# homework3-report-viz

November 17, 2020

author: Sylvain Laporte
matricule: C3874
date: 2020-11-17
course: IFT6113 - Geometric modeling and shape analysis

## 1 Homework 3 - Report

### 1.1 Problem 0 - Anchor vertices

We provide two different tools for selecting anchor vertices:

- `select_anchors.py`: a command line tool for selecting vertices one by one
- `select_anchors_by_range_visual.ipynb`: a Jupyter notebook for selecting groups of vertices based on bounds

Each tool stores the requested vertex informations – indices and new positions – in a JSON file in the `output/` directory as `{mesh-name}-anchors.json`.

#### 1.1.1 `select_anchors.py`

The `select_anchors.py` command line tool is available in the main directory.

This command contains a single option:

- `--input`, `-i`: to specify a mesh file, eg.: input/armadillo_1k.off (default: input/bar2.off)

#### 1.1.2 `select_anchors_by_range_visual.ipynb`

The `select_anchors_by_range_visual.ipynb` tool is a Jupyter notebook used to select groups of vertices based on bounds – greater than, less than – with some visual aid.

Instructions for this tool are included inside the notebook.

## 1.2 Problem 1 - Biharmonic deformations

### 1.2.1 1.

The derivations are available in `problem1_derivations.pdf`.

To implement this solution, we used Cholesky decomposition solver on the obtained derivation. We removed the rows associated to the user provided anchors from the linear system.

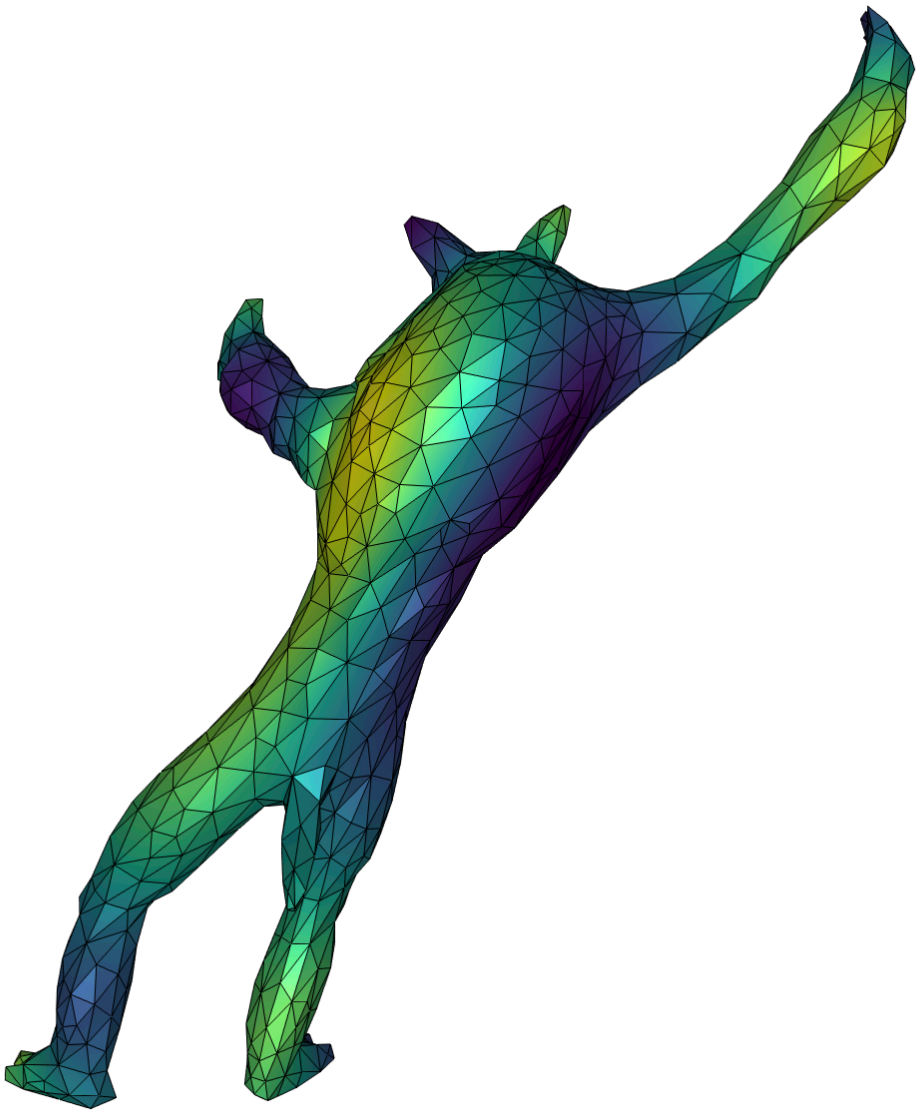
$$\boldsymbol{K}_{u \times u}\boldsymbol{d} = \boldsymbol{K}_{u \times k}\hat{\boldsymbol{d}}_{user}$$

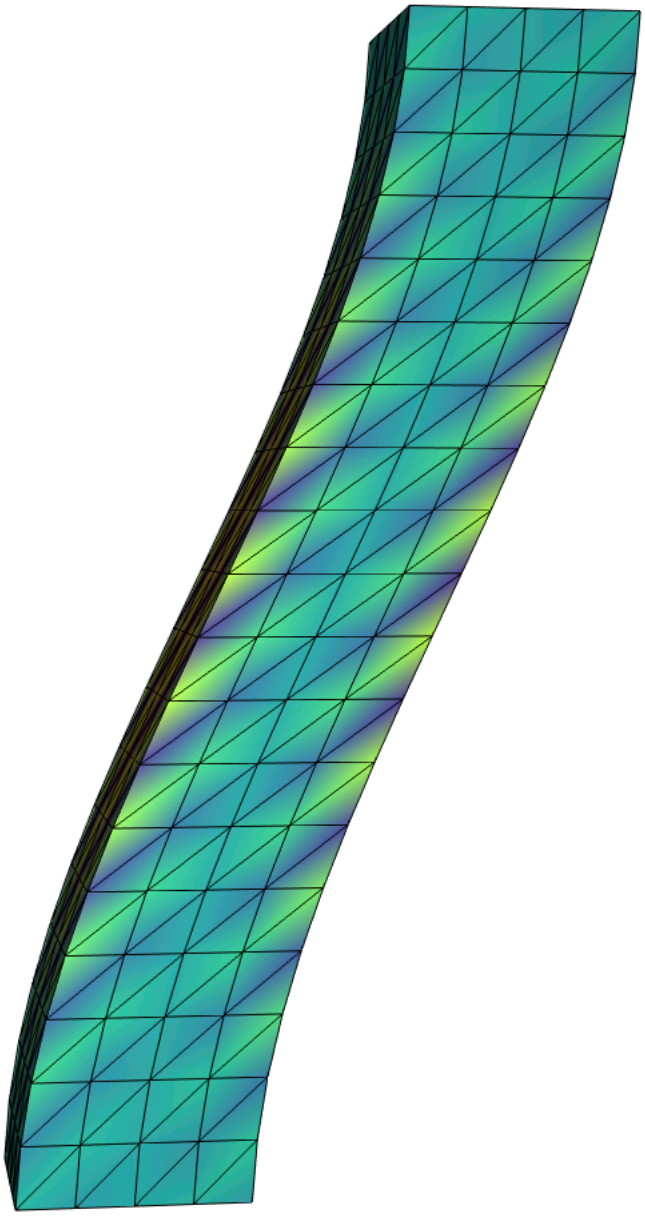
where $u$ is used for "unknown" displacements and $k$ for "known" displacements.

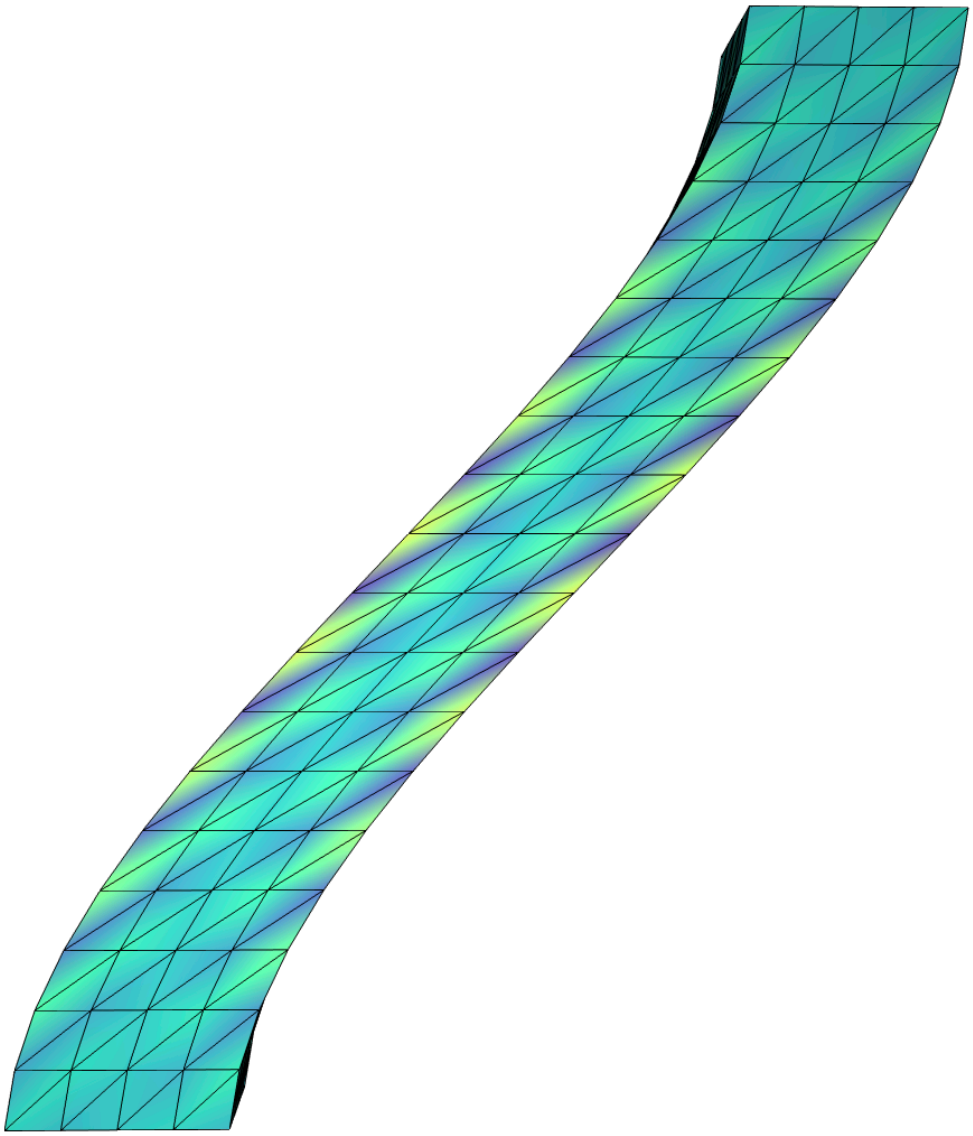
## 1.3 2.

Results obtained with our implementation of biharmonic deformation for armadillo and bar1.

A demonstration of victory…





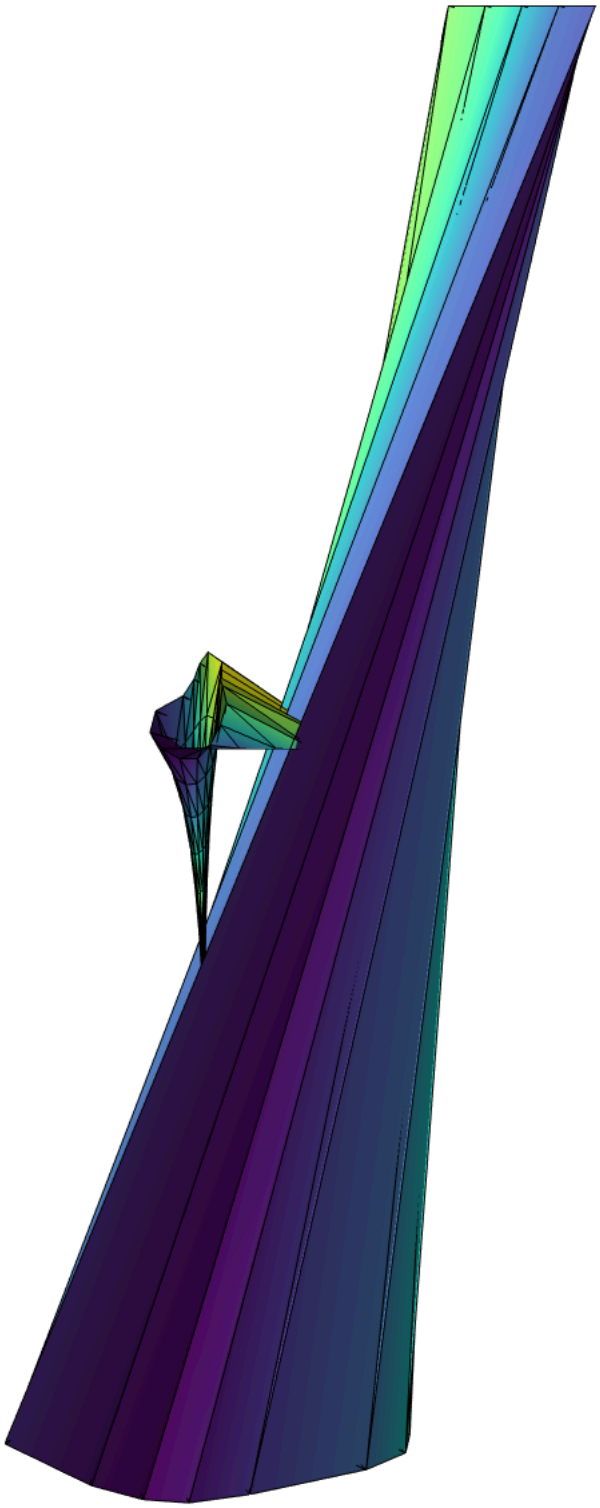


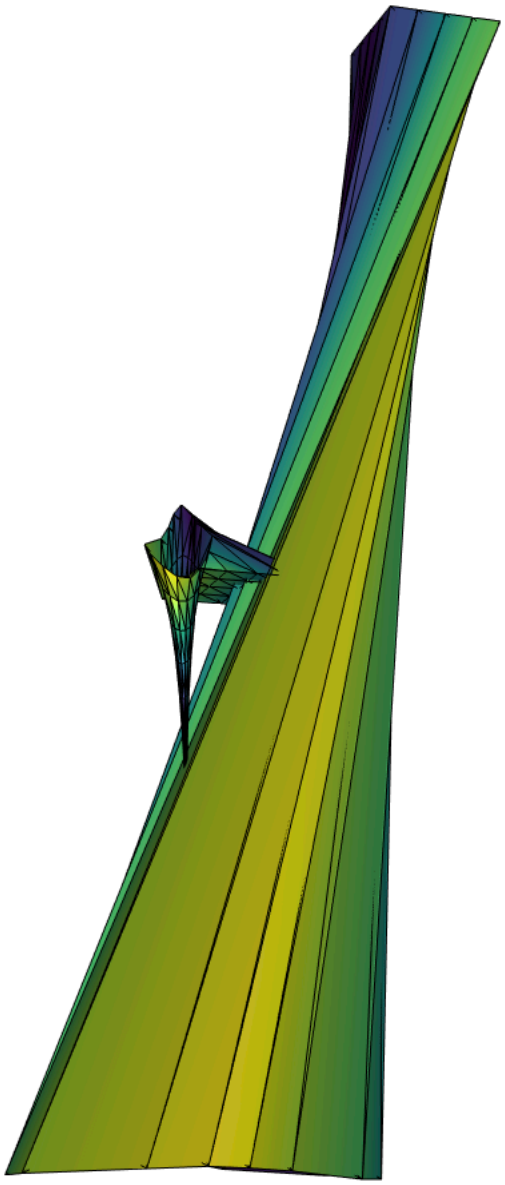
### 1.4 Problem 2 - As-Rigid-As-Possible deformation

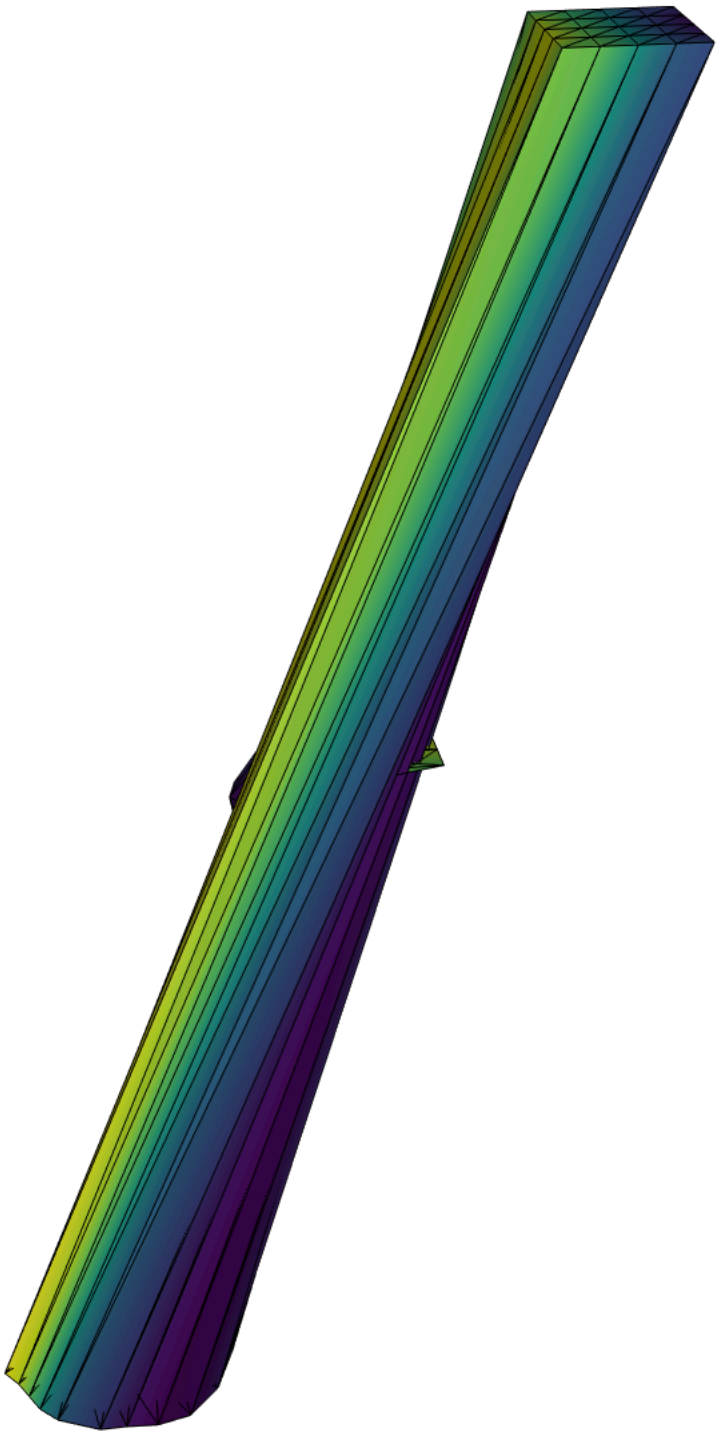
At this time, our implementation of ARAP deformation is still buggy and does not produce the expected result. We suspect some errors occur while computing the initial guess and/or the rotations.

For computing the initial guess, we tried a naive Laplacian editing method, then our working biharmonic deformation as a test. The following are the results as tested on `bar2.off`.

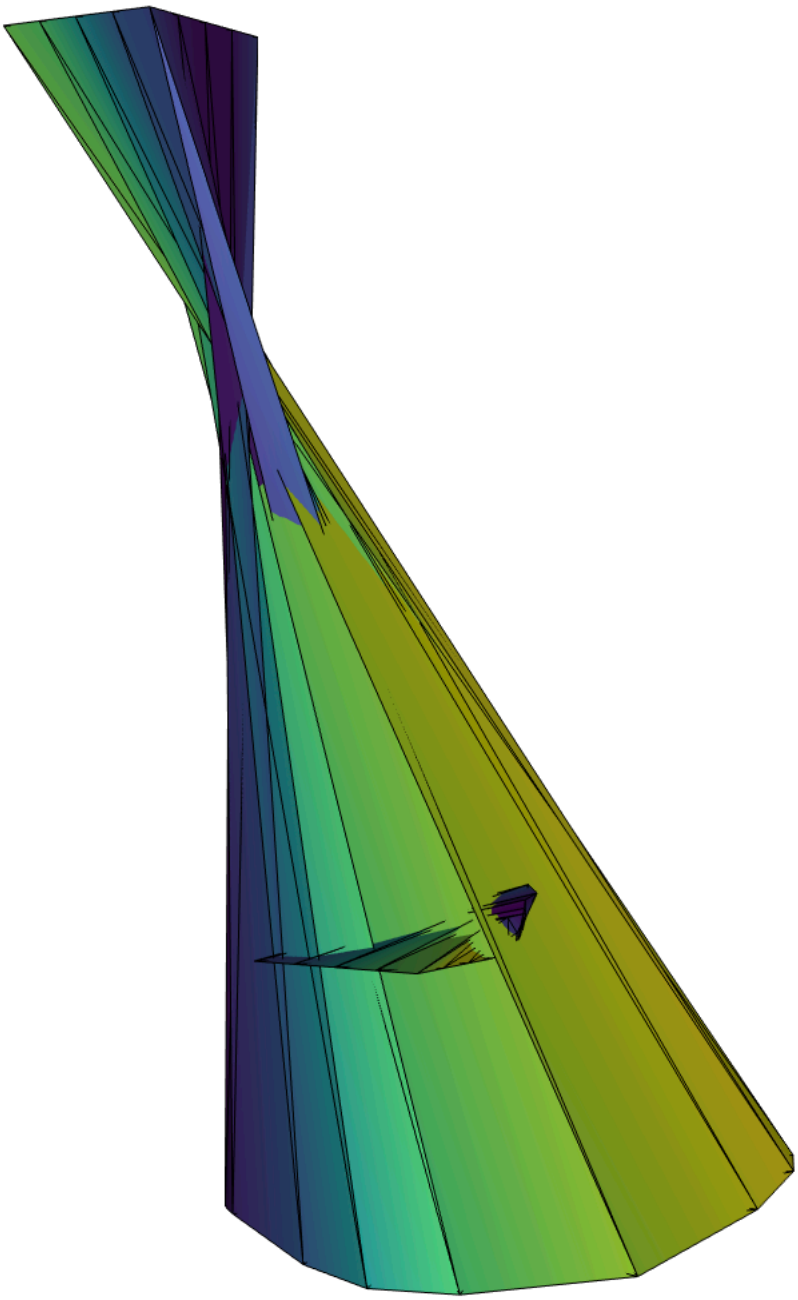
### 1.4.1 Naive Laplacian editing guess

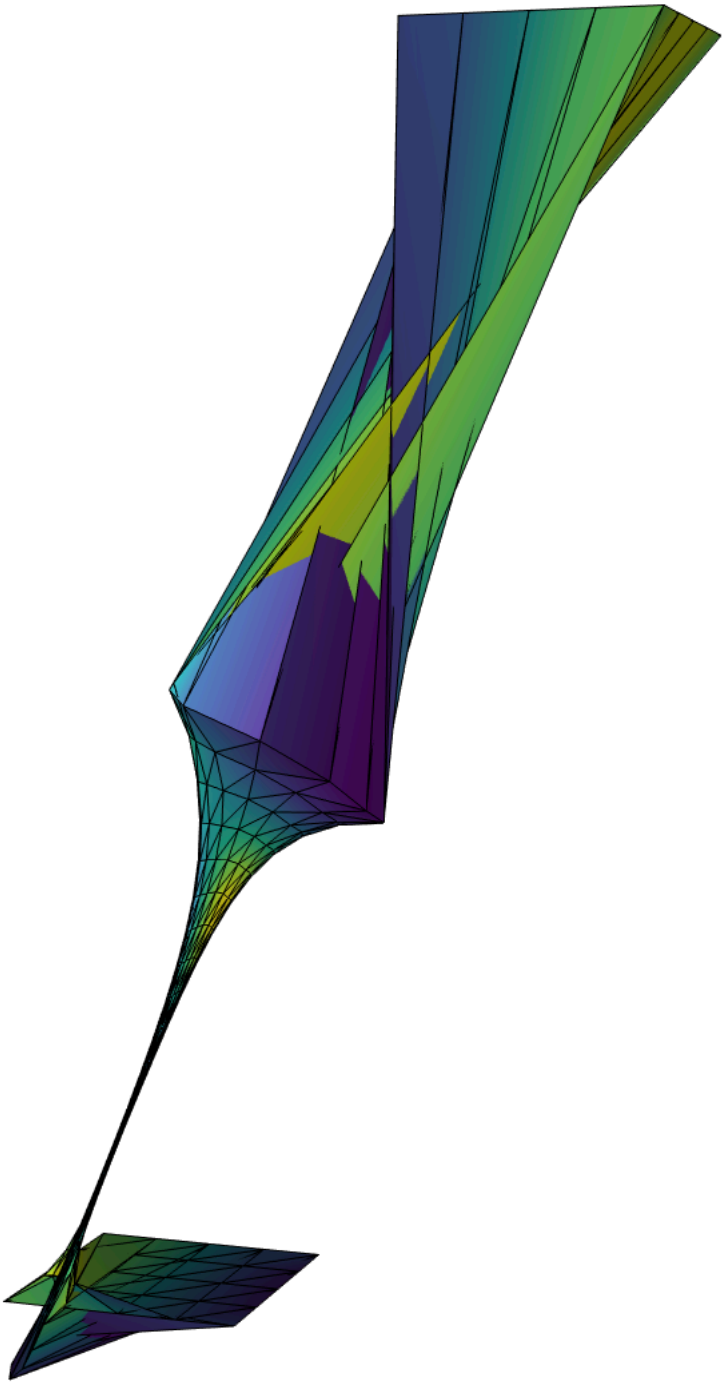






### 1.4.2 Biharmonic guess



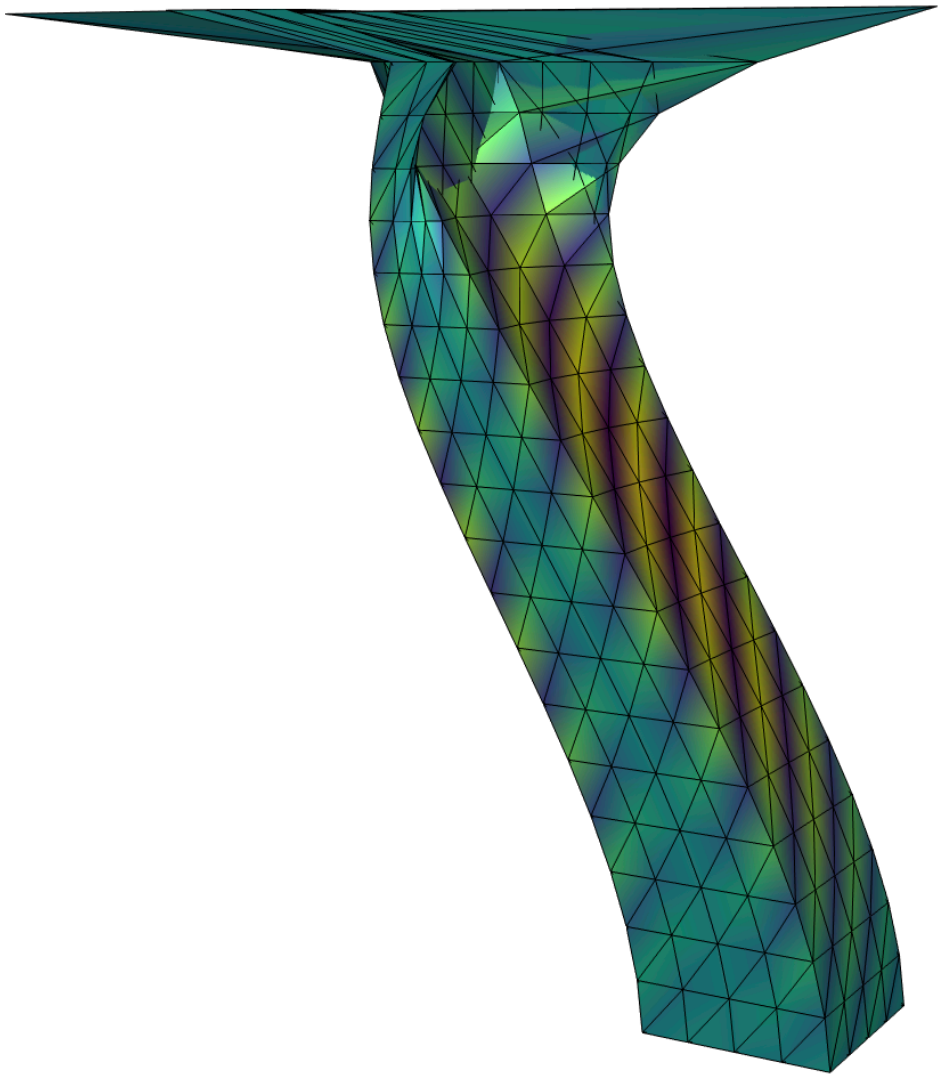


## 1.5 Bonus - Bloopers

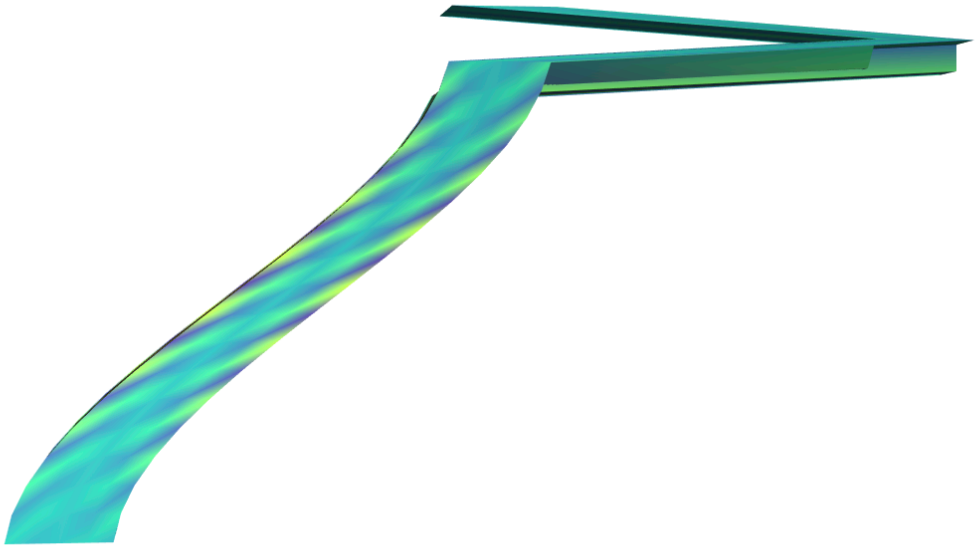
The following images are some “interesting” results collected while debugging the algorithms.

### 1.5.1 Biharmonic

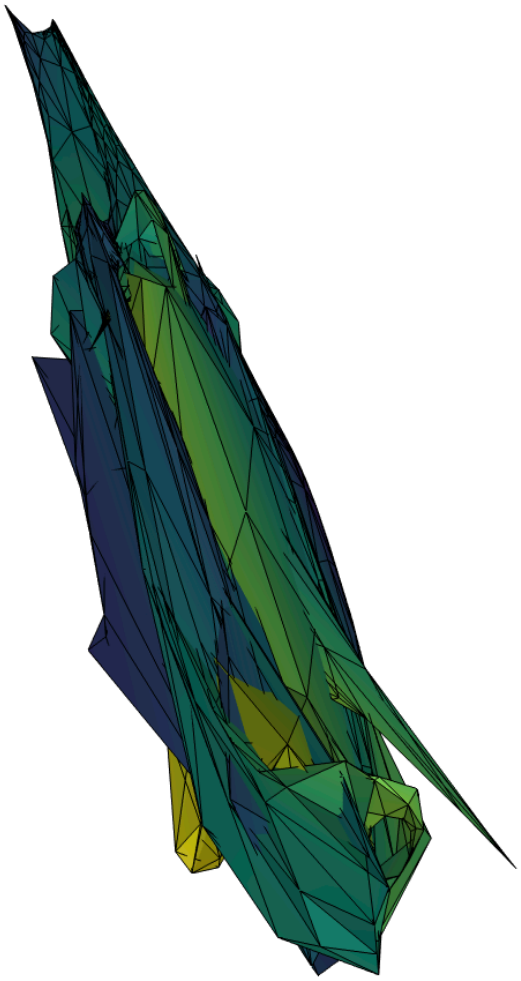
Didn’t take note on this one, one of these: didn’t divide by -1, mixed variables, or didn’t remove a debug multiplier. Kind of sexy though.



Some misselected vertices...

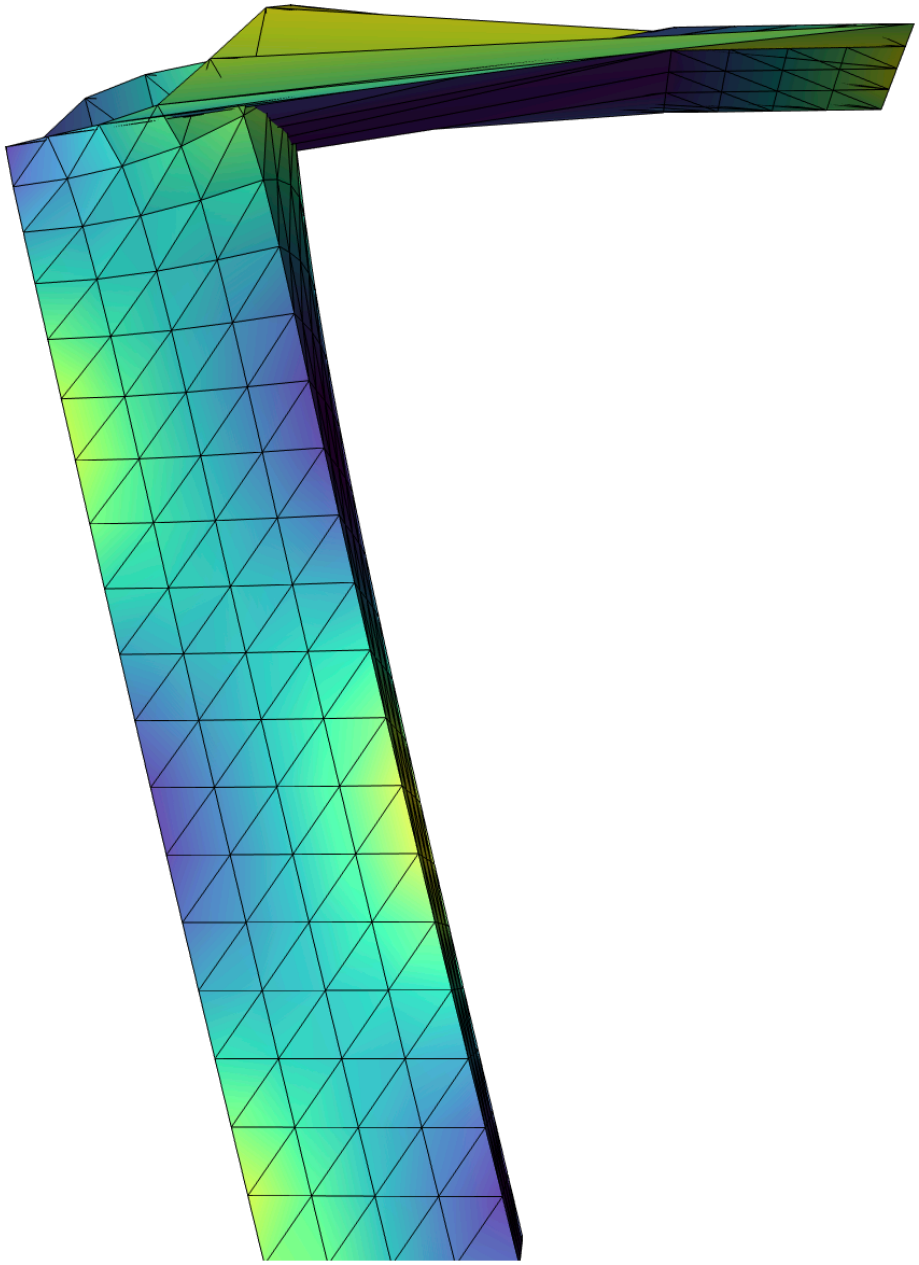


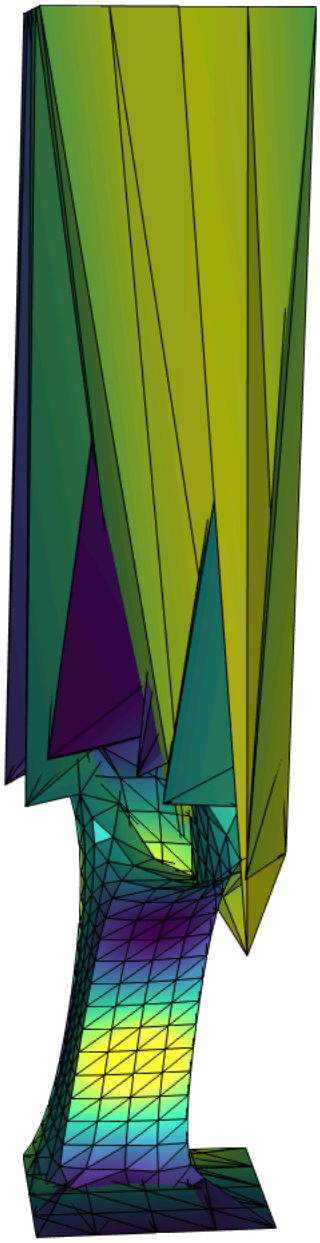
Eager to test on other meshes, I forgot to select new anchors points. Took me two attempts to realize it. This must be how it feels to pass through a black hole; the first is a bunny, the second is an armadillo.

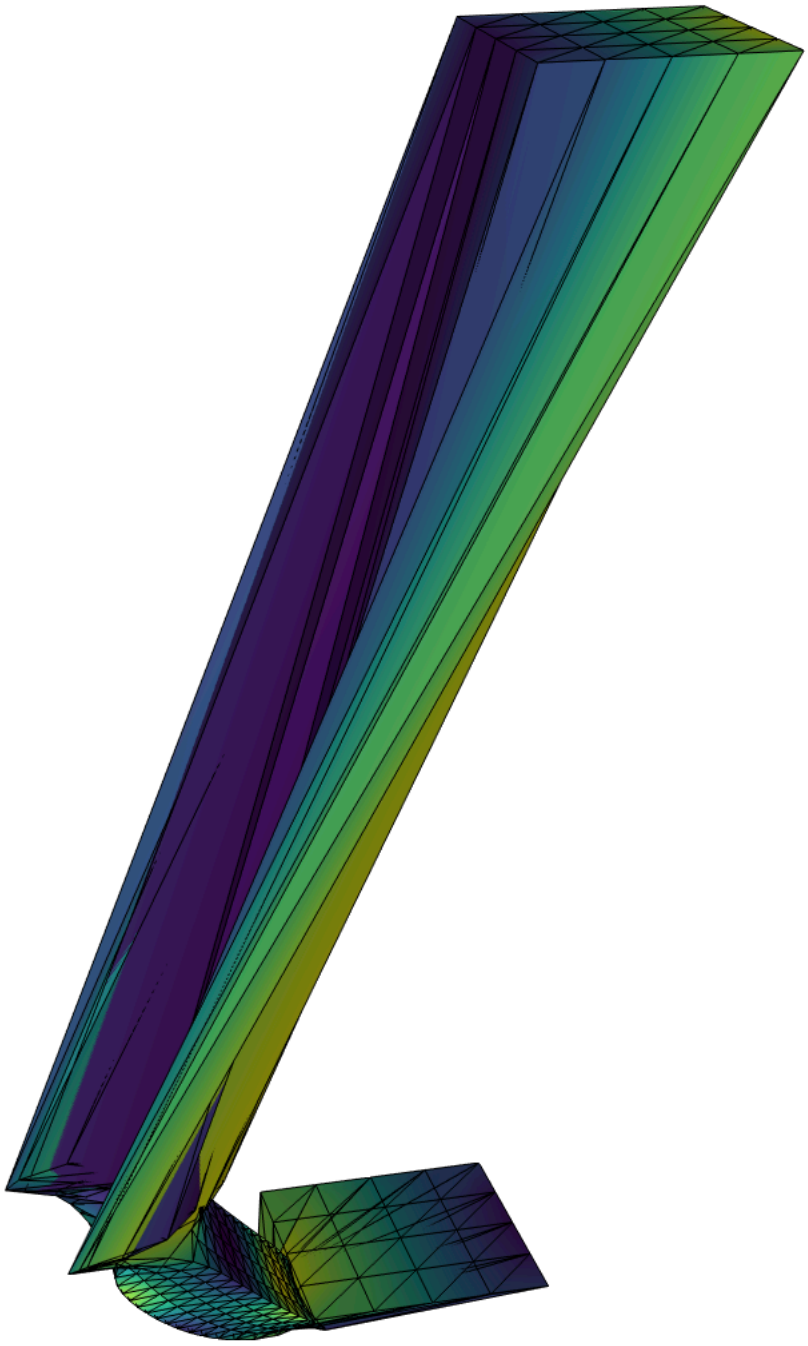


### 1.5.2 ARAP

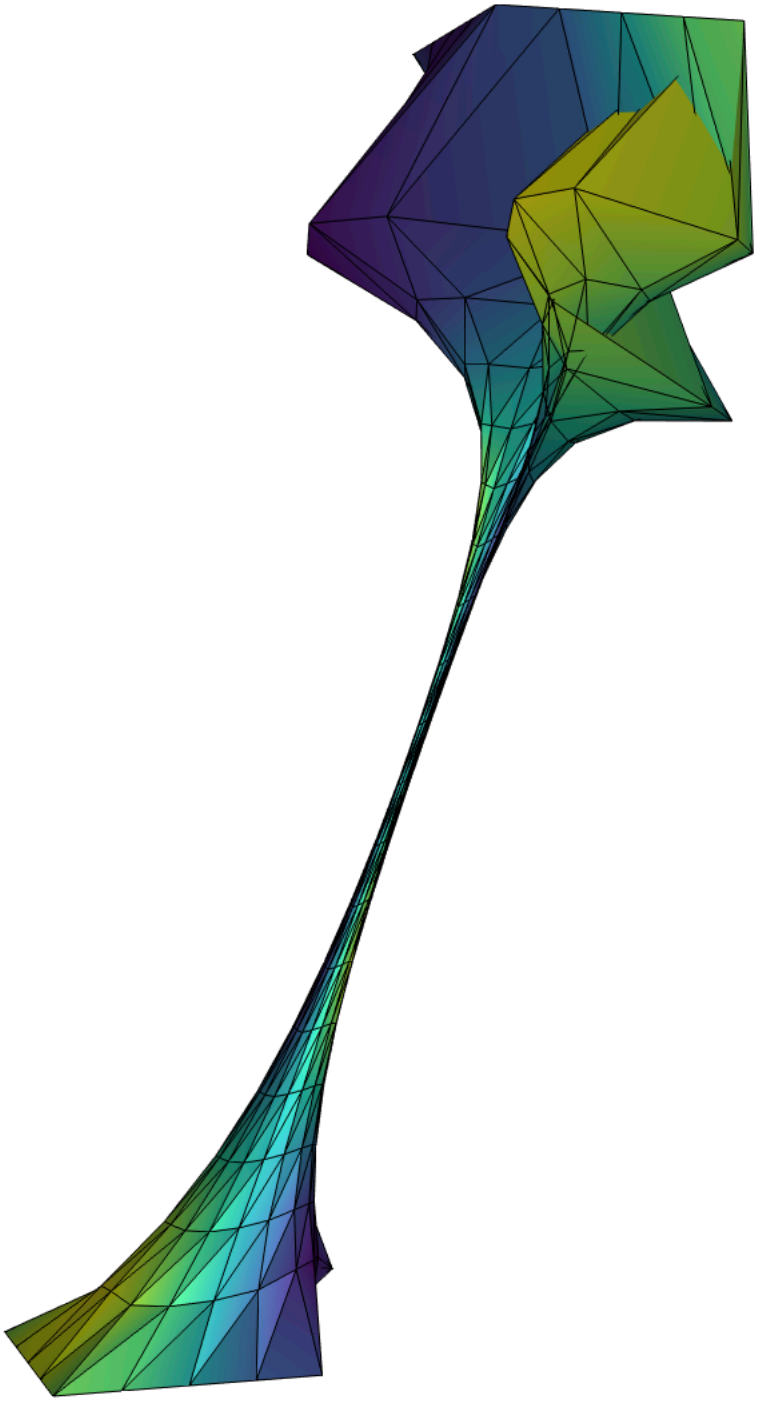
Lots of bugs. Kind of a great trophy designer though.







Inverted guessed vertices and current vertices whil computing rotations.



No real bunnies nor armadillos were harmed during this homework.

[ ]: