Machine Design Homework 5

June 29, 2022

Gabe Morris

[1]:
```
# Notebook Preamble
import matplotlib.pyplot as plt
import sympy as sp

plt.style.use('maroon_ipynb.mplstyle')

def von_mises(sx_, sy_, sz_, txy_, tyz_, tzx_):
    return(1/sp.sqrt(2)*sp.sqrt((sx_ - sy_)**2 + (sy_ - sz_)**2 + (sz_ -sx_)**2␣
 ↪+ 6*(txy_**2 + tyz_**2 + tzx_**2))).n()
```

Contents

1 Problem 8-1

1.1 Given

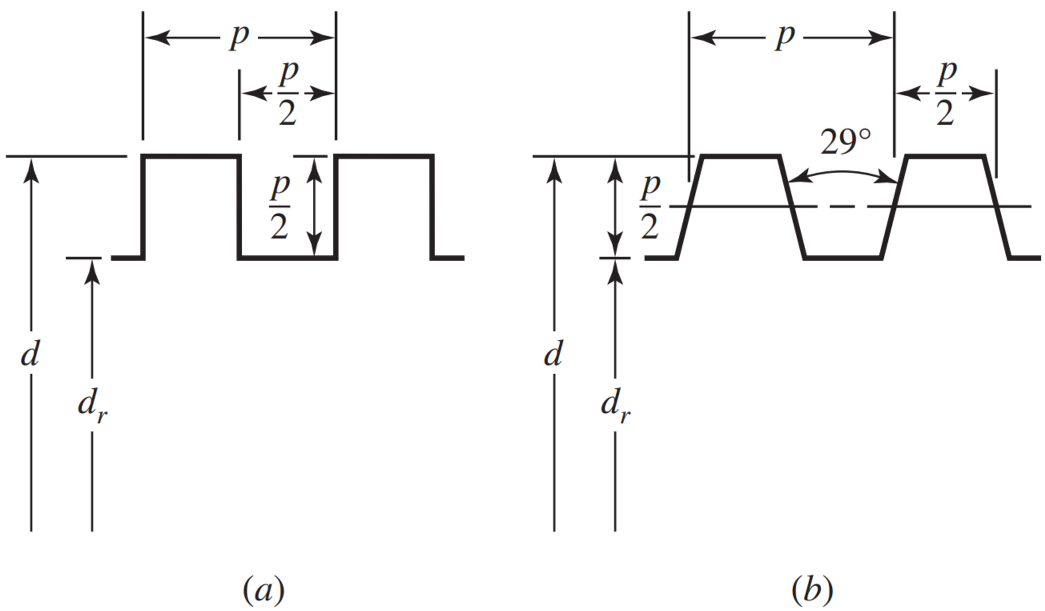
A power screw is 25 mm in diameter and has a thread pitch of 5 mm.

[2]:
```
d = sp.S(25)
p = sp.S(5)
```

1.2 Find

a. Find the thread depth, the thread width, the mean and root diameters, and the lead, provided square threads are used.
b. Repeat part (a) for Acme threads.

1.3 Solution



1.3.1 Part A

Thread Depth

From (a) in the figure above, the thread depth is half the pitch.

[3]:
```
# Thread depth
thread_depth = (p/2).n()
thread_depth  # mm
```

[3]: 2.5

Thread Width

The thread width is the same as the thread depth.

[4]:
```
thread_width = thread_depth
thread_width  # mm
```

[4]:

2.5

Mean and Root Diameters

[5]:
```
# Root diameter is the same as minor diameter
dr = d - 2*thread_depth
dr  # mm
```

[5]:

20.0

[6]:
```
# Mean diameter
dm = (d + dr)/2
dm  # mm
```

[6]:

22.5

Lead

[7]:
```
# The lead is the same as the pitch because it's single threaded
l = p
l  # mm
```

[7]:

5

1.3.2 Part B

The same procedure is done in Part B, but now we look at (b) in the above figure.

Thread Depth and Thread Width

[8]:
```
# The thread depth and thread width is still the same
thread_depth  # mm
```

[8]:

2.5

[9]:
```
thread_width  # mm
```

[9]:

2.5

Because the thread depth is the same, so will the minor and mean diameter. The lead also remains unchanged.

Mean Diameter, Root Diameter, and Lead

[10]:
```
dm  # mm
```

[10]:

22.5

[11]:
```
dr  # mm
```

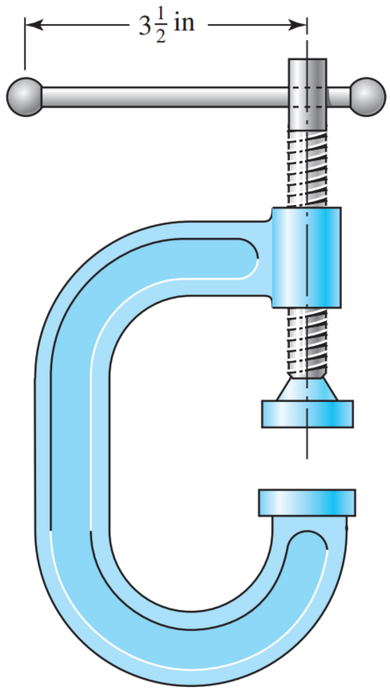
[11]: 20.0

[12]:
```
l  # mm
```

[12]: 5

2 Problem 8-7

2.1 Given



For the C clamp shown, a force is applied at the end of the $\frac{3}{8}\,in$ diameter handle. The screw is a $\frac{3}{4}\,in - 6$ Acme thread, and is 10 inches long overall, with a maximum of 8 inches possible in the clamping region. The handle and screw are both made from cold-drawn AISI 1006 steel. The coefficients of friction for the screw and the collar are 0.15. The collar, which in this case is the anvil striker's swivel joint, has a friction diameter of 1 inch. It is desired that the handle will yield before the screw will fail.

[13]:
```
# Cold Drawn Steel
Sy = sp.S(41)  # ksi
```

2.2 Find

Check this by the following steps:

a. Determine the maximum force that can be applied to the end of the handle to reach the point of yielding of the handle.
b. Using the force from part (a), determine the clamping force.
c. Using the force from part (a), determine the factor of safety for yielding at the interface of the screw body and the base of the first engaged thread, assuming the first thread carries 38% of the total clamping force.
d. Determine the factor of safety for buckling of the screw.

2.3 Solution

2.3.1 Part A

The handle is a cantilever beam.

[14]:
```
# Getting the maximum force
F = sp.Symbol('F')
M = (sp.S('3.5') - sp.S('0.375'))*F
d_handle = sp.Rational(3, 8)
I = sp.pi*d_handle**4/64
eq1 = sp.Eq((M*(d_handle/2)/I).nsimplify(), Sy)  # Force in kips
eq1
```

[14]:

$$\frac{51200F}{27\pi} = 41$$

[15]:
```
F_ = (sp.solve(eq1)[0]*1000).n()
F_  # lbf
```

[15]:

67.9246692875762

2.3.2 Part B

For a $\frac{3}{4}\,in - 6$ thread,

[16]:
```
p = l = sp.Rational(1, 6).n()
d = sp.S('0.75')
dm = d - p/2
dm # inches
```

[16]:

0.666666666666667

Equations 8-5 and 8-6 may be used to compute the force,

$$T_R = \frac{Fd_m}{2}\left(\frac{l + \pi f d_m \sec\alpha}{\pi d_m - fl\sec\alpha}\right)$$

$$T_c = \frac{Ff_c d_c}{2}$$

[17]:
```
fc = sp.S('0.15')
dc = 1
alpha = sp.rad(sp.S('14.5'))  # see Figure 8-7 (a)
sec_a = (1/sp.cos(alpha)).n()
T_ = sp.S('3.5')*F_
T = F*dm/2*((l + sp.pi*fc*dm*sec_a)/(sp.pi*dm - fc*l*sec_a)) + F*fc*dc/2
eq2 = sp.Eq(T, T_)
eq2
```

[17]:

$$0.075F + \frac{0.333333333333333F\,(0.166666666666667 + 0.103290031219769\pi)}{-0.0258225078049421 + 0.666666666666667\pi} = 237.736342506517$$

[18]:
```python
F_clamp = sp.solve(eq2)[0]
F_clamp  # lbf
```
[18]:

1542.27366292139

2.3.3 Part C

Use Eq. 8-8, 8-11, 8-7, and 8-12.

[19]:
```python
# Finding the axial stress
dr = d - p
sig_y = ((-4*F_clamp/(sp.pi*dr**2))/1000).n()
sig_y  # ksi
```
[19]:

−5.77082590952346

[20]:
```python
# Find the bending stress
sig_x = ((6*sp.S('0.38')*F_clamp/(sp.pi*dr*1*p))/1000).n()
sig_x  # ksi
```
[20]:

11.5127976894993

[21]:
```python
tau_yz = ((16*T_/(sp.pi*dr**3))/1000).n()
tau_yz  # ksi
```
[21]:

6.09979591836735

[22]:
```python
tau_zx = ((-4*sp.S('0.38')*T_/(sp.pi*dr**2*1*p))/1000).n()
tau_zx  # ksi
```
[22]:

−2.02818214285714

Now the von mises stress may be calculated.

[23]:
```python
sig_prime = von_mises(sig_x, sig_y, 0, 0, tau_yz, tau_zx)
sig_prime  # ksi
```
[23]:

18.874543508812

[24]:
```python
# Get the factor of safety
ny = Sy/sig_prime
ny
```
[24]:

2.17223796595972

2.3.4 Part D

The critical force, P_{cr}, may be found from Eq. 4-48,

$$\frac{P_{\mathrm{cr}}}{A} = S_y - \left(\frac{S_y}{2\pi}\frac{l}{k}\right)^2 \frac{1}{CE}$$

$C = 1.2$ from Table 4-2

[25]:
```
C = sp.S('1.2')
E = sp.S(30_000)  # ksi (steel)
k = dr/4
A = sp.pi/4*dr**2
P_cr = (A*(Sy - (Sy*8/(2*sp.pi*k))**2*1/(C*E))).n()
P_cr
```

[25]: 10.0061437991992

[26]:
```
# Factor of safety
P_cr*1000/F_clamp
```

[26]: 6.48791718341702