ZACH SKINNER

Thin-wall Pressure System
Problem Theory

\[ \sigma_1 = \sigma_h = \frac{pr}{t} \]

\[ \sigma_2 = \sigma_l = \frac{pr}{2t} \]

\[ \varepsilon_a = \varepsilon_x \cos^2 \theta_a + \varepsilon_y \sin^2 \theta_a + \gamma_{xy} \cos \theta_a \sin \theta_a \]

\[ \varepsilon_b = \varepsilon_x \cos^2 \theta_b + \varepsilon_y \sin^2 \theta_b + \gamma_{xy} \cos \theta_b \sin \theta_b \]

\[ \varepsilon_c = \varepsilon_x \cos^2 \theta_c + \varepsilon_y \sin^2 \theta_c + \gamma_{xy} \cos \theta_c \sin \theta_c \]

\[
\begin{bmatrix}
\varepsilon_a \\
\varepsilon_b \\
\varepsilon_c
\end{bmatrix} =
\begin{bmatrix}
\cos^2 \theta_a & \sin^2 \theta_a & \cos \theta_a \sin \theta_a \\
\cos^2 \theta_b & \sin^2 \theta_b & \cos \theta_b \sin \theta_b \\
\cos^2 \theta_c & \sin^2 \theta_c & \cos \theta_c \sin \theta_c
\end{bmatrix}
\begin{bmatrix}
\varepsilon_x \\
\varepsilon_y \\
\gamma_{xy}
\end{bmatrix}
\]

\[ \sigma_x = \frac{E}{1 - \nu^2} (\varepsilon_x + \nu \varepsilon_y) \]

\[ \sigma_y = \frac{E}{1 - \nu^2} (\varepsilon_y + \nu \varepsilon_x) \]

\[ \tau_{xy} = G \gamma_{xy} \]
Problem Theory

\[ \sigma_{1,2} = \frac{\sigma_x + \sigma_y}{2} \pm \sqrt{\left(\frac{\sigma_x - \sigma_y}{2}\right)^2 + \tau_{xy}^2} \]

\[ \tau_{\text{max, in-plane}} = \frac{\sigma_1 - \sigma_2}{2} \]

Mohr’s Circle:

\[ R = \sqrt{\left(\frac{\sigma_x - \sigma_y}{2}\right)^2 + \tau_{xy}^2} \]
## Problem Input and Output

### Input

<table>
<thead>
<tr>
<th>Thin Wall Stress Calculator</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Radius (in.)</td>
<td>1.75</td>
</tr>
<tr>
<td>Thickness (in.)</td>
<td>0.083</td>
</tr>
<tr>
<td>Pressure (psi)</td>
<td>250</td>
</tr>
<tr>
<td>E (psi)</td>
<td>28000000</td>
</tr>
<tr>
<td>Gage factor</td>
<td>2.05</td>
</tr>
<tr>
<td>$\theta_A$ (Degree)</td>
<td>45</td>
</tr>
<tr>
<td>$\theta_B$ (Degree)</td>
<td>0</td>
</tr>
<tr>
<td>$\theta_C$ (Degree)</td>
<td>315</td>
</tr>
<tr>
<td>$\varepsilon_A$ (Microstrain)</td>
<td>87</td>
</tr>
<tr>
<td>$\varepsilon_B$ (Microstrain)</td>
<td>33.3</td>
</tr>
<tr>
<td>$\varepsilon_C$ (Microstrain)</td>
<td>91.3</td>
</tr>
<tr>
<td>$\nu$</td>
<td>0.27</td>
</tr>
</tbody>
</table>

### Output

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$\sigma_L$ Theoretical (psi)</td>
<td>2510.5</td>
</tr>
<tr>
<td>$\sigma_H$ Theoretical (psi)</td>
<td>5021.1</td>
</tr>
<tr>
<td>$G$</td>
<td>11023622.0</td>
</tr>
<tr>
<td>$\varepsilon_X$ (Microstrain)</td>
<td>33.3</td>
</tr>
<tr>
<td>$\varepsilon_Y$ (Microstrain)</td>
<td>144.8</td>
</tr>
<tr>
<td>$\gamma_{XY}$ (Microstrain)</td>
<td>-4.0</td>
</tr>
<tr>
<td>$\sigma_L$ Measured (psi)</td>
<td>2186.2</td>
</tr>
<tr>
<td>$\sigma_H$ Measured (psi)</td>
<td>4643.7</td>
</tr>
<tr>
<td>$\tau_{XY}$ Measured (psi)</td>
<td>-43.8</td>
</tr>
<tr>
<td>$\sigma_{\text{Min}}$ (psi)</td>
<td>2185.4</td>
</tr>
<tr>
<td>$\sigma_{\text{Max}}$ (psi)</td>
<td>4644.4</td>
</tr>
<tr>
<td>$\tau_{\text{Max}}$ (psi)</td>
<td>1229.5</td>
</tr>
<tr>
<td>$\sigma_L$ Error (%)</td>
<td>7.5</td>
</tr>
<tr>
<td>$\sigma_H$ Error (%)</td>
<td>12.9</td>
</tr>
<tr>
<td>$\varepsilon_{\text{Min}}$ (Microstrain)</td>
<td>33.3</td>
</tr>
<tr>
<td>$\varepsilon_{\text{Max}}$ (Microstrain)</td>
<td>144.8</td>
</tr>
<tr>
<td>$\gamma_{\text{Max}}$ (Microstrain)</td>
<td>55.8</td>
</tr>
<tr>
<td>$\theta_P$ Stress (Degree)</td>
<td>1.0</td>
</tr>
<tr>
<td>$\theta_S$ Stress (Degree)</td>
<td>-44.0</td>
</tr>
<tr>
<td>$\theta_P$ Strain (Degree)</td>
<td>1.0</td>
</tr>
<tr>
<td>$\theta_S$ Strain (Degree)</td>
<td>-44.0</td>
</tr>
</tbody>
</table>
Problem Output

Stress Mohr's Circle

Strain Mohr's Circle
For i = 1 To 3
    angmat(i, 1) = (Cos(theta(i) * angconvert) ^ 2)
    angmat(i, 2) = (Sin(theta(i) * angconvert) ^ 2)
    angmat(i, 3) = (Cos(theta(i) * angconvert)) * (Sin(theta(i) * angconvert))
Next i

Call CramersRule(angmat(), strain(), strainxy())
Cells(17, 2) = strainxy(1) * 1000000
Cells(18, 2) = strainxy(2) * 1000000
Cells(19, 2) = strainxy(3) * 1000000

stressx = (e / (1 - v ^ 2)) * (strainxy(1) + v * strainxy(2))
Cells(20, 2) = stressx
stressy = (e / (1 - v ^ 2)) * (strainxy(2) + v * strainxy(1))
Cells(21, 2) = stressy
shstressxy = g * strainxy(3)
Cells(22, 2) = shstressxy

stressmin = ((stressx + stressy) / 2) - Sqr(((stressx - stressy) / 2) ^ 2) + (shstressxy ^ 2))
stressmax = ((stressx + stressy) / 2) + Sqr(((stressx - stressy) / 2) ^ 2) + (shstressxy ^ 2))
shstressmax = Sqr(((stressx - stressy) / 2) ^ 2) + (shstressxy ^ 2))
Cells(23, 2) = stressmin
Cells(24, 2) = stressmax
Cells(25, 2) = shstressmax
Code – Production of Mohr’s Circle

```plaintext
origin = (stressmin + stressmax) / 2
rad = (stressmax - stressmin) / 2
For i = 0 To 360
  normalstress = 0#
  shearstress = 0#
  normalstress = origin + rad * Cos(i * angconvert)
  shearstress = rad * Sin(i * angconvert)
  Cells(31 + i, 1) = i
  Cells(31 + i, 2) = normalstress
  Cells(31 + i, 3) = shearstress
Next i
```
Cramer’s Rule Subroutine

Sub CramersRule(a() As Double, b() As Double, c() As Double)
Dim i As Integer, a1(3, 3) As Double, a2(3, 3) As Double, a3(3, 3) As Double
Dim solution(3) As Single, a4(3, 3) As Double, j As Integer
Dim det1 As Double, det2 As Double, det3 As Double, det4 As Double
Dim x As Double, y As Double, z As Double

For i = 1 To 3
    For j = 1 To 3
        a2(i, j) = a(i, j)
        a3(i, j) = a(i, j)
        a4(i, j) = a(i, j)
    Next j
Next i

For i = 1 To 3
    a2(i, 1) = b(i)
    a3(i, 2) = b(i)
    a4(i, 3) = b(i)
Next i

Call determinate(a(), det1)
Call determinate(a2(), det2)
Call determinate(a3(), det3)
Call determinate(a4(), det4)

x = det2 / det1
y = det3 / det1
z = det4 / det1

c(1) = x
c(2) = y
c(3) = z

End Sub

Sub determinate(a() As Double, value As Double)
Dim term1 As Double, term2 As Double, term3 As Double
term1 = a(1, 1) * ((a(2, 2) * a(3, 3)) - (a(2, 3) * a(3, 2)))
term2 = -a(1, 2) * ((a(2, 1) * a(3, 3)) - (a(2, 3) * a(3, 1)))
term3 = a(1, 3) * ((a(2, 1) * a(3, 2)) - (a(2, 2) * a(3, 1)))
value = term1 + term2 + term3
End Sub
Zach Skinner

CIVL – 3110

Thin-Wall Pressure Tube Report

April 10, 2012

The purpose of the code is to take given data on a thin-wall pressure system to calculate principal stresses and strains, theoretical stresses, Poisson’s ratio, actual stresses and strains, orientation angles, and to produce Mohr’s circle. The code takes inputs of pressure, radius, thickness, strain gage orientation, and measured strain to produce the output. The code solves a matrix of sine and cosine terms to solve for strain values that are in turn used to calculate the measured stresses. These measured stresses are compared to theoretical calculations of the stress in the system. The measured stresses and strains are used to calculate the principal stresses and strains which are used to produce Mohr’s circle for the stress and strain of the system.

<table>
<thead>
<tr>
<th>Thin Wall Stress Calculator</th>
</tr>
</thead>
<tbody>
<tr>
<td>Radius (in.)</td>
</tr>
<tr>
<td>Thickness (in.)</td>
</tr>
<tr>
<td>Pressure (psi)</td>
</tr>
<tr>
<td>E (psi)</td>
</tr>
<tr>
<td>Gage factor</td>
</tr>
<tr>
<td>$\theta_A$ (Degree)</td>
</tr>
<tr>
<td>$\theta_B$ (Degree)</td>
</tr>
<tr>
<td>$\theta_C$ (Degree)</td>
</tr>
<tr>
<td>$\varepsilon_A$ (Microstrain)</td>
</tr>
<tr>
<td>$\varepsilon_B$ (Microstrain)</td>
</tr>
<tr>
<td>$\varepsilon_C$ (Microstrain)</td>
</tr>
<tr>
<td>$\nu$</td>
</tr>
</tbody>
</table>

Figure 1 – Project Inputs
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\sigma_L$ Theoretical (psi)</td>
<td>2510.5</td>
</tr>
<tr>
<td>$\sigma_H$ Theoretical (psi)</td>
<td>5021.1</td>
</tr>
<tr>
<td>$G$</td>
<td>11023622.0</td>
</tr>
<tr>
<td>$\varepsilon_L$ (Microstrain)</td>
<td>33.3</td>
</tr>
<tr>
<td>$\varepsilon_H$ (Microstrain)</td>
<td>144.8</td>
</tr>
<tr>
<td>$Y_{XY}$ (Microstrain)</td>
<td>-4.0</td>
</tr>
<tr>
<td>$\sigma_L$ Measured (psi)</td>
<td>2186.2</td>
</tr>
<tr>
<td>$\sigma_H$ Measured (psi)</td>
<td>4643.7</td>
</tr>
<tr>
<td>$\tau_{XY}$ Measured (psi)</td>
<td>-43.8</td>
</tr>
<tr>
<td>$\sigma_{Min}$ (psi)</td>
<td>2185.4</td>
</tr>
<tr>
<td>$\sigma_{Max}$ (psi)</td>
<td>4644.4</td>
</tr>
<tr>
<td>$\tau_{Max}$ (psi)</td>
<td>1229.5</td>
</tr>
<tr>
<td>$\sigma_L$ Error (%)</td>
<td>7.5</td>
</tr>
<tr>
<td>$\sigma_H$ Error (%)</td>
<td>12.9</td>
</tr>
<tr>
<td>$\varepsilon_{Min}$ (Microstrain)</td>
<td>33.3</td>
</tr>
<tr>
<td>$\varepsilon_{Max}$ (Microstrain)</td>
<td>144.8</td>
</tr>
<tr>
<td>$Y_{Max}$ (Microstrain)</td>
<td>55.8</td>
</tr>
<tr>
<td>$\theta_p$ Stress (Degree)</td>
<td>1.0</td>
</tr>
<tr>
<td>$\theta_q$ Stress (Degree)</td>
<td>-44.0</td>
</tr>
<tr>
<td>$\theta_p$ Strain (Degree)</td>
<td>1.0</td>
</tr>
<tr>
<td>$\theta_q$ Strain (Degree)</td>
<td>-44.0</td>
</tr>
</tbody>
</table>

**Figure 2 – Project Outputs**

![Stress Mohr's Circle](image)

**Figure 3 – Stress Mohr's Circle**
Figure 4 – Strain Mohr’s Circle

```vba
Option Explicit
Option Base 1

Private Sub CommandButton1_Click()

  Dim x As Single, t As Single, p As Single, e As Single, theta(3) As Single
  Dim strain(3) As Double, stress(3) As Double, strainmax As Double, angconvert(1, 3) As Double
  Dim ymax As Double, q As Double, strainmax(3) As Double, stresses As Double, rad As Single, origin As Single
  Dim compression As Double, shear As Double, t As Integer, qz As Single, t As Integer, angconvert(1, 3) As Double
  Dim stressmax As Double, stressmaxmax As Double, normalstress As Double, shearstress As Double
  Dim erro1 As Double, erro2 As Double, stressmin As Double, strainmaxmax As Double, shasstress As Double
  Dim stressthetap As Double, stresstheas As Double, stresstheas As Double, stressthetap As Double, stresstheas As Double
  Dim normalstress As Double, shmaxstress As Double

  angconvert = 3.14 / 180
  t = Cells(3, 2)
  p = Cells(4, 2)
  e = Cells(5, 2)
  qz = Cells(6, 2)
  theta(1) = Cells(7, 2)
  theta(2) = Cells(9, 2)
  theta(3) = Cells(9, 2)
  strain(1) = Cells(10, 2) * 0.000001
  strain(2) = Cells(11, 2) * 0.000001
  strain(3) = Cells(13, 2) * 0.000001
  v = Cells(13, 2)
  stress = (p + c) / t
  Cells(15, 2) = stress
  stress = (p + c) / (2 * t)
  Cells(14, 2) = stress
  q = e / (2 + v)
  Cells(16, 2) = q

  For i = 1 To 3
    angcon(1, 1) = (Cos(theta(i)) * angconvert) ^ 2
    angcon(1, 2) = (Sin(theta(i)) * angconvert) ^ 2
    angcon(1, 3) = (Cos(theta(i)) * angconvert) * (Sin(theta(i)) * angconvert)
  Next i

End Sub
```

Figure 5 – Project Code Part 1
```plaintext
Cell CreaseRule(angmat(), strain(), strainxy())
Cells[17, 2] = strainxy(1) * 1000000
Cells[18, 2] = strainxy(2) * 1000000
Cells[19, 2] = strainxy(3) * 1000000

stress = (ε / |1 - v^2|) * strainxy(1) + v * strainxy(2)
Cells[20, 2] = stress
stressxy = (ε / |1 - v^2|) * strainxy(2) + v * strainxy(3)
Cells[21, 2] = stressxy

atricexy = q * strainxy(3)
Cells[22, 2] = atrace

strainmin = ((stress + stressxy) / 2) - 0.5 * strainxy * (stress - stressxy) / 2 + strainxy * 2
strainmax = ((stress + stressxy) / 2) + 0.5 * strainxy * (stress - stressxy) / 2 + strainxy * 2

stratequ = 3 * ((strainxy - stressxy) / 2)^2 + strainxy * 2
Cells[23, 2] = strain
Cells[24, 2] = stress
Cells[25, 2] = atrace

strainmin = (strainxy + strainxy(2)) / 2 - 0.5 * (strainxy - strainxy(2)) / 2 + strainxy / 2
strainmax = (strainxy + strainxy(2)) / 2 + 0.5 * (strainxy - strainxy(2)) / 2 + strainxy / 2

stratequ = 3 * ((strainxy - strainxy(2)) / 2)^2 + (strainxy(2) / 2)^2
Cells[23, 4] = strain * 1000000
Cells[24, 4] = strain * 1000000
Cells[25, 4] = atrace * 1000000

origin = (strainmin + strainmax) / 2
rad = (strainmax - strainmin) / 2

For i = 0 To 360
    normalstress = 0
    shearstress = 0
    normalstress = origin + rad * Cos(i * angconvert)
    shearstress = rad * Sin(i * angconvert)
Cells[31 + i, 1] = normalstress
Cells[31 + i, 2] = shearstress
Cells[31 + i, 3] = shearstress

Next i
```

Figure 6 – Project Code Part 2

```plaintext
strainmin = strain * 1000000
strainmax = strain * 1000000
stratequ = strain * 1000000

origin = (strainmin + strainmax) / 2
rad = (strainmax - strainmin) / 2

For i = 0 To 360
    normalstrain = 0
    shearstrain = 0
    normalstrain = (origin + rad * Cos(i * angconvert))
    shearstrain = (rad * Sin(i * angconvert))
Cells[31 + i, 1] = normalstrain
Cells[31 + i, 2] = shearstrain
Cells[31 + i, 3] = shearstrain

Next i
```

Figure 7 – Project Code Part 3
Sub CramersRule(a() As Double, b() As Double, c() As Double)
Dim i As Integer, a1(3, 3) As Double, a2(3, 3) As Double, a3(3, 3) As Double
Dim solution(3) As Single, a4(3, 3) As Double, j As Integer
Dim det1 As Double, det2 As Double, det3 As Double, det4 As Double
Dim x As Double, y As Double, z As Double

For i = 1 To 3
    For j = 1 To 3
        a2(i, j) = a(i, j)
        a3(i, j) = a(i, j)
        a4(i, j) = a(i, j)
    Next j
Next i

For i = 1 To 3
    a2(i, 1) = b(i)
    a3(i, 2) = b(i)
    a4(i, 3) = b(i)
Next i

Call determinate(a(), det1)
Call determinate(a2(), det2)
Call determinate(a3(), det3)
Call determinate(a4(), det4)

x = det2 / det1
y = det3 / det1
z = det4 / det1

c(1) = x
c(2) = y
c(3) = z

End Sub

Sub determinate(a() As Double, value As Double)
Dim term1 As Double, term2 As Double, term3 As Double
term1 = a(1, 1) * ((a(2, 2) * a(3, 3)) - (a(2, 3) * a(3, 2)))
term2 = -a(1, 2) * ((a(2, 1) * a(3, 3)) - (a(2, 3) * a(3, 1)))
term3 = a(1, 3) * ((a(2, 1) * a(3, 2)) - (a(2, 2) * a(3, 1)))
value = term1 + term2 + term3
End Sub

Figure 8 – Project Code Part 4