Results

Class Script

```
classdef Material
   properties
       E             % Young's modulus (psi)
       nu            % Poisson's ratio
       su            % Tensile strength (psi)
       KIC           % Fracture toughness (psi*sqrt(in))
       sfatigue      % Fatigue strength for 10^6 cycles (psi)
       name          % Material name
   end
   methods
       function obj = Material(E, nu, su, KIC, sfatigue, name)
           obj.E = E;
           obj.nu = nu;
           obj.su = su;
           obj.KIC = KIC;
           obj.sfatigue = sfatigue;
           obj.name = name;
       end
   end
end
```

Hand Calculation Scripts

```
clear all;
clc;
% Global Parameters
M = 600;     % Max torque (in*lbf)
a = 0.04;    % Crack depth (in)
%% ========================= Testing =========================
L = 16;      % Length from drive to applied load (in)
h = 0.75;    % Necked width (in)
b = 0.5;     % necked thickness (in)
C = 1;       % Distance from drive to strain gauge (in)
h0 = 0.75;  % Root width (in)
b0 = 0.5;    % Root thickness (in)
L0 = L;      % Root Length (in)
P = M / L;  % Applied Load
% Define material and properties
M42_Steel = Material(32e6, 0.29, 370e3, 15e3, 115e3, 'M42 Steel');
% Stress and deflection | Safety factor | Stain gauge
[max_deflection, max_stress] = stress_and_deflection(M42_Steel, M, L, h, b, P,
h0, b0, L0);
[X_o, X_k, X_s] = safety_factor(M42_Steel, M, L, h, b, P, a, max_stress, h0,
b0, L0);
[strain_at_gauge, gauge_output] = strain_gauge(M42_Steel, M, L, h, b, P, C);
% Display results
disp([' ======================================== ' ...
      'Testing Results ' ...
      '========================================']);
T = table(max_deflection, max_stress, X_o, X_k, X_s, strain_at_gauge,
gauge_output);
disp(T);
% max_deflection = 0.091 in | max_stress = 12.80 ksi
% X_o = 28.9 | X_k = 2.95 | X_s = 8.98
% strain at gauge = 375 microstrain | gauge+output = 0.38 mV/V
%% ========================= Materials =========================
Steel_4140 = Material( ...
   mean([30.2 31.3]) * 1e6, ...    % E (psi)
   mean([0.285 0.295]), ...        % nu
   mean([86.3 104]) * 1e3, ...     % su (psi)
   mean([61.9 97.4]) * 1e3, ...    % KIC (psi*sqrt(in))
   mean([58.8 88.2]) * 1e3, ...    % fatigue strength (psi)
   'AISI 4140 Steel' );
Steel_4340 = Material( ...
   mean([29.7 30.9]) * 1e6, ...
   mean([0.285 0.295]), ...
   mean([112 138]) * 1e3, ...
   mean([51.9 87.4]) * 1e3, ...
   mean([69.5 103]) * 1e3, ...
   'AISI 4340 Steel' );
```

```
Steel_17_4PH = Material( ...
   mean([28.6 30]) * 1e6, ...
   mean([0.27 0.281]), ...
   mean([115 127]) * 1e3, ...
   mean([125 153]) * 1e3, ...
   mean([60.9 89.5]) * 1e3, ...
   '17-4PH Stainless Steel' );
Aluminum_7075_T6 = Material( ...
   mean([10 11]) * 1e6, ...
   mean([0.325 0.335]), ...
   mean([66.7 76.9]) * 1e3, ...
   mean([24.2 24.4]) * 1e3, ...
   mean([24.9 30.8]) * 1e3, ...
   'Aluminum 7075 T6' );
Aluminum_6061_T6 = Material( ...
   mean([9.66 10.2]) * 1e6, ...
   mean([0.325 0.335]), ...
   mean([34.8 40.6]) * 1e3, ...
   mean([27.3 32.8]) * 1e3, ...
   mean([17.4 23.8]) * 1e3, ...
   'Aluminum 6061 T6' );
Aluminum_2024_T3 = Material( ...
   mean([10.4 11]) * 1e6, ...
   mean([0.33 0.343]), ...
   mean([42.1 54]) * 1e3, ...
   mean([33.7 37.3]) * 1e3, ...
   mean([17.2 34]) * 1e3, ...
   'Aluminum 2024 T3' );
Titanium_6Al_4V = Material( ...
   mean([16.3 16.7]) * 1e6, ...
   mean([0.332 0.349]), ...
   mean([114 130]) * 1e3, ...
   mean([94.2 104]) * 1e3, ...
   mean([88.9 116]) * 1e3, ...
   'Titanium 6Al-4V' );
Titanium_6Al_4V_ELI = Material( ...
   mean([16 17]) * 1e6, ...
   mean([0.332 0.352]), ...
   mean([110 131]) * 1e3, ...
   mean([82.7 100]) * 1e3, ...
   mean([46.6 65.1]) * 1e3, ...
   'Titanium 6Al-4V ELI' );
Materials = [ ...
   Steel_4140, ...
   Steel_4340, ...
   Steel_17_4PH, ...
   Aluminum_7075_T6, ...
   Aluminum_6061_T6, ...
   Aluminum_2024_T3, ...
```

```
    Titanium_6Al_4V, ...
    Titanium_6Al_4V_ELI ...
];
%% ========================= Design Parameters & Dimensions
=========================
% Values were chosen as no materials fail at root
h0 = 0.8; % Root width (in)
b0 = 0.8; % Root thickness (in)
L = 15;   % Length from drive to applied load (in)
Distances = 1:1:(L-1); % Distance from drive to strain gauge (in)
Widths = 0.1:0.1:0.8;     % Necked widths & thicknesses
% Minimum Safety factors and strain gaige output
X_o_bench = 4;
X_k_bench = 2;
X_s_bench = 1.5;
strain_bench = 0.001;
P = M/L; % Applied load
% Loop through widths
for iW = 1:length(Widths)
    h = Widths(iW); % Set necked width
    b = Widths(iW); % Set necked thickness
    % Set up boolean array
    nM = length(Materials);
    nC = length(Distances);
    meets_safety = false(nC, nM);

    % Loop through distances
    for iD = 1:length(Distances)
        C = Distances(iD); % Set distance from drive to strain gauge
        L0 = C - 1; % Set root Length (end 1 in before strain gauge) (in)

        % Loop through Materials
        for iM = 1:length(Materials)
            material = Materials(iM); % Set material
            % Stress and deflection | Safety factor | Stain gauge
            [max_deflection, max_stress] = stress_and_deflection(material, M,
L, h, b, P, h0, b0, L0);
            [strain_at_gauge, gauge_output] = strain_gauge(material, M, L, h,
b, P, C);
            [X_o, X_k, X_s] = safety_factor(material, M, L, h, b, P, a,
max_stress, h0, b0, L0);
            % Update boolean array
            if X_o >= X_o_bench && X_k >= X_k_bench && X_s >= X_s_bench &&
gauge_output >= strain_bench
                meets_safety(iD, iM) = true;
            else
                meets_safety(iD, iM) = false;
            end
        end
```

```
    end

    % Create and display results table
    MaterialNames = string({Materials.name});
    safety_table = array2table(meets_safety, 'VariableNames', MaterialNames);
    safety_table = addvars(safety_table, Distances','Before', 1,
'NewVariableNames', 'Distance (in)');
    fprintf(['\n
================================================================= ' ...
        'Safety results for h = %.3f in (b = %.3f in) ' ...
        '=================================================================\n'],
h, b);
    disp(safety_table);
end
%% ========================= Selected Material =========================
% Using h = 0.5 in | b = 0.5 in | L = 15 in | C = 5 in | Titanium 6Al-4V
L = 15;      % Length from drive to applied load (in)
h = 0.5;     % Necked width (in)
b = h;       % Necked thickness (in)
C = 5;       % Distance from drive to strain gauge (in)
L0 = C - 1; % Root length (in)
P = M / L;  % Applied Load
% Stress and deflection | Safety factor | Stain gauge
[max_deflection, max_stress] = stress_and_deflection(Titanium_6Al_4V, M, L, h,
b, P, h0, b0, L0);
[X_o, X_k, X_s] = safety_factor(Titanium_6Al_4V, M, L, h, b,P, a, max_stress,
h0, b0, L0);
[strain_at_gauge, gauge_output] = strain_gauge(Titanium_6Al_4V, M, L, h, b, P,
C);
% Display results
T = table(max_deflection, max_stress, X_o, X_k, X_s, strain_at_gauge,
gauge_output);
fprintf(['\n  = ' ...
        'Results for h = b = %.3f in | L = %.3f in | C = %.3f | L0 = %.3f | %s
(Properties: E = %.3s psi, nu = %.3s) ' ...
        '=\n'], h, L, C, L0, Titanium_6Al_4V.name, Titanium_6Al_4V.E,
Titanium_6Al_4V.nu);
disp(T);
%% ========================= Stress and Deflection =========================
function [max_deflection, max_stress] = stress_and_deflection(material, M, L,
h, b, P, h0, b0, L0)
    % Extract material properties
    E = material.E;
    nu = material.nu;
    su = material.su;
    KIC = material.KIC;
    sfatigue = material.sfatigue;

    % Calculate stress and deflection
```

```
    I1 = b0*h0^3/12; % Root moment of inertia
    I2 = b*h^3/12;   % necked moment of inertia
    M_neck = P*(L-L0); % Moment at neck region
    neck_stress = M_neck*(h/2)/I2;   % Stress at neck
    root_stress = M*(h0/2)/I1;       % Stress at root
    max_stress = max(root_stress, neck_stress);
    % Energy method
    M = @(x) P.*(L-x);
    U = integral(@(x) M(x).^2/(2*E*I1), 0, L0) + integral(@(x)
M(x).^2/(2*E*I2), L0, L);
    max_deflection = (2/P)*U;
end
%% ========================= Safety Factor =========================
function [X_o, X_k, X_s] = safety_factor(material, M, L, h, b, P, a,
max_stress, h0, b0, L0)
    % Extract material properties
    E = material.E;
    nu = material.nu;
    su = material.su;
    KIC = material.KIC;
    sfatigue = material.sfatigue;
    % Calculate Safety factor for strength
    X_o = su/max_stress;
    % Calculate Safety factor for crack growth
    % For root
    Sg_root = 6*M/(b0*h0^2);
    KI_root = 1.12*Sg_root*sqrt(pi*a);

    % For neck
    M_neck = P*(L-L0);
    Sg_neck = 6*M_neck/(b*h^2);
    KI_neck = 1.12*Sg_neck*sqrt(pi*a);
    X_k = KIC/max(KI_root, KI_neck);

    % Calculate Safety factor for fatigue
    X_s = sfatigue/max_stress;
end
%% ========================= Strain Gauge =========================
function [strain_at_gauge, gauge_output] = strain_gauge(material, M, L, h, b,
P, C)
    % Extract material properties
    E = material.E;
    nu = material.nu;
    su = material.su;
    KIC = material.KIC;
    sfatigue = material.sfatigue;

    % Calculate strain at gauge
    M_gauge = P*(L-C);
```

```
    I = b*h^3/12;
    stress_gauge_1 = M_gauge*(h/2)/I;
    strain_gauge_1 = stress_gauge_1/E;
    stress_gauge_2 = M_gauge*(-h/2)/I;
    strain_gauge_2 = stress_gauge_2/E;

    strain_at_gauge = strain_gauge_1;
    % Calculate strain gauge output
    k = 2;
    gauge_output = (k/4)*(strain_gauge_1 - strain_gauge_2);
end
```

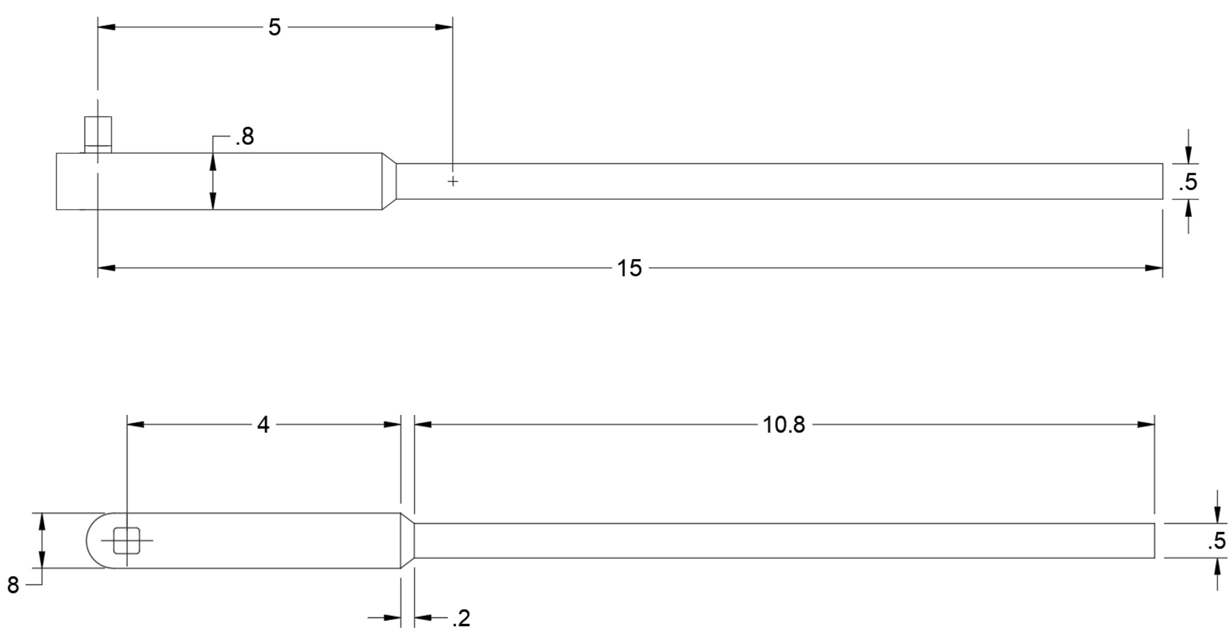
Hand Calculation Results

The MATLAB script iterated through the handle thickness and the strain-gauge distance from the drive for different materials. The design used a square handlebar with a thick root section (0.8 × 0.8 in) and a necked section (h × b in). The root region ended 1 in before the strain gauge, and the overall handle length was kept constant at 15 in. An example of one iteration table is shown below.

==================== Safety results for h = 0.500 in (b = 0.500 in) ====================

Distance (in)	AISI 4140 Steel	AISI 4340 Steel	17-4PH Stainless Steel	Aluminum 7075 T6	Aluminum 6061 T6	Aluminum 2024 T3	Titanium 6Al-4V	Titanium 6Al-4V ELI
1	false	false	false	false	false	false	true	true
2	false	false	false	false	false	false	true	true
3	false	false	false	false	false	false	true	true
4	false	false	false	false	false	false	true	true
5	false	false	false	false	false	false	true	true
6	false	false	false	false	false	false	true	true
7	false	false	false	true	false	false	false	false
8	false	false	false	true	false	false	false	false
9	false	false	false	true	false	false	false	false
10	false	false	false	false	false	false	false	false
11	false	false	false	false	false	false	false	false
12	false	false	false	false	false	false	false	false
13	false	false	false	false	false	false	false	false
14	false	false	false	false	false	false	false	false

Based on these results, the titanium alloy Ti-6Al-4V was selected as the material, and the final wrench dimensions are shown in the drawing below. In particular, the necked region measures 0.5 × 0.5 in, and the strain gauge is positioned 5 in from the drive.



The MATLAB script was then used to generate results for the maximum normal stress, the strain at the strain gauge location, and the deflection at the load point. It is important to note that although the drawing above includes a lofted region connecting the root and necked sections of the handle (to reduce stress concentrations), the hand calculations assumed an abrupt transition between the two sections.

```
= Results for h = b = 0.500 in | L = 15.000 in | C = 5.000 in | L0 = 4.000 in | Titanium 6Al-4V (Properties: E = 1.650e+07 psi, nu = 3.405e-01) =
  max_deflection    max_stress     X_o       X_k       X_s      strain_at_gauge    gauge_output
  ______________    __________    ______    ______    ______    _______________    ____________

      0.2549          21120       5.7765    11.818    4.8509       0.0011636         0.0011636
```

Hand Calculation Results		
Maximum Normal Stress (Ksi)	Strain at Strain Gauge (mε)	Load Point Deflection (in)
21.12	1.164	0.2549

The strain gauges selected for this design are the SGD-2/350-LY13 models from Dwyer Omega (Link). Each gauge measures 7.6 mm (0.3 in) in length and 5.8 mm (0.24 in) in width. The gauges will be arranged in a half-bridge configuration.

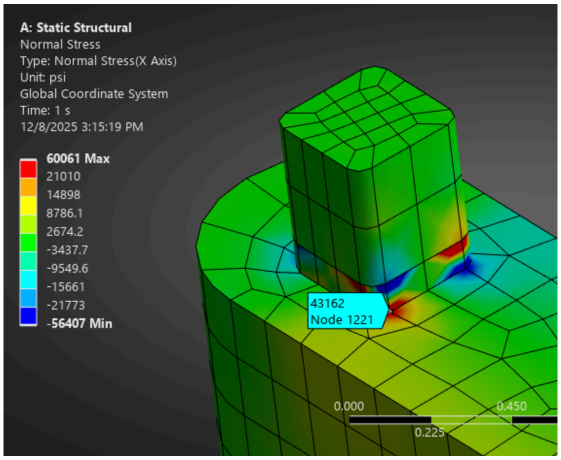
FEM Calculation Results

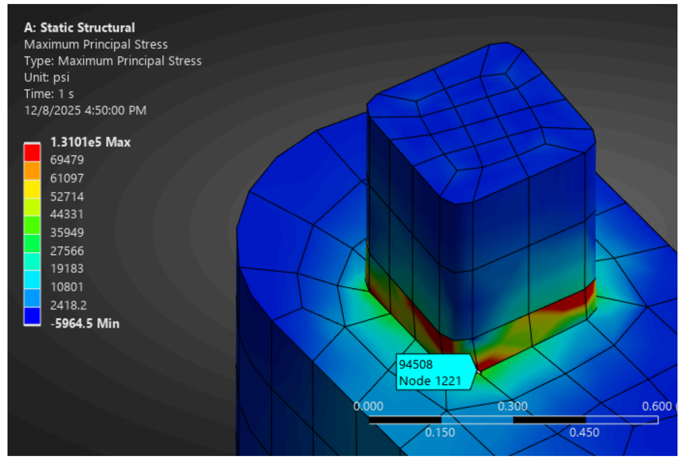
Initially, the analysis was performed using a mesh element size of 0.25 in. It was then repeated with a finer mesh size of 0.125 in. The strain at the strain-gauge location was obtained by using a strain probe at that point. An examination of the normal stress shows a maximum at the interface between the clamped region of the drive and the filleted region; however, this peak is caused by a singularity and was not used when reporting the maximum normal stress. The maximum principle stresses are also shown in the pictures below.

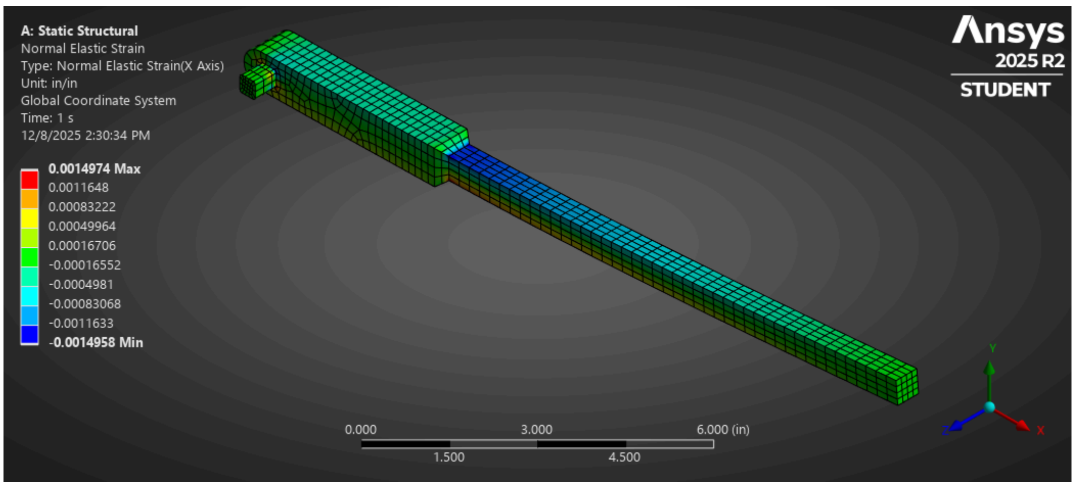
FEM Calculation Results (0.25 in element size)		
Maximum Normal Stress (Ksi)	Strain at Strain Gauge (mε)	Load Point Deflection (in)
43.16	1.163	0.3353

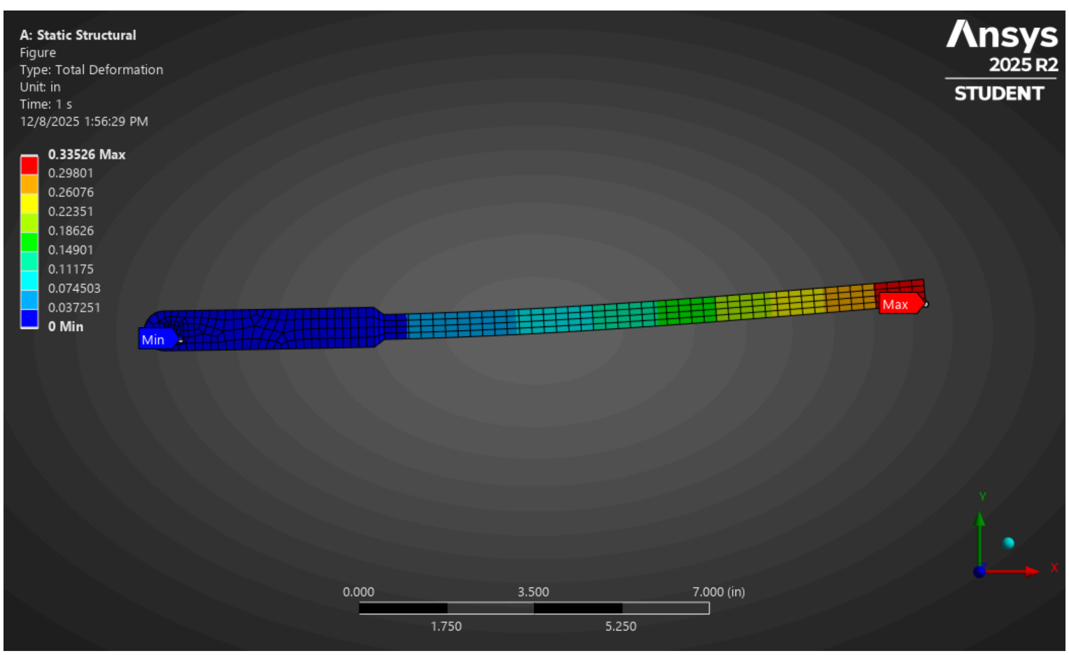
FEM Calculation Results (0.125 in element size)		
Maximum Normal Stress (Ksi)	Strain at Strain Gauge (mε)	Load Point Deflection (in)
44.99	1.164	0.3370

Element Size: 0.25 in

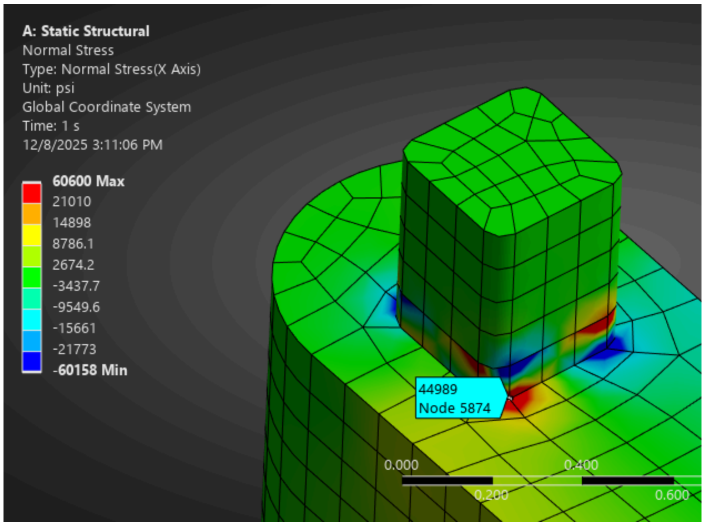


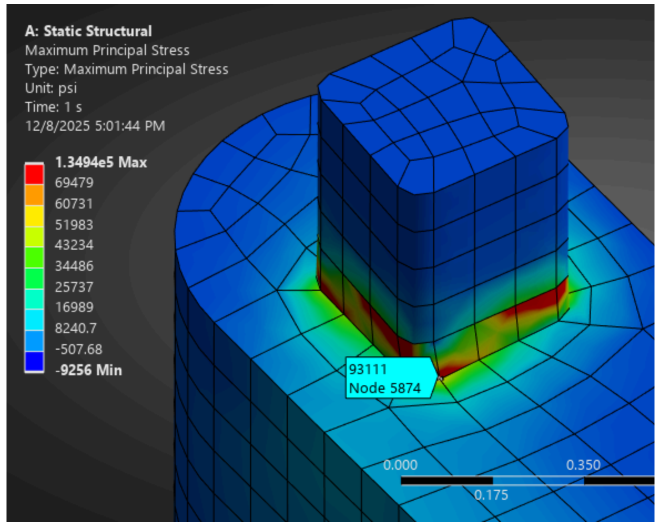


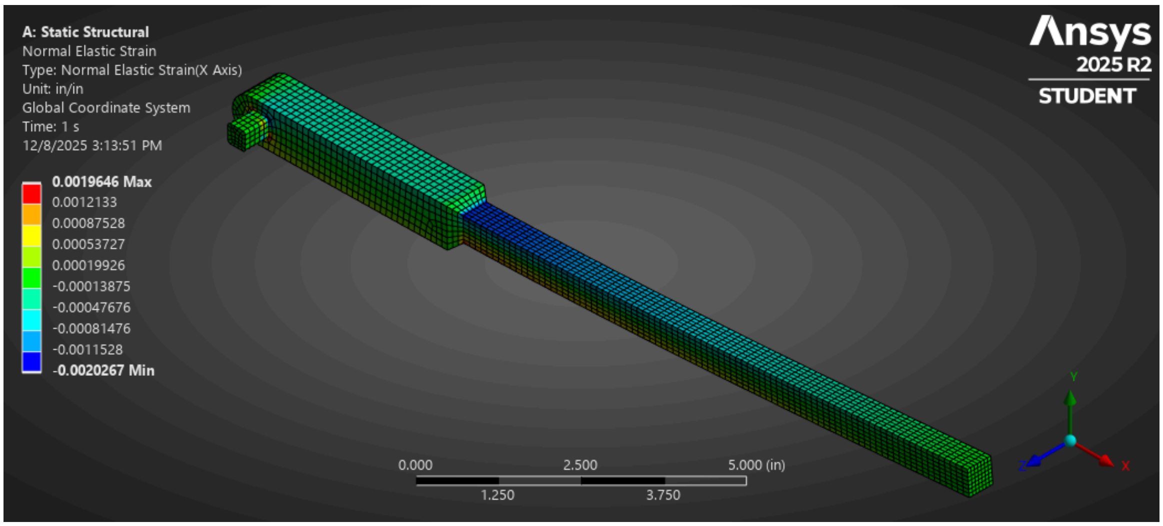


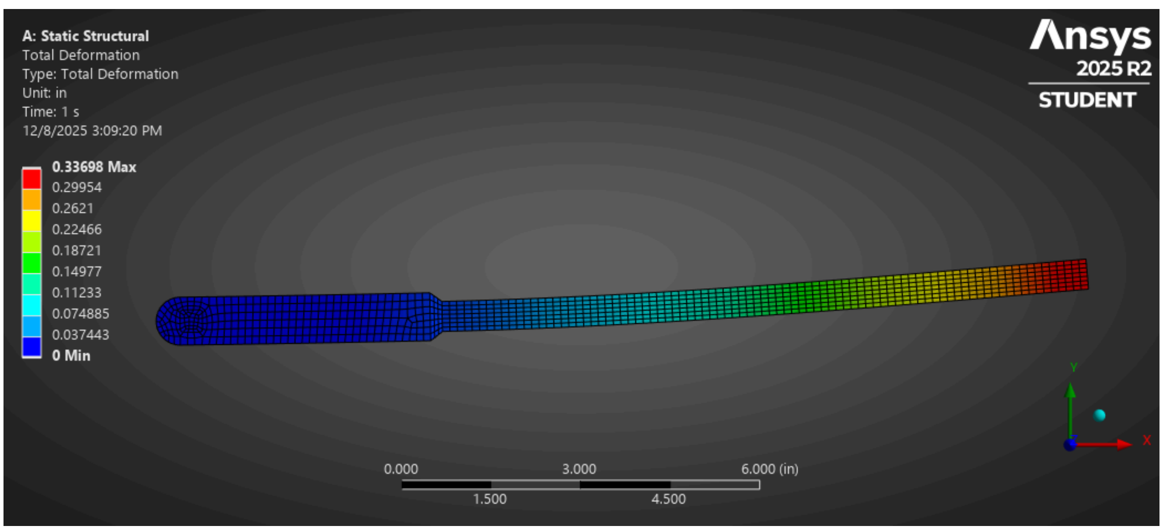


Element Size: 0.125 in





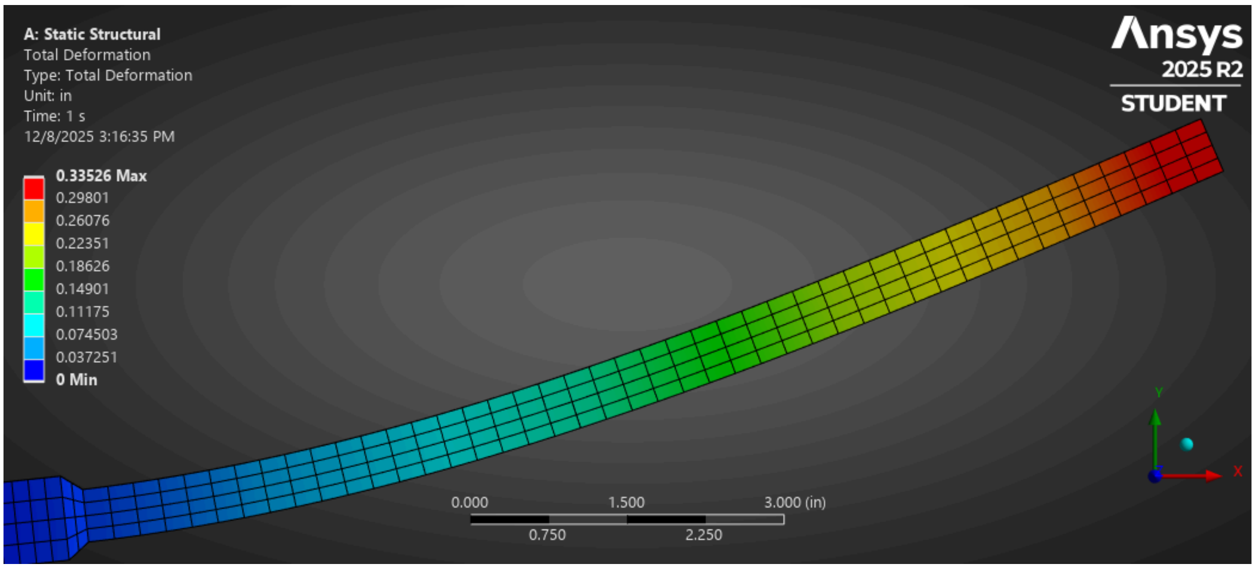


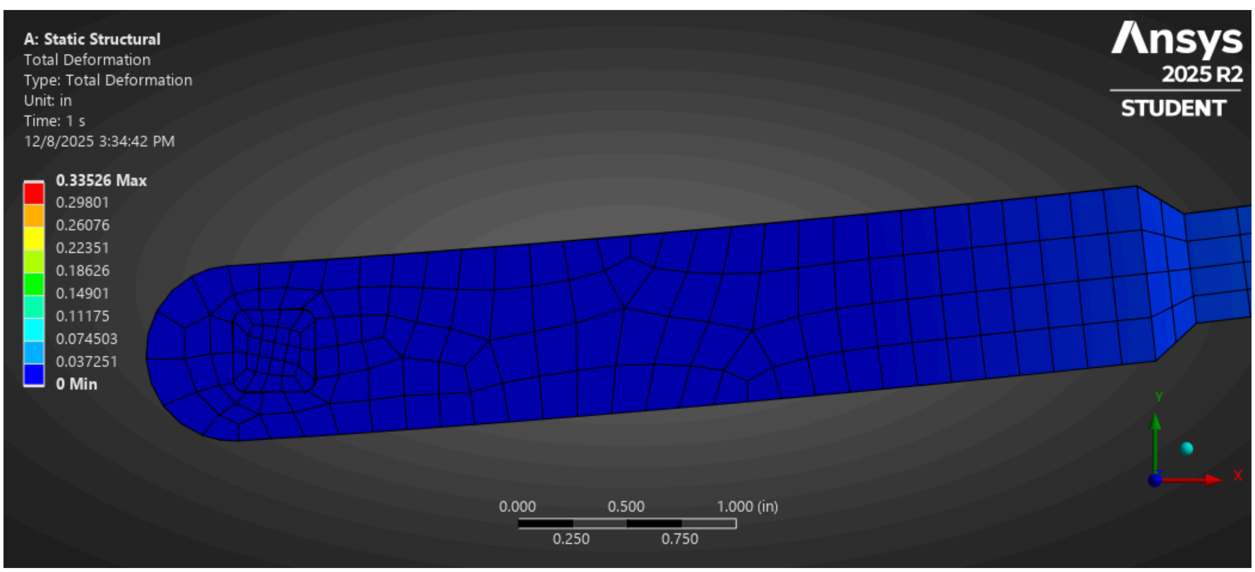


Reflection

Beam Theory

Overall, the mesh lines that cut across the beam handle remain straight. To check this, the deformation was exaggerated by a factor of five. In the necked part of the handle, the mesh lines show no warping at all. Closer to the drive, some slight warping appears, but it is very minimal. Overall, even with a small amount of warping near the drive, the model still provides a reasonably accurate estimate.





FEM vs Hand Calculated: Maximum Normal Stress

The FEM calculated a maximum normal stress of 44.99 ksi, while the hand calculation gave 21.12 ksi. In the hand calculation, this stress occurs at the start of the necked section of the handle. In the FEM model, the highest stress appears on the edge between the drive fillet and the handle. The higher value in the FEM results comes from stress concentrations that were not included in the hand calculations.

FEM vs Hand Calculated: Maximum Normal Stress

The FEM calculated a displacement of 0.3370 in, while the hand calculation gave 0.2549 in. For the hand calculations, displacement was found using energy methods. The handle was divided into two sections (the root and the necked regions) and an abrupt transition between them was assumed. In the CAD model, however, the two regions were connected by a 0.2-in lofted section. This should have caused the FEM to predict a lower displacement, but instead the FEM result was higher. This can be explained by the clamped constraint acting only on the upper part of the drive and not on the filleted area. As a result, the displacement comes not only from bending due to the applied moment but also from torsion in the drive, leading to a larger overall displacement.