# Progress Report for simulating the collision of UAVs with commercial airplanes

Xuan Gu & Shakil Rafi

University of Arkansas

May 7, 2022

# Table of Contents

# What to Solve

We seek to solve dynamic equilibrium equations. The formulation from Liu and Quek assumes among other things:

1. *Linear Elastic* Deformation grows proportionally with external force.
2. *Isotropic* Material property is direction independent.

We distinguish between four kinds of objects: beams, trusses, plates and shells.

# What to Solve

TypesOfObjects.png

Figure: The four kinds of objects. Liu & Quek, 2013

# What to Solve

Taking a queue from Liu and Quek, we start off by defining DEE for an idealized infinitesimal, linearly elastic, isotropic material.

Cube.png

Figure: Forces on an idealized cube. Liu & Quek, 2013

Observe that because of equilibrium $\sigma_{xy} = \sigma_{yx}; \sigma_{xz} = \sigma_{zx}; \sigma_{yz} = \sigma_{yz}$
Which give sus the stress tensors: $\sigma^{T} = \{\sigma_{xx}, \sigma_{yy}, \sigma_{zz}, \sigma_{yz}, \sigma xz, \sigma xy\}$

# What to Solve

We also get six strain components $\epsilon^T = \{\epsilon_{xx}, \epsilon_{yy}, \epsilon zz, \gamma_{yz}, \gamma_{xz}, \gamma_{xy}\}$
Where:

$$\epsilon_{xx} = \frac{\partial u}{\partial x}; \quad \epsilon_{yy} = \frac{\partial v}{\partial y}; \quad \epsilon_{zz} = \frac{\partial w}{\partial z};$$

$$\gamma_{xy} = 2\epsilon_{xy} = \frac{\partial u}{\partial_y} + \frac{\partial v}{\partial x};$$

$$\gamma_{xy} = 2\epsilon_{xz} = \frac{\partial u}{\partial_z} + \frac{\partial w}{\partial x};$$

$$\gamma_{yz} = 2\epsilon_{yz} = \frac{\partial v}{\partial_z} + \frac{\partial w}{\partial y};$$

# What to Solve

Provided $\mathbf{U} = \begin{pmatrix} u \\ v \\ w \end{pmatrix}$ the displacement matrix we get the system $\epsilon = \mathbf{LU}$

Where the differential operator is: $\mathbf{L} = \begin{pmatrix} \frac{\partial}{\partial x} & 0 & 0 \\ 0 & \frac{\partial}{\partial y} & 0 \\ 0 & 0 & \frac{\partial}{\partial z} \\ 0 & \frac{\partial}{\partial z} & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} & 0 & \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} & 0 \end{pmatrix}$

# What to solve

For an external force in the $y$ direction the equilibrium forces are:

$$\begin{aligned}(\sigma_{xx} + d\sigma_{xx})dydz - \sigma_{xx}dydz + (\sigma_{yx} + d\sigma_{yx})dxdz - \sigma_{yx}dxdz \\ + (\sigma_{zx} + d\sigma_{zx})dxdy - \sigma_{zx}dxdy + f_x dxdydz = \rho\ddot{u}dxdydz\end{aligned} \tag{1}$$

But since:

$$d\sigma_{xx} = \frac{\partial\sigma_{xx}}{\partial x}dx, d\sigma_{yx} = \frac{\partial\sigma_{yx}}{\partial y}dy, d\sigma_{zx} = \frac{\partial\sigma_{zx}}{\partial z}dz \tag{2}$$

Substituting (2) into (1) gives us:

$$\frac{\partial\sigma_{xx}}{\partial x} + \frac{\partial\sigma_{yx}}{\partial y} + \frac{\partial\sigma_{zx}}{\partial z} + f_x = \rho\ddot{u} \tag{3}$$

# What to solve

Using analogous equations for different directions,$y$ and $z$, we get:

$$\frac{\partial \sigma_{xy}}{\partial x} + \frac{\partial \sigma_{yy}}{\partial y} + \frac{\partial \sigma_{zy}}{\partial z} + f_y = \rho \ddot{v}$$
$$\frac{\partial \sigma_{xz}}{\partial x} + \frac{\partial \sigma_{yz}}{\partial y} + \frac{\partial \sigma_{zz}}{\partial z} + f_z = \rho \ddot{w}$$

Yielding the linear system:

$$L^T \sigma + f_b = \rho \ddot{U} \tag{4}$$

And since Hooke's Law states $\sigma = c\epsilon$ we can further simplify to:

$$L^T cLU + f_b = \rho \ddot{U}$$

# How to Solve It

General structure of a deal.ii step.

1. include deal.ii headers
2. Create `class StepN` with public methods `StepN()`, `run()` and private methods `make_grid()`,`setup_system()`,`assemble_system()`,`solve()`, `output_results()`, and attributes often `Triangulation<dim>`, `FE_Q <dim>`, `DoFHandler<dim>`
3. Define the methods outside, sometimes all encaspulated in a template for dimension independent programming.

# How to Solve It

`vscode.png`

Figure: VSCode of Step-3 with comments stripped, code folding, minor refactoring

# How to solve it

□grid-1.jpg

Figure: Grid refinement on a 2-simplex

Results of solving Laplace Equation on Different Domains:

PlateWithHole.png

Figure: Plate with one hole

# How to Solve It

Cheese.png

Figure: Cheese domain

# How to Solve It

More exotic domains are shown below:

MFDomain.png MFDomainLE.png

# How to solve it

Laplace3S.png

Figure: The Laplace problem solved on a tetrahedral domain

# How to solve it

CWH.png

Figure: Same but on a channel with holes

# Step-8

Step-8 presents the elastic equations, an extension of Laplace equations with vector-valued solutions:

$$-\mathrm{div}(C\nabla u) = f \tag{5}$$

# Step-8

The $x$-displacement on the teardrop domain is:

xdisp.png

Figure: x-displacement on the solution to the elastic equations on the teardrop domain

# Future Timeline

Our next step is to master step-44. Step-44 deals exclusively with a certain variation of the dynamic structure equations for a plate.

Mesh data for airplanes from UIUC Airfoil data is not in the format that deal.ii can understand. In particular converting from the Lednicer format in a text file proves challenging and prohibitively time consuming.

# Future Timeline

We hope to master Step-44 by the end of spring break by which time we will have usable mesh data. Our aim is to solve the dynamic equilibrium equations on the domains provided by the mesh data.

# Bibliography

1. Liu G. R. & S. S. Quek, *The Finite Element Method: A Practical Course*, 2nd Edition. Butterworth-Heinemann, 2013.
2. Bathe, K. J. *Finite Element Analysis of Solids and Fluids*. Online Lectures. MIT OpenCourseware. https://ocw.mit.edu/courses/mechanical-engineering/2-092-finite-element-analysis-of-solids-and-fluids-i-fall-2009/lecture-notes/