Table of Contents

AERO 433: Experimental Stress Analysis

Author: Justin Self, Will Crooks

```
% Housekeeping
clear all; close all; clc;
```

Constants

Aluminum alloy 3004

```
ElasticModWall = 69e9; % GPa to Pa
PoissonRatioWall = 0.33;
wallThickness = 0.1e-3; % mm to m
canDiamWill = 65.95e-3; % mm to m
ambientPressure = 101.1e3; % kPa to Pa
```

Bad Run Data - commented out since not needed

raw = readmatrix("bad_run2.TXT"); A = raw(:,2); B = raw(:,3);

% Time vectors for numeric values ta = find(~isnan(A)); tb = find(~isnan(B));

% Extract non NaN values from raw data A = A(~isnan(A)); B = B(~isnan(B));

% Remove the outlier in the data figure() % Hoop % Plot before the outlier p1 = plot(ta(1:10),A(1:10),'-square','LineWidth',1); p1.Color = 'b'; hold on % Plot after the outlier p2 = plot([ta(10) ta(12)],[A(10) A(12)],'-square','LineWidth',1); p2.Color = 'b';

% Longitudinal p3 = plot(tb,B,'-o','LineWidth',1); p3.Color = 'r';

% Annotation dim = [.2 .3 .3 .3]; str = 'One outlier removed'; annotation('textbox',dim,'String',str,'FitBoxToText','on');

% Graph pretty ylim padded xlim tight xLab = xlabel('Time [s]','Interpreter','latex'); yLab = ylabel('Microstrain [$\mu \epsilon$]','Interpreter','latex'); plotTitle = title('Run 2: Microstrain During Aluminum Can

Opening','interpreter','latex'); set(plotTitle,'FontSize',14,'FontWeight','bold') set(gca,'FontName','Palatino Linotype') set([xLab, yLab],'FontName','Palatino Linotype') set(gca,'FontSize', 9) set([xLab, yLab],'FontSize', 14) grid on legend('Hoop','','Longitudinal', 'interpreter','latex','Location', 'best')

Plot Savannah's dataraw = readmatrix("justin.TXT");

```
raw = readmatrix("Savannah_run4.TXT");
A = raw(:,2);
B = raw(:,3);

% Time vectors for numeric values
ta = find(~isnan(A));
tb = find(~isnan(B));

% Extract non NaN values from raw data
A = A(~isnan(A));
B = B(~isnan(B));

% Temperature correction factor
T = 72; % deg F (Wunderground; after interpolation)
Tref = 73; % deg F (MicroMeasurements Wheatstone Bridge)
cf = T/Tref;

% Correction
A = A*cf;
B = B*cf;

figure()
% Hoop
p1 = plot(ta,A,'-square','LineWidth',1);
p1.Color = 'b';

hold on
% Longitudinal
p2 = plot(tb,B,'-d','LineWidth',1);
p2.Color = '#808080';

% Graph pretty
ylim padded
xlim tight
xLab = xlabel('Time [s]','Interpreter','latex');
yLab = ylabel('Microstrain [$\mu \epsilon$]','Interpreter','latex');
plotTitle = title('Savannah: Microstrain During Aluminum Can
Opening','interpreter','latex');
set(plotTitle,'FontSize',14,'FontWeight','bold')
set(gca,'FontName','Palatino Linotype')
set([xLab, yLab],'FontName','Palatino Linotype')
set(gca,'FontSize', 9)
set([xLab, yLab],'FontSize', 14)
grid on
```

```
% attempted data correction of longitudinal stress
fudgeFactor = mean(B(1:5));
B_adjusted = B - fudgeFactor;
plot(ta,B_adjusted,'-o','LineWidth',1,'Color','r')

legend('Hoop','Longitudinal (raw)',"Longitudinal
(adjusted)",'interpreter','latex','Location', 'best')
```

savannah analysis

```
strainLong = B_adjusted*1e-6;
strainHoop = A*1e-6;
stressLong = (ElasticModWall*(strainLong+PoissonRatioWall*strainHoop))/(1-
PoissonRatioWall^2);
stressHoop = (ElasticModWall*(strainHoop+PoissonRatioWall*strainLong))/(1-
PoissonRatioWall^2);

strainLongBefore = mean(strainLong(1:5));
strainLongAfter = mean(strainLong(9:end));
strainHoopBefore = mean(strainHoop(1:5));
strainHoopAfter = mean(strainHoop(9:end));

stressLongBefore = mean(stressLong(1:5));
stressLongAfter = mean(stressLong(9:end));
stressHoopBefore = mean(stressHoop(1:5));
stressHoopAfter = mean(stressHoop(9:end));

disp("Savannah - Run 4")
disp("Before:")
disp("L micro-strain = " + strainLongBefore*1e6)
disp("H micro-strain = " + strainHoopBefore*1e6)
disp("L stress = " + stressLongBefore*1e-6 + " MPa")
disp("H stress = " + stressHoopBefore*1e-6 + " MPa")
disp("After:")
disp("L micro-strain = " + strainLongAfter*1e6)
disp("H micro-strain = " + strainHoopAfter*1e6)
disp("L stress = " + stressLongAfter*1e-6 + " MPa")
disp("H stress = " + stressHoopAfter*1e-6 + " MPa")

% stress ratio
stressRatio = mean(stressHoop(9:end)./stressLong(9:end));
disp("Run 4 - Stress Ratio (H/L) = " + stressRatio)

% Internal Pressure
BeforePressure_H = (4*wallThickness*ElasticModWall*strainHoopBefore)/
(canDiamWill*(2-PoissonRatioWall));
BeforePressure_L = (4*wallThickness*ElasticModWall*strainLongBefore)/
(canDiamWill*(1-2*PoissonRatioWall));
disp("H Pressure Before = " + BeforePressure_H*1e-3 + " kPa")
disp("L Pressure Before = " + BeforePressure_L*1e-3 + " kPa")
Pressure_H = (4*wallThickness*ElasticModWall*strainHoopAfter)/
(canDiamWill*(2-PoissonRatioWall));
Pressure_L = (4*wallThickness*ElasticModWall*strainLongAfter)/
```

```
(canDiamWill*(1-2*PoissonRatioWall));
disp("H Pressure After = " + Pressure_H*1e-3 + " kPa")
disp("L Pressure After = " + Pressure_L*1e-3 + " kPa")

deltaP_H = abs(BeforePressure_H - Pressure_H);
deltaP_L = abs(BeforePressure_L - Pressure_L);
P0_H = (ambientPressure + deltaP_H)*1e-3;
P0_L = (ambientPressure + deltaP_L)*1e-3;

disp("Delta P (H) = " + deltaP_H*1e-3 + " kPa")
disp("Delta P (L) = " + deltaP_L*1e-3 + " kPa")
disp("Internal Pressure (H) = " + P0_H + " kPa")
disp("Internal Pressure (L) = " + P0_L + " kPa")

disp("~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~")

% stress strain curves
figure("Name"," Longitudinal Stress-Strain Curve")
plot(strainLong,stressLong,'-o')
% Graph pretty
ylim padded
xlim tight
xLab = xlabel('$\epsilon_L$','Interpreter','latex');
yLab = ylabel('$\sigma_L$ [Pa]','Interpreter','latex');
plotTitle = title('Savannah Run: Longitudinal Stress-Strain
Curve','interpreter','latex');
set(plotTitle,'FontSize',14,'FontWeight','bold')
set(gca,'FontName','Palatino Linotype')
set([xLab, yLab],'FontName','Palatino Linotype')
set(gca,'FontSize', 9)
set([xLab, yLab],'FontSize', 14)
grid on

figure('Name',"Hoop Stress-Strain Curve")
plot(strainHoop,stressHoop,'-o')
% Graph pretty
ylim padded
xlim tight
xLab = xlabel('$\epsilon_H$','Interpreter','latex');
yLab = ylabel('$\sigma_H$ [Pa]','Interpreter','latex');
plotTitle = title('Savannah Run: Hoop Stress-Strain
Curve','interpreter','latex');
set(plotTitle,'FontSize',14,'FontWeight','bold')
set(gca,'FontName','Palatino Linotype')
set([xLab, yLab],'FontName','Palatino Linotype')
set(gca,'FontSize', 9)
set([xLab, yLab],'FontSize', 14)
grid on
```

plot run 1 - Will

```
raw = readmatrix("will_run1.TXT");
A = raw(:,2);
```

```
B = raw(:,3);

% Time vectors for numeric values
ta = find(~isnan(A));
tb = find(~isnan(B));

% Extract non NaN values from raw data
A = A(~isnan(A));
B = B(~isnan(B));

% Temperature correction factor
A = A*cf;
B = B*cf;

figure()
% Hoop
p1 = plot(ta,A,'-square','LineWidth',1);
p1.Color = 'b';

hold on
% Longitudinal
p2 = plot(tb,B,'-o','LineWidth',1);
p2.Color = 'r';

% Graph pretty
ylim padded
xlim tight
xLab = xlabel('Time [s]','Interpreter','latex');
yLab = ylabel('Microstrain [$\mu \epsilon$]','Interpreter','latex');
plotTitle = title('Will: Microstrain During Aluminum Can
Opening','interpreter','latex');
set(plotTitle,'FontSize',14,'FontWeight','bold')
set(gca,'FontName','Palatino Linotype')
set([xLab, yLab],'FontName','Palatino Linotype')
set(gca,'FontSize', 9)
set([xLab, yLab],'FontSize', 14)
grid on
legend('Hoop','Longitudinal','interpreter','latex','Location', 'best')
```

Analysis from Will's Run

stress - strain curve

```
strainLong = B*1e-6;
strainHoop = A*1e-6;
stressLong = (ElasticModWall*(strainLong+PoissonRatioWall*strainHoop))/(1-
PoissonRatioWall^2);
stressHoop = (ElasticModWall*(strainHoop+PoissonRatioWall*strainLong))/(1-
PoissonRatioWall^2);

strainLongBefore = mean(strainLong(1:4));
strainLongAfter = mean(strainLong(7:end));
strainHoopBefore = mean(strainHoop(1:4));
```

```
strainHoopAfter = mean(strainHoop(7:end));

stressLongBefore = mean(stressLong(1:4));
stressLongAfter = mean(stressLong(7:end));
stressHoopBefore = mean(stressHoop(1:4));
stressHoopAfter = mean(stressHoop(7:end));

disp("Will - Run 1")
disp("Before:")
disp("L micro-strain = " + strainLongBefore*1e6)
disp("H micro-strain = " + strainHoopBefore*1e6)
disp("L stress = " + stressLongBefore*1e-6 + " MPa")
disp("H stress = " + stressHoopBefore*1e-6 + " MPa")
disp("After:")
disp("L micro-strain = " + strainLongAfter*1e6)
disp("H micro-strain = " + strainHoopAfter*1e6)
disp("L stress = " + stressLongAfter*1e-6 + " MPa")
disp("H stress = " + stressHoopAfter*1e-6 + " MPa")

figure("Name"," Longitudinal Stress-Strain Curve")
plot(strainLong,stressLong,'-o')
% Graph pretty
ylim padded
xlim tight
xLab = xlabel('$\epsilon_L$','Interpreter','latex');
yLab = ylabel('$\sigma_L$ [Pa]','Interpreter','latex');
plotTitle = title('Will Run: Longitudinal Stress-Strain
Curve','interpreter','latex');
set(plotTitle,'FontSize',14,'FontWeight','bold')
set(gca,'FontName','Palatino Linotype')
set([xLab, yLab],'FontName','Palatino Linotype')
set(gca,'FontSize', 9)
set([xLab, yLab],'FontSize', 14)
grid on

figure('Name',"Hoop Stress-Strain Curve")
plot(strainHoop,stressHoop,'-o')
% Graph pretty
ylim padded
xlim tight
xLab = xlabel('$\epsilon_H$','Interpreter','latex');
yLab = ylabel('$\sigma_H$ [Pa]','Interpreter','latex');
plotTitle = title('Will Run: Hoop Stress-Strain
Curve','interpreter','latex');
set(plotTitle,'FontSize',14,'FontWeight','bold')
set(gca,'FontName','Palatino Linotype')
set([xLab, yLab],'FontName','Palatino Linotype')
set(gca,'FontSize', 9)
set([xLab, yLab],'FontSize', 14)
grid on

% stress ratio
stressRatio = mean(stressHoop(12:end)./stressLong(12:end));
disp("Run 1 - Stress Ratio (H/L) = " + stressRatio)
```

```
% Internal Pressure
BeforePressure_H = (4*wallThickness*ElasticModWall*strainHoopBefore)/
(canDiamWill*(2-PoissonRatioWall));
BeforePressure_L = (4*wallThickness*ElasticModWall*strainLongBefore)/
(canDiamWill*(1-2*PoissonRatioWall));
disp("H Pressure Before = " + BeforePressure_H*1e-3 + " kPa")
disp("L Pressure Before = " + BeforePressure_L*1e-3 + " kPa")
Pressure_H = (4*wallThickness*ElasticModWall*strainHoopAfter)/
(canDiamWill*(2-PoissonRatioWall));
Pressure_L = (4*wallThickness*ElasticModWall*strainLongAfter)/
(canDiamWill*(1-2*PoissonRatioWall));
disp("H Pressure After = " + Pressure_H*1e-3 + " kPa")
disp("L Pressure After = " + Pressure_L*1e-3 + " kPa")

deltaP_H = abs(BeforePressure_H - Pressure_H);
deltaP_L = abs(BeforePressure_L - Pressure_L);
P0_H = (ambientPressure + deltaP_H)*1e-3;
P0_L = (ambientPressure + deltaP_L)*1e-3;

disp("Delta P (H) = " + deltaP_H*1e-3 + " kPa")
disp("Delta P (L) = " + deltaP_L*1e-3 + " kPa")
disp("Internal Pressure (H) = " + P0_H + " kPa")
disp("Internal Pressure (L) = " + P0_L + " kPa")
```

plot Justin run 3

```
raw = readmatrix("justin_run3.TXT");
A = raw(:,2);
B = raw(:,3);

% Time vectors for numeric values
ta = find(~isnan(A));
tb = find(~isnan(B));

% Extract non NaN values from raw data
A = A(~isnan(A));
B = B(~isnan(B));

% Temperature correction
A = A*cf;
B = B*cf;

figure()
% Hoop
p1 = plot(ta,A,'-square','LineWidth',1);
p1.Color = 'b';

hold on
% Longitudinal
p2 = plot(tb,B,'-o','LineWidth',1);
p2.Color = 'r';
```

```
% Graph pretty
ylim padded
xlim tight
xLab = xlabel('Time [s]','Interpreter','latex');
yLab = ylabel('Microstrain [$\mu \epsilon$]','Interpreter','latex');
plotTitle = title('Justin: Microstrain During Aluminum Can
Opening','interpreter','latex');
set(plotTitle,'FontSize',14,'FontWeight','bold')
set(gca,'FontName','Palatino Linotype')
set([xLab, yLab],'FontName','Palatino Linotype')
set(gca,'FontSize', 9)
set([xLab, yLab],'FontSize', 14)
grid on
legend('Hoop','Longitudinal','interpreter','latex','Location', 'best')
```

Analysis from Justin's Run

stress - strain curve

```
strainLong = B(1:length(A))*1e-6;
strainHoop = A*1e-6;

strainLongBefore = mean(strainLong(1:9));
strainLongAfter = mean(strainLong(10:end));
strainHoopBefore = mean(strainHoop(1:9));
strainHoopAfter = mean(strainHoop(10:end));

stressLong = (ElasticModWall*(strainLong+PoissonRatioWall*strainHoop))/(1-
PoissonRatioWall^2);
stressHoop = (ElasticModWall*(strainHoop+PoissonRatioWall*strainLong))/(1-
PoissonRatioWall^2);

stressLongBefore = mean(stressLong(1:9));
stressLongAfter = mean(stressLong(10:end));
stressHoopBefore = mean(stressHoop(1:9));
stressHoopAfter = mean(stressHoop(10:end));

disp("~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~")
disp("Justin - Run 3")
disp("Before:")
disp("L micro-strain = " + strainLongBefore*1e6)
disp("H micro-strain = " + strainHoopBefore*1e6)
disp("L stress = " + stressLongBefore*1e-6 + " MPa")
disp("H stress = " + stressHoopBefore*1e-6 + " MPa")
disp("After:")
disp("L micro-strain = " + strainLongAfter*1e6)
disp("H micro-strain = " + strainHoopAfter*1e6)
disp("L stress = " + stressLongAfter*1e-6 + " MPa")
disp("H stress = " + stressHoopAfter*1e-6 + " MPa")

figure("Name"," Longitudinal Stress-Strain Curve")
plot(strainLong,stressLong,'-o')
ylim padded
```

```
xlim tight
xLab = xlabel('$\epsilon_L$','Interpreter','latex');
yLab = ylabel('$\sigma_L$ [Pa]','Interpreter','latex');
plotTitle = title('Justin Run: Longitudinal Stress-Strain
Curve','interpreter','latex');
set(plotTitle,'FontSize',14,'FontWeight','bold')
set(gca,'FontName','Palatino Linotype')
set([xLab, yLab],'FontName','Palatino Linotype')
set(gca,'FontSize', 9)
set([xLab, yLab],'FontSize', 14)
grid on

figure('Name',"Hoop Stress-Strain Curve")
plot(strainHoop,stressHoop,'-o')
% Graph pretty
ylim padded
xlim tight
xLab = xlabel('$\epsilon_H$','Interpreter','latex');
yLab = ylabel('$\sigma_H$ [Pa]','Interpreter','latex');
plotTitle = title('Justin Run: Hoop Stress-Strain
Curve','interpreter','latex');
set(plotTitle,'FontSize',14,'FontWeight','bold')
set(gca,'FontName','Palatino Linotype')
set([xLab, yLab],'FontName','Palatino Linotype')
set(gca,'FontSize', 9)
set([xLab, yLab],'FontSize', 14)
grid on


% stress ratio
stressRatio = mean(stressHoop(10:end)./stressLong(10:end));
disp("Run 3 - Stress Ratio (H/L) = " + stressRatio)

%mean([66.09, 65.92,65.90,65.92,65.94])

% Internal Pressure
BeforePressure_H = (4*wallThickness*ElasticModWall*strainHoopBefore)/
(canDiamWill*(2-PoissonRatioWall));
BeforePressure_L = (4*wallThickness*ElasticModWall*strainLongBefore)/
(canDiamWill*(1-2*PoissonRatioWall));
disp("H Pressure Before = " + BeforePressure_H*1e-3 + " kPa")
disp("L Pressure Before = " + BeforePressure_L*1e-3 + " kPa")
Pressure_H = (4*wallThickness*ElasticModWall*strainHoopAfter)/
(canDiamWill*(2-PoissonRatioWall));
Pressure_L = (4*wallThickness*ElasticModWall*strainLongAfter)/
(canDiamWill*(1-2*PoissonRatioWall));
disp("H Pressure After = " + Pressure_H*1e-3 + " kPa")
disp("L Pressure After = " + Pressure_L*1e-3 + " kPa")

deltaP_H = abs(BeforePressure_H - Pressure_H);
deltaP_L = abs(BeforePressure_L - Pressure_L);
P0_H = (ambientPressure + deltaP_H)*1e-3;
P0_L = (ambientPressure + deltaP_L)*1e-3;
```

```
disp("Delta P (H) = " + deltaP_H*1e-3 + " kPa")
disp("Delta P (L) = " + deltaP_L*1e-3 + " kPa")
disp("Internal Pressure (H) = " + P0_H + " kPa")
disp("Internal Pressure (L) = " + P0_L + " kPa")
```

Published with MATLAB® R2023b