

Process

Finite Element Analysis (FEA) is a technique used by engineers to computationally analyze the physical properties of solids. FEA uses Linear Algebra to discretize a part to a set of simple vectors, then apply simple stress strain equations to every generated vector to compute a final solution. This enables engineers to understand complex and normally impossible to analyze geometries (curved shapes, electromagnetic fields).

Meshing

Meshing takes a complex part and breaks it down into finite, connected elements. Meshing is an important part of accurate FEA and needs to be implemented correctly based on the objective and type of FEA.

Some important meshing parameters are:

- **Element size** - Smaller elements are more accurate but take longer to calculate
- **Growth pattern and rate** - Tuning what areas have small to large size is an important parameter for getting accurate and efficient results.

Ways to check mesh quality:

- **Aspect Ratio** - Measures how long corresponding sides of elements are, stretched elements are not good for accuracy.
- **Skewness** - Measures the angle deviation from ideal shape, distorted elements are bad for accuracy.

Partial Differential Equations for FEA

Overview: To approximate PDEs they are converted from the strong form to the weak form, which allows for approximations. Here the strong form for the one dimensional small strain relation is used.

Weak Form: Given a PDE, the weak form is derived by multiplying with a test function $v(x)$ and integrating over the domain Ω :

$$\int_{\Omega} \varepsilon v \, dx = \int_{\Omega} f v \, dx,$$

Approximation Using Basis Functions: We approximate the solution $u(x)$ and the test function $v(x)$ using finite basis functions:

$$u(x) \approx \phi_i(x), \quad u(x) \approx \sum_{j=1}^N U_j \phi_j(x),$$

Substitution into the Weak Form: Substitute the approximations into the weak form.

Assembly of the System of Equations: Separate the terms for each U_j and rewrite as a linear system:

$$\sum_{j=1}^N \left(\int_{\Omega} \frac{\partial \phi_j}{\partial x} \phi_i \, dx \right) U_j = \int_{\Omega} f \phi_i \, dx.$$

This can be written in matrix form as:

$$KU = F,$$

Boundary Conditions: Apply Neumann and Dirichlet boundary conditions onto the system.

Solve system: Solve the system of equations using appropriate numerical techniques.

$$\text{Stiffness matrix: } K_{ij} = \int_{\Omega} \frac{\partial \phi_j}{\partial x} \phi_i \, dx$$

$$\text{Force vector: } F_i = \int_{\Omega} f \phi_i \, dx$$

Stress Matrix

Using the mesh matrix and the partial differential form, FEA solvers generate element stress matrices which are then combined into a global stress matrix. Every element has a finite degrees of freedom (DoF), for example a 3D object has 6 DoF. This is represented by the form:

$$\{U\} = \begin{bmatrix} u_1 \\ v_1 \\ \theta_1 \\ u_2 \\ v_2 \\ \theta_2 \end{bmatrix}$$

The DoF matrix is used to derive an element stiffness matrix for each of the generated nodes. Most bodies behave as a spring modeled by $F = -kx$. Similarly, we can model the part as:

$$\{F\} = [K]\{U\}$$

where $[K]$ is the element stiffness matrix.

The element stiffness matrix is derived by converting the strong form of the governing equilibrium partial differential into the discrete weak form over elements of the mesh adjacency matrix.

$$K = \begin{bmatrix} k_{11} & k_{12} & k_{13} & 0 \\ k_{21} & k_{22} & k_{23} & 0 \\ k_{31} & k_{32} & k_{33} & 0 \end{bmatrix}$$

Each of the local stiffness matrices are assembled into a global stiffness matrix. for example, $K = k + k'$

$$K = \begin{bmatrix} k_{11} & k_{12} & k_{13} & 0 \\ k_{21} & k_{22} + k'_{11} & k_{23} + k'_{12} & k'_{13} \\ k_{31} & k_{32} + k'_{21} & k_{33} + k'_{22} & k'_{23} \\ 0 & k'_{31} & k'_{32} & k'_{33} \end{bmatrix}$$

The overlap between the local matrices are eliminated according to the adjacency matrix to construct the global stiffness matrix. The global stiffness matrix is NxN elements where N corresponds to the reference values in the DoF Matrix.

Solving the System

Solving the system is as simple as adding the boundary force conditions to the force matrix F and back solving for the DoF matrix U .

$$\{F\} = [K]^{-1}\{U\}$$

In practice, this is hard to do because of the size of the $[K]$ matrix, so other solving methods like **conjugate gradient method** are used to optimize it.

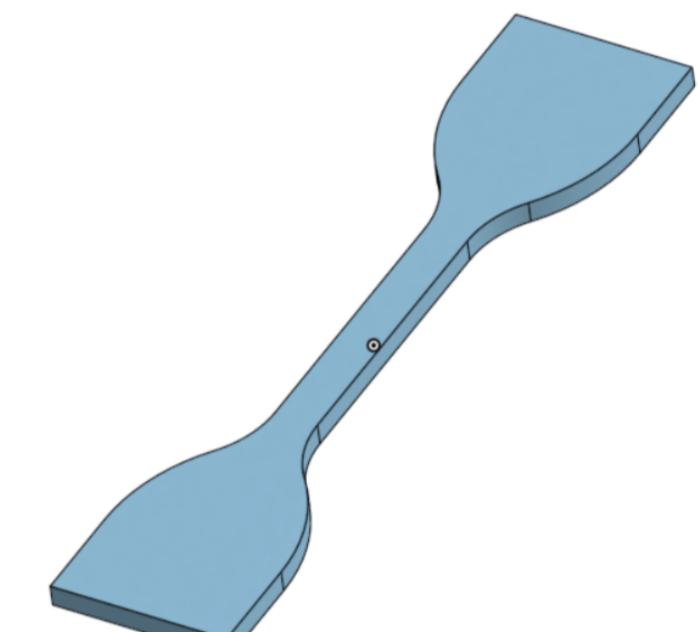
Testing and Validation

We implemented an FEA solver in MATLAB, and validated our simulation by 3D printing and testing it.

Test Piece



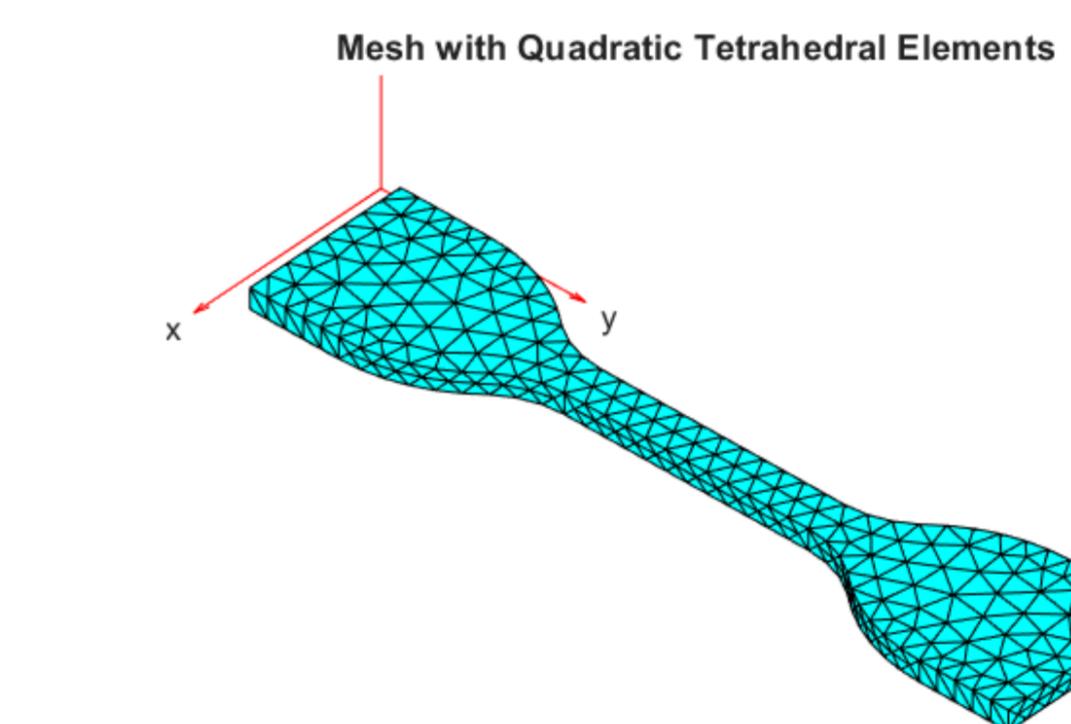
Dogbone in Strain Machine



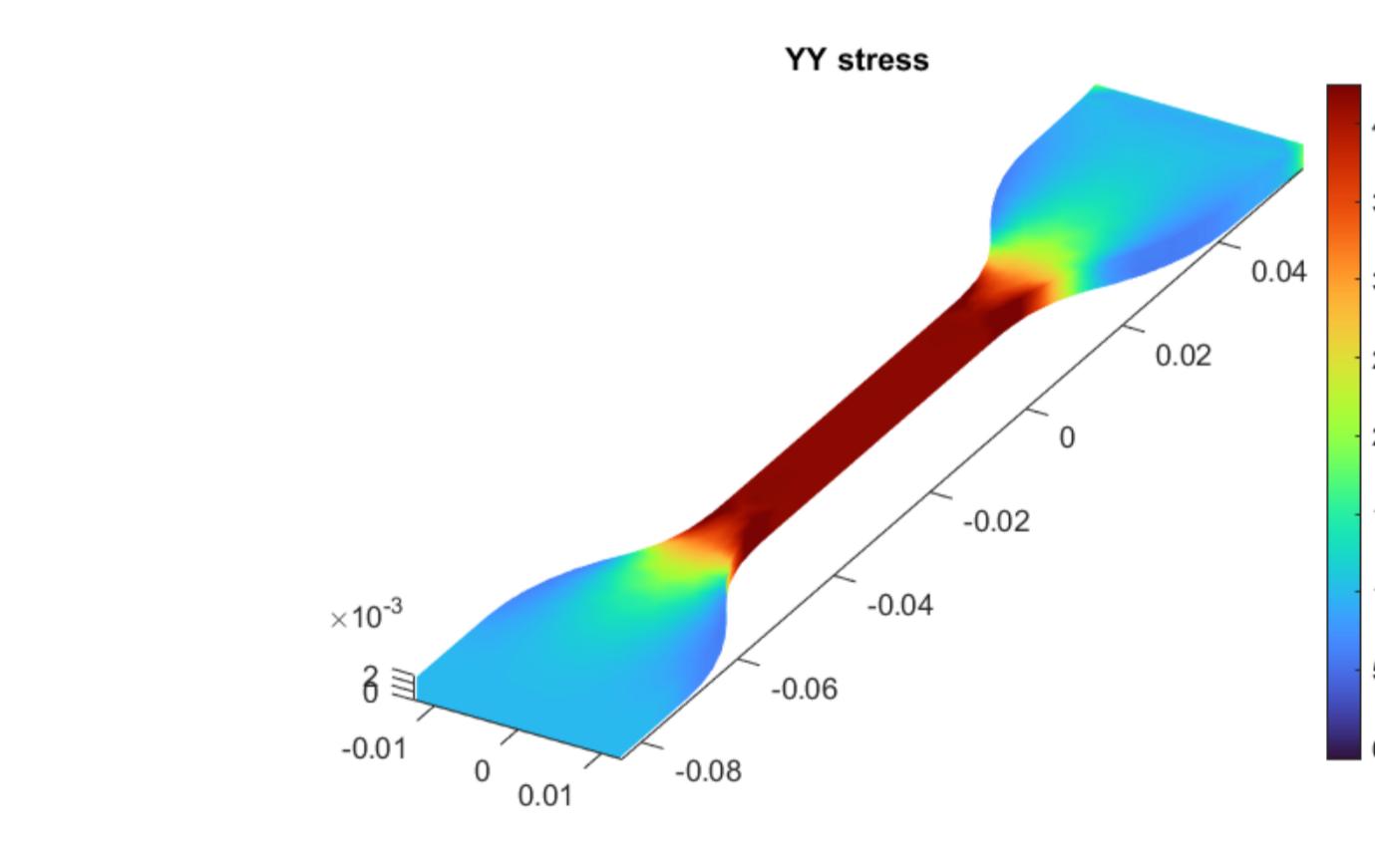
Dogbone CAD Model

FEA in MATLAB

We used the PDE toolbox to mesh and solve our part for strain.



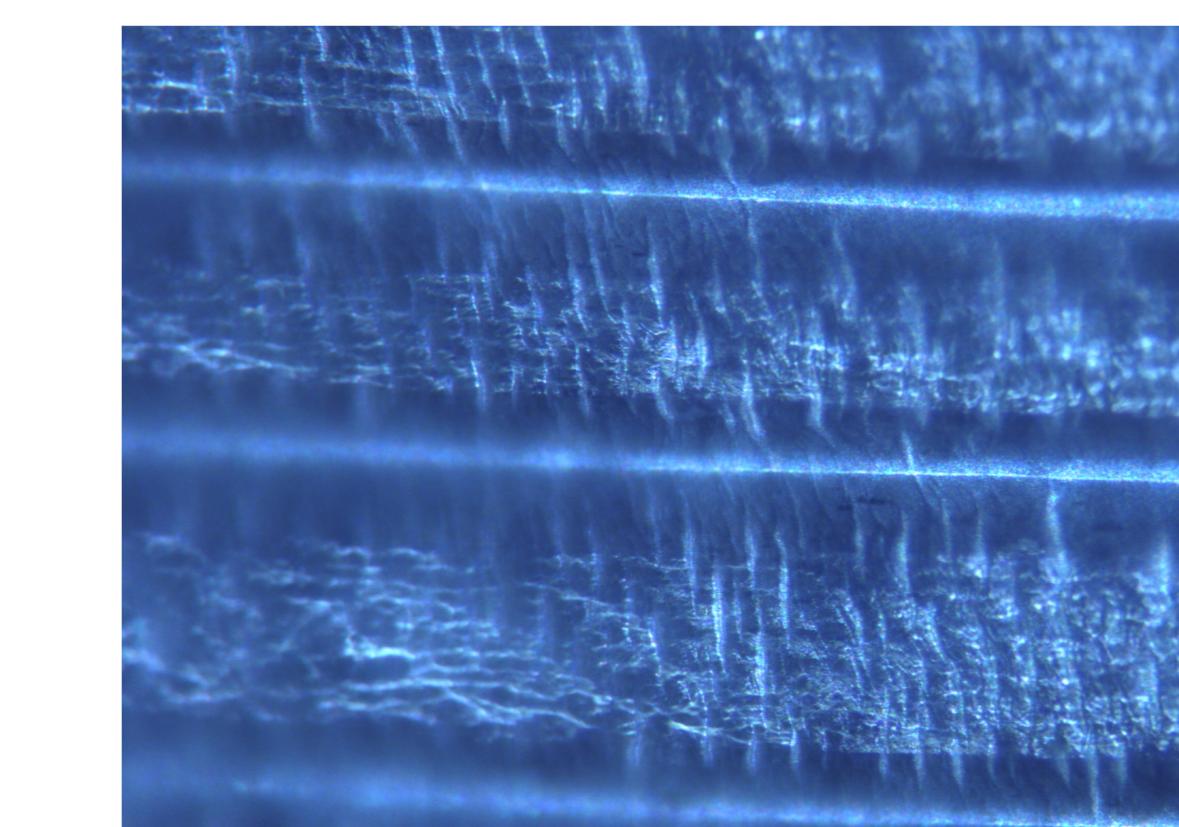
FEA Mesh



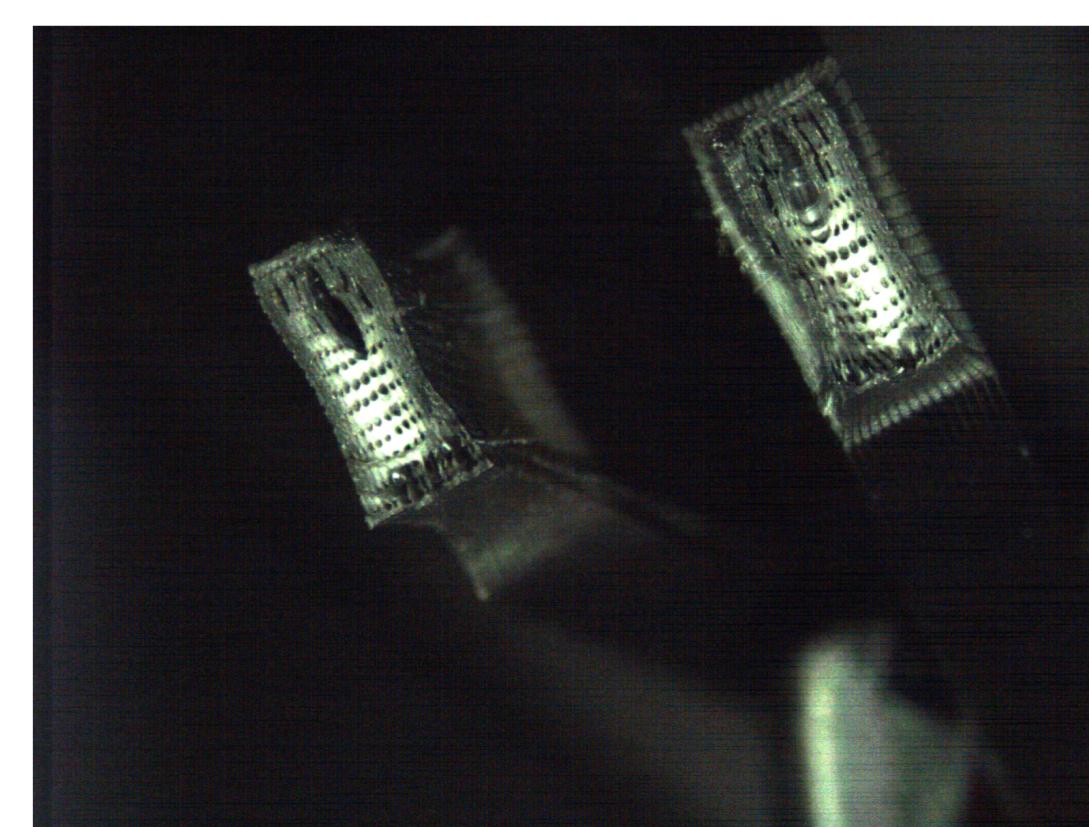
FEA Mesh Solution

Testing

We used the MatSci Instron Strain Machine to test our part.



Deformation at Neck



Necking at Break