Introduction and Basic Implementation for Finite Element Methods

Chapter 8: Finite elements for 2D unsteady Stokes and linear elasticity equations

Xiaoming He
Department of Mathematics & Statistics
Missouri University of Science & Technology

Outline

Outline

Target problem

- Consider the 2D unsteady Stokes equation

$$\begin{aligned}
&\mathbf{u}_t - \nabla \cdot \mathbb{T}(\mathbf{u}, p) = \mathbf{f}, \quad \text{in } \Omega \times [0, T], \\
&\nabla \cdot \mathbf{u} = 0 \qquad \text{in } \Omega \times [0, T], \\
&\mathbf{u} = \mathbf{g}, \quad \text{on } \partial\Omega \times [0, T], \\
&\mathbf{u} = \mathbf{u}_0, \ p = p_0, \quad \text{at } t = 0 \text{ and in } \Omega.
\end{aligned}$$

where Ω is a 2D domain, $[0, T]$ is the time interval, $\mathbf{f}(x, y, t)$ is a given function on $\Omega \times [0, T]$, $\mathbf{g}(x, y, t)$ is a given function on $\partial\Omega \times [0, T]$, $\mathbf{u}_0(x, y)$ and $p_0(x, y)$ are given functions on Ω at $t = 0$, $\mathbf{u}(x, y, t)$ and $p(x, y, t)$ are the unknown functions, and

$$\begin{aligned}
&\mathbf{u}(x, y, t) = (u_1, \ u_2)^t, \quad \mathbf{f}(x, y, t) = (f_1, \ f_2)^t, \\
&\mathbf{g}(x, y, t) = (g_1, \ g_2)^t, \quad \mathbf{u}_0(x, y) = (u_{10}, \ u_{20})^t.
\end{aligned}$$

Target problem

- The stress tensor $\mathbb{T}(\mathbf{u}, p)$ is defined as

$$\mathbb{T}(\mathbf{u}, p) = 2\nu\mathbb{D}(\mathbf{u}) - p\mathbb{I}$$

where ν is the viscosity and the deformation tensor

$$\mathbb{D}(\mathbf{u}) = \frac{1}{2}(\nabla\mathbf{u} + (\nabla\mathbf{u})^t).$$

- In more details, the deformation tensor can be written as

$$\mathbb{D}(\mathbf{u}) = \begin{pmatrix} \frac{\partial u_1}{\partial x} & \frac{1}{2}\left(\frac{\partial u_1}{\partial y} + \frac{\partial u_2}{\partial x}\right) \\ \frac{1}{2}\left(\frac{\partial u_1}{\partial y} + \frac{\partial u_2}{\partial x}\right) & \frac{\partial u_2}{\partial y} \end{pmatrix}.$$

- Hence the stress tensor can be written as

$$\mathbb{T}(\mathbf{u}, p) = \begin{pmatrix} 2\nu\frac{\partial u_1}{\partial x} - p & \nu\left(\frac{\partial u_1}{\partial y} + \frac{\partial u_2}{\partial x}\right) \\ \nu\left(\frac{\partial u_1}{\partial y} + \frac{\partial u_2}{\partial x}\right) & 2\nu\frac{\partial u_2}{\partial y} - p \end{pmatrix}.$$

Weak formulation

- First, take the inner product with a vector function $\mathbf{v}(x,y) = (v_1,\ v_2)^t$ on both sides of the Stokes equation:

$$
\begin{aligned}
& \mathbf{u}_t - \nabla \cdot \mathbb{T}(\mathbf{u}, p) = \mathbf{f} \quad \text{in } \Omega \\
\Rightarrow \quad & \mathbf{u}_t \cdot \mathbf{v} - \nabla \cdot \mathbb{T}(\mathbf{u}, p) \cdot \mathbf{v} = \mathbf{f} \cdot \mathbf{v} \quad \text{in } \Omega \\
\Rightarrow \quad & \int_\Omega \mathbf{u}_t \cdot \mathbf{v}\ dxdy - \int_\Omega (\nabla \cdot \mathbb{T}(\mathbf{u}, p)) \cdot \mathbf{v}\ dxdy = \int_\Omega \mathbf{f} \cdot \mathbf{v}\ dxdy
\end{aligned}
$$

- Second, multiply the divergence free equation by a function $q(x,y)$:

$$
\begin{aligned}
\nabla \cdot \mathbf{u} = 0 \quad \Rightarrow \quad & (\nabla \cdot \mathbf{u}) q = 0 \\
\Rightarrow \quad & \int_\Omega (\nabla \cdot \mathbf{u}) q\ dxdy = 0.
\end{aligned}
$$

- $\mathbf{u}(x,y,t)$ and $p(x,y,t)$ are called trail functions and $\mathbf{v}(x,y)$ and $q(x,y)$ are called test functions.

Weak formulation

- Using integration by parts in multi-dimension:

$$\int_\Omega (\nabla \cdot \mathbb{T}) \cdot \mathbf{v}\ dxdy = \int_{\partial\Omega} (\mathbb{T}\mathbf{n}) \cdot \mathbf{v}\ ds - \int_\Omega \mathbb{T} : \nabla \mathbf{v}\ dxdy,$$

where $\mathbf{n} = (n_1,\ n_2)^t$ is the unit outer normal vector of $\partial\Omega$, we obtain

$$\begin{aligned} & \int_\Omega \mathbf{u}_t \cdot \mathbf{v}\ dxdy - \int_\Omega \mathbb{T}(\mathbf{u}, p) : \nabla \mathbf{v}\ dxdy - \int_{\partial\Omega} (\mathbb{T}(\mathbf{u}, p)\mathbf{n}) \cdot \mathbf{v}\ ds \\ = & \int_\Omega \mathbf{f} \cdot \mathbf{v}\ dxdy. \end{aligned}$$

Here,

$$\begin{aligned} A : B &= \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} : \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix} \\ &= a_{11}b_{11} + a_{12}b_{12} + a_{21}b_{21} + a_{22}b_{22}. \end{aligned}$$

Weak formulation

- Using the above definition for $A : B$, it is not difficult to verify (an independent study project topic) that

$$\begin{aligned}\mathbb{T}(\mathbf{u}, p) : \nabla \mathbf{v} &= (2\nu\mathbb{D}(\mathbf{u}) - p\mathbb{I}) : \nabla \mathbf{v} \\ &= 2\nu\mathbb{D}(\mathbf{u}) : \mathbb{D}(\mathbf{v}) - p(\nabla \cdot \mathbf{v}).\end{aligned}$$

- Hence we obtain

$$\begin{aligned}&\int_\Omega \mathbf{u}_t \cdot \mathbf{v}\ dxdy + \int_\Omega 2\nu\mathbb{D}(\mathbf{u}) : \mathbb{D}(\mathbf{v})\ dxdy - \int_\Omega p(\nabla \cdot \mathbf{v})\ dxdy \\ &- \int_{\partial\Omega} (\mathbb{T}(\mathbf{u}, p)\mathbf{n}) \cdot \mathbf{v}\ ds \ = \ \int_\Omega \mathbf{f} \cdot \mathbf{v}\ dxdy, \\ &- \int_\Omega (\nabla \cdot \mathbf{u}) q\ dxdy = 0.\end{aligned}$$

Here we multiply the second equation by -1 in order to keep the matrix formulation symmetric later.

Weak formulation

- Since the solution on the domain boundary $\partial\Omega$ are given by $\mathbf{u}(x,y,t) = \mathbf{g}(x,y,t)$, then we can choose the test function $\mathbf{v}(x,y)$ such that $\mathbf{v} = 0$ on $\partial\Omega$.
- Hence

$$\int_\Omega \mathbf{u}_t \cdot \mathbf{v}\ dxdy + \int_\Omega 2\nu\mathbb{D}(\mathbf{u}) : \mathbb{D}(\mathbf{v})\ dxdy - \int_\Omega p(\nabla\cdot\mathbf{v})\ dxdy$$
$$= \int_\Omega \mathbf{f}\cdot\mathbf{v}\ dxdy,$$
$$-\int_\Omega (\nabla\cdot\mathbf{u})q\ dxdy = 0.$$

- Define

$$H^1(0,T;[H^1(\Omega)]^2) = \{\mathbf{v}(\cdot,t),\ \frac{\partial\mathbf{v}}{\partial t}(\cdot,t)\in[H^1(\Omega)]^2,\ \forall t\in[0,T]\},$$
$$L^2(0,T;L^2(\Omega)) = \{q(\cdot,t)\in L^2(\Omega),\ \forall t\in[0,T]\}.$$

where $[H^1(\Omega)]^2 = H^1(\Omega)\times H^1(\Omega)$.

Weak formulation

- Weak formulation in the vector format: find $\mathbf{u} \in H^1(0,T;[H^1(\Omega)]^2)$ and $p \in L^2(0,T;L^2(\Omega))$ such that

$$\int_\Omega \mathbf{u}_t \cdot \mathbf{v}\ dxdy + \int_\Omega 2\nu \mathbb{D}(\mathbf{u}) : \mathbb{D}(\mathbf{v})\ dxdy - \int_\Omega p(\nabla \cdot \mathbf{v})\ dxdy$$
$$= \int_\Omega \mathbf{f} \cdot \mathbf{v}\ dxdy,$$
$$-\int_\Omega (\nabla \cdot \mathbf{u}) q\ dxdy = 0,$$

for any $\mathbf{v} \in [H_0^1(\Omega)]^2$ and $q \in L^2(\Omega)$.

Weak formulation

- Define

$$a(\mathbf{u}, \mathbf{v}) = \int_\Omega 2\nu \mathbb{D}(\mathbf{u}) : \mathbb{D}(\mathbf{v}) \; dxdy,$$

$$b(\mathbf{u}, q) = -\int_\Omega (\nabla \cdot \mathbf{u}) q \; dxdy,$$

$$(\mathbf{f}, \mathbf{v}) = \int_\Omega \mathbf{f} \cdot \mathbf{v} \; dxdy.$$

- Weak formulation: find $\mathbf{u} \in H^1(0, T; [H^1(\Omega)]^2)$ and $p \in L^2(0, T; L^2(\Omega))$ such that

$$\begin{aligned} (\mathbf{u}_t, \mathbf{v}) + a(\mathbf{u}, \mathbf{v}) + b(\mathbf{v}, p) &= (\mathbf{f}, \mathbf{v}), \\ b(\mathbf{u}, q) &= 0, \end{aligned}$$

for any $\mathbf{v} \in [H_0^1(\Omega)]^2$ and $q \in L^2(\Omega)$.

Weak formulation

- In more details,

$$
\begin{aligned}
& \mathbb{D}(\mathbf{u}) : \mathbb{D}(\mathbf{v}) \\
= & \begin{pmatrix} \frac{\partial u_1}{\partial x} & \frac{1}{2}\left(\frac{\partial u_1}{\partial y} + \frac{\partial u_2}{\partial x}\right) \\ \frac{1}{2}\left(\frac{\partial u_1}{\partial y} + \frac{\partial u_2}{\partial x}\right) & \frac{\partial u_2}{\partial y} \end{pmatrix} \\
& : \begin{pmatrix} \frac{\partial v_1}{\partial x} & \frac{1}{2}\left(\frac{\partial v_1}{\partial y} + \frac{\partial v_2}{\partial x}\right) \\ \frac{1}{2}\left(\frac{\partial v_1}{\partial y} + \frac{\partial v_2}{\partial x}\right) & \frac{\partial v_2}{\partial y} \end{pmatrix} \\
= & \frac{\partial u_1}{\partial x}\frac{\partial v_1}{\partial x} + \frac{1}{4}\left(\frac{\partial u_1}{\partial y} + \frac{\partial u_2}{\partial x}\right)\left(\frac{\partial v_1}{\partial y} + \frac{\partial v_2}{\partial x}\right) \\
& + \frac{1}{4}\left(\frac{\partial u_1}{\partial y} + \frac{\partial u_2}{\partial x}\right)\left(\frac{\partial v_1}{\partial y} + \frac{\partial v_2}{\partial x}\right) + \frac{\partial u_2}{\partial y}\frac{\partial v_2}{\partial y}.
\end{aligned}
$$

Weak formulation

- Hence

$$
\begin{aligned}
& \mathbb{D}(\mathbf{u}) : \mathbb{D}(\mathbf{v}) \\
= & \frac{\partial u_1}{\partial x}\frac{\partial v_1}{\partial x} + \frac{\partial u_2}{\partial y}\frac{\partial v_2}{\partial y} + \frac{1}{2}\frac{\partial u_1}{\partial y}\frac{\partial v_1}{\partial y} \\
& + \frac{1}{2}\frac{\partial u_1}{\partial y}\frac{\partial v_2}{\partial x} + \frac{1}{2}\frac{\partial u_2}{\partial x}\frac{\partial v_1}{\partial y} + \frac{1}{2}\frac{\partial u_2}{\partial x}\frac{\partial v_2}{\partial x}.
\end{aligned}
$$

- Then

$$
\begin{aligned}
& \int_\Omega 2\nu \mathbb{D}(\mathbf{u}) : \mathbb{D}(\mathbf{v}) \; dxdy \\
= & \int_\Omega \nu \Big(2\frac{\partial u_1}{\partial x}\frac{\partial v_1}{\partial x} + 2\frac{\partial u_2}{\partial y}\frac{\partial v_2}{\partial y} + \frac{\partial u_1}{\partial y}\frac{\partial v_1}{\partial y} \\
& + \frac{\partial u_1}{\partial y}\frac{\partial v_2}{\partial x} + \frac{\partial u_2}{\partial x}\frac{\partial v_1}{\partial y} + \frac{\partial u_2}{\partial x}\frac{\partial v_2}{\partial x} \Big) \; dxdy.
\end{aligned}
$$

Weak formulation

- We also have

$$\int_\Omega \mathbf{u}_t \cdot \mathbf{v} \ dxdy = \int_\Omega \frac{\partial u_1}{\partial t} v_1 \ dxdy + \int_\Omega \frac{\partial u_2}{\partial t} v_2 \ dxdy,$$

$$\int_\Omega p(\nabla \cdot \mathbf{v}) \ dxdy = \int_\Omega \left(p\frac{\partial v_1}{\partial x} + p\frac{\partial v_2}{\partial y} \right) \ dxdy,$$

$$\int_\Omega \mathbf{f} \cdot \mathbf{v} \ dxdy = \int_\Omega (f_1 v_1 + f_2 v_2) \ dxdy,$$

$$\int_\Omega (\nabla \cdot \mathbf{u}) q \ dxdy = \int_\Omega \left(\frac{\partial u_1}{\partial x} q + \frac{\partial u_2}{\partial y} q \right) \ dxdy.$$

Weak formulation

- Weak formulation in the scalar format: find $u_1 \in H^1(0,T;[H^1(\Omega)]^2)$, $u_2 \in H^1(0,T;[H^1(\Omega)]^2)$, and $p \in L^2(0,T;L^2(\Omega))$ such that

$$
\begin{aligned}
&\int_\Omega \frac{\partial u_1}{\partial t} v_1 \ dxdy + \int_\Omega \frac{\partial u_2}{\partial t} v_2 \ dxdy + \int_\Omega \nu\Big(2\frac{\partial u_1}{\partial x}\frac{\partial v_1}{\partial x} \\
&+2\frac{\partial u_2}{\partial y}\frac{\partial v_2}{\partial y} + \frac{\partial u_1}{\partial y}\frac{\partial v_1}{\partial y} + \frac{\partial u_1}{\partial y}\frac{\partial v_2}{\partial x} + \frac{\partial u_2}{\partial x}\frac{\partial v_1}{\partial y} + \frac{\partial u_2}{\partial x}\frac{\partial v_2}{\partial x}\Big) \ dxdy \\
&- \int_\Omega \left(p\frac{\partial v_1}{\partial x} + p\frac{\partial v_2}{\partial y}\right) \ dxdy \\
&= \int_\Omega (f_1 v_1 + f_2 v_2) \ dxdy. \\
&- \int_\Omega \left(\frac{\partial u_1}{\partial x} q + \frac{\partial u_2}{\partial y} q\right) \ dxdy = 0.
\end{aligned}
$$

for any $v_1 \in H_0^1(\Omega)$, $v_2 \in H_0^1(\Omega)$, and $q \in L^2(\Omega)$.

Outline

Galerkin formulation

- Consider a finite element space $U_h \subset H^1(\Omega)$ for the velocity and a finite element space $W_h \subset L^2(\Omega)$ for the pressure. Define U_{h0} to be the space which consists of the functions of U_h with value 0 on the Dirichlet boundary.
- Then the Galerkin formulation is to find $\mathbf{u}_h \in H^1(0, T; [U_h]^2)$ and $p_h \in L^2(0, T; W_h)$ such that

$$\begin{aligned} (\mathbf{u}_{h_t}, \mathbf{v}) + a(\mathbf{u}_h, \mathbf{v}_h) + b(\mathbf{v}_h, p_h) &= (\mathbf{f}, \mathbf{v}_h), \\ b(\mathbf{u}_h, q_h) &= 0, \end{aligned}$$

for any $\mathbf{v}_h \in [U_{h0}]^2$ and $q_h \in W_h$.

Galerkin formulation

- For an easier implementation, we use the following Galerkin formulation (without considering the Dirichlet boundary condition, which will be handled later): find $\mathbf{u}_h \in H^1(0,T;[U_h]^2)$ and $p_h \in L^2(0,T;W_h)$ such that

$$
\begin{aligned}
(\mathbf{u}_{h_t}, \mathbf{v}) + a(\mathbf{u}_h, \mathbf{v}_h) + b(\mathbf{v}_h, p_h) &= (\mathbf{f}, \mathbf{v}_h), \\
b(\mathbf{u}_h, q_h) &= 0,
\end{aligned}
$$

for any $\mathbf{v}_h \in [U_h]^2$ and $q_h \in W_h$.

Galerkin formulation

- In more details of the vector format, the Galerkin formulation (without considering the Dirichlet boundary condition, which will be handled later) is to find $\mathbf{u}_h \in H^1(0, T; [U_h]^2)$ and $p_h \in L^2(0, T; W_h)$ such that

$$
\begin{aligned}
&\int_\Omega \mathbf{u}_{h_t} \cdot \mathbf{v}_h \; dxdy + \int_\Omega 2\nu \mathbb{D}(\mathbf{u}_h) : \mathbb{D}(\mathbf{v}_h) \; dxdy \\
&- \int_\Omega p_h (\nabla \cdot \mathbf{v}_h) \; dxdy = \int_\Omega \mathbf{f} \cdot \mathbf{v}_h \; dxdy, \\
&- \int_\Omega (\nabla \cdot \mathbf{u}_h) q_h \; dxdy = 0,
\end{aligned}
$$

for any $\mathbf{v}_h \in [U_h]^2$ and $q_h \in W_h$.

Galerkin formulation

- In our numerical example, $U_h = span\{\phi_j\}_{j=1}^{N_b}$ and $W_h = span\{\psi_j\}_{j=1}^{N_{bp}}$ are chosen to be the finite element spaces with the quadratic global basis functions $\{\phi_j\}_{j=1}^{N_b}$ and linear global basis functions $\{\psi_j\}_{j=1}^{N_{bp}}$, which are defined in Chapter 2. They are called Taylor-Hood finite elements.
- Why do we choose the pairs of finite elements in this way?
- Stability of mixed finite elements: inf-sup condition.

$$\inf_{0 \neq q_h \in W_h} \sup_{0 \neq \mathbf{u}_h \in U_h \times U_h} \frac{b(\mathbf{u}_h, q_h)}{\|\nabla \mathbf{u}_h\|_0 \, \|q_h\|_0} > \beta,$$

where $\beta > 0$ is a constant independent of mesh size h.
- See other course materials and references for the theory and more examples of stable mixed finite elements for Stokes equation.

Galerkin formulation

- In the scalar format, the Galerkin formulation (without considering the Dirichlet boundary condition, which will be handled later) is to find $u_{1h} \in H^1(0, T; U_h)$, $u_{2h} \in H^1(0, T; U_h)$, and $p_h \in L^2(0, T; W_h)$ such that

$$
\begin{aligned}
&\int_\Omega \frac{\partial u_{1h}}{\partial t} v_{1h}\ dxdy + \int_\Omega \frac{\partial u_{2h}}{\partial t} v_{2h}\ dxdy \\
&+\int_\Omega \nu \Big(2\frac{\partial u_{1h}}{\partial x}\frac{\partial v_{1h}}{\partial x} + 2\frac{\partial u_{2h}}{\partial y}\frac{\partial v_{2h}}{\partial y} + \frac{\partial u_{1h}}{\partial y}\frac{\partial v_{1h}}{\partial y} \\
&+\frac{\partial u_{1h}}{\partial y}\frac{\partial v_{2h}}{\partial x} + \frac{\partial u_{2h}}{\partial x}\frac{\partial v_{1h}}{\partial y} + \frac{\partial u_{2h}}{\partial x}\frac{\partial v_{2h}}{\partial x}\Big)\ dxdy \\
&-\int_\Omega \left(p_h\frac{\partial v_{1h}}{\partial x} + p_h\frac{\partial v_{2h}}{\partial y}\right)\ dxdy = \int_\Omega (f_1 v_{1h} + f_2 v_{2h})\ dxdy. \\
&-\int_\Omega \left(\frac{\partial u_{1h}}{\partial x}q_h + \frac{\partial u_{2h}}{\partial y}q_h\right)\ dxdy = 0.
\end{aligned}
$$

for any $v_{1h} \in U_h$, $v_{2h} \in U_h$, and $q_h \in W_h$.

Discretization formulation

Recall the following definitions from Chapter 2:

- N: number of mesh elements.
- N_m: number of mesh nodes.
- E_n $(n = 1, \cdots, N)$: mesh elements.
- Z_k $(k = 1, \cdots, N_m)$: mesh nodes.
- N_l: number of local mesh nodes in a mesh element.
- P:information matrix consisting of the coordinates of all mesh nodes.
- T: information matrix consisting of the global node indices of the mesh nodes of all the mesh elements.

Discretization formulation

- We only consider the nodal basis functions (Lagrange type) in this course.
- N_{lb}: number of local finite element nodes (=number of local finite element basis functions) in a mesh element.
- N_b: number of the finite element nodes (= the number of unknowns = the total number of the finite element basis functions).
- X_j $(j = 1, \cdots, N_b)$: finite element nodes.
- P_b: information matrix consisting of the coordinates of all finite element nodes.
- T_b: information matrix consisting of the global node indices of the finite element nodes of all the mesh elements.

Discretization formulation

- Since $u_{1h},\ u_{2h} \in H^1(0,T;U_h)$, $p_h \in L^2(0,T;W_h)$, $U_h = span\{\phi_j\}_{j=1}^{N_b}$, and $W_h = span\{\psi_j\}_{j=1}^{N_{bp}}$, then

$$u_{1h}(x,y,t) = \sum_{j=1}^{N_b} u_{1j}(t)\phi_j, \quad u_{2h}(x,y,t) = \sum_{j=1}^{N_b} u_{2j}(t)\phi_j,$$

$$p_h = \sum_{j=1}^{N_{bp}} p_j(t)\psi_j,$$

for some coefficients $u_{1j}(t)$, $u_{2j}(t)$ $(j=1,\cdots,N_b)$, and $p_j(t)$ $(j=1,\cdots,N_{bp})$.

- If we can set up a linear algebraic system for $u_{1j}(t)$, $u_{2j}(t)$ $(j=1,\cdots,N_b)$, and $p_j(t)$ $(j=1,\cdots,N_{bp})$, then we can solve it to obtain the finite element solution $\mathbf{u}_h = (u_{1h}, u_{2h})^t$ and p_h.

Discretization formulation

- For the first equation in the Galerkin formulation, we choose $\mathbf{v}_h = (\phi_i, 0)^t$ $(i = 1, \cdots, N_b)$ and $\mathbf{v}_h = (0, \phi_i)^t$ $(i = 1, \cdots, N_b)$. That is, in the first set of test functions, we choose $v_{1h} = \phi_i$ $(i = 1, \cdots, N_b)$ and $v_{2h} = 0$; in the second set of test functions, we choose $v_{1h} = 0$ and $v_{2h} = \phi_i$ $(i = 1, \cdots, N_b)$.

- For the second equation in the Galerkin formulation, we choose $q_h = \psi_i$ $(i = 1, \cdots, N_{bp})$.

Discretization formulation

- Set $\mathbf{v}_h = (\phi_i, 0)^t$, i.e., $v_{1h} = \phi_i$ and $v_{2h} = 0$ $(i = 1, \cdots, N_b)$, in the first equation of the Galerkin formulation. Then

$$
\begin{aligned}
&\int_\Omega \left(\sum_{j=1}^{N_b} u_{1j}(t)\phi_j \right)_t \phi_i \, dxdy + 2\int_\Omega \nu \left(\sum_{j=1}^{N_b} u_{1j}(t)\frac{\partial \phi_j}{\partial x} \right) \frac{\partial \phi_i}{\partial x} \, dxdy \\
&+ \int_\Omega \nu \left(\sum_{j=1}^{N_b} u_{1j}(t)\frac{\partial \phi_j}{\partial y} \right) \frac{\partial \phi_i}{\partial y} \, dxdy \\
&+ \int_\Omega \nu \left(\sum_{j=1}^{N_b} u_{2j}(t)\frac{\partial \phi_j}{\partial x} \right) \frac{\partial \phi_i}{\partial y} \, dxdy - \int_\Omega \left(\sum_{j=1}^{N_{bp}} p_j(t)\psi_j \right) \frac{\partial \phi_i}{\partial x} \, dxdy \\
&= \int_\Omega f_1 \phi_i dxdy.
\end{aligned}
$$

Discretization formulation

- Set $\mathbf{v}_h = (0, \phi_i)^t$, i.e., $v_{1h} = 0$ and $v_{2h} = \phi_i$ $(i = 1, \cdots, N_b)$, in the first equation of the Galerkin formulation. Then

$$\int_\Omega \left(\sum_{j=1}^{N_b} u_{2j}(t)\phi_j \right)_t \phi_i \ dxdy + 2\int_\Omega \nu \left(\sum_{j=1}^{N_b} u_{2j}(t)\frac{\partial \phi_j}{\partial y} \right) \frac{\partial \phi_i}{\partial y} \ dxdy$$

$$+ \int_\Omega \nu \left(\sum_{j=1}^{N_b} u_{1j}(t)\frac{\partial \phi_j}{\partial y} \right) \frac{\partial \phi_i}{\partial x} \ dxdy$$

$$+ \int_\Omega \nu \left(\sum_{j=1}^{N_b} u_{2j}(t)\frac{\partial \phi_j}{\partial x} \right) \frac{\partial \phi_i}{\partial x} \ dxdy - \int_\Omega \left(\sum_{j=1}^{N_{bp}} p_j(t)\psi_j \right) \frac{\partial \phi_i}{\partial y} \ dxdy$$

$$= \int_\Omega f_2 \phi_i dxdy.$$

Discretization formulation

- Set $q_h = \psi_i$ $(i = 1, \cdots, N_{bp})$ in the second equation of the Galerkin formulation. Then

$$
\begin{aligned}
& -\int_\Omega \left(\sum_{j=1}^{N_b} u_{1j}(t) \frac{\partial \phi_j}{\partial x} \right) \psi_i \; dxdy \\
& -\int_\Omega \left(\sum_{j=1}^{N_b} u_{2j}(t) \frac{\partial \phi_j}{\partial y} \right) \psi_i \; dxdy \\
= \quad & 0.
\end{aligned}
$$

Discretization formulation

- Simplify the above three sets of equations, we obtain

$$\sum_{j=1}^{N_b} u_{1j}'(t) \int_\Omega \phi_j \phi_i \ dxdy + \sum_{j=1}^{N_b} u_{1j}(t) \left(2 \int_\Omega \nu \frac{\partial \phi_j}{\partial x} \frac{\partial \phi_i}{\partial x} \ dxdy + \int_\Omega \nu \frac{\partial \phi_j}{\partial y} \frac{\partial \phi_i}{\partial y} \ dxdy \right)$$

$$+ \sum_{j=1}^{N_b} u_{2j}(t) \int_\Omega \nu \frac{\partial \phi_j}{\partial x} \frac{\partial \phi_i}{\partial y} \ dxdy + \sum_{j=1}^{N_{bp}} p_j(t) \left(- \int_\Omega \psi_j \frac{\partial \phi_i}{\partial x} \ dxdy \right) = \int_\Omega f_1 \phi_i dxdy,$$

$$\sum_{j=1}^{N_b} u_{2j}'(t) \int_\Omega \phi_j \phi_i \ dxdy + \sum_{j=1}^{N_b} u_{1j}(t) \int_\Omega \nu \frac{\partial \phi_j}{\partial y} \frac{\partial \phi_i}{\partial x} \ dxdy$$

$$+ \sum_{j=1}^{N_b} u_{2j}(t) \left(2 \int_\Omega \nu \frac{\partial \phi_j}{\partial y} \frac{\partial \phi_i}{\partial y} \ dxdy + \int_\Omega \nu \frac{\partial \phi_j}{\partial x} \frac{\partial \phi_i}{\partial x} \ dxdy \right)$$

$$+ \sum_{j=1}^{N_{bp}} p_j(t) \left(- \int_\Omega \psi_j \frac{\partial \phi_i}{\partial y} \ dxdy \right) = \int_\Omega f_2 \phi_i dxdy$$

$$\sum_{j=1}^{N_b} u_{1j}(t) \left(- \int_\Omega \frac{\partial \phi_j}{\partial x} \psi_i \ dxdy \right) + \sum_{j=1}^{N_b} u_{2j}(t) \left(- \int_\Omega \frac{\partial \phi_j}{\partial y} \psi_i \ dxdy \right) + \sum_{j=1}^{N_{bp}} p_j(t) * 0 = 0.$$

Matrix formulation

- Define

$$A_1 = \left[\int_\Omega \nu \frac{\partial \phi_j}{\partial x}\frac{\partial \phi_i}{\partial x}\, dxdy\right]_{i,j=1}^{N_b}, \quad A_2 = \left[\int_\Omega \nu \frac{\partial \phi_j}{\partial y}\frac{\partial \phi_i}{\partial y}\, dxdy\right]_{i,j=1}^{N_b},$$

$$A_3 = \left[\int_\Omega \nu \frac{\partial \phi_j}{\partial x}\frac{\partial \phi_i}{\partial y}\, dxdy\right]_{i,j=1}^{N_b}, \quad A_4 = \left[\int_\Omega \nu \frac{\partial \phi_j}{\partial y}\frac{\partial \phi_i}{\partial x}\, dxdy\right]_{i,j=1}^{N_b},$$

$$A_5 = \left[\int_\Omega -\psi_j \frac{\partial \phi_i}{\partial x}\, dxdy\right]_{i=1,j=1}^{N_b,N_{bp}}, \quad A_6 = \left[\int_\Omega -\psi_j \frac{\partial \phi_i}{\partial y}\, dxdy\right]_{i=1,j=1}^{N_b,N_{bp}},$$

$$A_7 = \left[\int_\Omega -\frac{\partial \phi_j}{\partial x}\psi_i\, dxdy\right]_{i=1,j=1}^{N_{bp},N_b}, \quad A_8 = \left[\int_\Omega -\frac{\partial \phi_j}{\partial y}\psi_i\, dxdy\right]_{i=1,j=1}^{N_{bp},N_b}.$$

- Define a zero matrix $\mathbb{O}_1 = [0]_{i=1,j=1}^{N_{bp},N_{bp}}$ whose size is $N_{bp} \times N_{bp}$. Then

$$A = \begin{pmatrix} 2A_1 + A_2 & A_3 & A_5 \\ A_4 & 2A_2 + A_1 & A_6 \\ A_7 & A_8 & \mathbb{O}_1 \end{pmatrix}$$

Matrix formulation

- Each matrix above can be obtained by Algorithm I-3 in Chapter 3.
- It is not difficult to verify (an independent study project topic) that
$$A_4 = A_3^t, \quad A_7 = A_5^t, \quad A_8 = A_6^t.$$
- Hence the matrix A is actually symmetric:

$$A = \begin{pmatrix} 2A_1 + A_2 & A_3 & A_5 \\ A_3^t & 2A_2 + A_1 & A_6 \\ A_5^t & A_6^t & \mathbb{O}_1 \end{pmatrix}$$

Matrix formulation

- Define the basic mass matrix

$$M_e = [m_{ij}]_{i,j=1}^{N_b} = \left[\int_\Omega \phi_j \phi_i \ dxdy\right]_{i,j=1}^{N_b}.$$

- The mass matrix M_e can be obtained by Algorithm I-3 in Chapter 3, with $r = s = p = q = 0$ and $c = 1$.
- Define zero matrices $\mathbb{O}_2 = [0]_{i=1,j=1}^{N_b,N_{bp}}$ and $\mathbb{O}_3 = [0]_{i=1,j=1}^{N_b,N_b}$. Then define the block mass matrix

$$M = \begin{pmatrix} M_e & \mathbb{O}_3 & \mathbb{O}_2 \\ \mathbb{O}_3 & M_e & \mathbb{O}_2 \\ \mathbb{O}_2^t & \mathbb{O}_2^t & \mathbb{O}_1 \end{pmatrix}$$

Matrix formulation

- Define the load vector

$$\vec{b}(t) = \begin{pmatrix} \vec{b}_1(t) \\ \vec{b}_2(t) \\ \vec{0} \end{pmatrix}$$

where

$$\vec{b}_1(t) = \left[\int_\Omega f_1 \phi_i dxdy \right]_{i=1}^{N_b}, \quad \vec{b}_2(t) = \left[\int_\Omega f_2 \phi_i dxdy \right]_{i=1}^{N_b}.$$

Here the size of the zero vector is $N_{bp} \times 1$. That is, $\vec{0} = [0]_{i=1}^{N_{bp}}$.

- Each of $\vec{b}_1(t)$ and $\vec{b}_2(t)$ can be obtained by Algorithm II-5 in Chapter 4.

Matrix formulation

- Define the unknown vector

$$\vec{X}(t) = \begin{pmatrix} \vec{X}_1(t) \\ \vec{X}_2(t) \\ \vec{X}_3(t) \end{pmatrix}$$

where

$$\vec{X}_1(t) = [u_{1j}(t)]_{j=1}^{N_b}, \quad \vec{X}_2(t) = [u_{2j}(t)]_{j=1}^{N_b}, \quad \vec{X}_3(t) = [p_j(t)]_{j=1}^{N_{bp}}.$$

Matrix formulation

- We obtain the first order ODE system

$$M\vec{X}'(t) + A\vec{X}(t) = \vec{b}(t).$$

- The structure of this ODE system is the same as that of the first order ODE system obtained for the second order parabolic equation in Chapter 3.
- Hence the same finite difference schemes in Chapter 3 can be directly utilized for this ODE system.
- The major differences between this ODE system and the one in Chapter 3 are the details in the definition of M, A, $\vec{X}$ and $\vec{b}$, which were discussed above.

Assembly of a time-independent matrix

Recall Algorithm I-3 from Chapter 3:

- Initialize the matrix: $A = sparse(N_b, N_b)$;
- Compute the integrals and assemble them into A:

FOR $n = 1, \cdots, N$:

FOR $\alpha = 1, \cdots, N_{lb}$:

FOR $\beta = 1, \cdots, N_{lb}$:

Compute $r = \int_{E_n} c \frac{\partial^{r+s}\psi_{n\alpha}}{\partial x^r \partial y^s} \frac{\partial^{p+q}\psi_{n\beta}}{\partial x^p \partial y^q}\, dxdy$;

Add r to $A(T_b(\beta, n), T_b(\alpha, n))$.

END

END

END

Assembly of the time-independent stiffness matrix

- Call Algorithm I-3 with $r = 1$, $s = 0$, $p = 1$, $q = 0$, $c = \nu$, basis type of $\mathbf{u}$ for trial function, and basis type of $\mathbf{u}$ for test function, to obtain A_1.
- Call Algorithm I-3 with $r = 0$, $s = 1$, $p = 0$, $q = 1$, $c = \nu$, basis type of $\mathbf{u}$ for trial function, and basis type of $\mathbf{u}$ for test function, to obtain A_2.
- Call Algorithm I-3 with $r = 1$, $s = 0$, $p = 0$, $q = 1$, $c = \nu$, basis type of $\mathbf{u}$ for trial function, and basis type of $\mathbf{u}$ for test function, to obtain A_3.
- Call Algorithm I-3 with $r = 0$, $s = 0$, $p = 1$, $q = 0$, $c = -1$, basis type of p for trial function, and basis type of $\mathbf{u}$ for test function, to obtain A_5.
- Call Algorithm I-3 with $r = 0$, $s = 0$, $p = 0$, $q = 1$, $c = -1$, basis type of p for trial function, and basis type of $\mathbf{u}$ for test function, to obtain A_6.
- Generate a zero matrix $\mathbb{O}$ whose size is $N_{bp} \times N_{bp}$.
- Then the stiffness matrix $A = [A_1 + 2A_2 \quad A_3 \quad A_5; A_3^t \quad 2A_2 + A_1 \quad A6; A_5^t \quad A_6^t \quad \mathbb{O}]$.

Assembly of the mass matrix

- Call Algorithm I-3 with $r = 0$, $s = 0$, $p = 0$, $q = 0$, $c = 1$, basis type of $\mathbf{u}$ for trial function, and basis type of $\mathbf{u}$ for test function, to obtain the basic mass matrix M_e.
- Generate three zero matrices $\mathbb{O}_1$, $\mathbb{O}_2$, and $\mathbb{O}_3$ whose sizes are $N_{bp} \times N_{bp}$, $N_b \times N_{bp}$, and $N_b \times N_b$, respectively.
- Then the block mass matrix $M = [M_e \quad \mathbb{O}_3 \quad \mathbb{O}_2; \mathbb{O}_3 \quad M_e \quad \mathbb{O}_2; \mathbb{O}_2^t \quad \mathbb{O}_2^t \quad \mathbb{O}_1]$.

Assembly of a time-independent vector

Recall Algorithm II-3 from Chapter 3:

- Initialize the matrix: $b = sparse(N_b, 1)$;
- Compute the integrals and assemble them into b:

$FOR\ n = 1, \cdots, N$:

$\quad FOR\ \beta = 1, \cdots, N_{lb}$:

$\quad\quad$ Compute $r = \int_{E_n} f \frac{\partial^{p+q}\psi_{n\beta}}{\partial x^p \partial y^q}\, dxdy$;

$\quad\quad b(T_b(\beta, n), 1) = b(T_b(\beta, n), 1) + r$;

$\quad END$

END

Assembly of a time-dependent vector

Recall Algorithm II-5 from Chapter 4:

- Specify a value for the time t based on the input time;
- Initialize the vector: $b = sparse(N_b, 1)$;
- Compute the integrals and assemble them into b:

$FOR\ n = 1, \cdots, N$:

$\quad FOR\ \beta = 1, \cdots, N_{lb}$:

$\quad\quad$ Compute $r = \int_{E_n} f(t) \frac{\partial^{p+q}\psi_{n\beta}}{\partial x^p \partial y^q}\, dxdy$;

$\quad\quad b(T_b(\beta, n), 1) = b(T_b(\beta, n), 1) + r$;

$\quad END$

END

Assembly of the load vector

- Call Algorithm II-5 with $p = q = 0$ and $f = f_1$ to obtain $b_1(t)$.
- Call Algorithm II-5 with $p = q = 0$ and $f = f_2$ to obtain $b_2(t)$.
- Define a zero column vector $\vec{0}$ whose size is $N_{bp} \times 1$.
- Then the load vector $\vec{b} = [b_1(t); b_2(t); \vec{0}]$.
- If f_1 and f_2 do not depend on t, then this part is exactly the same as the assembly of the load vector with Algorithm II-3 in Chapter 7.

Time-dependent Dirichlet boundary condition

Since Algorithm III-3 Chapter 6 is time-independent, it is not suitable for the time-dependent Dirichlet boundary condition in this chapter. Therefore, we will use the following Algorithm III-4:

- Specify a value for the time t based on the input time;
- Deal with the Dirichlet boundary conditions:

FOR $k = 1, \cdots, nbn$:

  If $boundarynodes(1, k)$ shows Dirichlet condition, then

    $i = boundarynodes(2, k)$;
    $\tilde{A}(i, :) = 0$;
    $\tilde{A}(i, i) = 1$;
    $\tilde{b}(i) = g_1(P_b(:, i), t)$;
    $\tilde{A}(N_b + i, :) = 0$;
    $\tilde{A}(N_b + i, N_b + i) = 1$;
    $\tilde{b}(N_b + i) = g_2(P_b(:, i), t)$;

  ENDIF

END

Outline

Temporal discretization for the ODE system

- Assume that we have a uniform partition of $[0, T]$ into M_m elements with mesh size $\triangle t$.
- The mesh nodes are $t_m = m\triangle t, \;\; m = 0, 1, \cdots, M_m$.
- Assume $\vec{X}^m$ is the numerical solution of $\vec{X}(t_m)$.
- Then the corresponding $\theta-$scheme is

$$M\frac{\vec{X}^{m+1} - \vec{X}^m}{\triangle t} + \theta A\vec{X}^{m+1} + (1-\theta)A\vec{X}^m = \theta\vec{b}(t_{m+1}) + (1-\theta)\vec{b}(t_m)$$

$$\Rightarrow \quad \left(\frac{M}{\triangle t} + \theta A\right)\vec{X}^{m+1} = \theta\vec{b}(t_{m+1}) + (1-\theta)\vec{b}(t_m) + \frac{M}{\triangle t}\vec{X}^m - (1-\theta)A\vec{X}^m.$$

Temporal discretization for the ODE system

- Iteration scheme 2:

$$\tilde{A}\vec{X}^{m+1} = \tilde{\vec{b}}^{m+1}, \ \ m = 0, \cdots, M_m - 1,$$

where

$$\tilde{A} = \frac{M}{\triangle t} + \theta A,$$

$$\tilde{\vec{b}}^{m+1} = \theta \vec{b}(t_{m+1}) + (1-\theta)\vec{b}(t_m) + \left[\frac{M}{\triangle t} - (1-\theta)A\right]\vec{X}^m.$$

Temporal discretization for the ODE system

Algorithm B:

- Generate the mesh information matrices P and T.
- Assemble the mass matrix M by using Algorithm I-3.
- Assemble the stiffness matrix A by using Algorithm I-3.
- Generate the initial vector $\vec{X}^0$.
- Iterate in time:

  FOR $m = 0, \cdots, M_m - 1$

  $t_{m+1} = (m+1)\triangle t;$

  $t_m = m\triangle t;$

  Assemble the load vectors $\vec{b}(t_{m+1})$ and $\vec{b}(t_m)$ by using Algorithm II-5 at $t = t_{m+1}$ and $t = t_m$;

  Deal with Dirichlet boundary conditions by using Algorithm III-4 for $\tilde{A}$ and $\tilde{\vec{b}}^{m+1}$ at $t = t_{m+1}$;

  Solve iteration scheme 2 for $\vec{X}^{m+1}$.

  END

Temporal discretization for the ODE system

- Define $\vec{X}^{m+\theta} = \theta\vec{X}^{m+1} + (1-\theta)\vec{X}^m$.
- Then $\vec{X}^{m+1} - \vec{X}^m = \frac{\vec{X}^{m+\theta} - \vec{X}^m}{\theta}$ if $\theta \neq 0$.
- Hence

$$
\begin{aligned}
& M\frac{\vec{X}^{m+1} - \vec{X}^m}{\triangle t} + \theta A\vec{X}^{m+1} + (1-\theta)A\vec{X}^m = \theta\vec{b}(t_{m+1}) + (1-\theta)\vec{b}(t_m) \\
\Rightarrow\quad & M\frac{\vec{X}^{m+1} - \vec{X}^m}{\triangle t} + A\left[\theta\vec{X}^{m+1} + (1-\theta)\vec{X}^m\right] = \theta\vec{b}(t_{m+1}) + (1-\theta)\vec{b}(t_m) \\
\Rightarrow\quad & M\frac{\vec{X}^{m+\theta} - \vec{X}^m}{\theta\triangle t} + A\vec{X}^{m+\theta} = \theta\vec{b}(t_{m+1}) + (1-\theta)\vec{b}(t_m) \\
\Rightarrow\quad & \left(\frac{M}{\theta\triangle t} + A\right)\vec{X}^{m+\theta} = \theta\vec{b}(t_{m+1}) + (1-\theta)\vec{b}(t_m) + \frac{M\vec{X}^m}{\theta\triangle t}.
\end{aligned}
$$

Temporal discretization for the ODE system

- Iteration scheme 3:

$$\tilde{A}^{\theta}\vec{X}^{m+\theta} = \tilde{\vec{b}}^{m+\theta}, \ \ m = 0, \cdots, M_m - 1,$$

where

$$\tilde{A}^{\theta} = \frac{M}{\theta \triangle t} + A,$$

$$\tilde{\vec{b}}^{m+\theta} = \theta \vec{b}(t_{m+1}) + (1-\theta)\vec{b}(t_m) + \frac{M}{\theta \triangle t}\vec{X}^m.$$

- Since $\vec{X}^{m+\theta} = \theta \vec{X}^{m+1} + (1-\theta)\vec{X}^m$, then

$$\vec{X}^{m+1} = \frac{\vec{X}^{m+\theta} - \vec{X}^m}{\theta} + \vec{X}^m.$$

Temporal discretization for the ODE system

Algorithm C:

- Generate the mesh information matrices P and T.
- Assemble the mass matrix M by using Algorithm I-3.
- Assemble the stiffness matrix A by using Algorithm I-3.
- Generate the initial vector $\vec{X}^0$.
- Iterate in time:

  FOR $m = 0, \cdots, M_m - 1$

  $t_{m+1} = (m+1)\triangle t;$

  $t_m = m\triangle t;$

  Assemble the load vectors $\vec{b}(t_{m+1})$ and $\vec{b}(t_m)$ by using Algorithm II-5 at $t = t_{m+1}$ and $t = t_m$;

  Deal with boundary conditions by using Algorithm III-4 for $\tilde{A}^\theta$ and $\tilde{\vec{b}}^{m+\theta}$ at $t = t_{m+\theta}$;

  Solve iteration scheme 3 for $\vec{X}^{m+1}$.

  END

Numerical example

- Example 1: Use the finite element method to solve the following equation on the domain $\Omega = [0,1] \times [-0.25, 0]$:

$$
\begin{aligned}
&\mathbf{u}_t - \nabla \cdot \mathbb{T}(\mathbf{u}, p) = \mathbf{f} \quad \text{in } \Omega \times [0,1], \\
&\nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega \times [0,1], \\
&u_1 = x^2 y^2 + e^{-y}, \quad \text{at } t = 0 \text{ and in } \Omega, \\
&u_2 = -\frac{2}{3} x y^3 + 2 - \pi \sin(\pi x), \quad \text{at } t = 0 \text{ and in } \Omega, \\
&p = -[2 - \pi \sin(\pi x)] \cos(2\pi y), \quad \text{at } t = 0 \text{ and in } \Omega,
\end{aligned}
$$

Numerical example

- Continued formulation:

$$
\begin{aligned}
u_1 &= e^{-y}cos(2\pi t) \quad \text{on } x = 0, \\
u_1 &= (y^2 + e^{-y})cos(2\pi t) \quad \text{on } x = 1, \\
u_1 &= \left(\frac{1}{16}x^2 + e^{0.25}\right) cos(2\pi t) \quad \text{on } y = -0.25, \\
u_1 &= cos(2\pi t) \quad \text{on } y = 0, \\
u_2 &= 2cos(2\pi t) \quad \text{on } x = 0, \\
u_2 &= \left(-\frac{2}{3}y^3 + 2\right) cos(2\pi t) \quad \text{on } x = 1, \\
u_2 &= \left[\frac{1}{96}x + 2 - \pi \sin(\pi x)\right] cos(2\pi t) \quad \text{on } y = -0.25, \\
u_2 &= [2 - \pi \sin(\pi x)]cos(2\pi t) \quad \text{on } y = 0.
\end{aligned}
$$

Numerical example

- Here

$$
\begin{aligned}
f_1 &= -2\pi(x^2y^2 + e^{-y})sin(2\pi t) \\
&\quad +[-2\nu x^2 - 2\nu y^2 - \nu e^{-y} + \pi^2 \cos(\pi x)\cos(2\pi y)]cos(2\pi t), \\
f_2 &= -2\pi\left[-\frac{2}{3}xy^3 + 2 - \pi\sin(\pi x)\right] sin(2\pi t) \\
&\quad +[4\nu xy - \nu\pi^3\sin(\pi x) \\
&\quad +2\pi(2 - \pi\sin(\pi x))\sin(2\pi y)]cos(2\pi t).
\end{aligned}
$$

Numerical example

- The analytic solution of this problem is

$$
\begin{aligned}
u_1 &= (x^2y^2 + e^{-y})cos(2\pi t), \\
u_2 &= \left[-\frac{2}{3}xy^3 + 2 - \pi \sin(\pi x)\right] cos(2\pi t), \\
p &= -[2 - \pi \sin(\pi x)] \cos(2\pi y) cos(2\pi t),
\end{aligned}
$$

which can be used to compute the errors between the numerical solution and the analytic solution. We can also verify f_1 and f_2 above by plugging the analytic solutions into the Stokes equation.

Numerical example

- Let's code for the Taylor-Hood finite elements for the 2D Stokes equation together!
- Taylor-Hood finite elements: linear finite elements for the pressure and quadratic finite elements for the velocity.
- We will use Algorithm B.
- Open your Matlab!

Numerical example

h	$\|\mathbf{u}-\mathbf{u}_h\|_\infty$	$\|\mathbf{u}-\mathbf{u}_h\|_0$	$\vert\mathbf{u}-\mathbf{u}_h\vert_1$
$1/8$	1.6676×10^{-3}	3.6290×10^{-4}	2.0487×10^{-2}
$1/16$	2.1848×10^{-4}	4.5026×10^{-5}	5.0726×10^{-3}
$1/32$	2.7448×10^{-5}	5.6114×10^{-6}	1.2626×10^{-3}
$1/64$	3.3781×10^{-6}	7.0079×10^{-7}	3.1525×10^{-4}

Table: Case 1: The numerical errors at $t=1$ for quadratic finite elements of the velocity and backward Euler scheme ($\theta=1$) with $\triangle t=8h^3$.

- Any Observation?

Numerical example

- Third order convergence $O(h^3)$ in L^2/L^∞ norm and second order convergence $O(h^2)$ in H^1 semi-norm.
- The backward Euler scheme has first order accuracy for temporal discretization.
- The quadratic finite element has third order accuracy in L^2/L^∞ norm and second order accuracy in H^1 semi-norm for spatial discretization.
- Hence the accuracy order is expected to be $O(\triangle t + h^3)$ in L^2/L^∞ norm and $O(\triangle t + h^2)$ in H^1 norm, which match the above observation since $\triangle t = 8h^3$ in case 1.

Numerical example

h	$\|p-p_h\|_\infty$	$\|p-p_h\|_0$	$|p-p_h|_1$
1/8	5.7967×10^{-1}	1.3909×10^{-1}	1.3489×10^{0}
1/16	9.4258×10^{-2}	2.3063×10^{-2}	6.3538×10^{-1}
1/32	1.8080×10^{-2}	4.2194×10^{-3}	3.1396×10^{-1}
1/64	3.8072×10^{-3}	8.6779×10^{-4}	1.5660×10^{-1}

Table: Case 1: The numerical errors at $t=1$ for linear finite elements of the pressure and backward Euler scheme ($\theta=1$) with $\triangle t=8h^3$.

- Any Observation?

Numerical example

- Second order convergence $O(h^2)$ in L^2/L^∞ norm and first order convergence $O(h)$ in H^1 semi-norm.
- The backward Euler scheme has second order accuracy for temporal discretization.
- The linear finite element has second order accuracy in L^2/L^∞ norm and first order accuracy in H^1 semi-norm for spatial discretization.
- Hence the accuracy order is expected to be $O(\triangle t + h^2)$ in L^2/L^∞ norm and $O(\triangle t + h)$ in H^1 norm, which match the above observation since $\triangle t = 8h^3$ in case 1.

Numerical example

- However, you will also observe high cost in time for this case since $\triangle t = 8h^3$ is much smaller than that of the previous cases.
- When the mesh becomes finer and finer or the problem becomes 3D, the situation is even worse.
- This is why we need temporal discretization with higher order accuracy and efficient methods to solve linear systems.

Numerical example

h	$\|\|\mathbf{u}-\mathbf{u}_h\|\|_\infty$	$\|\|\mathbf{u}-\mathbf{u}_h\|\|_0$	$\|\mathbf{u}-\mathbf{u}_h\|_1$
$1/8,\ 1/32$	1.6027×10^{-3}	3.5322×10^{-4}	2.0242×10^{-2}
$1/16,\ 1/64$	1.9654×10^{-4}	4.3845×10^{-5}	5.0469×10^{-3}
$1/32,\ 1/256$	2.5111×10^{-5}	5.4811×10^{-6}	1.2619×10^{-3}
$1/64,\ 1/512$	3.1014×10^{-6}	6.8432×10^{-7}	3.1519×10^{-4}

Table: Case 2: The numerical errors at $t=1$ for quadratic finite elements of the velocity and Crank-Nicolson scheme ($\theta=\frac{1}{2}$) with $\triangle t^2\leq h^3$.

- Any Observation?

Numerical example

- Third order convergence $O(h^3)$ in L^2/L^∞ norm and second order convergence $O(h^2)$ in H^1 semi-norm.
- The Crank-Nicolson scheme has second order accuracy for temporal discretization.
- The quadratic finite element has third order accuracy in L^2/L^∞ norm and second order accuracy in H^1 semi-norm for spatial discretization.
- Hence the accuracy order is expected to be $O(\triangle t^2 + h^3)$ in L^2/L^∞ norm and $O(\triangle t^2 + h^2)$ in H^1 norm, which match the above observation since $\triangle t^2 \approx h^3$ in case 2.

Numerical example

h	$\|p - p_h\|_\infty$	$\|p - p_h\|_0$	$\vert p - p_h\vert_1$
$1/8,\ 1/32$	2.0901×10^{-1}	3.8144×10^{-2}	1.2300×10^{0}
$1/16,\ 1/64$	5.9514×10^{-2}	9.5006×10^{-3}	6.2249×10^{-1}
$1/32,\ 1/256$	1.8457×10^{-2}	2.4493×10^{-3}	3.1202×10^{-1}
$1/64,\ 1/512$	5.1034×10^{-3}	6.0165×10^{-4}	1.5634×10^{-1}

Table: Case 2: The numerical errors at $t = 1$ for linear finite elements of the pressure and Crank-Nicolson scheme ($\theta = \frac{1}{2}$) with $\triangle t^2 \leq h^3$.

- Any Observation?

Numerical example

- Second order convergence $O(h^2)$ in L^2/L^∞ norm and first order convergence $O(h)$ in H^1 semi-norm.
- The Crank-Nicolson scheme has second order accuracy for temporal discretization.
- The linear finite element has second order accuracy in L^2/L^∞ norm and first order accuracy in H^1 semi-norm for spatial discretization.
- Hence the accuracy order is expected to be $O(\triangle t^2 + h^2)$ in L^2/L^∞ norm and $O(\triangle t^2 + h)$ in H^1 norm, which match the above observation since $\triangle t^2 \approx h^3$ in case 2.

Outline

Efficient methods

- Forward Euler: cheap at each time iteration step, but conditionally stable, which means that $\triangle t$ must be smaller enough.

- Multi-step methods for temporal discretization: two-step backward differentiation, three-step backward differentiation, Runge-Kutta method......

- Efficient solvers for linear systems: multi-grid, PCG, GMRES......

Mixed boundary conditions

- The treatment of the stress/Robin boundary conditions is similar to that of Chapter 7.
- If the functions in the stress/Robin boundary conditions are independent of time, then the same subroutines from Chapter 7 can be used before the time iteration starts.
- If the functions in the stress/Robin boundary conditions depend on time, then the same algorithms as those in Chapter 7 can be used at each time iteration step. But the time needs to be specified in these algorithms.

Mixed boundary conditions

- Consider

$$\mathbf{u}_t - \nabla \cdot \mathbb{T}(\mathbf{u}, p) = \mathbf{f} \quad \text{in } \Omega \times [0, T],$$
$$\nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega \times [0, T],$$
$$\mathbb{T}(\mathbf{u}, p)\mathbf{n} = \mathbf{p} \quad \text{on } \Gamma_S \times [0, T],$$
$$\mathbb{T}(\mathbf{u}, p)\mathbf{n} + r\mathbf{u} = \mathbf{q} \quad \text{on } \Gamma_R \times [0, T],$$
$$\mathbf{u} = \mathbf{g} \quad \text{on } \Gamma_D \times [0, T],$$
$$\mathbf{u} = \mathbf{u}_0, \quad \text{at } t = 0 \text{ and in } \Omega.$$

where $\Gamma_S, \ \Gamma_R \subset \partial\Omega$ and $\Gamma_D = \partial\Omega / (\Gamma_S \cup \Gamma_R)$.

Mixed boundary conditions

- Recall

$$
\int_\Omega \mathbf{u}_t \cdot \mathbf{v} \, dxdy + \int_\Omega 2\nu \mathbb{D}(\mathbf{u}) : \mathbb{D}(\mathbf{v}) \, dxdy - \int_\Omega p(\nabla \cdot \mathbf{v}) \, dxdy
$$

$$
- \int_{\partial\Omega} (\mathbb{T}(\mathbf{u}, p)\mathbf{n}) \cdot \mathbf{v} \, ds \; = \; \int_\Omega \mathbf{f} \cdot \mathbf{v} \, dxdy,
$$

$$
- \int_\Omega (\nabla \cdot \mathbf{u}) q \, dxdy = 0.
$$

- Since the solution on $\Gamma_D = \partial\Omega/(\Gamma_S \cup \Gamma_R)$ is given by $\mathbf{u} = \mathbf{g}$, then we can choose the test function $\mathbf{v}(x, y)$ such that $\mathbf{v} = 0$ on $\partial\Omega/(\Gamma_S \cup \Gamma_R)$.

Mixed boundary conditions

- Hence, similar to the treatment of the mixed boundary condition in Chapter 7, the weak formulation is to find $\mathbf{u} \in H^1(0,T;[H^1(\Omega)]^2)$ and $p \in L^2(0,T;L^2(\Omega))$ such that

$$
\begin{aligned}
&\int_\Omega \mathbf{u}_t \cdot \mathbf{v}\; dxdy + \int_\Omega 2\nu \mathbb{D}(\mathbf{u}) : \mathbb{D}(\mathbf{v})\; dxdy \\
&- \int_\Omega p(\nabla \cdot \mathbf{v})\; dxdy + \int_{\Gamma_R} r\mathbf{u} \cdot \mathbf{v}\; ds \\
&= \int_\Omega \mathbf{f} \cdot \mathbf{v}\; dxdy + \int_{\Gamma_R} \mathbf{q} \cdot \mathbf{v}\; ds + \int_{\Gamma_S} \mathbf{p} \cdot \mathbf{v}\; ds, \\
&- \int_\Omega (\nabla \cdot \mathbf{u}) q\; dxdy = 0.
\end{aligned}
$$

  for any $\mathbf{v} \in [H^1_{0D}(\Omega)]^2$ and $q \in L^2(\Omega)$ where $H^1_{0D}(\Omega) = \{w \in H^1(\Omega) : w = 0 \text{ on } \Gamma_D\}$.

- Code? Combine all of the subroutines for Dirichlet/Stress/Robin boundary conditions.

Mixed boundary conditions in normal/tangential directions

- Consider

$$
\begin{aligned}
&\mathbf{u}_t - \nabla \cdot \mathbb{T}(\mathbf{u}, p) = \mathbf{f} \quad \text{in } \Omega \times [0, T], \\
&\nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega \times [0, T], \\
&\mathbf{n}^t \mathbb{T}(\mathbf{u}, p)\mathbf{n} = p_n, \tau^t \mathbb{T}(\mathbf{u}, p)\mathbf{n} = p_\tau \text{ on } \Gamma_S \times [0, T], \\
&\mathbf{n}^t \mathbb{T}(\mathbf{u}, p)\mathbf{n} + r\mathbf{n}^t \mathbf{u} = q_n, \tau^t \mathbb{T}(\mathbf{u}, p)\mathbf{n} + r\tau^t \mathbf{u} = q_\tau \text{ on } \Gamma_R \times [0, T], \\
&\mathbf{u} = \mathbf{g} \quad \text{on } \Gamma_D \times [0, T], \\
&\mathbf{u} = \mathbf{u}_0, \quad \text{at } t = 0 \text{ and in } \Omega.
\end{aligned}
$$

where $\Gamma_S,\ \Gamma_R \subset \partial\Omega$, $\Gamma_D = \partial\Omega/(\Gamma_S \cup \Gamma_R)$, $\mathbf{n} = (n_1,\ n_2)^t$ is the unit outer normal vector of $\partial\Omega$, and $\tau = (\tau_1, \tau_2)^t$ is the corresponding unit tangential vector of $\partial\Omega$.

Mixed boundary conditions in normal/tangential directions

- Recall

$$
\begin{aligned}
&\int_\Omega \mathbf{u}_t \cdot \mathbf{v}\ dxdy + \int_\Omega 2\nu\mathbb{D}(\mathbf{u}) : \mathbb{D}(\mathbf{v})\ dxdy - \int_\Omega p(\nabla \cdot \mathbf{v})\ dxdy \\
&- \int_{\partial\Omega} (\mathbb{T}(\mathbf{u}, p)\mathbf{n}) \cdot \mathbf{v}\ ds \ = \ \int_\Omega \mathbf{f} \cdot \mathbf{v}\ dxdy, \\
&- \int_\Omega (\nabla \cdot \mathbf{u})q\ dxdy = 0.
\end{aligned}
$$

- Since the solution on $\Gamma_D = \partial\Omega/(\Gamma_S \cup \Gamma_R)$ is given by $\mathbf{u} = \mathbf{g}$, then we can choose the test function $\mathbf{v}(x, y)$ such that $\mathbf{v} = 0$ on $\partial\Omega/(\Gamma_S \cup \Gamma_R)$.

Mixed boundary conditions in normal/tangential directions

- Similar to the derivation of mixed boundary conditions in normal/tangential directions in Chapter 7, we obtain

$$
\begin{aligned}
&\int_{\partial\Omega} (\mathbb{T}(\mathbf{u}, p)\mathbf{n}) \cdot \mathbf{v} \; ds \\
= \;& \int_{\Gamma_S} (\mathbb{T}(\mathbf{u}, p)\mathbf{n}) \cdot \mathbf{v} \; ds + \int_{\Gamma_R} (\mathbb{T}(\mathbf{u}, p)\mathbf{n}) \cdot \mathbf{v} \; ds \\
&+ \int_{\partial\Omega/(\Gamma_S \cup \Gamma_R)} (\mathbb{T}(\mathbf{u}, p)\mathbf{n}) \cdot \mathbf{v} \; ds \\
= \;& \left[\int_{\Gamma_S} p_n(\mathbf{n}^t\mathbf{v}) \; ds + \int_{\Gamma_S} p_\tau(\tau^t\mathbf{v}) \; ds \right] \\
&+ \left[\int_{\Gamma_R} q_n(\mathbf{n}^t\mathbf{v}) \; ds + \int_{\Gamma_R} q_\tau(\tau^t\mathbf{v}) \; ds \right] \\
&- \left[\int_{\Gamma_R} (r\mathbf{n}^t\mathbf{u})(\mathbf{n}^t\mathbf{v}) \; ds + \int_{\Gamma_R} (r\tau^t\mathbf{u})(\tau^t\mathbf{v}) \; ds \right],
\end{aligned}
$$

Mixed boundary conditions in normal/tangential directions

- Hence, similar to the treatment of the mixed boundary conditions in normal/tangential directions in Chapter 7, the weak formulation is to find $\mathbf{u} \in H^1(0, T; [H^1(\Omega)]^2)$ and $p \in L^2(0, T; L^2(\Omega))$ such that

$$\int_\Omega \mathbf{u}_t \cdot \mathbf{v}\ dxdy + \int_\Omega 2\nu \mathbb{D}(\mathbf{u}) : \mathbb{D}(\mathbf{v})\ dxdy - \int_\Omega p(\nabla \cdot \mathbf{v})\ dxdy$$
$$+ \int_{\Gamma_R} (r\mathbf{n}^t\mathbf{u})(\mathbf{n}^t\mathbf{v})\ ds + \int_{\Gamma_R} (r\tau^t\mathbf{u})(\tau^t\mathbf{v})\ ds$$
$$= \int_\Omega \mathbf{f} \cdot \mathbf{v}\ dxdy + \int_{\Gamma_R} q_n(\mathbf{n}^t\mathbf{v})\ ds + \int_{\Gamma_R} q_\tau(\tau^t\mathbf{v})\ ds$$
$$+ \int_{\Gamma_S} p_n(\mathbf{n}^t\mathbf{v})\ ds + \int_{\Gamma_S} p_\tau(\tau^t\mathbf{v})\ ds,$$
$$- \int_\Omega (\nabla \cdot \mathbf{u})q\ dxdy = 0,$$

  for any $\mathbf{v} \in [H^1_{0D}(\Omega)]^2$ and $q \in L^2(\Omega)$.

- Code? Combine all of the subroutines for Dirichlet/Stress/Robin boundary conditions.

Another format of full discretization

- Recall the Galerkin formulation of the semi-discretization (without considering the Dirichlet boundary condition, which will be handled later): find $\mathbf{u}_h \in H^1(0,T;[U_h]^2)$ and $p_h \in L^2(0,T;W_h)$ such that

$$\begin{aligned}(\mathbf{u}_{h_t}, \mathbf{v}) + a(\mathbf{u}_h, \mathbf{v}_h) + b(\mathbf{v}_h, p_h) &= (\mathbf{f}, \mathbf{v}_h), \\ b(\mathbf{u}_h, q_h) &= 0,\end{aligned}$$

  for any $\mathbf{v}_h \in [U_h]^2$ and $q_h \in W_h$.

- Instead of obtaining the matrix formulation from this semi-discretization and proposing the full discretization based on the matrix formulation, we can first present the full discretization based on this semi-discretization and then obtain the matrix formulation for the full discretization.

Another format of full discretization

- In more details of the vector format, the Galerkin formulation (without considering the Dirichlet boundary condition, which will be handled later) is to find $\mathbf{u}_h \in H^1(0, T; [U_h]^2)$ and $p_h \in L^2(0, T; W_h)$ such that

$$\begin{aligned} &\int_\Omega \mathbf{u}_{h_t} \cdot \mathbf{v}_h \, dxdy + \int_\Omega 2\nu \mathbb{D}(\mathbf{u}_h) : \mathbb{D}(\mathbf{v}_h) \, dxdy \\ &- \int_\Omega p_h (\nabla \cdot \mathbf{v}_h) \, dxdy = \int_\Omega \mathbf{f} \cdot \mathbf{v}_h \, dxdy, \\ &- \int_\Omega (\nabla \cdot \mathbf{u}_h) q_h \, dxdy = 0, \end{aligned}$$

for any $\mathbf{v}_h \in [U_h]^2$ and $q_h \in W_h$.

Another format of full discretization

- In the scalar format, the Galerkin formulation (without considering the Dirichlet boundary condition, which will be handled later) is to find $u_{1h} \in H^1(0, T; U_h)$, $u_{2h} \in H^1(0, T; U_h)$, and $p_h \in L^2(0, T; W_h)$ such that

$$\int_\Omega \frac{\partial u_{1h}}{\partial t} v_{1h}\ dxdy + \int_\Omega \frac{\partial u_{2h}}{\partial t} v_{2h}\ dxdy$$
$$+ \int_\Omega \nu \Big(2\frac{\partial u_{1h}}{\partial x}\frac{\partial v_{1h}}{\partial x} + 2\frac{\partial u_{2h}}{\partial y}\frac{\partial v_{2h}}{\partial y} + \frac{\partial u_{1h}}{\partial y}\frac{\partial v_{1h}}{\partial y}$$
$$+ \frac{\partial u_{1h}}{\partial y}\frac{\partial v_{2h}}{\partial x} + \frac{\partial u_{2h}}{\partial x}\frac{\partial v_{1h}}{\partial y} + \frac{\partial u_{2h}}{\partial x}\frac{\partial v_{2h}}{\partial x}\Big)\ dxdy$$
$$- \int_\Omega \left(p_h \frac{\partial v_{1h}}{\partial x} + p_h \frac{\partial v_{2h}}{\partial y} \right)\ dxdy = \int_\Omega (f_1 v_{1h} + f_2 v_{2h})\ dxdy.$$
$$- \int_\Omega \left(\frac{\partial u_{1h}}{\partial x} q_h + \frac{\partial u_{2h}}{\partial y} q_h \right)\ dxdy = 0.$$

for any $v_{1h} \in U_h$, $v_{2h} \in U_h$, and $q_h \in W_h$.

Another format of full discretization

- Assume that we have a uniform partition of $[0, T]$ into M_m elements with mesh size $\triangle t$.
- The mesh nodes are $t_m = m\triangle t, \;\; m = 0, 1, \cdots, M_m$.
- Let $\mathbf{u}_h^0$ and p_h^0 denote the given initial condition at t_0.
- Let $\mathbf{u}_h^m$ and p_h^m denote the numerical solution at t_m.
- Then we consider the full discretization (without considering the Dirichlet boundary condition, which will be handled later): for $m = 0, \cdots, M_m - 1$, find $\mathbf{u}_h^{m+1} \in [U_h]^2$ and $p_h^{m+1} \in W_h$ such that

$$
\begin{aligned}
&\left(\frac{\mathbf{u}_h^{m+1} - \mathbf{u}_h^m}{\triangle t}, \mathbf{v}\right) + \theta a(\mathbf{u}_h^{m+1}, \mathbf{v}_h) + (1-\theta) a(\mathbf{u}_h^m, \mathbf{v}_h) \\
&+\theta b(\mathbf{v}_h, p_h^{m+1}) + (1-\theta) b(\mathbf{v}_h, p_h^m) \\
&= \theta(\mathbf{f}(t_{m+1}), \mathbf{v}_h) + (1-\theta)(\mathbf{f}(t_m), \mathbf{v}_h), \\
&\theta b(\mathbf{u}_h^{m+1}, q_h) + (1-\theta) b(\mathbf{u}_h^m, q_h) = 0,
\end{aligned}
$$

Another format of full discretization

- That is, for $m = 0, \cdots, M_m - 1$, find $\mathbf{u}_h^{m+1} \in [U_h]^2$ and $p_h^{m+1} \in W_h$ such that

$$\int_\Omega \frac{\mathbf{u}_h^{m+1} - \mathbf{u}_h^m}{\triangle t} \cdot \mathbf{v}_h \; dxdy + \theta \int_\Omega 2\nu \mathbb{D}(\mathbf{u}_h^{m+1}) : \mathbb{D}(\mathbf{v}_h) \; dxdy$$
$$+(1-\theta) \int_\Omega 2\nu \mathbb{D}(\mathbf{u}_h^m) : \mathbb{D}(\mathbf{v}_h) \; dxdy$$
$$-\theta \int_\Omega p_h^{m+1} (\nabla \cdot \mathbf{v}_h) \; dxdy - (1-\theta) \int_\Omega p_h^m (\nabla \cdot \mathbf{v}_h) \; dxdy$$
$$= \theta \int_\Omega \mathbf{f}(t_{m+1}) \cdot \mathbf{v}_h \; dxdy + (1-\theta) \int_\Omega \mathbf{f}(t_m) \cdot \mathbf{v}_h \; dxdy,$$
$$-\theta \int_\Omega (\nabla \cdot \mathbf{u}_h^{m+1}) q_h \; dxdy - (1-\theta) \int_\Omega (\nabla \cdot \mathbf{u}_h^m) q_h \; dxdy = 0,$$

for any $\mathbf{v}_h \in [U_h]^2$ and $q_h \in W_h$.

Another format of full discretization

- For $m = 0, \cdots, M_m - 1$, find $u_{1h}^{m+1},\ u_{2h}^{m+1} \in U_h$ and $p_h^{m+1} \in W_h$ such that

$$
\begin{aligned}
&\int_\Omega \frac{u_{1h}^{m+1} - u_{1h}^m}{\triangle t} v_{1h}\ dxdy + \int_\Omega \frac{u_{2h}^{m+1} - u_{2h}^m}{\triangle t} v_{2h}\ dxdy \\
&+\theta \int_\Omega \nu \Big(2\frac{\partial u_{1h}^{m+1}}{\partial x}\frac{\partial v_{1h}}{\partial x} + 2\frac{\partial u_{2h}^{m+1}}{\partial y}\frac{\partial v_{2h}}{\partial y} + \frac{\partial u_{1h}^{m+1}}{\partial y}\frac{\partial v_{1h}}{\partial y} \\
&+ \frac{\partial u_{1h}^{m+1}}{\partial y}\frac{\partial v_{2h}}{\partial x} + \frac{\partial u_{2h}^{m+1}}{\partial x}\frac{\partial v_{1h}}{\partial y} + \frac{\partial u_{2h}^{m+1}}{\partial x}\frac{\partial v_{2h}}{\partial x} \Big)\ dxdy \\
&+(1-\theta) \int_\Omega \nu \Big(2\frac{\partial u_{1h}^{m}}{\partial x}\frac{\partial v_{1h}}{\partial x} + 2\frac{\partial u_{2h}^{m}}{\partial y}\frac{\partial v_{2h}}{\partial y} + \frac{\partial u_{1h}^{m}}{\partial y}\frac{\partial v_{1h}}{\partial y} \\
&+ \frac{\partial u_{1h}^{m}}{\partial y}\frac{\partial v_{2h}}{\partial x} + \frac{\partial u_{2h}^{m}}{\partial x}\frac{\partial v_{1h}}{\partial y} + \frac{\partial u_{2h}^{m}}{\partial x}\frac{\partial v_{2h}}{\partial x} \Big)\ dxdy \\
&-\theta \int_\Omega \left(p_h^{m+1}\frac{\partial v_{1h}}{\partial x} + p_h^{m+1}\frac{\partial v_{2h}}{\partial y} \right)\ dxdy - (1-\theta)\int_\Omega \left(p_h^{m}\frac{\partial v_{1h}}{\partial x} + p_h^{m}\frac{\partial v_{2h}}{\partial y} \right)\ dxdy \\
&= \theta \int_\Omega (f_1(t_{m+1})v_{1h} + f_2(t_{m+1})v_{2h})\ dxdy + (1-\theta)\int_\Omega (f_1(t_m)v_{1h} + f_2(t_m)v_{2h})\ dxdy, \\
&-\theta \int_\Omega (\frac{\partial u_{1h}^{m+1}}{\partial x} q_h + \frac{\partial u_{2h}^{m+1}}{\partial y} q_h) dxdy - (1-\theta)\int_\Omega (\frac{\partial u_{1h}^{m}}{\partial x} q_h + \frac{\partial u_{2h}^{m}}{\partial y} q_h) dxdy = 0,
\end{aligned}
$$

for any $v_{1h} \in U_h$, $v_{2h} \in U_h$, and $q_h \in W_h$.

Another format of full discretization

- Since $u_{1h}^{m+1},\ u_{2h}^{m+1} \in U_h,\ p_h \in W_h,\ U_h = span\{\phi_j\}_{j=1}^{N_b}$, and $W_h = span\{\psi_j\}_{j=1}^{N_{bp}}$, then

$$u_{1h}^{m+1}(x,y) = \sum_{j=1}^{N_b} u_{1j}^{m+1}\phi_j(x,y),$$

$$u_{2h}^{m+1}(x,y) = \sum_{j=1}^{N_b} u_{2j}^{m+1}\phi_j(x,y),$$

$$p_h^{m+1}(x,y) = \sum_{j=1}^{N_{bp}} p_j^{m+1}\psi_j(x,y),$$

for some coefficients $u_{1j}^{m+1},\ u_{2j}^{m+1}$ $(j = 1, \cdots, N_b)$ and p_j^{m+1} $(j = 1, \cdots, N_{bp})$.

Another format of full discretization

- If we can set up a linear algebraic system for

$$u_{1j}^{m+1},\ u_{2j}^{m+1}\ (j = 1, \cdots, N_b) \text{ and } p_j^{m+1}\ (j = 1, \cdots, N_{bp})$$

  and solve it, then we can obtain the finite element solution u_{1h}^{m+1}, u_{2h}^{m+1}, and p_h^{m+1}.

Another format of full discretization

- For the first equation in the Galerkin formulation, we choose $\mathbf{v}_h = (\phi_i, 0)^t$ $(i = 1, \cdots, N_b)$ and $\mathbf{v}_h = (0, \phi_i)^t$ $(i = 1, \cdots, N_b)$. That is, in the first set of test functions, we choose $v_{1h} = \phi_i$ $(i = 1, \cdots, N_b)$ and $v_{2h} = 0$; in the second set of test functions, we choose $v_{1h} = 0$ and $v_{2h} = \phi_i$ $(i = 1, \cdots, N_b)$.

- For the second equation in the Galerkin formulation, we choose $q_h = \psi_i$ $(i = 1, \cdots, N_{bp})$.

Another format of full discretization

- Set $\mathbf{v}_h = (\phi_i, 0)^t$, i.e., $v_{1h} = \phi_i$ and $v_{2h} = 0$ $(i = 1, \cdots, N_b)$, in the first equation of the full discretization. Then

$$\int_\Omega \frac{\sum_{j=1}^{N_b} u_{1j}^{m+1}\phi_j - \sum_{j=1}^{N_b} u_{1j}^{m}\phi_j}{\triangle t}\phi_i \; dxdy + \theta \int_\Omega \nu \left[2\frac{\partial \left(\sum_{j=1}^{N_b} u_{1j}^{m+1}\phi_j\right)}{\partial x}\frac{\partial \phi_i}{\partial x} \right.$$
$$\left. + \frac{\partial \left(\sum_{j=1}^{N_b} u_{1j}^{m+1}\phi_j\right)}{\partial y}\frac{\partial \phi_i}{\partial y} + \frac{\partial \left(\sum_{j=1}^{N_b} u_{2j}^{m+1}\phi_j\right)}{\partial x}\frac{\partial \phi_i}{\partial y}\right] dxdy$$
$$+(1-\theta)\int_\Omega \nu \left[2\frac{\partial \left(\sum_{j=1}^{N_b} u_{1j}^{m}\phi_j\right)}{\partial x}\frac{\partial \phi_i}{\partial x} + \frac{\partial \left(\sum_{j=1}^{N_b} u_{1j}^{m}\phi_j\right)}{\partial y}\frac{\partial \phi_i}{\partial y} \right.$$
$$\left. + \frac{\partial \left(\sum_{j=1}^{N_b} u_{2j}^{m}\phi_j\right)}{\partial x}\frac{\partial \phi_i}{\partial y}\right] dxdy$$
$$-\theta \int_\Omega \left(\sum_{j=1}^{N_{bp}} p_j^{m+1}\psi_j\right)\frac{\partial \phi_i}{\partial x}\; dxdy - (1-\theta)\int_\Omega \left(\sum_{j=1}^{N_{bp}} p_j^{m}\psi_j\right)\frac{\partial \phi_i}{\partial x}\; dxdy$$
$$= \theta \int_\Omega f_1(t_{m+1})\phi_i \; dxdy + (1-\theta)\int_\Omega f_1(t_m)\phi_i \; dxdy.$$

Another format of full discretization

- Set $\mathbf{v}_h = (0, \phi_i)^t$, i.e., $v_{1h} = 0$ and $v_{2h} = \phi_i$ $(i = 1, \cdots, N_b)$, in the first equation of the full discretization. Then

$$
\begin{aligned}
&\int_\Omega \frac{\sum_{j=1}^{N_b} u_{2j}^{m+1}\phi_j - \sum_{j=1}^{N_b} u_{2j}^{m}\phi_j}{\triangle t}\phi_i \; dxdy + \theta \int_\Omega \nu \Big(2\frac{\partial \left(\sum_{j=1}^{N_b} u_{2j}^{m+1}\phi_j\right)}{\partial y}\frac{\partial \phi_i}{\partial y} \\
&+ \frac{\partial \left(\sum_{j=1}^{N_b} u_{1j}^{m+1}\phi_j\right)}{\partial y}\frac{\partial \phi_i}{\partial x} + \frac{\partial \left(\sum_{j=1}^{N_b} u_{2j}^{m+1}\phi_j\right)}{\partial x}\frac{\partial \phi_i}{\partial x}\Big) \; dxdy \\
&+(1-\theta)\int_\Omega \nu \Big(2\frac{\partial \left(\sum_{j=1}^{N_b} u_{2j}^{m}\phi_j\right)}{\partial y}\frac{\partial \phi_i}{\partial y} \\
&+ \frac{\partial \left(\sum_{j=1}^{N_b} u_{1j}^{m}\phi_j\right)}{\partial y}\frac{\partial \phi_i}{\partial x} + \frac{\partial \left(\sum_{j=1}^{N_b} u_{2j}^{m}\phi_j\right)}{\partial x}\frac{\partial \phi_i}{\partial x}\Big) \; dxdy \\
&-\theta \int_\Omega \left(\sum_{j=1}^{N_{bp}} p_j^{m+1}\psi_j\right)\frac{\partial \phi_i}{\partial y} \; dxdy - (1-\theta)\int_\Omega \left(\sum_{j=1}^{N_{bp}} p_j^{m}\psi_j\right)\frac{\partial \phi_i}{\partial y} \; dxdy \\
&= \theta \int_\Omega f_2(t_{m+1})\phi_i \; dxdy + (1-\theta)\int_\Omega f_2(t_m)\phi_i \; dxdy.
\end{aligned}
$$

Another format of full discretization

- Set $q_h = \psi_i$ $(i = 1, \cdots, N_{bp})$ in the second equation of the full discretization. Then

$$
\begin{aligned}
&-\theta \int_\Omega \left[\frac{\partial \left(\sum_{j=1}^{N_b} u_{1j}^{m+1} \phi_j \right)}{\partial x} \psi_i + \frac{\partial \left(\sum_{j=1}^{N_b} u_{2j}^{m+1} \phi_j \right)}{\partial y} \psi_i \right] dxdy \\
&-(1-\theta) \int_\Omega \left[\frac{\partial \left(\sum_{j=1}^{N_b} u_{1j}^{m} \phi_j \right)}{\partial x} \psi_i + \frac{\partial \left(\sum_{j=1}^{N_b} u_{2j}^{m} \phi_j \right)}{\partial y} \psi_i \right] dxdy \\
&= 0.
\end{aligned}
$$

Another format of full discretization

- Simplify the above three sets of equations, we obtain

$$
\begin{aligned}
&\sum_{j=1}^{N_b} u_{1j}^{m+1}\left(\frac{1}{\triangle t}\int_\Omega \phi_j\phi_i \; dxdy + 2\theta\int_\Omega \nu\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial x}\; dxdy + \theta\int_\Omega \nu\frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial y}\; dxdy\right)\\
&+\sum_{j=1}^{N_b} u_{2j}^{m+1}\left(\int_\Omega \nu\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial y}\; dxdy\right) + \sum_{j=1}^{N_{bp}} p_j^{m+1}\left(\theta\int_\Omega -\psi_j\frac{\partial\phi_i}{\partial x}\; dxdy\right)\\
&= \theta\int_\Omega f_1(t_{m+1})\phi_i\; dxdy + (1-\theta)\int_\Omega f_1(t_m)\phi_i\; dxdy\\
&+\sum_{j=1}^{N_b} u_{1j}^{m}\Big[\frac{1}{\triangle t}\int_\Omega \phi_j\phi_i\; dxdy - 2(1-\theta)\int_\Omega \nu\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial x}\; dxdy\\
&-(1-\theta)\int_\Omega \nu\frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial y}\; dxdy\Big]\\
&+\sum_{j=1}^{N_b} u_{2j}^{m}\left(-(1-\theta)\int_\Omega \nu\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial y}\; dxdy\right) + \sum_{j=1}^{N_{bp}} p_j^{m}\left(-(1-\theta)\int_\Omega -\psi_j\frac{\partial\phi_i}{\partial x}\; dxdy\right),
\end{aligned}
$$

Another format of full discretization

- and

$$
\begin{aligned}
&\sum_{j=1}^{N_b} u_{1j}^{m+1}\left(\theta\int_\Omega \nu\frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial x}\ dxdy\right)+\sum_{j=1}^{N_b} u_{2j}^{m+1}\Big[\frac{1}{\triangle t}\int_\Omega \phi_j\phi_i\ dxdy\\
&+2\theta\int_\Omega \nu\frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial y}\ dxdy+\theta\int_\Omega \nu\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial x}\ dxdy\Big]+\sum_{j=1}^{N_b} p_j^{m+1}\left(\theta\int_\Omega -\psi_j\frac{\partial\phi_i}{\partial y}\ dxdy\right)\\
&=\theta\int_\Omega f_2(t_{m+1})\phi_i\ dxdy+(1-\theta)\int_\Omega f_2(t_m)\phi_i\ dxdy\\
&+\sum_{j=1}^{N_b} u_{1j}^{m}\left(-(1-\theta)\int_\Omega \nu\frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial x}\ dxdy\right)\\
&+\sum_{j=1}^{N_b} u_{2j}^{m}\Big[\frac{1}{\triangle t}\int_\Omega \phi_j\phi_i\ dxdy-2(1-\theta)\int_\Omega \nu\frac{\partial\phi_j}{\partial y}\frac{\partial\phi_i}{\partial y}\ dxdy\\
&-(1-\theta)\int_\Omega \nu\frac{\partial\phi_j}{\partial x}\frac{\partial\phi_i}{\partial x}\ dxdy\Big]\\
&+\sum_{j=1}^{N_b} p_j^{m}\left(-(1-\theta)\int_\Omega -\psi_j\frac{\partial\phi_i}{\partial y}\ dxdy\right),
\end{aligned}
$$

Another format of full discretization

- and

$$
\begin{aligned}
& \sum_{j=1}^{N_b} u_{1j}^{m+1}\left(\theta \int_{\Omega} -\frac{\partial \phi_j}{\partial x}\psi_i \ dxdy\right) \\
& + \sum_{j=1}^{N_b} u_{2j}^{m+1}\left(\theta \int_{\Omega} -\frac{\partial \phi_j}{\partial y}\psi_i \ dxdy\right) \\
= \ & \sum_{j=1}^{N_b} u_{1j}^{m}\left(-(1-\theta) \int_{\Omega} -\frac{\partial \phi_j}{\partial x}\psi_i \ dxdy\right) \\
& + \sum_{j=1}^{N_b} u_{2j}^{m}\left(-(1-\theta) \int_{\Omega} -\frac{\partial \phi_j}{\partial y}\psi_i \ dxdy\right).
\end{aligned}
$$

Another format of full discretization

- Define

$$A_1 = \left[\int_\Omega \nu \frac{\partial \phi_j}{\partial x}\frac{\partial \phi_i}{\partial x}\ dxdy\right]_{i,j=1}^{N_b}, \quad A_2 = \left[\int_\Omega \nu \frac{\partial \phi_j}{\partial y}\frac{\partial \phi_i}{\partial y}\ dxdy\right]_{i,j=1}^{N_b},$$

$$A_3 = \left[\int_\Omega \nu \frac{\partial \phi_j}{\partial x}\frac{\partial \phi_i}{\partial y}\ dxdy\right]_{i,j=1}^{N_b}, \quad A_4 = \left[\int_\Omega \nu \frac{\partial \phi_j}{\partial y}\frac{\partial \phi_i}{\partial x}\ dxdy\right]_{i,j=1}^{N_b},$$

$$A_5 = \left[\int_\Omega -\psi_j \frac{\partial \phi_i}{\partial x}\ dxdy\right]_{i=1,j=1}^{N_b,N_{bp}}, \quad A_6 = \left[\int_\Omega -\psi_j \frac{\partial \phi_i}{\partial y}\ dxdy\right]_{i=1,j=1}^{N_b,N_{bp}},$$

$$A_7 = \left[\int_\Omega -\frac{\partial \phi_j}{\partial x}\psi_i\ dxdy\right]_{i=1,j=1}^{N_{bp},N_b}, \quad A_8 = \left[\int_\Omega -\frac{\partial \phi_j}{\partial y}\psi_i\ dxdy\right]_{i=1,j=1}^{N_{bp},N_b}.$$

- Define a zero matrix $\mathbb{O}_1 = [0]_{i=1,j=1}^{N_{bp},N_{bp}}$ whose size is $N_{bp} \times N_{bp}$. Then

$$A = \begin{pmatrix} 2A_1 + A_2 & A_3 & A_5 \\ A_4 & 2A_2 + A_1 & A_6 \\ A_7 & A_8 & \mathbb{O}_1 \end{pmatrix}$$

Another format of full discretization

- Each matrix above can be obtained by Algorithm I-3 in Chapter 3.
- It is not difficult to verify (an independent study project topic) that

$$A_4 = A_3^t, \quad A_7 = A_5^t, \quad A_8 = A_6^t.$$

- Hence the matrix A is actually symmetric:

$$A = \begin{pmatrix} 2A_1 + A_2 & A_3 & A_5 \\ A_3^t & 2A_2 + A_1 & A_6 \\ A_5^t & A_6^t & \mathbb{O}_1 \end{pmatrix}$$

Another format of full discretization

- Define the basic mass matrix

$$M_e = [m_{ij}]_{i,j=1}^{N_b} = \left[\int_\Omega \phi_j \phi_i \ dxdy\right]_{i,j=1}^{N_b}.$$

- The mass matrix M_e can be obtained by Algorithm I-3 in Chapter 3, with $r = s = p = q = 0$ and $c = 1$.
- Define zero matrices $\mathbb{O}_2 = [0]_{i=1,j=1}^{N_b,N_{bp}}$ and $\mathbb{O}_3 = [0]_{i=1,j=1}^{N_b,N_b}$. Then define the block mass matrix

$$M = \begin{pmatrix} M_e & \mathbb{O}_3 & \mathbb{O}_2 \\ \mathbb{O}_3 & M_e & \mathbb{O}_2 \\ \mathbb{O}_2^t & \mathbb{O}_2^t & \mathbb{O}_1 \end{pmatrix}$$

Another format of full discretization

- Define the load vector

$$\vec{b}(t) = \begin{pmatrix} \vec{b}_1(t) \\ \vec{b}_2(t) \\ \vec{0} \end{pmatrix}$$

where

$$\vec{b}_1(t) = \left[\int_\Omega f_1 \phi_i dxdy\right]_{i=1}^{N_b}, \quad \vec{b}_2(t) = \left[\int_\Omega f_2 \phi_i dxdy\right]_{i=1}^{N_b}.$$

Here the size of the zero vector is $N_{bp} \times 1$. That is, $\vec{0} = [0]_{i=1}^{N_{bp}}$.

- Each of $\vec{b}_1(t)$ and $\vec{b}_2(t)$ can be obtained by Algorithm II-5 in Chapter 4.
- In the matrix formulation of the full discretization, we will use $\vec{b}_1(t_{m+1})$, $\vec{b}_2(t_{m+1})$, $\vec{b}_1(t_m)$, and $\vec{b}_2(t_m)$.

Another format of full discretization

- Define the unknown vector

$$\vec{X}^{m+1} = \begin{pmatrix} \vec{X}_1^{m+1} \\ \vec{X}_2^{m+1} \\ \vec{X}_3^{m+1} \end{pmatrix}$$

where

$$\vec{X}_1^{m+1} = \left[u_{1j}^{m+1}\right]_{j=1}^{N_b}, \ \vec{X}_2^{m+1} = \left[u_{2j}^{m+1}\right]_{j=1}^{N_b}, \ \vec{X}_3^{m+1} = \left[p_j^{m+1}\right]_{j=1}^{N_{bp}}.$$

Another format of full discretization

- Then we obtain the following matrix formulation:

$$\begin{aligned}\left(\frac{M}{\triangle t}+\theta A\right)\vec{X}^{m+1} &= \theta\vec{b}(t_{m+1})+(1-\theta)\vec{b}(t_m)\\ &\quad +\frac{M}{\triangle t}\vec{X}^m-(1-\theta)A\vec{X}^m,\end{aligned}$$

  which is the same as the matrix formulation obtained in the last section.

- Hence the rest of the derivation and the pseudo code are the same as in the last section.

Outline

Target problem

- Consider

$$\mathbf{u}_{tt} - \nabla \cdot \sigma(\mathbf{u}) = \mathbf{f} \quad \text{in } \Omega \times [0, T],$$
$$\mathbf{u} = \mathbf{g} \quad \text{on } \partial\Omega \times [0, T],$$
$$\mathbf{u} = \mathbf{u}_0, \quad \frac{\partial \mathbf{u}}{\partial t} = \mathbf{u}_{00} \quad \text{at } t = 0 \text{ and in } \Omega.$$

- The stress tensor $\sigma(\mathbf{u})$ is defined as

$$\sigma(\mathbf{u}) = \begin{pmatrix} \sigma_{11}(\mathbf{u}) & \sigma_{12}(\mathbf{u}) \\ \sigma_{21}(\mathbf{u}) & \sigma_{22}(\mathbf{u}) \end{pmatrix}, \quad \sigma_{ij}(\mathbf{u}) = \lambda(\nabla \cdot \mathbf{u})\delta_{ij} + 2\mu\epsilon_{ij}(\mathbf{u}),$$

where λ and μ are Lamé parameters.

Target problem

- The strain tensor is defined as

$$\epsilon = \begin{pmatrix} \epsilon_{11} & \epsilon_{12} \\ \epsilon_{21} & \epsilon_{22} \end{pmatrix}, \qquad \epsilon_{ij}(\mathbf{u}) = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right),$$

where

$$\delta_{ij} = \begin{cases} 1, & i = j, \\ 0, & i \neq j. \end{cases}$$

- Hence the stress tensor can be written as

$$\sigma(\mathbf{u}) = \begin{pmatrix} \lambda\frac{\partial u_1}{\partial x_1} + 2\mu\frac{\partial u_1}{\partial x_1} + \lambda\frac{\partial u_2}{\partial x_2} & \mu\frac{\partial u_1}{\partial x_2} + \mu\frac{\partial u_2}{\partial x_1} \\ \mu\frac{\partial u_1}{\partial x_2} + \mu\frac{\partial u_2}{\partial x_1} & \lambda\frac{\partial u_1}{\partial x_1} + \lambda\frac{\partial u_2}{\partial x_2} + 2\mu\frac{\partial u_2}{\partial x_2} \end{pmatrix}$$

Weak formulation

- First, take the inner product with a vector function $\mathbf{v}(x_1, x_2) = (v_1,\ v_2)^t$ on both sides of the original equation:

$$
\begin{aligned}
& \mathbf{u}_{tt} - \nabla \cdot \sigma(\mathbf{u}) = \mathbf{f} \quad \text{in } \Omega \\
\Rightarrow\ & \mathbf{u}_{tt} \cdot \mathbf{v} - (\nabla \cdot \sigma(\mathbf{u})) \cdot \mathbf{v} = \mathbf{f} \cdot \mathbf{v} \quad \text{in } \Omega \\
\Rightarrow\ & \int_\Omega \mathbf{u}_{tt} \cdot \mathbf{v}\ dx_1 dx_2 - \int_\Omega (\nabla \cdot \sigma(\mathbf{u})) \cdot \mathbf{v}\ dx_1 dx_2 = \int_\Omega \mathbf{f} \cdot \mathbf{v}\ dx_1 dx_2.
\end{aligned}
$$

- $\mathbf{u}(x_1, x_2, t)$ is called a trail function and $\mathbf{v}(x_1, x_2)$ is called a test function.

Weak formulation

- Second, using integration by parts in multi-dimension:

$$\int_\Omega (\nabla \cdot \sigma(\mathbf{u})) \cdot \mathbf{v} \; dx_1 dx_2 = \int_{\partial\Omega} (\sigma(\mathbf{u})\mathbf{n}) \cdot \mathbf{v} \; ds - \int_\Omega \sigma(\mathbf{u}) : \nabla \mathbf{v} \; dx_1 dx_2,$$

where $\mathbf{n} = (n_1, \; n_2)^t$ is the unit outer normal vector of $\partial\Omega$, we obtain

$$\begin{aligned} & \int_\Omega \mathbf{u}_{tt} \cdot \mathbf{v} \; dx_1 dx_2 + \int_\Omega \sigma(\mathbf{u}) : \nabla \mathbf{v} \; dx_1 dx_2 - \int_{\partial\Omega} (\sigma(\mathbf{u})\mathbf{n}) \cdot \mathbf{v} \; ds \\ = & \int_\Omega \mathbf{f} \cdot \mathbf{v} \; dx_1 dx_2. \end{aligned}$$

Here,

$$\begin{aligned} A : B &= \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} : \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix} \\ &= a_{11} b_{11} + a_{12} b_{12} + a_{21} b_{21} + a_{22} b_{22}, \end{aligned}$$

Weak formulation

- and

$$\nabla \mathbf{v} = \begin{pmatrix} \frac{\partial v_1}{\partial x_1} & \frac{\partial v_1}{\partial x_2} \\ \frac{\partial v_2}{\partial x_1} & \frac{\partial v_2}{\partial x_2} \end{pmatrix}.$$

- Since the solution on the domain boundary $\partial\Omega$ are given by $\mathbf{u}(x_1, x_2, t) = \mathbf{g}(x_1, x_2, t)$, then we can choose the test function $\mathbf{v}(x_1, x_2)$ such that $\mathbf{v} = 0$ on $\partial\Omega$.

- Hence

$$\int_\Omega \mathbf{u}_{tt} \cdot \mathbf{v}\ dx_1 dx_2 + \int_\Omega \sigma(\mathbf{u}) : \nabla \mathbf{v}\ dx_1 dx_2 = \int_\Omega \mathbf{f} \cdot \mathbf{v}\ dx_1 dx_2.$$

- Define

$$H^2(0, T; [H^1(\Omega)]^2) = \{\mathbf{v}(\cdot, t), \frac{\partial \mathbf{v}}{\partial t}(\cdot, t), \frac{\partial^2 \mathbf{v}}{\partial t^2}(\cdot, t) \in [H^1(\Omega)]^2,\ \forall t \in [0, T]\}$$

where $[H^1(\Omega)]^2 = H^1(\Omega) \times H^1(\Omega)$.

Weak formulation

- Weak formulation for the unsteady linear elasticity equation: find $\mathbf{u} \in H^2(0, T; [H^1(\Omega)]^2)$ such that

$$\int_\Omega \mathbf{u}_{tt} \cdot \mathbf{v} \; dx_1 dx_2 + \int_\Omega \sigma(\mathbf{u}) : \nabla \mathbf{v} \; dx_1 dx_2 = \int_\Omega \mathbf{f} \cdot \mathbf{v} \; dx_1 dx_2.$$

for any $\mathbf{v} \in [H_0^1(\Omega)]^2$.

- Let $a(\mathbf{u}, \mathbf{v}) = \int_\Omega \sigma(\mathbf{u}) : \nabla \mathbf{v} \; dx_1 dx_2$ and $(\mathbf{f}, \mathbf{v}) = \int_\Omega \mathbf{f} \cdot \mathbf{v} \; dx_1 dx_2$.
- Weak formulation: find $\mathbf{u} \in H^2(0, T; [H^1(\Omega)]^2)$ such that

$$(\mathbf{u}_{tt}, v) + a(\mathbf{u}, \mathbf{v}) = (\mathbf{f}, \mathbf{v})$$

for any $\mathbf{v} \in [H_0^1(\Omega)]^2$.

Weak formulation

- In details,

$$\begin{aligned}
\sigma(\mathbf{u}) : \nabla \mathbf{v} &= \begin{pmatrix} \sigma_{11}(\mathbf{u}) & \sigma_{12}(\mathbf{u}) \\ \sigma_{21}(\mathbf{u}) & \sigma_{22}(\mathbf{u}) \end{pmatrix} : \begin{pmatrix} \frac{\partial v_1}{\partial x_1} & \frac{\partial v_1}{\partial x_2} \\ \frac{\partial v_2}{\partial x_1} & \frac{\partial v_2}{\partial x_2} \end{pmatrix} \\
&= \sigma_{11}(\mathbf{u})\frac{\partial v_1}{\partial x_1} + \sigma_{12}(\mathbf{u})\frac{\partial v_1}{\partial x_2} + \sigma_{21}(\mathbf{u})\frac{\partial v_2}{\partial x_1} + \sigma_{22}(\mathbf{u})\frac{\partial v_2}{\partial x_2} \\
&= \left(\lambda\frac{\partial u_1}{\partial x_1} + 2\mu\frac{\partial u_1}{\partial x_1} + \lambda\frac{\partial u_2}{\partial x_2}\right)\frac{\partial v_1}{\partial x_1} \\
&\quad + \left(\mu\frac{\partial u_1}{\partial x_2} + \mu\frac{\partial u_2}{\partial x_1}\right)\frac{\partial v_1}{\partial x_2} + \left(\mu\frac{\partial u_1}{\partial x_2} + \mu\frac{\partial u_2}{\partial x_1}\right)\frac{\partial v_2}{\partial x_1} \\
&\quad + \left(\lambda\frac{\partial u_1}{\partial x_1} + \lambda\frac{\partial u_2}{\partial x_2} + 2\mu\frac{\partial u_2}{\partial x_2}\right)\frac{\partial v_2}{\partial x_2}
\end{aligned}$$

Weak formulation

- Then

$$
\begin{aligned}
&\int_\Omega \sigma(\mathbf{u}) : \nabla \mathbf{v} \;dx_1 dx_2 \\
= \;& \int_\Omega \Big(\lambda \frac{\partial u_1}{\partial x_1}\frac{\partial v_1}{\partial x_1} + 2\mu \frac{\partial u_1}{\partial x_1}\frac{\partial v_1}{\partial x_1} + \lambda \frac{\partial u_2}{\partial x_2}\frac{\partial v_1}{\partial x_1} \\
& + \mu \frac{\partial u_1}{\partial x_2}\frac{\partial v_1}{\partial x_2} + \mu \frac{\partial u_2}{\partial x_1}\frac{\partial v_1}{\partial x_2} + \mu \frac{\partial u_1}{\partial x_2}\frac{\partial v_2}{\partial x_1} + \mu \frac{\partial u_2}{\partial x_1}\frac{\partial v_2}{\partial x_1} \\
& + \lambda \frac{\partial u_1}{\partial x_1}\frac{\partial v_2}{\partial x_2} + \lambda \frac{\partial u_2}{\partial x_2}\frac{\partial v_2}{\partial x_2} + 2\mu \frac{\partial u_2}{\partial x_2}\frac{\partial v_2}{\partial x_2}\Big) \;dx_1 dx_2.
\end{aligned}
$$

- Also, we have

$$
\int_\Omega \mathbf{f} \cdot \mathbf{v} \;dx_1 dx_2 = \int_\Omega (f_1 v_1 + f_2 v_2) \;dx_1 dx_2,
$$

$$
\int_\Omega \mathbf{u}_{tt} \cdot \mathbf{v} \;dx_1 dx_2 = \int_\Omega \left(\frac{\partial^2 u_1}{\partial t^2} v_1 + \frac{\partial^2 u_2}{\partial t^2} v_2\right) \;dx_1 dx_2.
$$

Weak formulation

- Weak formulation in the scalar format: find $u_1 \in H^2(0,T;H^1(\Omega))$ and $u_2 \in H^2(0,T;H^1(\Omega))$ such that

$$
\begin{aligned}
& \int_\Omega \left(\frac{\partial^2 u_1}{\partial t^2} v_1 + \frac{\partial^2 u_2}{\partial t^2} v_2 \right) \; dx_1 dx_2 \\
& + \int_\Omega \Big(\lambda \frac{\partial u_1}{\partial x_1} \frac{\partial v_1}{\partial x_1} + 2\mu \frac{\partial u_1}{\partial x_1} \frac{\partial v_1}{\partial x_1} + \lambda \frac{\partial u_2}{\partial x_2} \frac{\partial v_1}{\partial x_1} \\
& + \mu \frac{\partial u_1}{\partial x_2} \frac{\partial v_1}{\partial x_2} + \mu \frac{\partial u_2}{\partial x_1} \frac{\partial v_1}{\partial x_2} + \mu \frac{\partial u_1}{\partial x_2} \frac{\partial v_2}{\partial x_1} + \mu \frac{\partial u_2}{\partial x_1} \frac{\partial v_2}{\partial x_1} \\
& + \lambda \frac{\partial u_1}{\partial x_1} \frac{\partial v_2}{\partial x_2} + \lambda \frac{\partial u_2}{\partial x_2} \frac{\partial v_2}{\partial x_2} + 2\mu \frac{\partial u_2}{\partial x_2} \frac{\partial v_2}{\partial x_2} \Big) \; dx_1 dx_2 \\
= \; & \int_\Omega (f_1 v_1 + f_2 v_2) \; dx_1 dx_2 .
\end{aligned}
$$

for any $v_1 \in H_0^1(\Omega)$ and $v_2 \in H_0^1(\Omega)$.

Galerkin formulation

- Assume there is a finite dimensional subspace $U_h \subset H^1(\Omega)$. Define U_{h0} to be the space which consists of the functions of U_h with value 0 on the Dirichlet boundary.
- Then the Galerkin formulation is to find $\mathbf{u}_h \in H^2(0, T; [U_h]^2)$ such that

$$(\mathbf{u}_{h_{tt}}, v) + a(\mathbf{u}_h, \mathbf{v}_h) = (\mathbf{f}, \mathbf{v}_h)$$
$$\Leftrightarrow \quad \int_\Omega \sigma(\mathbf{u}_h) : \nabla \mathbf{v}_h \, dx_1 dx_2 = \int_\Omega \mathbf{f} \cdot \mathbf{v}_h \, dx_1 dx_2$$

  for any $\mathbf{v}_h \in [U_{h0}]^2$.
- Basic idea of Galerkin formulation: use finite dimensional space to approximate infinite dimensional space.
- Here $U_h = span\{\phi_j\}_{j=1}^{N_b}$ is chosen to be a finite element space where $\{\phi_j\}_{j=1}^{N_b}$ are the global finite element basis functions, such as those defined in Chapter 2.

Galerkin formulation

- For an easier implementation, we use the following Galerkin formulation (without considering the Dirichlet boundary condition, which will be handled later): find $\mathbf{u}_h \in H^2(0, T; [U_h]^2)$ such that

$$
\begin{aligned}
& (\mathbf{u}_{h_{tt}}, v) + a(\mathbf{u}_h, \mathbf{v}_h) = (\mathbf{f}, \mathbf{v}_h) \\
\Leftrightarrow \quad & \int_\Omega \sigma(\mathbf{u}_h) : \nabla \mathbf{v}_h \; dx_1 dx_2 = \int_\Omega \mathbf{f} \cdot \mathbf{v}_h \; dx_1 dx_2
\end{aligned}
$$

for any $\mathbf{v}_h \in [U_h]^2$.

Galerkin formulation

- In details, the Galerkin formulation is to find $u_{1h} \in H^2(0,T;U_h)$ and $u_{2h} \in H^2(0,T;U_h)$ such that

$$\int_\Omega \left(\frac{\partial^2 u_{1h}}{\partial t^2} v_{1h} + \frac{\partial^2 u_{2h}}{\partial t^2} v_{2h} \right) \, dx_1 dx_2$$
$$+ \int_\Omega \Big(\lambda \frac{\partial u_{1h}}{\partial x_1} \frac{\partial v_{1h}}{\partial x_1} + 2\mu \frac{\partial u_{1h}}{\partial x_1} \frac{\partial v_{1h}}{\partial x_1} + \lambda \frac{\partial u_{2h}}{\partial x_2} \frac{\partial v_{1h}}{\partial x_1}$$
$$+\mu \frac{\partial u_{1h}}{\partial x_2} \frac{\partial v_{1h}}{\partial x_2} + \mu \frac{\partial u_{2h}}{\partial x_1} \frac{\partial v_{1h}}{\partial x_2} + \mu \frac{\partial u_{1h}}{\partial x_2} \frac{\partial v_{2h}}{\partial x_1} + \mu \frac{\partial u_{2h}}{\partial x_1} \frac{\partial v_{2h}}{\partial x_1}$$
$$+\lambda \frac{\partial u_{1h}}{\partial x_1} \frac{\partial v_{2h}}{\partial x_2} + \lambda \frac{\partial u_{2h}}{\partial x_2} \frac{\partial v_{2h}}{\partial x_2} + 2\mu \frac{\partial u_{2h}}{\partial x_2} \frac{\partial v_{2h}}{\partial x_2} \Big) \, dx_1 dx_2$$
$$= \int_\Omega (f_1 v_{1h} + f_2 v_{2h}) \, dx_1 dx_2.$$

for any $v_{1h} \in U_h$ and $v_{2h} \in U_h$.

Discretization formulation

- Since $u_{1h},\ u_{2h} \in H^2(0,T;U_h)$ and $U_h = span\{\phi_j\}_{j=1}^{N_b}$, then

$$u_{1h}(x,y,t) = \sum_{j=1}^{N_b} u_{1j}(t)\phi_j, \quad u_{2h}(x,y,t) = \sum_{j=1}^{N_b} u_{2j}(t)\phi_j,$$

  for some coefficients $u_{1j}(t)$ and $u_{2j}(t)\ \ (j=1,\cdots,N_b)$.

- If we can set up a linear algebraic system for $u_{1j}(t)$ and $u_{2j}(t)\ \ (j=1,\cdots,N_b)$, then we can solve it to obtain the finite element solution $\mathbf{u}_h = (u_{1h}, u_{2h})^t$.

- We choose $\mathbf{v}_h = (\phi_i, 0)^t\ (i=1,\cdots,N_b)$ and $\mathbf{v}_h = (0,\phi_i)^t\ (i=1,\cdots,N_b)$. That is, in the first set of test functions, we choose $v_{1h} = \phi_i\ (i=1,\cdots,N_b)$ and $v_{2h}=0$; in the second set of test functions, we choose $v_{1h}=0$ and $v_{2h} = \phi_i\ (i=1,\cdots,N_b)$.

Discretization formulation

- Set $\mathbf{v}_h = (\phi_i, 0)^t$, i.e., $v_{1h} = \phi_i$ and $v_{2h} = 0$ $(i = 1, \cdots, N_b)$. Then

$$\int_\Omega \left(\sum_{j=1}^{N_b} u_{1j}(t)\phi_j\right)_{tt} \phi_i \, dxdy + \int_\Omega \lambda \left(\sum_{j=1}^{N_b} u_{1j} \frac{\partial \phi_j}{\partial x_1}\right) \frac{\partial \phi_i}{\partial x_1} dx_1 dx_2$$

$$+2\int_\Omega \mu \left(\sum_{j=1}^{N_b} u_{1j} \frac{\partial \phi_j}{\partial x_1}\right) \frac{\partial \phi_i}{\partial x_1} dx_1 dx_2 + \int_\Omega \lambda \left(\sum_{j=1}^{N_b} u_{2j} \frac{\partial \phi_j}{\partial x_2}\right) \frac{\partial \phi_i}{\partial x_1} dx_1 dx_2$$

$$+\int_\Omega \mu \left(\sum_{j=1}^{N_b} u_{1j} \frac{\partial \phi_j}{\partial x_2}\right) \frac{\partial \phi_i}{\partial x_2} dx_1 dx_2 + \int_\Omega \mu \left(\sum_{j=1}^{N_b} u_{2j} \frac{\partial \phi_j}{\partial x_1}\right) \frac{\partial \phi_i}{\partial x_2} dx_1 dx_2$$

$$= \int_\Omega f_1 \phi_i dx_1 dx_2.$$

Discretization formulation

- Set $\mathbf{v}_h = (0, \phi_i)^t$, i.e., $v_{1h} = 0$ and $v_{2h} = \phi_i$ $(i = 1, \cdots, N_b)$. Then

$$\int_\Omega \left(\sum_{j=1}^{N_b} u_{2j}(t)\phi_j \right)_{tt} \phi_i \; dxdy + \int_\Omega \mu \left(\sum_{j=1}^{N_b} u_{1j} \frac{\partial \phi_j}{\partial x_2} \right) \frac{\partial \phi_i}{\partial x_1} dx_1 dx_2$$
$$+ \int_\Omega \mu \left(\sum_{j=1}^{N_b} u_{2j} \frac{\partial \phi_j}{\partial x_1} \right) \frac{\partial \phi_i}{\partial x_1} dx_1 dx_2 + \int_\Omega \lambda \left(\sum_{j=1}^{N_b} u_{1j} \frac{\partial \phi_j}{\partial x_1} \right) \frac{\partial \phi_i}{\partial x_2} dx_1 dx_2$$
$$+ \int_\Omega \lambda \left(\sum_{j=1}^{N_b} u_{2j} \frac{\partial \phi_j}{\partial x_2} \right) \frac{\partial \phi_i}{\partial x_2} dx_1 dx_2 + 2 \int_\Omega \mu \left(\sum_{j=1}^{N_b} u_{2j} \frac{\partial \phi_j}{\partial x_2} \right) \frac{\partial \phi_i}{\partial x_2} dx_1 dx_2$$
$$= \int_\Omega f_2 \phi_i dx_1 dx_2.$$

Discretization formulation

- Simplify the above two sets of equations, we obtain

$$
\begin{aligned}
&\sum_{j=1}^{N_b} u_{1j}''(t) \int_\Omega \phi_j\phi_i \; dxdy + \sum_{j=1}^{N_b} u_{1j}\Big(\int_\Omega \lambda \frac{\partial \phi_j}{\partial x_1}\frac{\partial \phi_i}{\partial x_1} dx_1 dx_2 + 2\int_\Omega \mu \frac{\partial \phi_j}{\partial x_1}\frac{\partial \phi_i}{\partial x_1} dx_1 dx_2 \\
&+ \int_\Omega \mu \frac{\partial \phi_j}{\partial x_2}\frac{\partial \phi_i}{\partial x_2} dx_1 dx_2 \Big) + \sum_{j=1}^{N_b} u_{2j} \left(\int_\Omega \lambda \frac{\partial \phi_j}{\partial x_2}\frac{\partial \phi_i}{\partial x_1} dx_1 dx_2 + \int_\Omega \mu \frac{\partial \phi_j}{\partial x_1}\frac{\partial \phi_i}{\partial x_2} dx_1 dx_2 \right) \\
&= \int_\Omega f_1 \phi_i dx_1 dx_2 \\
&\sum_{j=1}^{N_b} u_{2j}''(t) \int_\Omega \phi_j\phi_i \; dxdy + \sum_{j=1}^{N_b} u_{1j} \left(\int_\Omega \lambda \frac{\partial \phi_j}{\partial x_1}\frac{\partial \phi_i}{\partial x_2} dx_1 dx_2 + \int_\Omega \mu \frac{\partial \phi_j}{\partial x_2}\frac{\partial \phi_i}{\partial x_1} dx_1 dx_2 \right) \\
&+ \sum_{j=1}^{N_b} u_{2j} \left(\int_\Omega \lambda \frac{\partial \phi_j}{\partial x_2}\frac{\partial \phi_i}{\partial x_2} dx_1 dx_2 + 2\int_\Omega \mu \frac{\partial \phi_j}{\partial x_2}\frac{\partial \phi_i}{\partial x_2} dx_1 dx_2 + \int_\Omega \mu \frac{\partial \phi_j}{\partial x_1}\frac{\partial \phi_i}{\partial x_1} dx_1 dx_2 \right) \\
&= \int_\Omega f_2 \phi_i dx_1 dx_2.
\end{aligned}
$$

Matrix formulation

- Define

$$A_1 = \left[\int_\Omega \lambda \frac{\partial \phi_j}{\partial x_1}\frac{\partial \phi_i}{\partial x_1} dx_1 dx_2\right]_{i,j=1}^{N_b}, \quad A_2 = \left[\int_\Omega \mu \frac{\partial \phi_j}{\partial x_1}\frac{\partial \phi_i}{\partial x_1} dx_1 dx_2\right]_{i,j=1}^{N_b},$$

$$A_3 = \left[\int_\Omega \mu \frac{\partial \phi_j}{\partial x_2}\frac{\partial \phi_i}{\partial x_2} dx_1 dx_2\right]_{i,j=1}^{N_b}, \quad A_4 = \left[\int_\Omega \lambda \frac{\partial \phi_j}{\partial x_2}\frac{\partial \phi_i}{\partial x_1} dx_1 dx_2\right]_{i,j=1}^{N_b},$$

$$A_5 = \left[\int_\Omega \mu \frac{\partial \phi_j}{\partial x_1}\frac{\partial \phi_i}{\partial x_2} dx_1 dx_2\right]_{i,j=1}^{N_b}, \quad A_6 = \left[\int_\Omega \lambda \frac{\partial \phi_j}{\partial x_1}\frac{\partial \phi_i}{\partial x_2} dx_1 dx_2\right]_{i,j=1}^{N_b},$$

$$A_7 = \left[\int_\Omega \mu \frac{\partial \phi_j}{\partial x_2}\frac{\partial \phi_i}{\partial x_1} dx_1 dx_2\right]_{i,j=1}^{N_b}, \quad A_8 = \left[\int_\Omega \lambda \frac{\partial \phi_j}{\partial x_2}\frac{\partial \phi_i}{\partial x_2} dx_1 dx_2\right]_{i,j=1}^{N_b}.$$

- Each matrix above can be obtained by Algorithm I-3 in Chapter 3.
- Then

$$A = \begin{pmatrix} A_1 + 2A_2 + A_3 & A_4 + A_5 \\ A_6 + A_7 & A_8 + 2A_3 + A_2 \end{pmatrix}$$

Matrix formulation

- Define the basic mass matrix

$$M_e = [m_{ij}]_{i,j=1}^{N_b} = \left[\int_\Omega \phi_j \phi_i \, dxdy\right]_{i,j=1}^{N_b}.$$

- The mass matrix M_e can be obtained by Algorithm I-3 in Chapter 3, with $r = s = p = q = 0$ and $c = 1$.
- Define a zero matrix $\mathbb{O}_4 = [0]_{i=1,j=1}^{N_b,N_b}$. Then define the block mass matrix

$$M = \begin{pmatrix} M_e & \mathbb{O}_4 \\ \mathbb{O}_4 & M_e \end{pmatrix}$$

Matrix formulation

- Define the load vector

$$\vec{b}(t) = \begin{pmatrix} \vec{b}_1(t) \\ \vec{b}_2(t) \end{pmatrix}$$

where

$$\vec{b}_1(t) = \left[\int_\Omega f_1 \phi_i \, dx_1 dx_2 \right]_{i=1}^{N_b}, \quad \vec{b}_2(t) = \left[\int_\Omega f_2 \phi_i \, dx_1 dx_2 \right]_{i=1}^{N_b}.$$

- Each of $\vec{b}_1(t)$ and $\vec{b}_2(t)$ can be obtained by Algorithm II-5 in Chapter 4.

Matrix formulation

- Define the unknown vector

$$\vec{X}(t) = \begin{pmatrix} \vec{X}_1(t) \\ \vec{X}_2(t) \end{pmatrix}$$

where

$$\vec{X}_1(t) = [u_{1j}(t)]_{j=1}^{N_b}, \quad \vec{X}_2(t) = [u_{2j}(t)]_{j=1}^{N_b}.$$

Matrix formulation

- We obtain the second order ODE system
$$M\vec{X}''(t) + A\vec{X}(t) = \vec{b}(t).$$
- The structure of this ODE system is the same as that of the second order ODE system obtained for the second order hyperbolic equation in Chapter 3.
- Hence the same finite difference schemes in Chapter 3 can be directly utilized for this ODE system.
- The major differences between this ODE system and the one in Chapter 3 are the details in the definition of M, A, $\vec{X}$ and $\vec{b}$, which were discussed above.

Assembly of a time-independent matrix

Recall Algorithm I-3 from Chapter 3:

- Initialize the matrix: $A = sparse(N_b, N_b)$;
- Compute the integrals and assemble them into A:

FOR $n = 1, \cdots, N$:
  FOR $\alpha = 1, \cdots, N_{lb}$:
    FOR $\beta = 1, \cdots, N_{lb}$:
      Compute $r = \int_{E_n} c \frac{\partial^{r+s}\psi_{n\alpha}}{\partial x^r \partial y^s} \frac{\partial^{p+q}\psi_{n\beta}}{\partial x^p \partial y^q} \, dxdy$;
      Add r to $A(T_b(\beta, n), T_b(\alpha, n))$.
    END
  END
END

Assembly of the time-independent stiffness matrix

- Call Algorithm I-3 with $r = 1$, $s = 0$, $p = 1$, and $q = 0$ and $c = \lambda$ to obtain A_1.
- Call Algorithm I-3 with $r = 1$, $s = 0$, $p = 1$, and $q = 0$ and $c = \mu$ to obtain A_2.
- Call Algorithm I-3 with $r = 0$, $s = 1$, $p = 0$, and $q = 1$ and $c = \mu$ to obtain A_3.
- Call Algorithm I-3 with $r = 0$, $s = 1$, $p = 1$, and $q = 0$ and $c = \lambda$ to obtain A_4.
- Call Algorithm I-3 with $r = 1$, $s = 0$, $p = 0$, and $q = 1$ and $c = \mu$ to obtain A_5.
- Call Algorithm I-3 with $r = 1$, $s = 0$, $p = 0$, and $q = 1$ and $c = \lambda$ to obtain A_6.
- Call Algorithm I-3 with $r = 0$, $s = 1$, $p = 1$, and $q = 0$ and $c = \mu$ to obtain A_7.
- Call Algorithm I-3 with $r = 0$, $s = 1$, $p = 0$, and $q = 1$ and $c = \lambda$ to obtain A_8.
- Then the stiffness matrix $A = [A_1 + 2A_2 + A_3 \quad A_4 + A_5; A_6 + A_7 \quad A_8 + 2A_3 + A_2]$.

Assembly of the mass matrix

- Call Algorithm I-3 with $r = 0$, $s = 0$, $p = 0$, $q = 0$, $c = 1$, to obtain the basic mass matrix M_e.
- Generate a zero matrix $\mathbb{O}_4$ whose size is $N_b \times N_b$.
- Then the block mass matrix $M = [M_e \quad \mathbb{O}_4 \ ; \mathbb{O}_4 \quad M_e]$.

Assembly of a time-independent vector

Recall Algorithm II-3 from Chapter 3:

- Initialize the matrix: $b = sparse(N_b, 1)$;
- Compute the integrals and assemble them into b:

$FOR\ n = 1, \cdots, N$:

$\quad FOR\ \beta = 1, \cdots, N_{lb}$:

$\quad\quad$ Compute $r = \int_{E_n} f \frac{\partial^{p+q}\psi_{n\beta}}{\partial x^p \partial y^q}\, dxdy$;

$\quad\quad b(T_b(\beta, n), 1) = b(T_b(\beta, n), 1) + r$;

$\quad END$

END

Assembly of a time-dependent vector

Recall Algorithm II-5 from Chapter 4:

- Specify a value for the time t based on the input time;
- Initialize the vector: $b = sparse(N_b, 1)$;
- Compute the integrals and assemble them into b:

$FOR\ n = 1, \cdots, N$:

$\quad FOR\ \beta = 1, \cdots, N_{lb}$:

$\quad\quad$ Compute $r = \int_{E_n} f(t) \frac{\partial^{p+q}\psi_{n\beta}}{\partial x^p \partial y^q}\, dxdy$;

$\quad\quad b(T_b(\beta, n), 1) = b(T_b(\beta, n), 1) + r$;

$\quad END$

END

Assembly of the load vector

- Call Algorithm II-5 with $p = q = 0$ and $f = f_1$ to obtain $b_1(t)$.
- Call Algorithm II-5 with $p = q = 0$ and $f = f_2$ to obtain $b_2(t)$.
- Then the load vector $\vec{b} = [b_1(t); b_2(t)]$.
- If f_1 and f_2 do not depend on t, then this part is exactly the same as the assembly of the load vector with Algorithm II-3 in Chapter 6.

Time-dependent Dirichlet boundary condition

Recall Algorithm III-4 from this chapter:

- Specify a value for the time t based on the input time;
- Deal with the Dirichlet boundary conditions:

$FOR\ k = 1, \cdots, nbn$:

    If $boundarynodes(1, k)$ shows Dirichlet condition, then

        $i = boundarynodes(2, k)$;

        $\tilde{A}(i, :) = 0$;

        $\tilde{A}(i, i) = 1$;

        $\tilde{b}(i) = g_1(P_b(:, i), t)$;

        $\tilde{A}(N_b + i, :) = 0$;

        $\tilde{A}(N_b + i, N_b + i) = 1$;

        $\tilde{b}(N_b + i) = g_2(P_b(:, i), t)$;

    $ENDIF$

END

Temporal discretization for the ODE system

- Consider the centered finite difference scheme for the system of ODEs:
$$M\vec{X}''(t) + A\vec{X}(t) = \vec{b}(t).$$
- Assume that we have a uniform partition of $[0, T]$ into M_m elements with mesh size $\triangle t$.
- The mesh nodes are $t_m = m\triangle t, \ m = 0, 1, \cdots, M_m$.
- Assume $\vec{X}^m$ is the numerical solution of $\vec{X}(t_m)$.
- Then the centered finite difference scheme is
$$\begin{aligned} & M\frac{\vec{X}^{m+1} - 2\vec{X}^m + \vec{X}^{m-1}}{\triangle t^2} + A\frac{\vec{X}^{m+1} + 2\vec{X}^m + \vec{X}^{m-1}}{4} \\ = \ & \vec{b}(t_m), \ m = 1, \cdots, M_m. \end{aligned}$$

Temporal discretization for the ODE system

- Iteration scheme 2:

$$\tilde{A}\vec{X}^{m+1} = \tilde{\vec{b}}^{m+1}, \ m = 1, \cdots, M_m,$$

where

$$\tilde{A} = \frac{M}{\triangle t^2} + \frac{A}{4},$$

$$\tilde{\vec{b}}^{m+1} = \vec{b}(t_m) + \left[\frac{2M}{\triangle t^2} - \frac{A}{2}\right]\vec{X}^m - \left[\frac{M}{\triangle t^2} + \frac{A}{4}\right]\vec{X}^{m-1}.$$

Temporal discretization for the ODE system

Algorithm B:

- Generate the mesh information matrices P and T.
- Assemble the mass matrix M by using Algorithm I-3.
- Assemble the stiffness matrix A by using Algorithm I-3.
- Generate the initial vector $\vec{X}^0$ and $\vec{X}^1$ based on the initial conditions.
- Iterate in time:

  FOR $m = 1, \cdots, M_m - 1$:

  $t_m = m\triangle t$;

  Assemble the load vectors $\vec{b}(t_m)$ by using Algorithm II-5 at $t = t_m$;

  Deal with Dirichlet boundary conditions by using Algorithm III-4 for $\tilde{A}$ and $\tilde{\vec{b}}^{m+1}$ at $t = t_{m+1}$;

  Solve iteration scheme 2 for $\vec{X}^{m+1}$.

  END

Mixed boundary conditions for unsteady linear elasticity equations

- Consider

$$\begin{aligned}
&\mathbf{u}_{tt} - \nabla\cdot\sigma(\mathbf{u}) = \mathbf{f} \quad \text{in } \Omega\times[0,\,T],\\
&\sigma(\mathbf{u})\mathbf{n} = \mathbf{p} \quad \text{on } \Gamma_S\times[0,\,T],\\
&\sigma(\mathbf{u})\mathbf{n} + r\mathbf{u} = \mathbf{q} \quad \text{on } \Gamma_R\times[0,\,T],\\
&\mathbf{u} = \mathbf{g} \quad \text{on } \Gamma_D\times[0,\,T],\\
&\mathbf{u} = \mathbf{u}_0,\quad \frac{\partial \mathbf{u}}{\partial t} = \mathbf{u}_{00},\quad \text{at } t = 0 \text{ and in } \Omega.
\end{aligned}$$

  where $\Gamma_S,\ \Gamma_R \subset \partial\Omega$ and $\Gamma_D = \partial\Omega/(\Gamma_S\cup\Gamma_R)$.
- Recall

$$\begin{aligned}
&\int_\Omega \mathbf{u}_{tt}\cdot\mathbf{v}\ dx_1dx_2 + \int_\Omega \sigma(\mathbf{u}) : \nabla\mathbf{v}\ dx_1dx_2 - \int_{\partial\Omega}(\sigma(\mathbf{u})\mathbf{n})\cdot\mathbf{v}\ ds\\
=\ &\int_\Omega \mathbf{f}\cdot\mathbf{v}\ dx_1dx_2.
\end{aligned}$$

Mixed boundary conditions for unsteady linear elasticity equations

- Since the solution on $\Gamma_D = \partial\Omega/(\Gamma_S \cup \Gamma_R)$ is given by $\mathbf{u} = \mathbf{g}$, then we can choose the test function $\mathbf{v}(x_1, x_2)$ such that $\mathbf{v} = 0$ on $\partial\Omega/(\Gamma_S \cup \Gamma_R)$.
- Hence, similar to the treatment of the mixed boundary condition in Chapter 6, the weak formulation is to find $\mathbf{u} \in H^2(0, T; [H^1(\Omega)]^2)$ such that

$$\int_\Omega \mathbf{u}_{tt} v \; dxdy + \int_\Omega \sigma(\mathbf{u}) : \nabla \mathbf{v} \; dx_1 dx_2 + \int_{\Gamma_R} r\mathbf{u} \cdot \mathbf{v} \; ds$$
$$= \int_\Omega \mathbf{f} \cdot \mathbf{v} \; dx_1 dx_2 + \int_{\Gamma_R} \mathbf{q} \cdot \mathbf{v} \; ds + \int_{\Gamma_S} \mathbf{p} \cdot \mathbf{v} \; ds$$

  for any $\mathbf{v} \in [H^1_{0D}(\Omega)]^2$ where $H^1_{0D}(\Omega) = \{w \in H^1(\Omega) : w = 0 \text{ on } \Gamma_D\}$.
- Code? Combine all of the subroutines for Dirichlet/Stress/Robin boundary conditions.

Mixed boundary conditions in normal/tangential directions for unsteady linear elasticity equations

- Consider

$$\begin{aligned}
&\mathbf{u}_{tt} - \nabla \cdot \sigma(\mathbf{u}) = \mathbf{f} \ \text{ in } \Omega \times [0, T], \\
&\mathbf{n}^t \sigma(\mathbf{u})\mathbf{n} = p_n, \ \tau^t \sigma(\mathbf{u})\mathbf{n} = p_\tau \ \text{ on } \Gamma_S \times [0, T], \\
&\mathbf{n}^t \sigma(\mathbf{u})\mathbf{n} + r\mathbf{n}^t \mathbf{u} = q_n, \ \tau^t \sigma(\mathbf{u})\mathbf{n} + r\tau^t \mathbf{u} = q_\tau \ \text{ on } \Gamma_R \times [0, T], \\
&\mathbf{u} = \mathbf{g} \ \text{ on } \Gamma_D \times [0, T], \\
&\mathbf{u} = \mathbf{u}_0, \ \frac{\partial \mathbf{u}}{\partial t} = \mathbf{u}_{00}, \ \text{ at } t = 0 \text{ and in } \Omega.
\end{aligned}$$

where $\Gamma_S, \ \Gamma_R \subset \partial\Omega$, $\Gamma_D = \partial\Omega/(\Gamma_S \cup \Gamma_R)$, $\mathbf{n} = (n_1, \ n_2)^t$ is the unit outer normal vector of $\partial\Omega$, and $\tau = (\tau_1, \tau_2)^t$ is the corresponding unit tangential vector of $\partial\Omega$.

Mixed boundary conditions in normal/tangential directions for unsteady linear elasticity equations

- Recall

$$\int_\Omega \mathbf{u}_{tt} \cdot \mathbf{v} \; dx_1 dx_2 + \int_\Omega \sigma(\mathbf{u}) : \nabla \mathbf{v} \; dx_1 dx_2 - \int_{\partial\Omega} (\sigma(\mathbf{u})\mathbf{n}) \cdot \mathbf{v} \; ds$$
$$= \int_\Omega \mathbf{f} \cdot \mathbf{v} \; dx_1 dx_2.$$

- Since the solution on $\Gamma_D = \partial\Omega/(\Gamma_S \cup \Gamma_R)$ is given by $\mathbf{u} = \mathbf{g}$, then we can choose the test function $\mathbf{v}(x_1, x_2)$ such that $\mathbf{v} = 0$ on $\partial\Omega/(\Gamma_S \cup \Gamma_R)$.

Dirichlet/stress/Robin mixed boundary condition in normal/tangential directions

- Similar to the derivation of mixed boundary conditions in normal/tangential directions in Chapter 6, we obtain

$$
\begin{aligned}
& \int_{\partial\Omega} (\sigma(\mathbf{u})\mathbf{n}) \cdot \mathbf{v} \; ds \\
= \; & \int_{\Gamma_S} (\sigma(\mathbf{u})\mathbf{n}) \cdot \mathbf{v} \; ds + \int_{\Gamma_R} (\sigma(\mathbf{u})\mathbf{n}) \cdot \mathbf{v} \; ds + \int_{\partial\Omega/(\Gamma_S \cup \Gamma_R)} (\sigma(\mathbf{u})\mathbf{n}) \cdot \mathbf{v} \; ds \\
= \; & \left[\int_{\Gamma_S} p_n(\mathbf{n}^t\mathbf{v}) \; ds + \int_{\Gamma_S} p_\tau(\tau^t\mathbf{v}) \; ds \right] \\
& + \left[\int_{\Gamma_R} q_n(\mathbf{n}^t\mathbf{v}) \; ds + \int_{\Gamma_R} q_\tau(\tau^t\mathbf{v}) \; ds \right] \\
& - \left[\int_{\Gamma_R} (r\mathbf{n}^t\mathbf{u})(\mathbf{n}^t\mathbf{v}) \; ds + \int_{\Gamma_R} (r\tau^t\mathbf{u})(\tau^t\mathbf{v}) \; ds \right],
\end{aligned}
$$

Mixed boundary conditions in normal/tangential directions for unsteady linear elasticity equations

- Hence, similar to the treatment of the mixed boundary conditions in normal/tangential directions in Chapter 6, the weak formulation is to find $\mathbf{u} \in H^2(0, T; [H^1(\Omega)]^2)$ such that

$$
\begin{aligned}
& \int_\Omega \mathbf{u}_{tt} \cdot \mathbf{v} \; dx_1 dx_2 + \int_\Omega \sigma(\mathbf{u}) : \nabla \mathbf{v} \; dx_1 dx_2 \\
& + \int_{\Gamma_R} (r \mathbf{n}^t \mathbf{u})(\mathbf{n}^t \mathbf{v}) \; ds + \int_{\Gamma_R} (r \tau^t \mathbf{u})(\tau^t \mathbf{v}) \; ds \\
= \quad & \int_\Omega \mathbf{f} \cdot \mathbf{v} \; dx_1 dx_2 + \int_{\Gamma_R} q_n (\mathbf{n}^t \mathbf{v}) \; ds + \int_{\Gamma_R} q_\tau (\tau^t \mathbf{v}) \; ds \\
& + \int_{\Gamma_S} p_n (\mathbf{n}^t \mathbf{v}) \; ds + \int_{\Gamma_S} p_\tau (\tau^t \mathbf{v}) \; ds.
\end{aligned}
$$

  for any $\mathbf{v} \in [H^1_{0D}(\Omega)]^2$.

- Code? Combine all of the subroutines for Dirichlet/Stress/Robin boundary conditions.