\phi) \tag{2.8}$$

Definite expression of the mixing free energy density $f_m(\phi)$ is Flory-Huggins type or Ginzburg-Landau type:

- Flory-Huggins type

$$f_m = (1-\phi)\ln(1-\phi) + \chi\phi(1-\phi) \tag{2.9}$$

- Ginzburg-Landau type

$$f_m = (\frac{1}{2} - \chi)\phi^2 + \frac{1}{6}\phi^3 + \frac{1}{12}\phi^4 \tag{2.10}$$

F_{ion} is the translational entropy of counter ions of polyelectrolytes. Here, ν_{i0} denotes the density of counter ions in the reference state.

$$F_{ion} = \int_V d^d x \nu_{i0} \Big(\frac{\phi}{\phi_0}\Big) \ln \Big(\frac{\phi}{\phi_0}\Big) \tag{2.11}$$

F_{el} is the elastic energy term for general network deformation.

$$\begin{aligned} F_{el} &= \int_{V_0} d^d x_0 \frac{\nu_0}{2} (\mathrm{tr}W - 2B \ln \frac{\phi}{\phi_0}) && (2.12)\\ &= \int_V d^d x \frac{\phi}{\phi_0} \frac{\nu_0}{2} (\mathrm{tr}W - 2B \ln \frac{\phi}{\phi_0}) && (2.13)\\ &= \int_V d^d x \frac{\nu}{2} (\mathrm{tr}W - 2B \ln \frac{\phi}{\phi_0}) && (2.14) \end{aligned}$$

In this term, W_{ij} is the finger strain tensor defined as follows.

$$W_{ij} = \frac{dx_i}{dx_{0k}} \cdot \frac{dx_j}{dx_{0k}} \tag{2.15}$$

Here, $\boldsymbol{x}_0$ denotes the cartesian coordinates in the reference state, ν the crosslinking number density, ν_0 the crosslinking number density in the reference state, B the magnitude of elastic logarithmic term, ϕ the network volume fraction and ϕ_0 the network volume fraction in the reference state. Volume fraction of the polymer network is related to the finger strain tensor as $\phi = \phi_0 det \boldsymbol{W}^{-\frac{1}{2}}$.

We deform the gel infinitesimally as $\boldsymbol{x} \to \boldsymbol{x} + \boldsymbol{u}$. Then the change of the free energy may be expressed in the term of the Cauchy stress of polymer network $\boldsymbol{\sigma}$ in the form,

$$\delta F = \int d^{d-1} x n_j (\sigma_{ij}) u_i - \int d^d x \nabla_j \sigma_{ij} u_i = \int d^d x \sigma_{ij} \frac{\partial u_i}{\partial x_j} \tag{2.16}$$

Therefore, the stress tensor of polymer network is described as follows.

$$-\sigma_{ij} = \Big[\ \phi f'_m - f_m \ \Big] \delta_{ij} - \nu_0 \frac{\phi}{\phi_0} (W_{ij} - (B + \frac{\nu_{i0}}{\nu_0}) \delta_{ij}) \tag{2.17}$$

Isotropic bulk modulus K and shear modulus μ of gels, which are needed in the implicit scheme of gels, are described as follows.

$$\begin{aligned} K &= \phi^2 f''_m + \nu_0 \Big[(\frac{2}{d} - 1)(\frac{\phi}{\phi_0})^{1-2/d} - (B + \frac{\nu_{i0}}{\nu_0}) \frac{\phi}{\phi_0} \Big] && (2.18)\\ \mu &= \nu_0 (\frac{\phi}{\phi_0})^{1-2/d} && (2.19) \end{aligned}$$

2.3 Boundary Conditions of Geldyn

There are the following in the boundary condition (partial region conditions) which can be imposed on a field in Geldyn.

2.3.1 Overview of boundary conditions

- Periodic boundary condition :

  Possible only in the mesh of the UNSTRUCTURED_RECT type. In Geldyn, since the periodic boundary is treated geometrically and a periodic boundary condition will be automatically applied to all physical quantity if a mesh is made into a periodic, it is not necessary to specify clearly in Input UDF.

- Dirichlet condition :

  The conditions which impose a fixed value to a subregion.

  We use the large penalty number method for dirichlet boundary conditions in the implicit solvers. [7]

- Neumann condition :

  Give the direction component of a normal of the surface of the gradient vector of physical quantity. It is not necessary to give clearly the Neumann conditions (natural boundary conditions) whose component of a normal direction at the border plane of the form of $\boldsymbol{n} \cdot \nabla f = 0$ is zero in the discretization by the finite element method.

2.3.2 Boundary conditions of Pressure p

The boundary conditions which can be applied to a pressure field are as follows.

- **Periodic boundary condition**
  It can be applied, only when a mesh configuration type is UNSTRUCTURED_RECT. The following equation is imposed when a periodic boundary condition is imposed in the x directions.

  $$p(x, y, z) = p(x + L_x, y, z)$$

  Imposing a periodic boundary condition also in the direction of y, or the direction of z, the same equation is imposed to each direction.

- **Permeable surfaces (Dirichlet condition)**
  Apply value p_0 of a pressure of solvent for a permeable surface.

  $$p(\boldsymbol{x})|_{Boundary} = p_0$$

  In order to give such a Dirichlet boundary condition to a pressure field, input "D_PERMEABLE" into a condition name and the value of the pressure field on a boundary into the head data of the value of conditions.

  This boundary condition is applied in a method of Polymer Stress field for the explicit stress-diffusion coupling solver and in a method of Displacement field for the implicit stress-diffusion coupling solver.

- **Impermeable surfaces (Neumann condition)**
  Apply zero pressure gradient for a impermeable surface.

  $$\boldsymbol{n} \cdot \nabla p(\boldsymbol{x})|_{Boundary} = 0$$

  In order to give such a Neumann boundary condition to a pressure field, input "N" into a condition name ("N_IMPERMEABLE" for impermeable surfaces) and the value of the gradient of pressure field on a boundary into the head data of the value of conditions.

  This boundary condition is applied in a method of Flux field for the explicit stress-diffusion coupling solver and in a method of Displacement field for the implicit stress-diffusion coupling solver.

2.3.3 Boundary conditions of Displacement $\boldsymbol{u}$ (Polymer Velocity $\boldsymbol{v}_p$)

The boundary conditions which can be set up to a displacement field are as follows.

- **Periodic boundary condition**
  It can be applied, only when a mesh configuration type is UNSTRUCTURED_RECT. The following equation is imposed when a periodic boundary condition is imposed in the x directions.

  $$\boldsymbol{u}(x, y, z) = \boldsymbol{u}(x + L_x, y, z)$$

  When a periodic boundary condition is also imposed in the direction of y, or the direction of z, the same equation is imposed to each direction.

- **Fixed surfaces or surfaces moving with constant velocity (Dirichlet condition)**
  The boundary fixed or moved by constant velocity $\boldsymbol{v}_{p0}$ can be expressed with the following Dirichlet boundary conditions.

$$\boldsymbol{v}_p(x,y,z)|_{wall} = \boldsymbol{v}_{p0}$$

  For specifying the velocity of the boundary, input "D_VX"(Specify X component), "D_VY"(Specify Y component) or "D_VZ"(Specify Z component) into a condition name, and input a velocity into the value of conditions. Or input "D_VEC" into a condition name and input three values, X, Y, and Z component of a velocity into a value in an array.

  This boundary condition is applied in a method of Displacement field for any solver.

- **Load of surface forces (Neumann condition)**
  The load of surface force on gels can be expressed with the following Neumann boundary conditions.

$$\boldsymbol{n}\cdot(\boldsymbol{\sigma} - p\boldsymbol{I})|_{Boundary} = \boldsymbol{T}$$

  For specifying the load of the boundary, input "N_LOAD" into a condition name, and input three values, X, Y, and Z component of a force into the value of conditions.

  This boundary condition is applied in a method of Polymer Stress field for the explicit stress-diffusion coupling solver and in a method of Displacement field for the implicit stress-diffusion coupling solver.

2.4 Simulation Scheme for Dynamics of Gels

The two simulation schemes for the stress-diffusion coupling model are expected as follows.

1. Separation of time scale of the diffusion of solvent and elastic deformation. [4]
   → The explicit scheme (solver) for the stress-diffusion coupling model.
2. Minimize Rayleighian based on two fluid model by Onsager's theorem for energy dissipation. [3, 8]
   → The implicit scheme (solver) for the stress-diffusion coupling model.

2.4.1 Explicit scheme for collective diffusion model of gel networks

The collective diffusion model of gel networks described as eq.(2.2) is simple relaxation process of free energy of gels. Therefore eq.(2.2) is rewritten as the follow interal equation.

$$\frac{\partial \boldsymbol{x}}{\partial t} = -\zeta^{-1}\frac{\delta}{\delta \boldsymbol{x}}\Big[F - \int_V d^d x \rho \boldsymbol{g}\cdot\delta\boldsymbol{x} - \int_S d^{d-1}x\boldsymbol{T}\cdot\delta\boldsymbol{x}\Big] \tag{2.20}$$

From eq.(2.7), using the linear interpolation of the finite element method and the explicit time evolution, eq.(2.20) is rewritten as follows.

$$\begin{aligned}
\dot{x}_i^J &= u_i^J/\delta t \\
&= -\frac{\sum_{e(\ni J)}\Big[-\frac{1}{V_e^0}\frac{\delta V_e}{\delta x_i^J}\int_e d^d x_0(\phi f_m'(\phi) - f_m(\phi)) + (\frac{\delta}{\delta x_i^J}tr\boldsymbol{W})\sum_{I(\in e)}\frac{\nu_0^I}{2}\int_e d^d x_0 L_I(\boldsymbol{x}_0)\Big]}{\sum_{e(\ni J)}\frac{V_e}{d+1}} \\
&\quad + \frac{\sum_{e(\in V)}\sum_{I(\in e)}\rho^I g_i\int_e d^d x L_I(\boldsymbol{x})L_J(\boldsymbol{x})}{\sum_{e(\ni J)}\frac{V_e}{d+1}} + \frac{\sum_{e(\in S)}\sum_{I(\in e)}T_i^I\int_e d^{d-1}x L_I(\boldsymbol{x})L_J(\boldsymbol{x})}{\sum_{e(\ni J)}\frac{V_e}{d+1}}
\end{aligned} \tag{2.21}$$

Here, I and J denote the vertex number and V_e a volume (3D) or an area (2D) of a finite element.

Using eq.(2.21), we have developed the explicit solver for the collective diffusion of gel networks. (ref. application examples 01, 02 and 03)

2.4.2 Implicit scheme for collective diffusion model of gel networks

The Rayleighian for the collective diffusion model of gel networks eq.(2.2) is described as follows.

$$R\{\boldsymbol{v}_p(t), \boldsymbol{x}(t)\} = \int_V d^d x[\frac{1}{2}\zeta \boldsymbol{v}_p^2 + \boldsymbol{\sigma} : \boldsymbol{D}] - \int_V d^d x \rho \boldsymbol{g} \cdot \boldsymbol{v}_p - \int_S d^{d-1} x \boldsymbol{T} \cdot \boldsymbol{v}_p \tag{2.22}$$

Here, the first term of right hand side denotes the energy dissipation by friction between polymer and solvent, the second term the change of free energy, the third term the energy change by the volume force (gravity) and the forth term the energy change by the surface force. $\boldsymbol{D}$ is deformation rate tensor of polymer and defined as

$$D_{ij} \equiv \frac{1}{2}(\frac{\partial v_{pi}}{\partial x_j} + \frac{\partial v_{pj}}{\partial x_i}) \tag{2.23}$$

For the semi-implicit time evolution, polymer stress tensor σ_{ij} is written as follows.

$$\sigma_{ij}(t+\delta t) = \sigma_{ij}(t) + \delta t[2\mu(t) D_{ij} + (K(t) - \frac{2}{d}\mu(t)) D_{ll}\delta_{ij}] \tag{2.24}$$

Here, K, μ and $K - 2/d\mu$ express the bulk modulus, shear modulus and lame number of gels at time t. From eq.(2.22) and eq.(2.24), the Rayleighian for the semi-implicit collective diffusion model linearized for displacement of polymer $\boldsymbol{u}$ in time $t + \delta t$ is described as follows.

$$R\{\boldsymbol{u}, \boldsymbol{x}(t), \delta t\}\delta t = \int_V d^d x \Big[\frac{1}{2}\frac{\zeta}{\delta t}\boldsymbol{u}^2 + \mu(e_{ij} - \frac{1}{d}\delta_{ij} e_{ll})^2 - \frac{K}{2} e_{ll}^2\Big] - \int_V d^d x(\boldsymbol{\nabla}\boldsymbol{\sigma} + \rho \boldsymbol{g}) \cdot \boldsymbol{u} - \int_S d^{d-1} x \boldsymbol{T} \cdot \boldsymbol{u} \tag{2.25}$$

Here, $\boldsymbol{e}$ is strain tensor for linear elasticity and defined as follows.

$$e_{ij} \equiv \frac{1}{2}(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}) \tag{2.26}$$

Using the linear elasticity (matrix) solver for FEM, we can minimize above functional and solve $\boldsymbol{u}$ and shape in δt. Therefore, we have developed the semi-implicit solver for the collective diffusion of gel networks. (ref. application examples 02)

2.4.3 Explicit scheme for stress-diffusion coupling model of gels

Here, by separating the time scale of diffusion of solvent and elastic deformation, [4] we have physically reduced two fluid model eqs.(2.3,2.4, 2.5) and constructed the simple non-linear simulation scheme for large deformation of gels using the stress-diffusion coupling model of gels as the follows. [5]

From eq.(2.4), the solvent velocity related to the polymer velocity in gels is proportional to the gradient of the solvent pressure from the Darcy's law:

$$\boldsymbol{v}_s - \boldsymbol{v}_p = -\zeta^{-1}(1-\phi)\boldsymbol{\nabla} p \tag{2.27}$$

Here, from the mechanical balance between solvent and polymer, the next equation is satisfied.

$$\nabla_i p = \nabla_j \sigma_{ij} \tag{2.28}$$

The flux of solvent $\boldsymbol{J}$ related to the polymer is described as follows.

$$\begin{aligned} J_i &= (\phi v_{pi} + (1-\phi) v_{si}) - v_{pi} \\ &= -\zeta^{-1}(1-\phi)^2 \nabla_j \sigma_{ij} \end{aligned} \tag{2.29}$$

Therefore, the time-evolution of volume fraction of the polymer network is calculated by the following continuity equation using the euler picture:

$$\frac{\partial}{\partial t}\phi = -\nabla_i L_{ij}(\phi)\nabla_k \sigma_{jk} = \int_V d^d x L(\phi)(\nabla_j \sigma_{ji})\frac{\delta}{\delta \sigma_{ik}}(\nabla_l \sigma_{kl}) + \int_S d^{d-1} x J_i n_i \tag{2.30}$$

where $L_{ij}(\phi) = L(\phi)\delta_{ij}$ is the isotropic Onsager coefficient.

The deformation of the network at each time is determined by the elastic force balance while keeping a local volume fraction of gel in the moment, since the elastic force balance are satisfied instantaneously, on the other hand the diffusion process is so slow. This procedure for the elastic force balance is realized by minimizing the following elastic free energy using the langrange picture with the constraint term:

$$\frac{D}{Dt'}x_i(\boldsymbol{x}_0,t') = -\zeta^{-1}\frac{\delta F'}{\delta x_i} \tag{2.31}$$

$$F'(\boldsymbol{x}) = \int_V d^dx' \Big[\alpha(\phi'-\phi)^2 + \frac{\nu_0}{2}\frac{\phi'}{\phi_0}\mathrm{tr}W\Big] \tag{2.32}$$

Here, ϕ', t' denotes the virtual volume fraction of the polymer network and the virtual time for the elastic force balance calculation process. The first term of the right hand side is the constraint term for keeping a local volume fraction $\phi(\boldsymbol{x}_0,t)$ of gel, when the coefficient α is large.

We formulate the explicit time evolution solver for eqs.(2.30, 2.31, 2.29) using the linear interpolation of the finite element method.

eq.(2.31) using FEM is similarly described as eq.(2.21), therefore, we describe, here, the FEM formulation of eq.(2.30) as follows.

$$\begin{aligned}\dot{\phi}^J \sum_{e(\ni J)}\frac{V_e}{d+1} &= \sum_{e(\ni J)}\sum_{I(\in e)}\sum_{i,j}(\nabla_j L_I)(\nabla_i L_J)\sigma^I_{ij}\sum_{K(\in e)}L(\phi^K)\int_e d^dx L_K \\ &+ \sum_{e(\in S)}\sum_{I(\in e)}\sum_i J^I_i n_i \int_S d^{d-1}x L_I \end{aligned}\tag{2.33}$$

Here, I and J denote the vertex number and V_e a volume (3D) or an area (2D) of a finite element.

Using above equations, we have developed the explicit solver for the stress-diffusion coupling of gels. (ref. application examples 03)

2.4.4 Implicit scheme for stress-diffusion coupling model of gels

Here, we have constructed the Rayleighian for the stress-diffusion coupling model of gels described as eqs.(2.3,2.4, 2.5) [3] and have developed the semi-implicit solver for the stress-diffusion coupling dynamics of gels.

$\boldsymbol{v}_p\cdot$ eq.(2.3) $+\boldsymbol{v}_s\cdot$ eq.(2.4) leads the Rayleighian for the stress-diffusion coupling dynamics of gels as follows.

$$\begin{aligned}R\{\dot{p}(\boldsymbol{x}(t)),\boldsymbol{v}_p(t),p(\boldsymbol{x}(t)),\boldsymbol{x}(t)\} &= \int_V d^dx\Big[\frac{1}{2}\zeta(\boldsymbol{v}_p-\boldsymbol{v}_s)^2 - p\boldsymbol{\nabla}\cdot(\phi\boldsymbol{v}_p+(1-\phi)\boldsymbol{v}_s)+\boldsymbol{\sigma}:\boldsymbol{D}\Big] \\ &- \int_V d^dx(\rho_p\phi\boldsymbol{v}_p+\rho_s(1-\phi)\boldsymbol{v}_s)\cdot\boldsymbol{g} \\ &+ \int_S d^{d-1}x[p(\phi\boldsymbol{v}_p+(1-\phi)\boldsymbol{v}_s)\cdot\boldsymbol{n}-(\boldsymbol{\sigma}\cdot\boldsymbol{n})\cdot\boldsymbol{v}_p]\end{aligned}\tag{2.34}$$

Here, the first term of right hand side denotes the energy dissipation by friction between polymer and solvent, the second term the incompressibility condition, the third term the change of free energy. $\boldsymbol{D}$ is deformation rate tensor of polymer and defined as eq.(2.23). Using the boundary condition between polymer stress, solvent pressure and surface forces described as

$$(\boldsymbol{\sigma}-p\boldsymbol{I})\cdot\boldsymbol{n} = \boldsymbol{T}$$

, eq.(2.34) is rewritten as follows.

$$\begin{aligned}R\{\dot{p}(\boldsymbol{x}(t)),\boldsymbol{v}_p(t),p(\boldsymbol{x}(t)),\boldsymbol{x}(t)\} &= \int_V d^dx\Big[-\frac{1}{2\zeta}(1-\phi)^2(\boldsymbol{\nabla}p-\rho_s\boldsymbol{g})^2 - p\boldsymbol{\nabla}\cdot\boldsymbol{v}_p+\boldsymbol{\sigma}:\boldsymbol{D}\Big] \\ &- \int_V d^dx(\rho_p\phi+\rho_s(1-\phi))\boldsymbol{g}\cdot\boldsymbol{v}_p \\ &- \int_S d^{d-1}x\boldsymbol{T}\cdot\boldsymbol{v}_p\end{aligned}\tag{2.35}$$

For the semi-implicit time evolution, polymer stress tensor σ_{ij} is written as follows.

$$\sigma_{ij}(t+\delta t) = \sigma_{ij}(t) + \delta t[2\mu(t)D_{ij} + (K(t) - \frac{2}{d}\mu(t))D_{ll}\delta_{ij}] \tag{2.36}$$

Here, K, μ and $K - 2/d\mu$ express the bulk modulus, shear modulus and lame number of gels at time t. Therefore, Rayleighian for the semi-implicit stress-diffusion coupling dynamics linearized for pressure of solvent p and displacement of polymer $\boldsymbol{u}$ in time $t + \delta t$ is described as follows.

$$\begin{aligned} R\{\dot{p}(\boldsymbol{x}(t)), \boldsymbol{v}_p(t), p(\boldsymbol{x}(t)), \boldsymbol{x}(t)\} &= \int_V d^d x \Big[-\frac{1}{2\zeta}(1-\phi)^2(\boldsymbol{\nabla} p)^2 + \frac{1}{\zeta}(1-\phi)^2(\boldsymbol{\nabla} p)\cdot(\rho_s \boldsymbol{g}) - p\boldsymbol{\nabla}\cdot\boldsymbol{u}/\delta t + \boldsymbol{\sigma} : \boldsymbol{e}/\delta t \Big] \\ &+ \int_V d^d x \{\mu(e_{ij} - \frac{1}{d}\delta_{ij}e_{ll})^2 - \frac{K}{2}e_{ll}^2\}/\delta t \\ &- \int_V d^d x(\rho_p\phi + \rho_s(1-\phi))\boldsymbol{g}\cdot\boldsymbol{u}/\delta t \\ &- \int_S d^{d-1}x \boldsymbol{T}\cdot\boldsymbol{u}/\delta t \end{aligned} \tag{2.37}$$

Here, $\boldsymbol{e}$ is strain tensor for linear elasticity and defined as eq.(2.26). Using the linear (matrix) solver for FEM, we can minimize above functional and solve p, $\boldsymbol{u}$ and shape in time $t + \delta t$. Therefore, we have developed the semi-implicit solver for the stress-diffusion coupling of gels.
(ref. application examples 04 and 05)

Chapter 3

Sample problems of GelDyn

3.1 Sample Problems for Swelling Dynamics of Geldyn

This chapter shows the applications of gel dynamics simulator - Geldyn - by the finite element method for deformation dynamics of polymer gels. Input and output UDF files corresponding to these applications are dedicated to the directory `MUFFIN/sample/Geldyn/EX01,EX02,..` according to the problem, below the directory `MUFFIN/sample/Geldyn` of the distribution version of MUFFIN.

3.1.1 Application 01: Swelling of 2-dimensional slab gels and surface folding

Input UDF file:

`MUFFIN/sample/Geldyn/EX01/EX01_in.udf`

The description of Input UDF file:

Here, a simulation is performed using the value of the nondimensional parameter explained in the theoretical section.

- Mesh . . . type : UNSTRUCTURED_RECT, dimension : 2D, size : 64x8, division : 64x16, X-Periodicity.
- Parameters for swelling

Name of Parameters(KEY)	Values
NOIZE_OF_INITIAL_DEFORMATION	$1.0e-2$
UNIFORM_CROSSLINKING_DENSITY_IN_REFERENCE_STATE	0.025
UNIFORM_VOLUME_FRACTION_IN_REFERENCE_STATE	0.5
TYPE_OF_MIXING_FREE_ENERGY	FLORY_HUGGINS
UNIFORM_CHI_PARAMETER_IN_INITIAL_STATE	-20.0

- Boundary conditions

  Fix the lower boundary "BOUNDARY_VERTEX_YMIN".

partial region	field	condition	values
BOUNDARY_VERTEX_YMIN	Displacement	D_VEC	0.0, 0.0

- Fields to be used

  Chi, Concentration, DerivedFreeEnergy, Displacement, FingerTensor, FreeEnergy, Moduli, and VolumeFraction

- Dynamics and Method . . . Collective Diffusion Model of Gel Networks using Explicit Solver.

  Initialization procedure"INITIALIZE:COLLECTIVE_DIFFUSION_OF_GEL_NETWORKS" is defined as follows.

field	command for initialization
Chi	INITIALIZE:CHI_PARAMETER
Concentration	INITIALIZE:ION_CONCENTRATION
Moduli	INITIALIZE:MODULI
VolumeFraction	INITIALIZE:VOLUME_FRACTION
FingerTensor	INITIALIZE:FINGER_STRAIN_TENSOR
FreeEnergy	SOLVE:TOTAL_ENERGY
Displacement	MOVE:POSITION_OF_VERTEX_BY_RANDOM

Time evolution procedure"EVOLVE:COLLECTIVE_DIFFUSION_OF_GEL_NETWORKS" is defined as follows.

field	command for evolution
FingerTensor	SOLVE:FINGER_STRAIN_TENSOR
VolumeFraction	SOLVE:BY_FINGER_STRAIN_TENSOR
FreeEnergy	SOLVE:TOTAL_ENERGY
DerivedFreeEnergy	SOLVE:DERIVED_TOTAL_ENERGY
Displacement	MOVE:BY_COLLECTIVE_DIFFUSION

Results of simulation

The example which displays the volume fraction field by the view function of GOURMET is shown in Fig.3.1.1. From the left figure in time $t = 0.0, 120.0, 140.0$, and 160.0 are displayed. Action named "SHOW_SWELLING_RATIO" on GOURMET is used for a display.

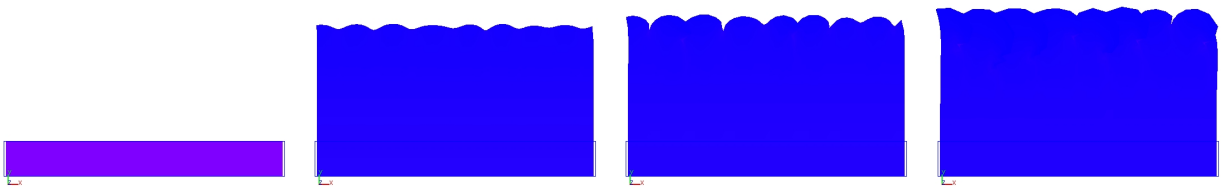

Figure 3.1: Application 01 of Geldyn : Swelling of 2-dimensional slab gels and folding pattern formations on surface

3.1.2 Application 02: Swelling of 3D plate gels and pattern formation comparison with implicit solver

Input UDF file:

For the explicit solver, `MUFFIN/sample/Geldyn/EX02/EX02-1_in.udf`.
For the implicit (matrix) solver, `MUFFIN/sample/Geldyn/EX02/EX02-2_in.udf`.

The description of Input UDF file:

Here, a simulation is performed using the value of the nondimensional parameter explained in the theoretical section.

- Mesh . . . type : UNSTRUCTURED_RECT, dimension : 3D, size : 16x16x8, division : 16x16x4, XY-Periodicity.
- Parameters for swelling

Name of Parameters(KEY)	Values
NOIZE_OF_INITIAL_DEFORMATION	$1.0e-2$
UNIFORM_CROSSLINKING_DENSITY_IN_REFERENCE_STATE	0.025
UNIFORM_VOLUME_FRACTION_IN_REFERENCE_STATE	0.5
TYPE_OF_MIXING_FREE_ENERGY	FLORY_HUGGINS
UNIFORM_CHI_PARAMETER_IN_INITIAL_STATE	-20.0

- Boundary conditions

  For "Explicit Solver", fix the lower boundary "BOUNDARY_VERTEX_ZMIN".

partial region	field	condition	values
BOUNDARY_VERTEX_ZMIN	Displacement	D_VEC	0.0, 0.0, 0.0

  For "Implicit Solver", fix the lower boundary "BOUNDARY_VERTEX_ZMIN" and set permeable condition on boundary "BOUNDARY_VERTEX_ZMAX".

partial region	field	condition	values
BOUNDARY_VERTEX_ZMIN	Displacement	D_VEC	0.0, 0.0, 0.0
BOUNDARY_VERTEX_ZMAX	Pressure	D_PERMEABLE	0.0

- Fields to be used

  For "Explicit Solver", Chi, Concentration, DerivedFreeEnergy, Displacement, FingerTensor, FreeEnergy, Moduli, and VolumeFraction are used.

  For "Implicit Solver", Chi, Concentration, Displacement, FingerTensor, FreeEnergy, Moduli, VolumeFraction, PolymerStress, VolumeForce, and Pressure are used.

- Dynamics and Method
  ... "Collective Diffusion Model using Explicit Solver" is the same as "Application 01".
  ... "Collective Diffusion Model using Implicit Solver"

  Initialization procedure"INITIALIZE:COLLECTIVE_DIFFUSION_OF_GEL_NETWORKS" is defined as follows.

field	command for initialization
Chi	INITIALIZE:CHI_PARAMETER
Concentration	INITIALIZE:ION_CONCENTRATION
Moduli	INITIALIZE:MODULI
VolumeFraction	INITIALIZE:VOLUME_FRACTION
FingerTensor	INITIALIZE:FINGER_STRAIN_TENSOR
FreeEnergy	SOLVE:TOTAL_ENERGY
Displacement	MOVE:POSITION_OF_VERTEX_BY_RANDOM

  Time evolution procedure"EVOLVE:COLLECTIVE_DIFFUSION_OF_GEL_NETWORKS:IMPLICIT" is defined as follows.

field	command for evolution
FingerTensor	SOLVE:FINGER_STRAIN_TENSOR
VolumeFraction	SOLVE:BY_FINGER_STRAIN_TENSOR
FreeEnergy	SOLVE:TOTAL_ENERGY
Moduli	SOLVE:MODULI_OF_GELS
PolymerStress	SOLVE:POLYMER_STRESS
PolymerStress	APPLY:BOUNDARY_CONDITION
VolumeForce	SOLVE:VOLUME_FORCE_BY_STRESS_AND_GRAVITY
Displacement	MOVE:BY_COLLECTIVE_DIFFUSION:LINEAR_ELASTICITY

Results of simulation

The example which displays the volume fraction field is shown in Fig.3.1.2. From the left figure in time $t = 0.0, 140.0$, and 180.0 are displayed. Action named "SHOW_SWELLING_RATIO" on GOURMET is used for a display.

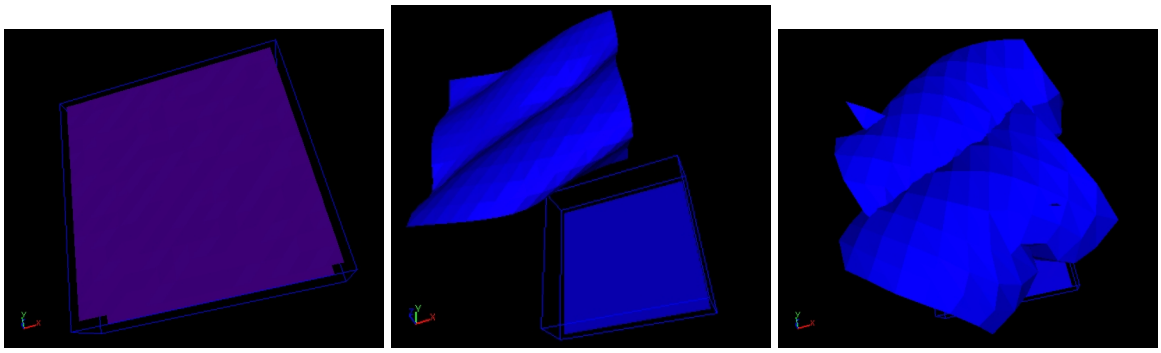

Figure 3.2: Application 02 of Geldyn : Swelling of 3D plate gels and Pattern formation

3.1.3 Application 03: Free swelling of 2D slab gels comparison of two dynamics

Input UDF file:

For the explicit collective diffusion of gel networks,
`MUFFIN/sample/Geldyn/EX03/EX03-1_in.udf`.
For the explicit stress-diffusion coupling of gels,
`MUFFIN/sample/Geldyn/EX03/EX03-2_in.udf`.

The description of Input UDF file:

Here, a simulation is performed using the value of the nondimensional parameter explained in the theoretical section.

- Mesh . . . type : UNSTRUCTURED_RECT, dimension : 2D, size : 128x16, division : 128x16.
- Parameters for swelling

Name of Parameters(KEY)	Values
NOIZE_OF_INITIAL_DEFORMATION	$1.0e-3$
UNIFORM_CROSSLINKING_DENSITY_IN_REFERENCE_STATE	0.025
UNIFORM_VOLUME_FRACTION_IN_REFERENCE_STATE	0.9
TYPE_OF_MIXING_FREE_ENERGY	GINZBURG_LANDAU
UNIFORM_CHI_PARAMETER_IN_INITIAL_STATE	0.6

- Boundary conditions

  For "Explicit Collective Diffusion Dynamics Solver", you need to input nothing. (in default, all boundaries are permeable.)

  For "Explicit Stress-Diffusion Coupling Dynamics Solver", set all boundaries to permeable condition for free swelling.

partial region	field	condition	values
BOUNDARY_VERTEX_XMIN	Pressure	D_PERMEABLE	0.0
BOUNDARY_VERTEX_XMAX	Pressure	D_PERMEABLE	0.0
BOUNDARY_VERTEX_YMIN	Pressure	D_PERMEABLE	0.0
BOUNDARY_VERTEX_YMAX	Pressure	D_PERMEABLE	0.0

- Fields to be used

  For "Explicit Collective Diffusion Solver of Gel Networks", Chi, Concentration, DerivedFreeEnergy, Displacement, FingerTensor, FreeEnergy, Moduli, and VolumeFraction are used.

  For "Explicit Stress-Diffusion Coupling Solver", Chi, Concentration, DerivedFreeEnergy, DerivedIncompressibility, Displacement, Flux, FingerTensor, FreeEnergy, Incompressibility, Moduli, VolumeFraction, PolymerStress, and Pressure.

- Dynamics and Method
  ... "Collective Diffusion Model using Explicit Solver" is the same as "Application 01".
  ... "Stress-Diffusion Coupling Model using Explicit Solver"

  Initialization procedure"INITIALIZE:STRESS_DIFFUSION_COUPLING_OF_GELS" is defined as follows.

field	command for initialization
Chi	INITIALIZE:CHI_PARAMETER
Concentration	INITIALIZE:ION_CONCENTRATION
Moduli	INITIALIZE:MODULI
VolumeFraction	INITIALIZE:VOLUME_FRACTION
FingerTensor	INITIALIZE:FINGER_STRAIN_TENSOR
FreeEnergy	SOLVE:TOTAL_ENERGY
Displacement	INITIALIZE:MINIMIZER_FOR_LOCAL_EQUILIBRIUM
Displacement	MOVE:POSITION_OF_VERTEX_BY_RANDOM

  Time evolution procedure"EVOLVE:STRESS_DIFFUSION_COUPLING_OF_GELS" is defined as follows.

field	command for evolution
FingerTensor	SOLVE:FINGER_STRAIN_TENSOR
VolumeFraction	SOLVE:BY_FINGER_STRAIN_TENSOR
FreeEnergy	SOLVE:TOTAL_ENERGY
PolymerStress	SOLVE:POLYMER_STRESS
PolymerStress	APPLY:BOUNDARY_CONDITION
Flux	SOLVE:BY_PRESSURE_COUPLING_WITH_STRESS_AND_GRAVITY
VolumeFraction	SOLVE:BY_SOLVENT_FLUX
Displacement	MOVE:LOCAL_EQUILIBRIUM_WITH_INCOMPRESSIBILITY

Results of simulation

The example which displays the volume fraction field is shown in Fig.3.1.3. From the left figure in time $t = 0.0$ and 2000.0 are displayed. Action named "SHOW_SWELLING_RATIO" on GOURMET is used for a display.

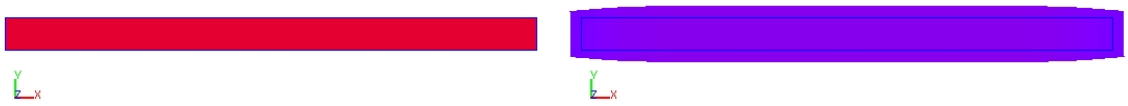

Figure 3.3: Application 03 of Geldyn : Free swelling of 2D slab gels.

3.1.4 Application 04: Free Swelling of 3D long rod gels comparison of two dynamics

Input UDF file:

For the explicit collective diffusion of gel networks,
`MUFFIN/sample/Geldyn/EX04/EX04-1_in.udf`.

For the implicit stress-diffusion coupling of gels,
`MUFFIN/sample/Geldyn/EX04/EX04-2_in.udf`.

The description of Input UDF file:

Here, a simulation is performed using the value of the nondimensional parameter explained in the theoretical section.

- Mesh . . . type : UNSTRUCTURED_RECT, dimension : 3D, size : 4x4x32, division : 16x16x32.
- Parameters for swelling

Name of Parameters(KEY)	Values
NOIZE_OF_INITIAL_DEFORMATION	$1.0e-3$
UNIFORM_CROSSLINKING_DENSITY_IN_REFERENCE_STATE	0.025
UNIFORM_VOLUME_FRACTION_IN_REFERENCE_STATE	0.9
TYPE_OF_MIXING_FREE_ENERGY	GINZBURG_LANDAU
UNIFORM_CHI_PARAMETER_IN_INITIAL_STATE	0.6

- Boundary conditions

  For "Explicit Collective Diffusion Dynamics Solver", you need to input no boundary condition. (in default, all boundaries are permeable.)

  For "Implicit Stress-Diffusion Coupling Dynamics Solver", set all boundaries to permeable condition for free swelling.

partial region	field	condition	values
BOUNDARY_VERTEX_XMIN	Pressure	D_PERMEABLE	0.0
BOUNDARY_VERTEX_XMAX	Pressure	D_PERMEABLE	0.0
BOUNDARY_VERTEX_YMIN	Pressure	D_PERMEABLE	0.0
BOUNDARY_VERTEX_YMAX	Pressure	D_PERMEABLE	0.0
BOUNDARY_VERTEX_ZMIN	Pressure	D_PERMEABLE	0.0
BOUNDARY_VERTEX_ZMAX	Pressure	D_PERMEABLE	0.0

- Fields to be used

  For "Explicit Collective Diffusion Solver of Gel Networks", Chi, Concentration, DerivedFreeEnergy, Displacement, FingerTensor, FreeEnergy, Moduli, and VolumeFraction are used.

  For "Implicit Stress-Diffusion Coupling Solver", Chi, Concentration, Displacement, FingerTensor, FreeEnergy, Moduli, VolumeFraction, PolymerStress, and Pressure.

- Dynamics and Method
  . . . "Collective Diffusion Model using Explicit Solver" is the same as "Application 01".
  . . . "Stress-Diffusion Coupling Model using Implicit Solver"

  Initialization procedure"INITIALIZE:STRESS_DIFFUSION_COUPLING_OF_GELS:IMPLICIT" is defined as follows.

field	command for initialization
Chi	INITIALIZE:CHI_PARAMETER
Concentration	INITIALIZE:ION_CONCENTRATION
Moduli	INITIALIZE:MODULI
VolumeFraction	INITIALIZE:VOLUME_FRACTION
FingerTensor	INITIALIZE:FINGER_STRAIN_TENSOR
FreeEnergy	SOLVE:TOTAL_ENERGY
Displacement	MOVE:POSITION_OF_VERTEX_BY_RANDOM

  Time evolution procedure"EVOLVE:STRESS_DIFFUSION_COUPLING_OF_GELS:IMPLICIT" is defined as follows.

field	command for evolution
FingerTensor	SOLVE:FINGER_STRAIN_TENSOR
VolumeFraction	SOLVE:BY_FINGER_STRAIN_TENSOR
FreeEnergy	SOLVE:TOTAL_ENERGY
Moduli	SOLVE:MODULI_OF_GELS
PolymerStress	SOLVE:POLYMER_STRESS
Displacement	MOVE:BY_STRESS_DIFFUSION_COUPLING:LINEAR_ELASTICITY

Results of simulation

The example which displays the volume fraction field is shown in Fig.3.1.4. From the left figure in time $t = 0.0$ and 1000.0 are displayed. Action named "SHOW_SWELLING_RATIO" on GOURMET is used for a display.

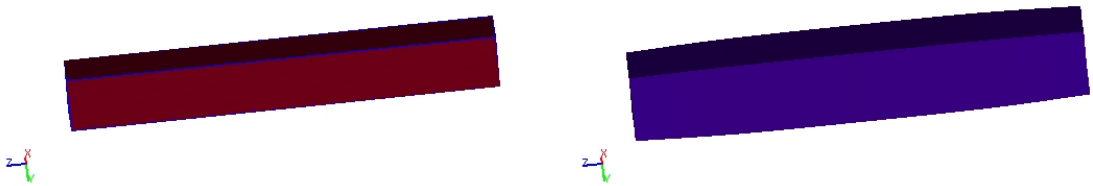

Figure 3.4: Application 04 of Geldyn : Free swelling of 3D long rod gels.

3.1.5 Application 05: Free Swelling of 3D large disk gels comparison of two dynamics

Input UDF file:

For the explicit collective diffusion of gel networks,
`MUFFIN/sample/Geldyn/EX05/EX05-1_in.udf`.
For the implicit stress-diffusion coupling of gels,
`MUFFIN/sample/Geldyn/EX05/EX05-2_in.udf`.

The description of Input UDF file:

Here, a simulation is performed using the value of the nondimensional parameter explained in the theoretical section.

- Mesh . . . type : UNSTRUCTURED_RECT, dimension : 3D, size : 32x32x4, division : 32x32x16.
- Parameters for swelling . . . same as Application 04.
- Boundary conditions . . . same as Application 04.
- Fields to be used . . . same as Application 04.
- Dynamics and Method
  . . . "Collective Diffusion Model using Explicit Solver" is the same as "Application 01".
  . . . "Stress-Diffusion Coupling Model using Implicit Solver" is the same as "Application 04".

Results of simulation

The example which displays the volume fraction field is shown in Fig.3.1.5. From the left figure in time $t = 0.0$ and 1000.0 are displayed. Action named "SHOW_SWELLING_RATIO" on GOURMET is used for a display.

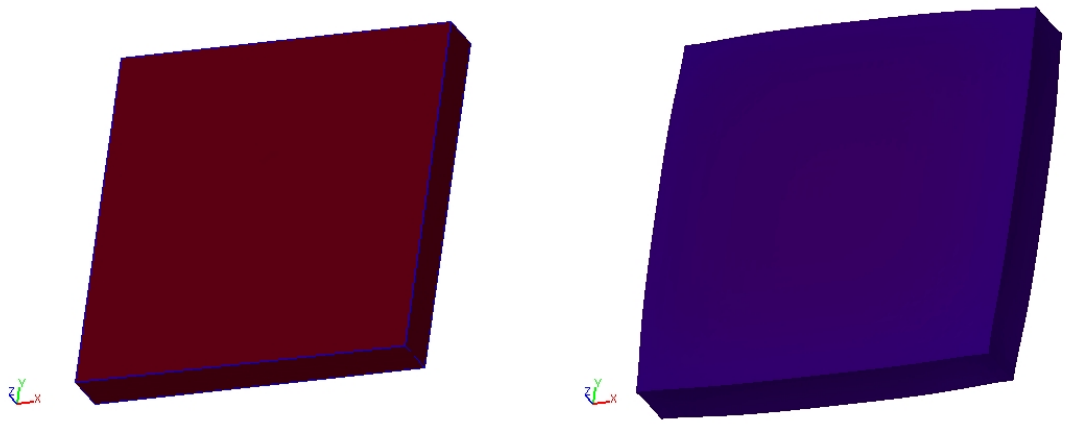

Figure 3.5: Application 05 of Geldyn : Free swelling of 3D large disk gels.

Chapter 4

Operation Guide of GelDyn

4.1 Input Parameters of Geldyn

4.1.1 Solver control parameters of Geldyn

Name of Parameters	Meanings and notations in theory
INTERVAL_OF_MINIMIZER _OUTPUT	Interval of monitoring for minimizer.
MAX_ITERATION_OF_MINIMIZER	Max iteration of minimizer.
DT_FOR_LOCAL_EQUILIBRIUM _MINIMIZER	DT for minimizer in the explicit stress-diffusion coupling solver.
DT_FOR_STATIC_EQUILIBRIUM _MINIMIZER	DT for "SIMPLEMIN" minimizer in the explicit static equilibrium solver.
ENERGY_WEIGHT_IN_MINIMIZER	Weight of elastic energy in the explicit stress-diffusion coupling minimizer.
INCOMPRESSIBILITY_WEIGHT _IN_MINIMIZER	Weight of error of incompressibility (large number) in the explicit stress-diffusion coupling minimizer.
MATRIX_SOLVER	Linear equation (matrix equation) solver name for implicit solvers to be used. Either "ICCG" or "CG". Default is "ICCG".
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.
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.

4.1.2 Physical parameters of Geldyn

Name of Parameters	Meanings and notations in theory
SEED_OF_RANDOM_NUMBER	Seed of rundom number for initial deformation by random.
NOIZE_OF_INITIAL_DEFORMATION	Magnitude of initial deformation by random.
EXPONENT_NU	Exponent ν for volume fraction dependency to blob size. ν equals 1/2 for θ-solvent and 3/5 for good-solvent.
MAGNITUDE_OF_ELASTIC_LOGTERM	Magnitude of elastic logarithmic term. (default 0.0)
TYPE_OF_MIXING_FREE_ENERGY	Type of mixing free energy of gels to be used. Either "FLORY_HUGGINS" or "GINZBURG_RANDAU" Default is "GINZBURG_RANDAU".
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 polymer and solvent (mass density when volume fraction is 1.0).
UNIFORM_VOLUME_FRACTION _IN_REFERENCE_STATE	Uniform volume fraction in reference state. If this parameter does't exist, UDF input field data ("VolumeFraction") are applied.
UNIFORM_CROSSLINKING_DENSITY _IN_REFERENCE_STATE	Uniform crosslingking number density in reference state. If this parameter does't exist, UDF input field data (1st component of "Moduli") are applied.
UNIFORM_SHEAR_MODULUS	Uniform isotropic shear modulus for implicit solver. If this parameter does't exist, UDF input field data (2nd component of "Moduli") are applied.
UNIFORM_BULK_MODULUS	Uniform isotropic bulk modulus for implicit solver. If this parameter does't exist, UDF input field data (3rd component of "Moduli") are applied.
UNIFORM_CHI_PARAMETER _IN_INITIAL_STATE	Uniform polymer-solvent interaction parameter (in initial state). If this parameter does't exist, UDF input field data (1st component of "Chi") are applied.
STIMULI_OF_QUENCH	Type of stimuli for quench to be used. Either "CHI" (change χ-parameter field) or "TEMPERATURE"(change temperature). Default is "CHI".
TEMPERATURE_IN_INITIAL_STATE	Temperature of system in initial state (defore quench). Input this parameter, if "TEMPERATURE" is selected as STIMULI_OF_QUENCH.
TEMPERATURE_IN_FINAL_STATE	Temperature of system in final state (after quench). Input this parameter, if "TEMPERATURE" is selected as STIMULI_OF_QUENCH.
UNIFORM_CHI_PARAMETER _IN_FINAL_STATE	Uniform polymer-solvent interaction parameter in final state (after quench), when "CHI" is selected as STIMULI_OF_QUENCH. If this parameter does't exist, UDF input field data (2nd component of "Chi") are applied.
UNIFORM_ION_CONCENTRATION _IN_REFERENCE_STATE	Uniform ion concentration (in initial state). If this parameter does't exist, UDF input field data (1st component of "Concentration") are applied.

4.1.3 Analysis results of Geldyn

Name of Results	Meanings and notations in theory
TOTAL_FREE_ENERGY	Total free energy of gels
TOTAL_ELASTIC_FREE_ENERGY	Total elastic energy of gels
TOTAL_ERROR_OF_INCOMPRESSIBILITY	Total error of incompressibility of gels for the explicit stress-diffusion coupling solver

4.2 Fields and Commands for Fields of Geldyn

4.2.1 List of Fields for Geldyn

Name of Field	Meanings and notations in theory
Displacement	Displacement (Velocity) of polymer network (essential)
FingerTensor	Finger tensor of polymer network (essential)
VolumeFraction	Volume fraction of polymer (essential)
Moduli	The crosslinking number density, bulk and shear modulus (essential)
Chi	Polymer-solvent interaction parameter (essential)
Concentration	Counter ions concentration (essential)
FreeEnergy	Free energy density of gels (for the explicit solvers)
PolymerStress	Polymer network stress tensor (for the stress-diffusion coupling solvers and the implicit solver of the collective diffusion solver)
Pressure	Pressure of solvent (for the stress-diffusion coupling solvers)
Flux	Flux of solvent (for the stress-diffusion coupling solvers)
VolumeForce	Volume force of gels (for the implicit collective diffusion solver)
DerivedFreeEnergy	Derived free energy on vertex (for the explicit stress-diffusion coupling solver)
Incompressibility	Error of incompressibility of gels (for the explicit stress-diffusion coupling solver)
DerivedIncompressibility	Derived error of incompressibility of gels on vertex (for the explicit stress-diffusion coupling solver)

4.2.2 List of Commands for Geldyn

VolumeFraction : Volume fraction of polymer - commands

VolumeFraction	Name
Initialization	"INITIALIZE:VOLUME_FRACTION"
Time evolution	"SOLVE:BY_FINGER_STRAIN_TENSOR"
Time evolution	"SOLVE:BY_SOLVENT_FLUX"

1. **VolumeFraction - initialization commands**

Name	**"INITIALIZE:VOLUME_FRACTION"**
Function	Initialize volume fraction of polymer from morphology input data or uniform parameter. If uniform parameter doesn't exist, field data is inputted.
Dependent parameter	UNIFORM_VOLUME_FRACTION_IN_REFERENCE_STATE

2. **VolumeFraction - time evolution commands**

Name	**"SOLVE:BY_FINGER_STRAIN_TENSOR"**
Function	Solve volume fraction of polymer by finger strain tensor
Dependent field	FingerTensor

Name	**"SOLVE:BY_SOLVENT_FLUX"**
Function	One step time integration of equation for volume fraction
Dependent field	Flux
Dependent parameter	DT

Displacement : Displacement of polymer network - commands

Displacement	Name
Initialization	"INITIALIZE:MINIMIZER_FOR_LOCAL_EQUILIBRIUM"
Time evolution	"MOVE:POSITION_OF_VERTEX"
Time evolution	"MOVE:POSITION_OF_VERTEX_BY_RANDOM"
Time evolution	"MOVE:BY_COLLECTIVE_DIFFUSION"
Time evolution	"MOVE:GO_TO_STATIC_EQUILIBRIUM"
Time evolution	"MOVE:LOCAL_EQUILIBRIUM_WITH_INCOMPRESSIBILITY"
Time evolution	"MOVE:BY_COLLECTIVE_DIFFUSION:LINEAR_ELASTICITY"
Time evolution	"MOVE:BY_STRESS_DIFFUSION_COUPLING:LINEAR_ELASTICITY"
Time evolution	"MOVE:BY_LINEAR_ELASTICITY_DYNAMICS"

1. **Displacement - initialization commands**

Name	**"INITIALIZE:MINIMIZER_FOR_LOCAL_EQUILIBRIUM"**
Function	Initialize minimizer for the explicit stress-diffusion coupling solver
Dependent parameter	MINIMIZER
Dependent parameter	MINIMIZER_REGION_MIN_X
Dependent parameter	MINIMIZER_REGION_MAX_X
Dependent parameter	MINIMIZER_REGION_MIN_Y
Dependent parameter	MINIMIZER_REGION_MAX_Y
Dependent parameter	MINIMIZER_REGION_MIN_Z
Dependent parameter	MINIMIZER_REGION_MAX_Z

2. **Displacement - time evolution commands**

Name	**"MOVE:POSITION_OF_VERTEX"**
Function	Move position of vertices and deform mesh with dirichlet BC

Name	**"MOVE:POSITION_OF_VERTEX_BY_RANDOM"**
Function	Move position of vertices by randomize with dirichlet BC
Dependent parameter	SEED_OF_RANDOM_NUMBER
Dependent parameter	NOIZE_OF_INITIAL_DEFORMATION

Name	**"MOVE:BY_COLLECTIVE_DIFFUSION"**
Function	Solve the explicit collective diffusion solver with dirichlet BC
Dependent field	FreeEnergy
Dependent field	DerivedFreeEnergy
Dependent field	VolumeFraction
Dependent field	FingerTensor
Dependent parameter	DT
Dependent parameter	EXPONENT_NU

Name	**"MOVE:GO_TO_STATIC_EQUILIBRIUM"**
Function	Solve the static equilibrium shape and deform with dirichlet BC
Dependent field	FreeEnergy
Dependent field	DerivedFreeEnergy
Dependent field	VolumeFraction
Dependent field	FingerTensor
Dependent parameter	MINIMIZER
Dependent parameter	MINIMIZER_REGION_MIN_X
Dependent parameter	MINIMIZER_REGION_MAX_X
Dependent parameter	MINIMIZER_REGION_MIN_Y
Dependent parameter	MINIMIZER_REGION_MAX_Y
Dependent parameter	MINIMIZER_REGION_MIN_Z
Dependent parameter	MINIMIZER_REGION_MAX_Z
Dependent parameter	INTERVAL_OF_MINIMIZER_OUTPUT
Dependent parameter	MAX_ITERATION_OF_MINIMIZER
Dependent parameter	DT_FOR_STATIC_EQUILIBRIUM_MINIMIZER

Name	**"MOVE:LOCAL_EQUILIBRIUM _WITH_INCOMPRESSIBILITY"**
Function	Solve the explicit stress-diffusion coupling solver with dirichlet BC
Dependent field	FreeEnergy
Dependent field	DerivedFreeEnergy
Dependent field	Incompressibility
Dependent field	DerivedIncompressibility
Dependent field	VolumeFraction
Dependent field	FingerTensor
Dependent parameter	ENERGY_WEIGHT_IN_MINIMIZER
Dependent parameter	INCOMPRESSIBILITY_WEIGHT_IN_MINIMIZER
Dependent parameter	INTERVAL_OF_MINIMIZER_OUTPUT
Dependent parameter	MAX_ITERATION_OF_MINIMIZER
Dependent parameter	DT_FOR_LOCAL_EQUILIBRIUM_MINIMIZER

Name	**"MOVE:BY_COLLECTIVE_DIFFUSION :LINEAR_ELASTICITY"**
Function	Solve the implicit collective diffusion solver with BC
Dependent field	VolumeForce
Dependent field	Moduli
Dependent field	VolumeFraction
Dependent parameter	MATRIX_SOLVER
Dependent parameter	ELEMENTS_PER_MATRIX_MERGE
Dependent parameter	PENALTY_NUMBER_FOR_DIRICHLET_BC
Dependent parameter	DT
Dependent parameter	EXPONENT_NU
Dependent parameter	CONVERGENCE_CRITERION_FOR_CG_1
Dependent parameter	CONVERGENCE_CRITERION_FOR_CG_2

Name	**"MOVE:BY_STRESS_DIFFUSION_COUPLING :LINEAR_ELASTICITY"**
Function	Solve pressure and deformation by the implicit stress-diffusion coupling solver with BC
Dependent field	Pressure
Dependent field	Moduli
Dependent field	VolumeFraction
Dependent field	PolymerStress
Dependent parameter	MATRIX_SOLVER
Dependent parameter	ELEMENTS_PER_MATRIX_MERGE
Dependent parameter	PENALTY_NUMBER_FOR_DIRICHLET_BC
Dependent parameter	DT
Dependent parameter	EXPONENT_NU
Dependent parameter	CONVERGENCE_CRITERION_FOR_CG_1
Dependent parameter	CONVERGENCE_CRITERION_FOR_CG_2
Dependent parameter	GRAVITY
Dependent parameter	GRAVITY_X
Dependent parameter	GRAVITY_Y
Dependent parameter	GRAVITY_Z
Dependent parameter	MASS_DENSITY

Name	**"MOVE:BY_LINEAR_ELASTICITY_DYNAMICS**
Function	Solve the linear elasticity dynamics with BC
Dependent field	VolumeForce
Dependent field	Moduli
Dependent parameter	MATRIX_SOLVER
Dependent parameter	ELEMENTS_PER_MATRIX_MERGE
Dependent parameter	PENALTY_NUMBER_FOR_DIRICHLET_BC
Dependent parameter	DT
Dependent parameter	CONVERGENCE_CRITERION_FOR_CG_1
Dependent parameter	CONVERGENCE_CRITERION_FOR_CG_2

3. **Displacement - partial region condition (boundary condition) commands**

Partial region condition	meanings and parameters
D_VEC	Set velocity 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 velocity vector.
D_VY	Y component of fixed velocity vector.
D_VZ	Z component of fixed velocity vector.
N_LOAD	Set load on vertices in specified partial region(fixed load condition). Give a 3-dimensional vector.
N_LOAD_NORMAL	Set load on vertices normal direction of specified partial region(fixed load condition). positive: outer direction, negative: inner direction.

FingerTensor : Finger tensor of polymer network - commands

FingerTensor	Name
Initialization	"INITIALIZE:FINGER_STRAIN_TENSOR"
Time evolution	"SOLVE:FINGER_STRAIN_TENSOR"

1. **FingerTensor - initialization commands**

Name	**"INITIALIZE:FINGER_STRAIN_TENSOR"**
Function	Initialize finger strain tensor and set reference mesh shape

2. **FingerTensor - time evolution commands**

Name	**"SOLVE:FINGER_STRAIN_TENSOR"**
Function	Solve finger strain tensor by present mesh shape

PolymerStress : Polymer network stress tensor - commands

PolymerStress	Name
Time evolution	"SOLVE:POLYMER_STRESS"
Time evolution	"APPLY:BOUNDARY_CONDITION"

1. **PolymerStress - time evolution commands**

Name	**"SOLVE:POLYMER_STRESS"**
Function	Solve polymer network stress tensor by finger strain tensor
Dependent field	VolumeFraction
Dependent field	FingerTensor
Dependent field	Chi
Dependent field	Moduli
Dependent field	Concentration
Dependent parameter	MAGNITUDE_OF_ELASTIC_LOGTERM
Dependent parameter	TYPE_OF_MIXING_FREE_ENERGY

Name	**"APPLY:BOUNDARY_CONDITION"**
Function	Apply boundary condition from BC of pressure and load of displacement
Dependent field	Pressure
Dependent field	Displacement

Pressure : Pressure of solvent - commands

1. **Pressure - partial region condition (boundary condition) commands**

Partial region condition	meanings and parameters
D_PERMEABLE	set a constant value (Dirichlet condition) for permeable surface. (only stress-diffusion coupling) Other surfaces are considered as impermeable wall. (cf. In the collective diffusion model, all surfaces are considered as permeable ones.)

Flux : Flux of solvent - commands

Flux	Name
Time evolution	"SOLVE:BY_PRESSURE_COUPLING_WITH_STRESS_AND_GRAVITY"

1. **Flux - time evolution commands**

Name	**"SOLVE:BY_PRESSURE_COUPLING _WITH_STRESS_AND_GRAVITY"**
Function	Solve flux of solvent with gravity and BC
Dependent field	VolumeFraction
Dependent field	PolymerStress
Dependent field	Pressure
Dependent field	Displacement
Dependent parameter	EXPONENT_NU
Dependent parameter	GRAVITY
Dependent parameter	GRAVITY_X
Dependent parameter	GRAVITY_Y
Dependent parameter	GRAVITY_Z
Dependent parameter	MASS_DENSITY

VolumeForce : Volume force of gels - commands

VolumeForce	Name
Time evolution	"SOLVE:VOLUME_FORCE_BY_STRESS_AND_GRAVITY"
Time evolution	"SOLVE:VOLUME_FORCE_BY_GRAVITY"

1. **VolumeForce - time evolution commands**

Name	**"SOLVE:VOLUME_FORCE_BY_STRESS_AND_GRAVITY"**
Function	Solve volume force of gels by gravity and gradient of polymer stress
Dependent field	VolumeFraction
Dependent field	PolymerStress
Dependent parameter	GRAVITY
Dependent parameter	GRAVITY_X
Dependent parameter	GRAVITY_Y
Dependent parameter	GRAVITY_Z
Dependent parameter	MASS_DENSITY

Name	**"SOLVE:VOLUME_FORCE_BY_GRAVITY"**
Function	Solve volume force of gels by gravity
Dependent field	VolumeFraction
Dependent parameter	GRAVITY
Dependent parameter	GRAVITY_X
Dependent parameter	GRAVITY_Y
Dependent parameter	GRAVITY_Z
Dependent parameter	MASS_DENSITY

Moduli : The crosslinking number density, bulk and shear modulus - commands

Moduli	Name
Initialization	"INITIALIZE:MODULI"
Time evolution	"SOLVE:MODULI_OF_GELS"

1. **Moduli - initialization commands**

Name	**"INITIALIZE:MODULI"**
Function	Initialize crosslingking number density in reference state, or shear and bulk modulus from input field data or uniform parameter If uniform parameter doesn't exist, field data is inputted.
Dependent parameter	UNIFORM_CROSSLINKING_DENSITY_IN_REFERENCE_STATE
Dependent parameter	UNIFORM_SHEAR_MODULUS
Dependent parameter	UNIFORM_BULK_MODULUS

2. **Moduli - time evolution commands**

Name	**"SOLVE:MODULI_OF_GELS"**
Function	Solve isotropic bulk and shear modulus field of gels.
Dependent field Dependent field Dependent field	VolumeFraction Chi Concentration
Dependent parameter Dependent parameter	MAGNITUDE_OF_ELASTIC_LOGTERM TYPE_OF_MIXING_FREE_ENERGY

Chi : Polymer-solvent interaction parameter - commands

Chi	Name
Initialization	"INITIALIZE:CHI_PARAMETER"
Time evolution	"QUENCH:CHANGE_CHI_PARAMETER"

1. **Chi - initialization commands**

Name	**"INITIALIZE:CHI_PARAMETER"**
Function	Initialize χ-parameter from input field data or uniform parameter If uniform parameter doesn't exist, field data is inputted.
Dependent parameter	UNIFORM_CHI_PARAMETER_IN_INITIAL_STATE

2. **Chi - time evolution commands**

Name	**"QUENCH:CHANGE_CHI_PARAMETER"**
Function	Change temperature or χ-parameter field and quench the system.
Dependent parameter Dependent parameter Dependent parameter Dependent parameter	STIMULI_OF_QUENCH TEMPERATURE_IN_INITIAL_STATE TEMPERATURE_IN_FINAL_STATE UNIFORM_CHI_PARAMETER_IN_FINAL_STATE

Concentration : Counter ions concentration - commands

Concentration	Name
Initialization	"INITIALIZE:ION_CONCENTRATION"

1. **Concentration - initialization commands**

Name	**"INITIALIZE:ION_CONCENTRATION"**
Function	Initialize counter ions concentration from input field data or uniform parameter If uniform parameter doesn't exist, field data is inputted.
Dependent parameter	UNIFORM_ION_CONCENTRATION_IN_REFERENCE_STATE

FreeEnergy : Free energy and elastic energy of gels - commands

FreeEnergy	Name
Time evolution	"SOLVE:TOTAL_ENERGY"
Time evolution	"SOLVE:ELASTIC_ENERGY"

1. **FreeEnergy - time evolution commands**

Name	**"SOLVE:TOTAL_ENERGY"**
Function	Solve free energy of gels and analyze the total free energy
Dependent field	VolumeFraction
Dependent field	FingerTensor
Dependent field	Chi
Dependent field	Moduli
Dependent field	Concentration
Dependent parameter	MAGNITUDE_OF_ELASTIC_LOGTERM
Dependent parameter	TYPE_OF_MIXING_FREE_ENERGY
Dependent result	TOTAL_FREE_ENERGY

Name	**"SOLVE:ELASTIC_ENERGY"**
Function	Solve elastic energy of gels and analyze the total elastic energy
Dependent field	VolumeFraction
Dependent field	FingerTensor
Dependent field	Moduli
Dependent parameter	MAGNITUDE_OF_ELASTIC_LOGTERM
Dependent result	TOTAL_ELASTIC_FREE_ENERGY

Incompressibility : Error of incompressibility of gels - commands

Incompressibility	Name
Time evolution	"SOLVE:INCOMPRESSIBILITY"

1. **Incompressibility - initialization commands**

2. **Incompressibility - time evolution commands**

Name	**"SOLVE:INCOMPRESSIBILITY"**
Function	Solve error of incompressibility of gels and analyze the total error of incompressibility
Dependent field	VolumeFraction
Dependent result	TOTAL_ERROR_OF_INCOMPRESSIBILITY

DerivedFreeEnergy : Derived free energy on vertex - commands

DerivedFreeEnergy	Name
Time evolution	"SOLVE:DERIVED_TOTAL_ENERGY"
Time evolution	"SOLVE:DERIVED_ELASTIC_ENERGY"

1. **DerivedFreeEnergy - time evolution commands**

Name	**"SOLVE:DERIVED_TOTAL_ENERGY"**
Function	Solve derived free energy of gels on vertices
Dependent field	VolumeFraction
Dependent field	FingerTensor
Dependent field	Chi
Dependent field	Moduli
Dependent field	Concentration
Dependent parameter	MAGNITUDE_OF_ELASTIC_LOGTERM
Dependent parameter	TYPE_OF_MIXING_FREE_ENERGY

Name	**"SOLVE:DERIVED_ELASTIC_ENERGY"**
Function	Solve derived elastic energy of gels on vertices
Dependent field	VolumeFraction
Dependent field	FingerTensor
Dependent field	Moduli
Dependent parameter	MAGNITUDE_OF_ELASTIC_LOGTERM

DerivedIncompressibility : Derived error of incompressibility of gels - commands

DerivedIncompressibility	Name
Time evolution	"SOLVE:DERIVED_INCOMPRESSIBILITY"

1. **DerivedIncompressibility - time evolution commands**

Name	**"SOLVE:DERIVED_INCOMPRESSIBILITY"**
Function	Solve derived error of incompressibility of gels on vertices
Dependent field	VolumeFraction
Dependent field	FingerTensor

References

1) T.Tanaka, L. and G.B.Benedek, : *J.Chem.Phys*, Vol. 59, p. 5151 (1973).

2) T.Tanaka, and D.J.Fillmore, : *J.Chem.Phys*, Vol. 70, p. 1214 (1979).

3) M.Doi, A. and K.Kawasaki, e. eds.: *Dynamics and Patters in Complex Fluids*, Springer (1990).

4) Y.Li, and T.Tanaka, : *J.Chem.Phys*, Vol. 92, p. 1365 (1990).

5) T.Yamaue, T. and M.Doi, : *AIP Conference Proceedings*, Vol. 519, p. 584 (2000).

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

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

8) M.Doi, and A.Onuki, : *J.Phys.II France*, Vol. 2, p. 1631 (1992).