Coupled reaction-diffusion equations

A solver for general coupled reaction-diffusion equations has recently been written.

$$\frac{\partial u_i}{\partial t} - \nabla \cdot (D_i \boldsymbol{\nabla} u_i) = f_i(\mathbf{x}, u_1, \ldots, u_p, v_1, \ldots, v_p),$$

$$\frac{\mathrm{d}v_i}{\mathrm{d}t} = g_i(\mathbf{x}, u_1, \ldots, u_p, v_1, \ldots, v_p),$$

where u_i and v_i denote the extracellular and intracellular concentrations of solute i respectively, and with BCs

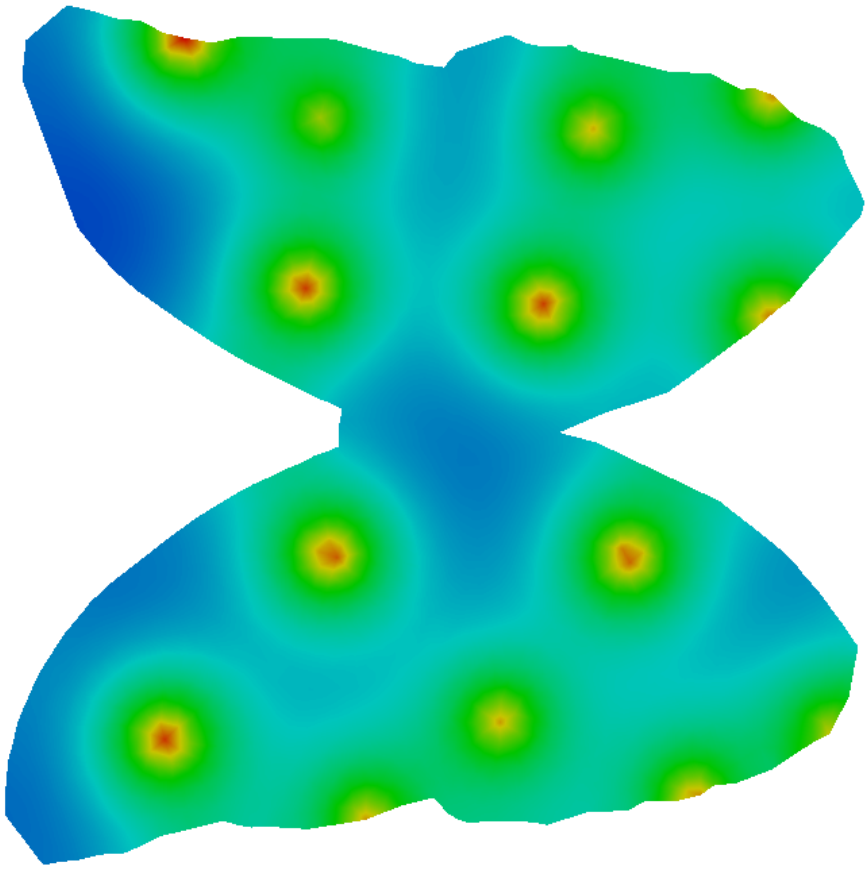
$$u_i = u_i^*(\mathbf{x}), \qquad \text{on } \Gamma_1$$

$$\mathbf{n} \cdot (D_i(\mathbf{x}) \nabla u_i) = g_i(\mathbf{x}), \qquad \text{on } \Gamma_2$$

$$u_i(0, \mathbf{x}) = u_i^{(0)}(\mathbf{x}),$$

$$v_i(0, \mathbf{x}) = v_i^{(0)}(\mathbf{x}),$$

Coupled reaction-diffusion equations



See the tutorial `LinearParabolicPdeSystemsWithCoupledOdeSystems` for more details

An overview of alternative methods for solving PDEs

The finite difference method

- Finite differences are conceptually the simplest method for solving PDEs
- Set up a (generally regular) grid on the geometry, aim to compute the solution at the gridpoints
- All derivatives in the PDE (and in Neumann BCs) are replaced with difference formulas

For a regular grid in 1D $x_0, x_1, \ldots, x_N$, stepsize h: some possible difference formulas and corresponding error introduced are: forward, backward and central differences:

$$\begin{aligned}
\frac{\mathrm{d}u}{\mathrm{d}x}(x_i) &= \frac{x_{i+1} - x_i}{h} + \mathcal{O}(h) \\
\frac{\mathrm{d}u}{\mathrm{d}x}(x_i) &= \frac{x_i - x_{i-1}}{h} + \mathcal{O}(h) \\
\frac{\mathrm{d}u}{\mathrm{d}x}(x_i) &= \frac{x_{i+1} - x_{i-1}}{2h} + \mathcal{O}(h^2)
\end{aligned}$$

and

$$\frac{\mathrm{d}^2u}{\mathrm{d}x^2}(x_i) = \frac{x_{i+1} - 2x_i + x_{i-1}}{h^2} + \mathcal{O}(h^2)$$

The finite difference method

- Finite differences are conceptually the simplest method for solving PDEs
- Set up a (generally regular) grid on the geometry, aim to compute the solution at the gridpoints
- All derivatives in the PDE (and in Neumann BCs) are replaced with difference formulas

For a regular grid in 1D $x_0, x_1, \ldots, x_N$, stepsize h: some possible difference formulas and corresponding error introduced are: forward, backward and central differences:

$$\begin{aligned}
\frac{\mathrm{d}u}{\mathrm{d}x}(x_i) &= \frac{x_{i+1} - x_i}{h} + \mathcal{O}(h) \\
\frac{\mathrm{d}u}{\mathrm{d}x}(x_i) &= \frac{x_i - x_{i-1}}{h} + \mathcal{O}(h) \\
\frac{\mathrm{d}u}{\mathrm{d}x}(x_i) &= \frac{x_{i+1} - x_{i-1}}{2h} + \mathcal{O}(h^2)
\end{aligned}$$

and

$$\frac{\mathrm{d}^2u}{\mathrm{d}x^2}(x_i) = \frac{x_{i+1} - 2x_i + x_{i-1}}{h^2} + \mathcal{O}(h^2)$$

The finite difference method

For example, for the heat equation $u_t = u_{xx} + f$, suppose we choose a fully explicit time-discretisation, and then discretise in space:

Heat equation:	$u_t = u_{xx} + f$
Semi-discretised:	$u^{n+1} - u^n = \Delta t\, u^n_{xx} + \Delta t\, f(t_n)$
Fully discretised:	$u_i^{n+1} - u_i^n = \frac{\Delta t}{h^2}(u^n_{i+1} - 2u^n_i + u^n_{i-1}) + \Delta t\, f(t_n, x_i)$

(Since this is an explicit scheme there will be a condition required for stability: $\frac{\Delta t}{h^2} \leq \frac{1}{2}$. Such results are obtained using *Von Neumann (Fourier) stability analysis*).

Let us write the above as a linear system: $\mathbf{U}^{n+1} = \mathbf{U}^n + \frac{\Delta t}{h^2} D\mathbf{U}^n + \Delta t\, \mathbf{F}^n$, where $U^n = [U_1^n, \ldots, U_N^n]$, $F^n = [f(t_n, x_1), \ldots, f(t_n, x_N)]$, and D is a matrix with -2s on the diagonal and 1s above and below the diagonal.

The first and last rows of the linear system have to be altered to take into account Dirichlet and/or **Neumann** boundary conditions.

The finite difference method

For example, for the heat equation $u_t = u_{xx} + f$, suppose we choose a fully explicit time-discretisation, and then discretise in space:

Heat equation:	$u_t = u_{xx} + f$
Semi-discretised:	$u^{n+1} - u^n = \Delta t\, u^n_{xx} + \Delta t\, f(t_n)$
Fully discretised:	$u_i^{n+1} - u_i^n = \frac{\Delta t}{h^2}(u^n_{i+1} - 2u^n_i + u^n_{i-1}) + \Delta t\, f(t_n, x_i)$

(Since this is an explicit scheme there will be a condition required for stability: $\frac{\Delta t}{h^2} \leq \frac{1}{2}$. Such results are obtained using *Von Neumann (Fourier) stability analysis*).

Let us write the above as a linear system: $\mathbf{U}^{n+1} = \mathbf{U}^n + \frac{\Delta t}{h^2} D\mathbf{U}^n + \Delta t\, \mathbf{F}^n$, where $U^n = [U_1^n, \ldots, U_N^n]$, $F^n = [f(t_n, x_1), \ldots, f(t_n, x_N)]$, and D is a matrix with -2s on the diagonal and 1s above and below the diagonal.

The first and last rows of the linear system have to be altered to take into account Dirichlet and/or **Neumann** boundary conditions.

The finite difference method

For example, for the heat equation $u_t = u_{xx} + f$, suppose we choose a fully explicit time-discretisation, and then discretise in space:

$$\begin{aligned}
&\text{Heat equation:} && u_t = u_{xx} + f \\
&\text{Semi-discretised:} && u^{n+1} - u^n = \Delta t\, u^n_{xx} + \Delta t\, f(t_n) \\
&\text{Fully discretised:} && u^{n+1}_i - u^n_i = \frac{\Delta t}{h^2}(u^n_{i+1} - 2u^n_i + u^n_{i-1}) + \Delta t\, f(t_n, x_i)
\end{aligned}$$

(Since this is an explicit scheme there will be a condition required for stability: $\frac{\Delta t}{h^2} \leq \frac{1}{2}$. Such results are obtained using *Von Neumann (Fourier) stability analysis*).

Let us write the above as a linear system: $\mathbf{U}^{n+1} = \mathbf{U}^n + \frac{\Delta t}{h^2} D\mathbf{U}^n + \Delta t\, \mathbf{F}^n$, where $U^n = [U^n_1, \ldots, U^n_N]$, $F^n = [f(t_n, x_1), \ldots, f(t_n, x_N)]$, and D is a matrix with -2s on the diagonal and 1s above and below the diagonal.

The first and last rows of the linear system have to be altered to take into account Dirichlet and/or **Neumann** boundary conditions.

The finite difference method

For example, for the heat equation $u_t = u_{xx} + f$, suppose we choose a fully explicit time-discretisation, and then discretise in space:

Heat equation:	$u_t = u_{xx} + f$
Semi-discretised:	$u^{n+1} - u^n = \Delta t\, u^n_{xx} + \Delta t\, f(t_n)$
Fully discretised:	$u_i^{n+1} - u_i^n = \frac{\Delta t}{h^2}(u^n_{i+1} - 2u^n_i + u^n_{i-1}) + \Delta t\, f(t_n, x_i)$

(Since this is an explicit scheme there will be a condition required for stability: $\frac{\Delta t}{h^2} \leq \frac{1}{2}$. Such results are obtained using *Von Neumann (Fourier) stability analysis*).

Let us write the above as a linear system: $\mathbf{U}^{n+1} = \mathbf{U}^n + \frac{\Delta t}{h^2} D\mathbf{U}^n + \Delta t\, \mathbf{F}^n$, where $U^n = [U_1^n, \ldots, U_N^n]$, $F^n = [f(t_n, x_1), \ldots, f(t_n, x_N)]$, and D is a matrix with -2s on the diagonal and 1s above and below the diagonal.

The first and last rows of the linear system have to be altered to take into account Dirichlet and/or **Neumann** boundary conditions.

The finite difference method

Compare the finite difference equation:

$$\mathbf{U}^{n+1} = \mathbf{U}^n + \frac{\Delta t}{h^2} D\mathbf{U}^n + \Delta t \mathbf{F}^n$$

with the equivalent finite element equation:

$$M\mathbf{U}^{n+1} = M\mathbf{U}^n + \Delta t\, K\mathbf{U}^n + \Delta t \mathbf{b}^n$$

In fact, for a regular grid in 1D and with linear basis functions, $K = D/h$ (except for first/last rows).

The big advantages of FE over FD are

- FD is difficult to write down on *irregular geometries*, but FE works ON any valid mesh
- Neumann boundary conditions are handled very naturally in FE — require an integral over surfaces — but nothing required for zero-Neumann BCs. Much more difficult in FD.
- Error analysis

The finite difference method

Compare the finite difference equation:

$$\mathbf{U}^{n+1} = \mathbf{U}^n + \frac{\Delta t}{h^2} D\mathbf{U}^n + \Delta t \mathbf{F}^n$$

with the equivalent finite element equation:

$$M\mathbf{U}^{n+1} = M\mathbf{U}^n + \Delta t\, K\mathbf{U}^n + \Delta t \mathbf{b}^n$$

In fact, for a regular grid in 1D and with linear basis functions, $K = D/h$ (except for first/last rows).

The big advantages of FE over FD are

- FD is difficult to write down on irregular geometries, but FE works ON any valid mesh
- Neumann boundary conditions are handled very naturally in FE—require an integral over surface, and nothing required for zero-Neumann BCs. Much more difficult in FD
- Error analysis

The finite difference method

Compare the finite difference equation:

$$\mathbf{U}^{n+1} = \mathbf{U}^n + \frac{\Delta t}{h^2} D\mathbf{U}^n + \Delta t \mathbf{F}^n$$

with the equivalent finite element equation:

$$M\mathbf{U}^{n+1} = M\mathbf{U}^n + \Delta t\, K\mathbf{U}^n + \Delta t \mathbf{b}^n$$

In fact, for a regular grid in 1D and with linear basis functions, $K = D/h$ (except for first/last rows).

The big advantages of FE over FD are

- FD is difficult to write down on **irregular geometries**, but FE works ON any valid mesh
- **Neumann boundary conditions** are handled very naturally in FE—require an integral over surface, and nothing required for zero-Neumann BCs. Much more difficult in FD
- **Error analysis**

The finite difference method

Compare the finite difference equation:

$$\mathbf{U}^{n+1} = \mathbf{U}^n + \frac{\Delta t}{h^2} D\mathbf{U}^n + \Delta t \mathbf{F}^n$$

with the equivalent finite element equation:

$$M\mathbf{U}^{n+1} = M\mathbf{U}^n + \Delta t\, K\mathbf{U}^n + \Delta t \mathbf{b}^n$$

In fact, for a regular grid in 1D and with linear basis functions, $K = D/h$ (except for first/last rows).

The big advantages of FE over FD are

- FD is difficult to write down on **irregular geometries**, but FE works ON any valid mesh
- **Neumann boundary conditions** are handled very naturally in FE—require an integral over surface, and nothing required for zero-Neumann BCs. Much more difficult in FD
- Error analysis

The finite difference method

Compare the finite difference equation:

$$\mathbf{U}^{n+1} = \mathbf{U}^n + \frac{\Delta t}{h^2} D\mathbf{U}^n + \Delta t \mathbf{F}^n$$

with the equivalent finite element equation:

$$M\mathbf{U}^{n+1} = M\mathbf{U}^n + \Delta t\, K\mathbf{U}^n + \Delta t \mathbf{b}^n$$

In fact, for a regular grid in 1D and with linear basis functions, $K = D/h$ (except for first/last rows).

The big advantages of FE over FD are

- FD is difficult to write down on **irregular geometries**, but FE works ON any valid mesh
- **Neumann boundary conditions** are handled very naturally in FE—require an integral over surface, and nothing required for zero-Neumann BCs. Much more difficult in FD
- **Error analysis**

Finite volume methods

- Very commonly used for hyperbolic PDEs (for which the FE method tends to have trouble) and in computational fluid dynamics
- As with FE, FV is based on *integral formulations*.
- The domain is broken down into *control volumes* (similar to 'elements').
- One unknown computed per element (i.e. no need for 'node')—this can be considered to be the average value of u in the control volume.

Consider the advection equation $u_t + \nabla \cdot \mathbf{f}(u) = 0$. Integrate over a control volume Ω_i of volume V_i:

$$\int_{\Omega_i} u_t \, \mathrm{d}V = \int_{\Omega_i} -\nabla \cdot \mathbf{f}(u) \, \mathrm{d}V = -\int_{\partial\Omega_i} \mathbf{f}(u) \cdot \mathbf{n} \, \mathrm{d}S$$

Using an explicit time-discretisation, and $\int_{\Omega_i} U^n \, \mathrm{d}V \approx V_i U_i^n$, we obtain

$$U_i^{n+1} = U_i^n - \frac{\Delta t}{V_i} \int_{\partial\Omega_i} \mathbf{f}(u^n) \cdot \mathbf{n} \, \mathrm{d}S$$

See eg `http://www.comp.leeds.ac.uk/meh/Talks/FVTutorial.pdf` for more details

Finite volume methods

- Very commonly used for hyperbolic PDEs (for which the FE method tends to have trouble) and in computational fluid dynamics
- As with FE, FV is based on *integral formulations*.
- The domain is broken down into *control volumes* (similar to 'elements').
- One unknown computed per element (i.e. no need for 'node')—this can be considered to be the average value of u in the control volume.

Consider the advection equation $u_t + \nabla \cdot \mathbf{f}(u) = 0$. Integrate over a control volume Ω_i of volume V_i:

$$\int_{\Omega_i} u_t \, \mathrm{d}V = \int_{\Omega_i} -\nabla \cdot \mathbf{f}(u) \, \mathrm{d}V = -\int_{\partial\Omega_i} \mathbf{f}(u) \cdot \mathbf{n} \, \mathrm{d}S$$

Using an explicit time-discretisation, and $\int_{\Omega_i} U^n \, \mathrm{d}V \approx V_i U_i^n$, we obtain

$$U_i^{n+1} = U_i^n - \frac{\Delta t}{V_i} \int_{\partial\Omega_i} \mathbf{f}(u^n) \cdot \mathbf{n} \, \mathrm{d}S$$

See eg `http://www.comp.leeds.ac.uk/meh/Talks/FVTutorial.pdf` for more details

Finite volume methods

- Very commonly used for hyperbolic PDEs (for which the FE method tends to have trouble) and in computational fluid dynamics
- As with FE, FV is based on *integral formulations*.
- The domain is broken down into *control volumes* (similar to 'elements').
- One unknown computed per element (i.e. no need for 'node')—this can be considered to be the average value of u in the control volume.

Consider the advection equation $u_t + \nabla \cdot \mathbf{f}(u) = 0$. Integrate over a control volume Ω_i of volume V_i:

$$\int_{\Omega_i} u_t \, \mathrm{d}V = \int_{\Omega_i} -\nabla \cdot \mathbf{f}(u) \, \mathrm{d}V = -\int_{\partial\Omega_i} \mathbf{f}(u) \cdot \mathbf{n} \, \mathrm{d}S$$

Using an explicit time-discretisation, and $\int_{\Omega_i} U^n \, \mathrm{d}V \approx V_i U_i^n$, we obtain

$$U_i^{n+1} = U_i^n - \frac{\Delta t}{V_i} \int_{\partial\Omega_i} \mathbf{f}(u^n) \cdot \mathbf{n} \, \mathrm{d}S$$

See eg `http://www.comp.leeds.ac.uk/meh/Talks/FVTutorial.pdf` for more details

Finite volume methods

- Very commonly used for hyperbolic PDEs (for which the FE method tends to have trouble) and in computational fluid dynamics
- As with FE, FV is based on *integral formulations*.
- The domain is broken down into *control volumes* (similar to 'elements').
- One unknown computed per element (i.e. no need for 'node')—this can be considered to be the average value of u in the control volume.

Consider the advection equation $u_t + \nabla \cdot \mathbf{f}(u) = 0$. Integrate over a control volume Ω_i of volume V_i:

$$\int_{\Omega_i} u_t \, \mathrm{d}V = \int_{\Omega_i} -\nabla \cdot \mathbf{f}(u) \, \mathrm{d}V = -\int_{\partial\Omega_i} \mathbf{f}(u) \cdot \mathbf{n} \, \mathrm{d}S$$

Using an explicit time-discretisation, and $\int_{\Omega_i} U^n \, \mathrm{d}V \approx V_i U_i^n$, we obtain

$$U_i^{n+1} = U_i^n - \frac{\Delta t}{V_i} \int_{\partial\Omega_i} \mathbf{f}(u^n) \cdot \mathbf{n} \, \mathrm{d}S$$

See eg `http://www.comp.leeds.ac.uk/meh/Talks/FVTutorial.pdf` for more details

Finite volume methods

- Very commonly used for hyperbolic PDEs (for which the FE method tends to have trouble) and in computational fluid dynamics
- As with FE, FV is based on *integral formulations*.
- The domain is broken down into *control volumes* (similar to 'elements').
- One unknown computed per element (i.e. no need for 'node')—this can be considered to be the average value of u in the control volume.

Consider the advection equation $u_t + \nabla \cdot \mathbf{f}(u) = 0$. Integrate over a control volume Ω_i of volume V_i:

$$\int_{\Omega_i} u_t \,\mathrm{d}V = \int_{\Omega_i} -\nabla \cdot \mathbf{f}(u) \,\mathrm{d}V = -\int_{\partial\Omega_i} \mathbf{f}(u) \cdot \mathbf{n} \,\mathrm{d}S$$

Using an explicit time-discretisation, and $\int_{\Omega_i} U^n \,\mathrm{d}V \approx V_i U_i^n$, we obtain

$$U_i^{n+1} = U_i^n - \frac{\Delta t}{V_i} \int_{\partial\Omega_i} \mathbf{f}(u^n) \cdot \mathbf{n} \,\mathrm{d}S$$

See eg `http://www.comp.leeds.ac.uk/meh/Talks/FVTutorial.pdf` for more details

Finite volume methods

- Very commonly used for hyperbolic PDEs (for which the FE method tends to have trouble) and in computational fluid dynamics
- As with FE, FV is based on *integral formulations*.
- The domain is broken down into *control volumes* (similar to 'elements').
- One unknown computed per element (i.e. no need for 'node')—this can be considered to be the average value of u in the control volume.

Consider the advection equation $u_t + \nabla \cdot \mathbf{f}(u) = 0$. Integrate over a control volume Ω_i of volume V_i:

$$\int_{\Omega_i} u_t \,\mathrm{d}V = \int_{\Omega_i} -\nabla \cdot \mathbf{f}(u) \,\mathrm{d}V = -\int_{\partial\Omega_i} \mathbf{f}(u) \cdot \mathbf{n} \,\mathrm{d}S$$

Using an explicit time-discretisation, and $\int_{\Omega_i} U^n \,\mathrm{d}V \approx V_i U_i^n$, we obtain

$$U_i^{n+1} = U_i^n - \frac{\Delta t}{V_i} \int_{\partial\Omega_i} \mathbf{f}(u^n) \cdot \mathbf{n} \,\mathrm{d}S$$

See eg `http://www.comp.leeds.ac.uk/meh/Talks/FVTutorial.pdf` for more details

Methods of weight residuals

In FE we used an *integral formulation of the PDE*, eg: *find* $u \in \mathcal{V}$ *such that:*

$$\int_\Omega \nabla u \cdot \nabla v \, \mathrm{d}V = \int_\Omega f v \, \mathrm{d}V + \int_{\Gamma_2} g v \, \mathrm{d}S \quad \forall v \in \mathcal{V}$$

Write this as: *find* $u \in \mathcal{V}$ *such that:* $a(u, v) = l(v) \quad \forall v \in \mathcal{V}$

To *discretise* the integral equation, we replace $\mathcal{V}$ by *finite-dimensional subspaces* (of dimension N): *find* $u^{approx} \in \mathcal{W}_1$ *such that:*

$$a(u^{approx}, v) = l(v) \quad \forall v \in \mathcal{W}_2$$

Choosing bases:

$$\begin{aligned} \mathcal{W}_1 &= \mathrm{span}\{\phi_1, \ldots, \phi_N\} \\ \mathcal{W}_2 &= \mathrm{span}\{\chi_1, \ldots, \chi_N\} \end{aligned}$$

(ie $u^{approx} = \sum \alpha_i \phi_i$), we can obtain N equations for N unknowns.

Different methods are based on different choices of $\mathcal{W}_1$ and $\mathcal{W}_2$.

Methods of weight residuals

In FE we used an *integral formulation of the PDE*, eg: *find* $u \in \mathcal{V}$ *such that:*

$$\int_\Omega \boldsymbol{\nabla} u \cdot \boldsymbol{\nabla} v \, \mathrm{d}V = \int_\Omega f v \, \mathrm{d}V + \int_{\Gamma_2} g v \, \mathrm{d}S \quad \forall v \in \mathcal{V}$$

Write this as: *find* $u \in \mathcal{V}$ *such that:* $a(u, v) = l(v) \quad \forall v \in \mathcal{V}$

To *discretise* the integral equation, we replace $\mathcal{V}$ by *finite-dimensional subspaces* (of dimension N): *find* $u^{approx} \in \mathcal{W}_1$ *such that:*

$$a(u^{\text{approx}}, v) = l(v) \quad \forall v \in \mathcal{W}_2$$

Choosing bases:

$$\begin{aligned} \mathcal{W}_1 &= \text{span}\{\phi_1, \ldots, \phi_N\} \\ \mathcal{W}_2 &= \text{span}\{\chi_1, \ldots, \chi_N\} \end{aligned}$$

(ie $u^{\text{approx}} = \sum \alpha_i \phi_i$), we can obtain N equations for N unknowns.

Different methods are based on different choices of $\mathcal{W}_1$ and $\mathcal{W}_2$.

Methods of weight residuals

In FE we used an *integral formulation of the PDE*, eg: *find* $u \in \mathcal{V}$ *such that:*

$$\int_\Omega \boldsymbol{\nabla} u \cdot \boldsymbol{\nabla} v \, \mathrm{d}V = \int_\Omega f v \, \mathrm{d}V + \int_{\Gamma_2} g v \, \mathrm{d}S \quad \forall v \in \mathcal{V}$$

Write this as: *find* $u \in \mathcal{V}$ *such that:* $a(u, v) = l(v) \quad \forall v \in \mathcal{V}$

To *discretise* the integral equation, we replace $\mathcal{V}$ by *finite-dimensional subspaces* (of dimension N): *find* $u^{approx} \in \mathcal{W}_1$ *such that:*

$$a(u^{\text{approx}}, v) = l(v) \quad \forall v \in \mathcal{W}_2$$

Choosing bases:

$$\begin{aligned} \mathcal{W}_1 &= \text{span}\{\phi_1, \ldots, \phi_N\} \\ \mathcal{W}_2 &= \text{span}\{\chi_1, \ldots, \chi_N\} \end{aligned}$$

(ie $u^{\text{approx}} = \sum \alpha_i \phi_i$), we can obtain N equations for N unknowns.

Different methods are based on different choices of $\mathcal{W}_1$ and $\mathcal{W}_2$.

Methods of weight residuals

In FE we used an *integral formulation of the PDE*, eg: *find* $u \in \mathcal{V}$ *such that:*

$$\int_\Omega \boldsymbol{\nabla} u \cdot \boldsymbol{\nabla} v \, \mathrm{d}V = \int_\Omega f v \, \mathrm{d}V + \int_{\Gamma_2} g v \, \mathrm{d}S \quad \forall v \in \mathcal{V}$$

Write this as: *find* $u \in \mathcal{V}$ *such that:* $a(u, v) = l(v) \quad \forall v \in \mathcal{V}$

To *discretise* the integral equation, we replace $\mathcal{V}$ by *finite-dimensional subspaces* (of dimension N): *find* $u^{approx} \in \mathcal{W}_1$ *such that:*

$$a(u^{\text{approx}}, v) = l(v) \quad \forall v \in \mathcal{W}_2$$

Choosing bases:

$$\begin{aligned} \mathcal{W}_1 &= \text{span}\{\phi_1, \ldots, \phi_N\} \\ \mathcal{W}_2 &= \text{span}\{\chi_1, \ldots, \chi_N\} \end{aligned}$$

(ie $u^{\text{approx}} = \sum \alpha_i \phi_i$), we can obtain N equations for N unknowns.

Different methods are based on different choices of $\mathcal{W}_1$ and $\mathcal{W}_2$.

Methods of weight residuals

Find $u^{\text{approx}} \in \mathcal{W}_1$ such that:

$$a(u^{\text{approx}}, v) = l(v) \quad \forall v \in \mathcal{W}_2$$

with:

$$\begin{aligned} \mathcal{W}_1 &= \text{span}\{\phi_1, \ldots, \phi_N\}. \\ \mathcal{W}_2 &= \text{span}\{\chi_1, \ldots, \chi_N\} \end{aligned}$$

- **Galerkin methods**: use $\phi = \chi$, i.e. $\mathcal{W}_1 = \mathcal{W}_2$
- **Collocation methods**: use δ-functions for χ's (i.e. replace integrals with point evaluations (at N collocation points $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$)
- **(Continuous) Galerkin FEM**: use $\mathcal{W}_1 = \mathcal{W}_2$ and take the ϕ_i to be continuous and **piecewise polynomial** on element
  - As we know in practice we just consider 1 canonical element and define the basis functions on this (the *shape functions*)
  - Elements could be tetrahedral/hexahedral, shape functions could be linear, quadratic, cubic Hermite and more...
- **Discontinuous Galerkin FEM**: ϕ_i piecewise polynomial but no longer continuous across elements
- **Spectral methods**: ϕ_k globally continuous and infinitely differentiable (for example, $\phi_k(x) = exp(ikx)$)

Methods of weight residuals

Find $u^{\text{approx}} \in \mathcal{W}_1$ such that:

$$a(u^{\text{approx}}, v) = l(v) \quad \forall v \in \mathcal{W}_2$$

with:

$$\begin{aligned} \mathcal{W}_1 &= \text{span}\{\phi_1, \ldots, \phi_N\}. \\ \mathcal{W}_2 &= \text{span}\{\chi_1, \ldots, \chi_N\} \end{aligned}$$

- **Galerkin methods**: use $\phi = \chi$, i.e. $\mathcal{W}_1 = \mathcal{W}_2$
- **Collocation methods**: use δ-functions for χ's (i.e. replace integrals with point evaluations (at N collocation points $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$)
- **(Continuous) Galerkin FEM**: use $\mathcal{W}_1 = \mathcal{W}_2$ and take the ϕ_i to be continuous and **piecewise polynomial** on element
  - As we know in practice we just consider 1 canonical element and define the basis functions on this (the *shape functions*)
  - Elements could be tetrahedral/hexahedral, shape functions could be linear, quadratic, cubic Hermite and more...
- **Discontinuous Galerkin FEM**: ϕ_i piecewise polynomial but no longer continuous across elements
- **Spectral methods**: ϕ_k globally continuous and infinitely differentiable (for example, $\phi_k(x) = exp(ikx)$)

Methods of weight residuals

Find $u^{\text{approx}} \in \mathcal{W}_1$ such that:

$$a(u^{\text{approx}}, v) = l(v) \quad \forall v \in \mathcal{W}_2$$

with:

$$\begin{aligned} \mathcal{W}_1 &= \text{span}\{\phi_1, \ldots, \phi_N\}. \\ \mathcal{W}_2 &= \text{span}\{\chi_1, \ldots, \chi_N\} \end{aligned}$$

- **Galerkin methods**: use $\phi = \chi$, i.e. $\mathcal{W}_1 = \mathcal{W}_2$
- **Collocation methods**: use δ-functions for χ's (i.e. replace integrals with point evaluations (at N collocation points $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$)
- **(Continuous) Galerkin FEM**: use $\mathcal{W}_1 = \mathcal{W}_2$ and take the ϕ_i to be continuous and **piecewise polynomial** on element
  - As we know in practice we just consider 1 canonical element and define the basis functions on this (the *shape functions*)
  - Elements could be tetrahedral/hexahedral, shape functions could be linear, quadratic, cubic Hermite and more...
- **Discontinuous Galerkin FEM**: ϕ_i piecewise polynomial but no longer continuous across elements
- **Spectral methods**: ϕ_k globally continuous and infinitely differentiable (for example, $\phi_k(x) = exp(ikx)$)

Methods of weight residuals

Find $u^{\text{approx}} \in \mathcal{W}_1$ such that:

$$a(u^{\text{approx}}, v) = l(v) \quad \forall v \in \mathcal{W}_2$$

with:

$$\begin{aligned} \mathcal{W}_1 &= \text{span}\{\phi_1, \ldots, \phi_N\}. \\ \mathcal{W}_2 &= \text{span}\{\chi_1, \ldots, \chi_N\} \end{aligned}$$

- **Galerkin methods**: use $\phi = \chi$, i.e. $\mathcal{W}_1 = \mathcal{W}_2$
- **Collocation methods**: use δ-functions for χ's (i.e. replace integrals with point evaluations (at N collocation points $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$)
- **(Continuous) Galerkin FEM**: use $\mathcal{W}_1 = \mathcal{W}_2$ and take the ϕ_i to be continuous and **piecewise polynomial** on element
  - As we know in practice we just consider 1 canonical element and define the basis functions on this (the *shape functions*)
  - Elements could be tetrahedral/hexahedral, shape functions could be linear, quadratic, cubic Hermite and more...
- **Discontinuous Galerkin FEM**: ϕ_i piecewise polynomial but no longer continuous across elements
- **Spectral methods**: ϕ_k globally continuous and infinitely differentiable (for example, $\phi_k(x) = exp(ikx)$)

Methods of weight residuals

Find $u^{\text{approx}} \in \mathcal{W}_1$ such that:

$$a(u^{\text{approx}}, v) = l(v) \quad \forall v \in \mathcal{W}_2$$

with:

$$\begin{aligned} \mathcal{W}_1 &= \text{span}\{\phi_1, \ldots, \phi_N\}. \\ \mathcal{W}_2 &= \text{span}\{\chi_1, \ldots, \chi_N\} \end{aligned}$$

- **Galerkin methods**: use $\phi = \chi$, i.e. $\mathcal{W}_1 = \mathcal{W}_2$
- **Collocation methods**: use δ-functions for χ's (i.e. replace integrals with point evaluations (at N collocation points $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$)
- **(Continuous) Galerkin FEM**: use $\mathcal{W}_1 = \mathcal{W}_2$ and take the ϕ_i to be continuous and **piecewise polynomial** on element
  - As we know in practice we just consider 1 canonical element and define the basis functions on this (the *shape functions*)
  - Elements could be tetrahedral/hexahedral, shape functions could be linear, quadratic, cubic Hermite and more...
- **Discontinuous Galerkin FEM**: ϕ_i piecewise polynomial but no longer continuous across elements
- **Spectral methods**: ϕ_k globally continuous and infinitely differentiable (for example, $\phi_k(x) = exp(ikx)$)

Spectral methods

- There are both spectral-collocation methods (work with the strong form) or spectral-Galerkin methods (work with the weak form)

- Various choices of basis functions ($\mathcal{W}_1$) are possible, for example
- For problems with **periodic boundary conditions**, use $\phi_k(x) = exp(ikx)$
  - i.e. $u^{approx} = \sum \alpha_k \phi_k$ approximates u with a cut-off Fourier series
- For problems with **non-periodic boundary conditions**: use a set of 'orthogonal polynomials' for ϕ_k, such a Legendre or Chebychev polynomials

- For problems with *smooth data* (initial condition, boundary conditions, forces etc are smooth functions), spectral methods give exceptional rates of convergence.
- For more info, see e.g. `http://www.lorene.obspm.fr/palma.pdf`

Spectral methods

- There are both spectral-collocation methods (work with the strong form) or spectral-Galerkin methods (work with the weak form)

- Various choices of basis functions ($\mathcal{W}_1$) are possible, for example
- For problems with **periodic boundary conditions**, use $\phi_k(x) = exp(ikx)$
  - i.e. $u^{\text{approx}} = \sum \alpha_k \phi_k$ approximates u with a cut-off Fourier series
- For problems with **non-periodic boundary conditions**: use a set of 'orthogonal polynomials' for ϕ_k, such a Legendre or Chebychev polynomials

- For problems with *smooth data* (initial condition, boundary conditions, forces etc are smooth functions), spectral methods give exceptional rates of convergence.
- For more info, see e.g. `http://www.lorene.obspm.fr/palma.pdf`

Spectral methods

- There are both spectral-collocation methods (work with the strong form) or spectral-Galerkin methods (work with the weak form)

- Various choices of basis functions ($\mathcal{W}_1$) are possible, for example
- For problems with **periodic boundary conditions**, use $\phi_k(x) = exp(ikx)$
  - i.e. $u^{\text{approx}} = \sum \alpha_k \phi_k$ approximates u with a cut-off Fourier series
- For problems with **non-periodic boundary conditions**: use a set of 'orthogonal polynomials' for ϕ_k, such a Legendre or Chebychev polynomials

- For problems with *smooth data* (initial condition, boundary conditions, forces etc are smooth functions), spectral methods give exceptional rates of convergence.
- For more info, see e.g. `http://www.lorene.obspm.fr/palma.pdf`

Continuum Mechanics

Overview

1. Introduction: solids and fluids
2. Kinematics*
3. Balance equations*
4. Material laws*
5. Overall governing equations*
6. Weak problem and numerical method**
7. Objected-oriented design in Chaste**

* Focussing on nonlinear elasticity, but also mentioning linear elasticity & fluids
** Nonlinear elasticity only

Overview

1. Introduction: solids and fluids
2. Kinematics*
3. Balance equations*
4. Material laws*
5. Overall governing equations*
6. Weak problem and numerical method**
7. Objected-oriented design in Chaste**

* Focussing on nonlinear elasticity, but also mentioning linear elasticity & fluids
** Nonlinear elasticity only

Overview

1. Introduction: solids and fluids
2. Kinematics*
3. Balance equations*
4. Material laws*
5. Overall governing equations*
6. Weak problem and numerical method**
7. Objected-oriented design in Chaste**

* Focussing on nonlinear elasticity, but also mentioning linear elasticity & fluids
** Nonlinear elasticity only

Overview

1. Introduction: solids and fluids
2. Kinematics*
3. Balance equations*
4. Material laws*
5. Overall governing equations*
6. Weak problem and numerical method**
7. Objected-oriented design in Chaste**

* Focussing on nonlinear elasticity, but also mentioning linear elasticity & fluids
** Nonlinear elasticity only

Overview

1. Introduction: solids and fluids
2. Kinematics*
3. Balance equations*
4. Material laws*
5. Overall governing equations*
6. Weak problem and numerical method**
7. Objected-oriented design in Chaste**

* Focussing on nonlinear elasticity, but also mentioning linear elasticity & fluids
** Nonlinear elasticity only

Overview

1. Introduction: solids and fluids
2. Kinematics*
3. Balance equations*
4. Material laws*
5. Overall governing equations*
6. Weak problem and numerical method**
7. Objected-oriented design in Chaste**

* Focussing on nonlinear elasticity, but also mentioning linear elasticity & fluids
** Nonlinear elasticity only

Introduction: solids and fluids

Solids versus fluids

Solids versus fluids

Pitchdrop experiment, Queensland. Experiment begun 1927 (1930). Drops fell in: 1938, 1947, 1954, 1962, 1970, 1979, 1988, 2000

Solids versus fluids

"Fluids cannot resist deformation force". Shape will change as long as the force is applied. Whereas a solid can change shape but not indefinitely.

More specifically, fluids cannot resist shear forces

- For solids, force is related to *deformation* (coefficient: *stiffness*)
- For fluids, force is related to *deformation-rate* (coefficient: *viscosity*)

Some materials are fluid under some conditions (excl. temperature) and solid under others (see, for example, `youtube:walking on custard`)

Solids versus fluids

"Fluids cannot resist deformation force". Shape will change as long as the force is applied. Whereas a solid can change shape but not indefinitely.

More specifically, fluids cannot resist shear forces

- For solids, force is related to *deformation* (coefficient: *stiffness*)
- For fluids, force is related to *deformation-rate* (coefficient: *viscosity*)

Some materials are fluid under some conditions (excl. temperature) and solid under others (see, for example, `youtube:walking on custard`)

Types of solid

For solids, force is related to deformation (stress related to strain):

Elastic

- When an applied force is removed, the solid returns to its original shape
- For small enough forces/strains, stress is usually proportional to strain (*linear elasticity*)

Visco-elastic

- Also exhibit a *viscous response*, for example, slow change of shape if a force is held constant / slow decrease of stress if strain held constant
- Stress becomes a function of strain and *strain rate*

Plastic

- Once a large enough stress is applied (the *yield* stress), the material undergoes permanent deformation (*flows*), due to internal rearrangement. If the stress is removed it won't return back to original state.

Types of solid

For solids, force is related to deformation (stress related to strain):

Elastic

- When an applied force is removed, the solid returns to its original shape
- For small enough forces/strains, stress is usually proportional to strain (*linear elasticity*)

Visco-elastic

- Also exhibit a *viscous response*, for example, slow change of shape if a force is held constant / slow decrease of stress if strain held constant
- Stress becomes a function of strain and *strain rate*

Plastic

- Once a large enough stress is applied (the *yield* stress), the material undergoes permanent deformation (*flows*), due to internal rearrangement. If the stress is removed it won't return back to original state.

Types of solid

For solids, force is related to deformation (stress related to strain):

Elastic

- When an applied force is removed, the solid returns to its original shape
- For small enough forces/strains, stress is usually proportional to strain (*linear elasticity*)

Visco-elastic

- Also exhibit a *viscous response*, for example, slow change of shape if a force is held constant / slow decrease of stress if strain held constant
- Stress becomes a function of strain and *strain rate*

Plastic

- Once a large enough stress is applied (the *yield* stress), the material undergoes permanent deformation (*flows*), due to internal rearrangement. If the stress is removed it won't return back to original state.

Types of fluid

For fluids force is related to deformation-rate (stress is related to strain-rate)

Newtonian

- Stress is related *linearly* to the strain-rate

Non-Newtonian

- Stress is related *non-linearly* to the strain-rate

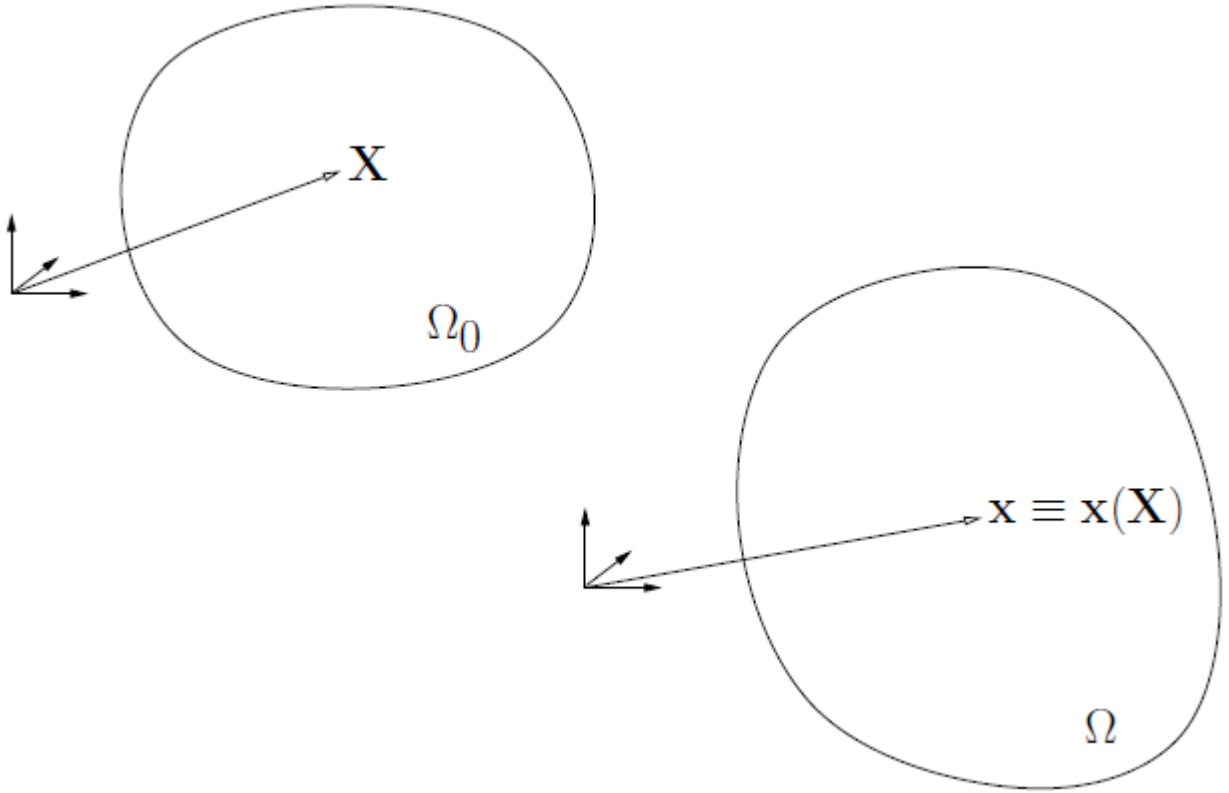
Kinematics of solids

Kinematics of solids

In the later section we will write down balance equations relating the internal stresses in the body to external forces.

What are the internal stresses a function of?

Undeformed and deformed states



- Let Ω_0 represent the unloaded, unstressed body
- Let Ω_t represent the deformed body at time t
  - For time-independent problems, we denote the deformed body Ω
- Let $\mathbf{X}$ represent a point in the undeformed body
- Let $\mathbf{x} \equiv \mathbf{x}(t, \mathbf{X})$ represent the corresponding deformed position
- Let the displacement be denoted $\mathbf{u} = \mathbf{x} - \mathbf{X}$

The deformation gradient

Let $F_{iM} = \frac{\partial x_i}{\partial X_M}$ be the deformation gradient. This describes the deformation, excluding rigid body translations.

Any deformation can be decomposed into a (local) translation, rotation, and stretch. Correspondingly, F can be decomposed into a rotation and a stretch: $F = RU$, where R is an rotation matrix, and U is a positive-definite symmetric matrix representing stretch.

Examples, in 2D:

- let $\mathbf{x} = \begin{bmatrix} \alpha X \\ \beta Y \end{bmatrix}$, then $F = \begin{bmatrix} \alpha & 0 \\ 0 & \beta \end{bmatrix}$ (simple bi-axial stretch)
- let $\mathbf{x} = \begin{bmatrix} X - \alpha Y \\ Y \end{bmatrix}$, then $F = \begin{bmatrix} 1 & -\alpha \\ 0 & 1 \end{bmatrix}$ (simple shear)

The deformation gradient

Let $F_{iM} = \frac{\partial x_i}{\partial X_M}$ be the deformation gradient. This describes the deformation, excluding rigid body translations.

Any deformation can be decomposed into a (local) translation, rotation, and stretch. Correspondingly, F can be decomposed into a rotation and a stretch: $F = RU$, where R is an rotation matrix, and U is a positive-definite symmetric matrix representing stretch.

Examples, in 2D:

- let $\mathbf{x} = \begin{bmatrix} \alpha X \\ \beta Y \end{bmatrix}$, then $F = \begin{bmatrix} \alpha & 0 \\ 0 & \beta \end{bmatrix}$ (simple bi-axial stretch)
- let $\mathbf{x} = \begin{bmatrix} X - \alpha Y \\ Y \end{bmatrix}$, then $F = \begin{bmatrix} 1 & -\alpha \\ 0 & 1 \end{bmatrix}$ (simple shear)

The deformation gradient

Let $F_{iM} = \frac{\partial x_i}{\partial X_M}$ be the deformation gradient. This describes the deformation, excluding rigid body translations.

Any deformation can be decomposed into a (local) translation, rotation, and stretch. Correspondingly, F can be decomposed into a rotation and a stretch: $F = RU$, where R is an rotation matrix, and U is a positive-definite symmetric matrix representing stretch.

Examples, in 2D:

- let $\mathbf{x} = \begin{bmatrix} \alpha X \\ \beta Y \end{bmatrix}$, then $F = \begin{bmatrix} \alpha & 0 \\ 0 & \beta \end{bmatrix}$ (simple bi-axial stretch)
- let $\mathbf{x} = \begin{bmatrix} X - \alpha Y \\ Y \end{bmatrix}$, then $F = \begin{bmatrix} 1 & -\alpha \\ 0 & 1 \end{bmatrix}$ (simple shear)

The deformation gradient

det F

F is the jacobian of the mapping from Ω_0 to Ω, therefore det F represents the change in local volume. Hence:

- $\det F > 0$ for all deformations
- For incompressible deformations (also known as *isochoric* or *isovolumetric* deformations), $\det F = 1$ (everywhere)

Define $J = \det F$

Principal stretches

The eigenvalues of U are of the *principal stretches*, denoted $\lambda_1, \lambda_2, \lambda_3$

The deformation gradient

det F

F is the jacobian of the mapping from Ω_0 to Ω, therefore det F represents the change in local volume. Hence:

- $\det F > 0$ for all deformations
- For incompressible deformations (also known as *isochoric* or *isovolumetric* deformations), $\det F = 1$ (everywhere)

Define $J = \det F$

Principal stretches

The eigenvalues of U are of the *principal stretches*, denoted $\lambda_1, \lambda_2, \lambda_3$

Lagrangian measures of strain

The *(right) Cauchy-Green deformation tensor* is

$$C = F^{\mathsf{T}} F$$

Note that

- $F = RU \quad \Rightarrow \quad C = U^{\mathsf{T}} R^{\mathsf{T}} RU = U^{\mathsf{T}} U = U^2$
- i.e. C is independent of the rotation
- the eigenvalues of C are λ_1^2, λ_2^2, λ_3^2.

The *Green-Lagrange strain tensor* is

$$E = \frac{1}{2}(C - I)$$

and is the nonlinear generalisation of the simple 1d strain measure $(l - l_0)/l_0$

Can work with either C or E. Note: for no deformation $C = I$ vs $E = 0$.

Lagrangian measures of strain

The *(right) Cauchy-Green deformation tensor* is

$$C = F^{\mathsf{T}} F$$

Note that

- $F = RU \quad \Rightarrow \quad C = U^{\mathsf{T}} R^{\mathsf{T}} RU = U^{\mathsf{T}} U = U^2$
- i.e. C is independent of the rotation
- the eigenvalues of C are λ_1^2, λ_2^2, λ_3^2.

The *Green-Lagrange strain tensor* is

$$E = \frac{1}{2}(C - I)$$

and is the nonlinear generalisation of the simple 1d strain measure $(l - l_0)/l_0$

Can work with either C or E. Note: for no deformation $C = I$ vs $E = 0$.

Lagrangian measures of strain

The *(right) Cauchy-Green deformation tensor* is

$$C = F^{\mathsf{T}} F$$

Note that

- $F = RU \quad \Rightarrow \quad C = U^{\mathsf{T}} R^{\mathsf{T}} RU = U^{\mathsf{T}} U = U^2$
- i.e. C is independent of the rotation
- the eigenvalues of C are λ_1^2, λ_2^2, λ_3^2.

The *Green-Lagrange strain tensor* is

$$E = \frac{1}{2}(C - I)$$

and is the nonlinear generalisation of the simple 1d strain measure $(l - l_0)/l_0$

Can work with either C or E. Note: for no deformation $C = I$ vs $E = 0$.

Lagrangian measures of strain

The *(right) Cauchy-Green deformation tensor* is

$$C = F^{\mathsf{T}} F$$

Note that

- $F = RU \quad \Rightarrow \quad C = U^{\mathsf{T}} R^{\mathsf{T}} RU = U^{\mathsf{T}} U = U^2$
- i.e. C is independent of the rotation
- the eigenvalues of C are λ_1^2, λ_2^2, λ_3^2.

The *Green-Lagrange strain tensor* is

$$E = \frac{1}{2}(C - I)$$

and is the nonlinear generalisation of the simple 1d strain measure $(l - l_0)/l_0$

Can work with either C or E. Note: for no deformation $C = I$ vs $E = 0$.