

# Project Report on collision modeling of UAVs on commercial airplanes.

Shakil Rafi

March 2022

## 1 Introduction

This is a collaborative project between Xuan Gu and the author. We seek to simulate the collision of drones (also called unmanned aerial vehicles, or UAVs) using finite element methods. Our tool of choice will be deal.ii, a C++ library developed by Wolfgang Bangerth et.al at the University of Hiedelberg.

## 2 What to solve

We seek to solve the dynamic equilibrium equation. We take a queue from Liu and Quek and start by defining the equation in its full generality and then restrict ourself. For an infinitesimal cube of plastic anisotropic material is:

$$\frac{\partial \sigma_{xy}}{\partial x} + \frac{\partial \sigma_{xy}}{\partial y} + \frac{\partial \sigma_{xy}}{\partial z} + f_x = \rho \ddot{u}$$

using the standar engineering notation for  $\sigma$ . In general while these are the most general form of the dynamic equilibrium equations we will tend to reduce them by one dimension in the case of plates. But before that we need to clarify that there are two major categories of plates: *Kirchoff plate* theory for thin plates and *Reissner-Mindlin* theory for thick plates. The author and the main author of the project Gu is unsure what theory to use as the equations are slightly different. The author proceeds with Kirchoff Plate Theory (KPT) as follows:

$$D \left( \frac{\partial^4 w}{\partial x^4} + 2 \frac{\partial^4 w}{\partial x^2 \partial y^2} + \frac{\partial^4 w}{\partial y^4} \right) + \rho h \ddot{w} = f_z$$

Where  $D = \frac{Eh^3}{12(1-\nu^2)}$ , where  $E$  is the Young modulus,  $\nu$  is the Poisson's ratio. Recall that similar to trusses, beams, and the idealized cube of anisotropic material the static version of this is acheived by dropping the dynamic term resulting in:

$$D \left( \frac{\partial^4 w}{\partial x^4} + 2 \frac{\partial^4 w}{\partial x^2 \partial y^2} + \frac{\partial^4 w}{\partial y^4} \right) = f_z$$

The author points this out with the realizing that a possible stepping stone to dynamic equilibrium equations could be to first implement the static version of these equations.

### 3 How to solve

Bangerth provides video tutorials on the Colorado State University website in collaboration with KUAW. A total of forty-three lectures is given, while the deal.ii website takes the unusual approach of dividing lessons into 79 steps with only the first several have videos associated with them.

The author has learned steps-1-6 of these videos, and made several modifications to the existing codebase. The author was able to change the step-1 to resemble a standard 2-simplex in 2D, and the standard "Cheese" domain.

This was done by first realizing that all the code for the steps followed a certain pattern. The code starts by define a class for that step `class StepN` with public methods `StepN()`, `run()`, and private methods `setup system()`, `assemble system()`, `solve()`, `output results()`, and with attributes that are often `Triangulation` representing a specific triangulation pattern, `FE Q` handling the finite element matrices and `DoFHandler` for the degrees of freedom at each node.

Whereas the Laplace solver used in steps 3 is only implemented on the square domain, the author has managed to make it work on domains such as the "Plate with Hole" and the "Cheese" domain, and whereas steps-5-6 was done on an octagonal domain, the author changes the domain to different more exotic domains and observes smoothing for iterations  $\geq 3$ .

While all these are rudimentary for dealii, step-5 presents an interesting challenge in that for the first time, a mesh is actually input into deal.ii using the `circle-grid.inp` file, this presents us with a method of actually interpreting mesh data from the outside. While author has developed some amount of skill in editing `inp` files by hand, and while the author realizes that mesh data for most commercial airplanes are available at the UIUC Aerofoil Database in Lednicer format, converting the text file to a suitable `INP` is laborious enough to be prohibitive. While there are promising software out there like Blender and gmsh that *may* be able to help, a significant bottleneck exists in this are, that the author hopes to solve in the future.

The author has also learned to vary vector and initial starting points for Step-8, the elastic equation, a necessary stepping stone to solving systems of equations, and a step that Step-44 explicitly references

## 4 Roadmap

The author seeks first to understand Step-8 where the question of input is handled. Having mastered Step-8 the author seeks to go to Step-44. Step-44 represents an interesting case study in how to use deal.ii to tackle problems involving linear systems of equations. Then the author seeks to take steps to model DEE on beams following Liu and Quek and the followed by plates.