Beam optimization with SciPY

joseph.morlier

EMSM-207, March 2025

1 Presentation

Here we will deal with problem 2.17 of the book of Arora (Introduction to Optimum Design).

A beam (cantilever) of rectangular section is subjected to a maximum bending moment M and to a maximum shear V. The admissible materials are respectively $sigma_a$ and tau_a. The normal stress is expressed by $sigma = 6M/bd^2$ and shear by $tau = 3V/2bd$

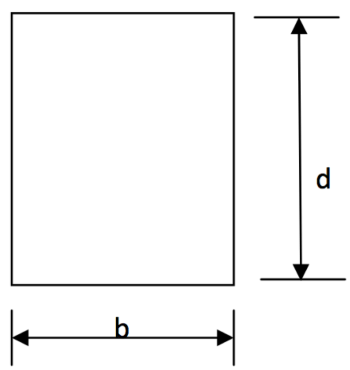

Figure 1: Variables: d height , b width

A geometric constraint that is added is that the height should not exceed 2 times the width.

- Formulate the problem. **please try to normalize the constraints**
- Visualize the domain (objective function and contrainsts) on $[b, d] = [0 : 300, 0 : 300]$. **! Units, b and d in [mm]**.
- Mass minimization or cross section minimization as , L is constant?

  $M = 140kN.m$, $V = 24kN$, $\sigma_a = 165MPa$, $\tau_a = 50MPa$

2 Data structuration

You shoud use *SciPy's minimize*. First:

Create a vector function (x aggregates the 2 variables b and d, the famous **objective function**)

```python
# Objective function (replace this with your own)
def objfun(x):
    b = x[0]
    d = x[1]
    return ? # Objective function
```

Then **constraints** for inequalities constraints.

```python
    # Nonlinear constraints function
def confun(x):
    b = x[0]
    d = x[1]
    c = [
        ?, #positive null form
        ? #positive null form
    ]
    return c

# Linear constraints
def linear_constraint(x):
    return ?  #positive null form
```

Of course you'll need a main **to solve this problem**

```python
    # Optimization PLAY WITH method, tol, maxiter
bounds = [(Lb[0], Ub[0]), (Lb[1], Ub[1])]
options = {'disp': True, 'maxiter': 1000}
result = minimize(objfun, x0, method='SLSQP', tol=1e-6, bounds=bounds, constraints=[
    {'type': 'ineq', 'fun': confun},
    {'type': 'ineq', 'fun': linear_constraint}
], options=options)

optimal_values = result.x
optimal_area = objfun(optimal_values)

print(f"Optimal design: b = {optimal_values[0]} mm, d = {optimal_values[1]} mm, surface = {optimal_area} mm^2")
```

QUESTIONS

Q0: Have a look to SLSQP documentation (Sequential least square programming: equality and inequality constraints)

Q1: What is the optimum?

Q2: What are the active constraints?

Q3: Is there a sensitivity to x_0 ? Try to program a "multistart", google it

Q4: List all the options of *SLSQP*?

Q5: Please give to *SLSPQ*, the analytical gradients (objectives and constraints?). Is it Faster?

Check all available methods

Tips and Tricks

3 Possible extension

- Use different optimizers and compare the results
- Optimizing the Thickness Distribution of a Cantilever Beam at fixed L

4 Simply with sciPy

Please check the notebook !!