بسمه تعالی

دانشکده مهندسی مکانیک

دانشکدگان فنی

دانشگاه تهران

بهینه سازی سیستم‌های مکانیکی

تکلیف شماره یک

محمد مهدی خجسته ۸۱۰۶۹۷۲۸۰

استاد: دکتر شریعت پناهی

بهار ۱۴۰۲

۱- طبق معادله شماره یک در تصویر زیر مشاهده می‌شود، معادله جرم تیر (hollow shaft) را می‌نویسیم.

Eq. 1

Weight of hollow shaft cylinder

$$\boxed{w_h = \pi \, [\, (D_o / 2)^2 - (D_i / 2)^2 \,] \, L \, \rho}$$

For the weight of a solid shaft enter D_i = 0.00 or,

Eq. 2

Weight of solid shaft cylinder

$$w_h = \pi \, (D_o / 2)^2 \, L \, \rho$$

Where

w_h = weight kg (lbs)

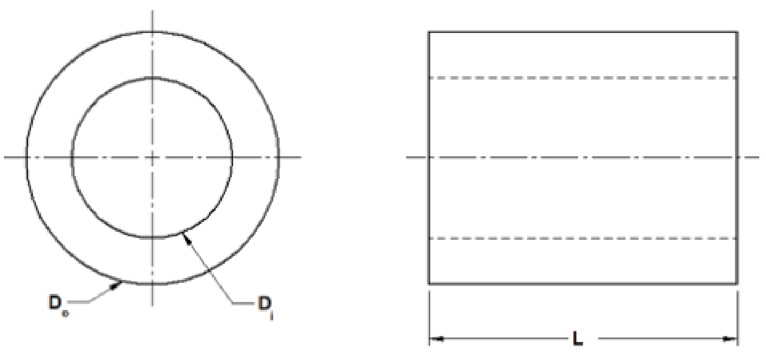
D_o = outside diameter of hollow shaft, m (in)

D_i = inside diameter of hollow shaft, m (in)

D = diameter of solid shaft, m (in)

ρ = density kg/m^3 , (lbs/in^3)

L = length of cylinder, m (in)



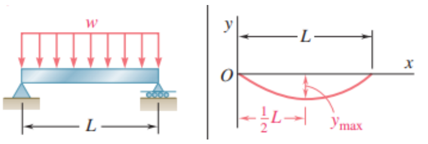
$$W_h = \pi\left[\left(\frac{D_o}{2}\right)^2 - \left(\frac{D_i}{2}\right)^2\right] L\rho \xrightarrow{\pi,\ L,\ \rho \text{ are cons.}} \left[\left(\frac{D_o}{2}\right)^2 - \left(\frac{D_i}{2}\right)^2\right]$$

حال که معادله مورد نظر مشخص شده است، به سراغ اعمال قیدهای مسئله می‌رویم. در حال حاضر می‌توانیم در مورد دو قید زیر که در صورت سوال آمده است تصمیم گیری کنیم.

- قطر خارجی از ۶۰ میلی‌متر بیشتر نشود.
- حداقل ضخامت دیواره تیر ۱ میلی‌متر باشد.

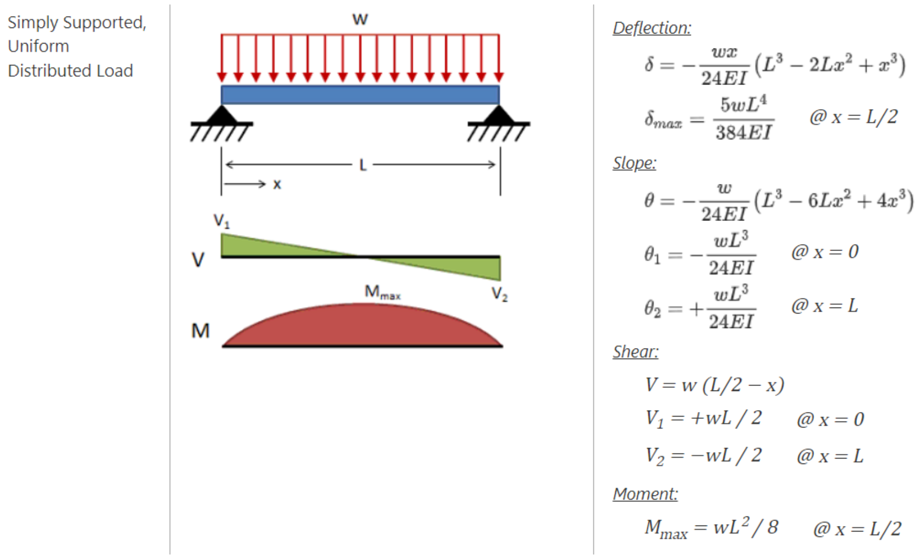
$$g1 = D_i > 0 \rightarrow -D_i < 0$$
$$g2 = D_o \leq 60\ mm \rightarrow D_o - 60 \leq 0$$
$$g3 = 1 - D_o + D_i \leq 0$$

در ادامه برای بررسی خیز و سایر پارامترهای یاد شده به سراغ جدول‌هایی که در انتهای کتاب مقاومت مصالح Beer آمده است می‌رویم. برای این نوع بارگذاری، برای صندلی مورد استفاده در فرودگاه‌ها شرایطی مانند شکل زیر را خواهیم داشت.



$-\dfrac{5wL^4}{384EI}$ | $\pm\dfrac{wL^3}{24EI}$ | $y = -\dfrac{w}{24EI}(x^4 - 2Lx^3 + L^3x)$

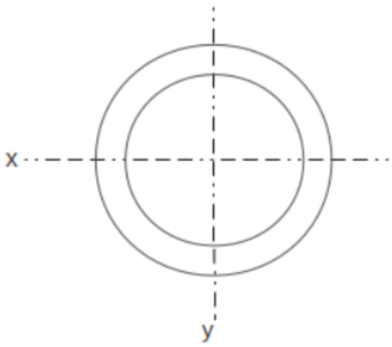
بررسی کامل معادلات حاکم بر این نوع بارگذاری در شکل زیر آمده است.



$$\delta_{\max} = \frac{5wL^4}{384EI}$$

برای Area Moment of Inertia هم مطابق شکل زیر خواهیم داشت.

Hollow Cylindrical Cross Section



The Area Moment of Inertia for a hollow cylindrical section can be calculated as

$I_x = \pi (d_o^4 - d_i^4) / 64$ *(5)*

where

d_o = cylinder outside diameter

d_i = cylinder inside diameter

$I_y = \pi (d_o^4 - d_i^4) / 64$ *(5b)*

$$\Rightarrow \delta_{\max} = \frac{5\left(\frac{500}{1000}\right)(2100)^4}{384\left(200\times10^9\times10^{-6}\right)\left(\frac{\pi\left(D_o^4 - D_i^4\right)}{64}\right)}$$

حال می‌توانیم به سراغ این قید دیگری از صورت مسئله برویم.

- خیز بیشینه تیر از ۱۰میلی‌متر تجاوز ننماید.

$$g4=\delta_{\max}\le 10\ \ mm \to \delta_{\max}-10\le 0$$

$$\to \frac{5\left(\frac{500}{1000}\right)(2100)^4}{384\left(200\times10^9\times10^{-6}\right)\left(\frac{\pi\left(D_o^4-D_i^4\right)}{64}\right)}-10\le 0$$

$$\to \frac{40516875}{\pi\left(D_o^4-D_i^4\right)}-10\le 0$$

$$\to \frac{12896921.8697}{\left(D_o^4-D_i^4\right)}-10\le 0$$

$$\to 12896921.8697-10\left(D_o^4-D_i^4\right)\le 0$$

$$\to 1289692.1869-\left(D_o^4-D_i^4\right)\le 0$$

$$\to 1289692.1869-D_o^4+D_i^4\le 0$$

تنها قید باقی مانده از صورت مسئله قید زیر می‌باشد.

- مقدار بیشینه تنش عمودی در این تیر از تنش تسلیم تجاوز ننماید.

$$g5=\sigma_{\max}\le\sigma_y \to \sigma_{\max}-240\le 0$$

$$\sigma_{\max}=\frac{M_{\max}\left(\frac{D_0}{2}\right)}{I}=\frac{\left(\frac{wL^2}{8}\right)\left(\frac{D_0}{2}\right)}{\frac{\pi\left(D_o^4-D_i^4\right)}{64}}$$

$$\to \frac{\left(\frac{wL^2}{8}\right)\left(\frac{D_0}{2}\right)}{\frac{\pi\left(D_o^4-D_i^4\right)}{64}}-240\le 0$$

$$\rightarrow \frac{4\left(wL^2\right)\left(D_0\right)}{\pi\left(D_o^4 - D_i^4\right)} - 240 \le 0$$

$$\rightarrow 4\left(wL^2\right)\left(D_0\right) - 240 \times \pi\left(D_o^4 - D_i^4\right) \le 0$$

$$\rightarrow 4\left(\frac{500}{1000} \times \left(2100\right)^2\right)\left(D_0\right) - 240 \times \pi\left(D_o^4 - D_i^4\right) \le 0$$

$$\rightarrow 11697.8883\left(D_0\right) - D_o^4 + D_i^4 \le 0$$

نکته مهم! کلیه‌ی روابط بدست آمده بر حسب قطر نوشته شده است نه بر حسب شعاع.

۲. با استفاده از کد نویسی در نرم افزار MATLAB ناحیه مجاز مساله را رسم کنید و محدب یا غیر محدب بودن آن را مشخص کنید.

HW1 - Khojasteh - 810697280.mlx

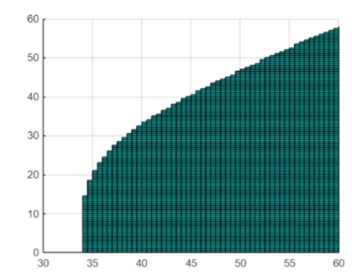
```
clc
clear all
syms D_o D_i
v = 0:0.5:100;
[D_o, D_i] = meshgrid(v);

g1 = -D_i <= 0;
g2 = D_o - 60 <= 0;
g3 = 1 + D_i - D_o <= 0;
g4 = 1289692.1869 - D_o.^4 + D_i.^4 <= 0;
g5 = 11697.8883.*D_o - D_o.^4 + D_i.^4 <= 0;

g1 = double(g1);
g2 = double(g2);
g3 = double(g3);
g4 = double(g4);
g5 = double(g5);

g1(g1==0) = NaN;
g2(g2==0) = NaN;
g3(g3==0) = NaN;
g4(g4==0) = NaN;
g5(g5==0) = NaN;

g = g1.*g2.*g3.*g4.*g5;
surf(D_o, D_i, g)
view(0, 90)
```



کدهای مربوط به هر بخش در فایل پیوست تقدیم شده است.

با توجه به شکل ناحیه مجاز می‌توان گفت که این ناحیه محدب است ولی با چک کردن خود هسین تابع در نقطه زیر از بازه می‌توان گفت که خود تابع محدب نیست، به دلیل اینکه هسین تابع مثبت نیمه معین هم نشد:

HW1 - Khojasteh - 810697280.mlx

```
clc
clear all
syms D_o D_i
L = 2100;
ro = 7800.*(10.^-9);
W_h = pi.*L.*ro.*[(D_o./2).^2 - (D_i./2).^2];
Hf = hessian(W_h);

PM1 = Hf(1,1)
```

PM1 =

$$-\frac{819\,\pi}{100000}$$

```
PM2 = det(Hf)
```

PM2 =

$$-\frac{670761\,\pi^2}{10000000000}$$

```

PM1_test = subs(PM1,[D_o,D_i],[0.05,6])
```

PM1_test =

$$-\frac{819\,\pi}{100000}$$

```
PM2_test = subs(PM2,[D_o,D_i],[0.05,6])
```

PM2_test =

$$-\frac{670761\,\pi^2}{10000000000}$$

ادامه پاسخ سوال‌ها در صفحه‌های بعد...

۳. ابتدا لاگرانژین را تشکیل داده و سپس دستگاه معادلات $\nabla L = 0$ را در نرم‌افزار MATLAB حل کنید و نقاط بحرانی را به دست آورید، سپس با بررسی شرایط بهینگی، نقطه یا نقاط کمینه را پیدا کنید.

HW1 - Khojasteh - 810697280.mlx

```
clc
clear all
syms D_o D_i u1 u2 u3 u4 u5 s1 s2 s3 s4 s5 L
L = 2100;
ro = 7800.*(10.^-9);

Lagrangian = (pi.*L.*ro.*[(D_o./2).^2 - (D_i./2).^2])+u1.*(-D_i + s1.^2)+u2.*(D_o - 60 + s2.^2)+u3.*(1 + D_i - D_o + s3.^2)+u4.*(1289692.1869 - D_o.^4 + D_i.^4 + s4.^2)+u5.*(11697.8883.*D_o - D_o.^4 + D_i.^4 + s5.^2);
```

```

eqn1 = s1.^2 >= 0;
eqn2 = s2.^2 >= 0;
eqn3 = s3.^2 >= 0;
eqn4 = s4.^2 >= 0;
eqn5 = s5.^2 >= 0;

eqn6 = diff(Lagrangian, 'D_o') == 0;
eqn7 = diff(Lagrangian, 'D_i') == 0;
eqn8 = diff(Lagrangian, 'u1') == 0;
eqn9 = diff(Lagrangian, 'u2') == 0;
eqn10 = diff(Lagrangian, 'u3') == 0;
eqn11 = diff(Lagrangian, 'u4') == 0;
eqn12 = diff(Lagrangian, 'u5') == 0;
eqn13 = diff(Lagrangian, 's1') == 0;
eqn14 = diff(Lagrangian, 's2') == 0;
eqn15 = diff(Lagrangian, 's3') == 0;
eqn16 = diff(Lagrangian, 's4') == 0;
eqn17 = diff(Lagrangian, 's5') == 0;

eqn18 = u1 >= 0;
eqn19 = u2 >= 0;
eqn20 = u3 >= 0;
eqn21 = u4 >= 0;
eqn22 = u5 >= 0;

eqns = [eqn1 eqn2 eqn3 eqn4 eqn5 eqn6 eqn7 eqn8 eqn9 eqn10 eqn11 eqn12 eqn13 eqn14 eqn15 eqn16 eqn17 eqn18 eqn19 eqn20 eqn21 eqn22];

s = solve(eqns);

D_o_min = vpa(s.D_o);
D_i_min = vpa(s.D_i);
u1_min = vpa(s.u1);
u2_min = vpa(s.u2);
u3_min = vpa(s.u3);
u4_min = vpa(s.u4);
u5_min = vpa(s.u5);
s1_min = vpa(s.s1);
s2_min = vpa(s.s2);
s3_min = vpa(s.s3);
s4_min = vpa(s.s4);
s5_min = vpa(s.s5);
```

```
Lagrangian_min = subs((pi.*L.*ro.*[(D_o./2).^2 - (D_i./2).^2])+u1.*(-D_i + s1.^2)+u2.*(D_o - 60 + s2.^2)+u3.*(1 + D_i - D_o + s3.^2)+u4.*(1289692.1869 - D_o.^4 + D_i.^4 + s4.^2)+u5.*(11697.8883.*D_o - D_o.^4 + D_i.^4 + s5.^2),{D_o,D_i,u1,u2,u3,u4,u5,s1,s2,s3,s4,s5},{D_o_min,D_i_min,u1_min,u2_min,u3_min,u4_min,u5_min,s1_min,s2_min,s3_min,s4_min,s5_min});
```

```

fprintf("D_o_min = %.8f\n",D_o_min(1))
```

```
D_o_min = 60.00000000
```

```
fprintf("D_i_min = %.8f\n",D_i_min(1))
```

```
D_i_min = 58.44812420
```

```
fprintf("Minimum of cost function = %f\n",Lagrangian_min(1))
```

```
Minimum of cost function = 2.364770
```

کدهای مربوط به هر بخش در فایل پیوست تقدیم شده است.

۴. با استفاده از کتابخانه بهینه‌سازی نرم‌افزار MATLAB، نقطه یا نقاط کمینه را بیابید. (از دستور fmincon استفاده کنید و نوع حلگر را interior point قرار دهید)

```matlab
clc
clear all
syms D_o D_i u1 u2 u3 u4 u5 s1 s2 s3 s4 s5
L = 2100;
ro = 7800.*(10.^-9);

Lagrangian = (pi.*L.*ro.*[(D_o./2).^2 - (D_i./2).^2])+u1.*(-D_i + s1.^2)+u2.*(D_o - 60 + s2.^2)+u3.*(1 + D_i - D_o + s3.^2)+u4.*(1289692.1869 - D_o.^4 + D_i.^4 + s4.^2)+u5.*(11697.8883.*D_o - D_o.^4 + D_i.^4 + s5.^2);

gradL= gradient(Lagrangian);
s= subs(gradL,[u1,u2,u3,u4,u5],[0,0,0,0,0]);
te = vpasolve(s==0);

test = te.D_o(1)
```

test = 0

```matlab
test = te.D_i(1)
```

test = 0

```matlab
test = te.s1(1)
```

test = 0

```matlab
test = te.s2(1)
```

test = −7.7459666924148337703585307995648

```matlab
test = te.s3(1)
```

test = −1.0 i

```matlab
test = te.s4(1)
```

test = −1135.6461539141494789535829008484 i

```matlab
test = te.s5(1)
```

test = 0

۵. با استفاده از کتابخانه Scipy در پایتون، مساله بهینه‌سازی را حل کنید. (نوع الگوریتم حلگر را Trust-region Constrained قرار دهید، برای محاسبه گرادیان از روش تفاضل محدود دو نقطه‌ای و برای محاسبه هسین از روش BFGS استفاده نمایید)

HW1 - Khojasteh - 810697280.py × | Release Notes: 1.77.1

D: > 1401.02 > Optimization > HW > HW1 - Khojasteh - 810697280 > HW1 - Khojasteh - 810697280.py

```python
import numpy as np
import scipy

from scipy.optimize import minimize
from scipy.optimize import NonlinearConstraint
from scipy.optimize import BFGS

L = 2100
pi = 3.141592653589793
ro = 7800*(10^-9)

def f(x):
    return [pi*L*ro*[(x[0]/2)^2 - (x[1]/2)^2]]

from scipy.optimize import Bounds
bounds = Bounds([0,0], [np.inf,np.inf])

from scipy.optimize import LinearConstraint
linear_constraint = LinearConstraint([1,-1],-np.inf, 1)

def con_f(x):
    return [1289692.1869 - x[0]^4 + x[1]^4, 11697.8883*x[0] - x[0]^4 + x[1]^4]

nonlinear_constraint = NonlinearConstraint(cons_f, -np.inf, -10, jac='2-point', hess=BFGS())

x0 = np.array([0.4,2.5])
res = minimize(f, x0, method='trust_constr', jac='2-point', hess=BFGS(),
               constraints=[linear_constraint, nonlinear_constraint],
               options={'verbose:1'}, bounds=bounds)

print(res.x)
```