Stability of Coal Pillar

Pradeep Kumar S

March 15, 2025

1 Introduction

The purpose of this project is to develop a Python script that calculates the pillar strength and the Factor of Safety (FoS) for underground mining applications. In mining design, it is essential to ensure that coal pillars can safely support the overburden. The Factor of Safety is calculated by comparing the pillar strength to the stress acting on the pillar. This script offers flexibility by allowing users to choose between two strength formulas and by considering different pillar geometries (square or rectangular). The script also incorporates tributary area theory to estimate the pillar stress.

2 Objectives

- **Pillar Geometry Selection:** Allow the user to specify if the pillar is square or rectangular. For a square pillar, only the width is needed, while for a rectangular pillar, both width and length are required.
- **Strength Calculation:** Offer two formulas for calculating pillar strength:
  - **Formula 1:**
  $$S_p = 0.27\,\sigma_c\,h^{-0.36} + \left(1 + \frac{H}{250}\right)\left(\frac{w}{h} - 1\right)$$
  where σ_c is the Uniaxial Compressive Strength (UCS) in MPa, h is the pillar height, w is the pillar effective width, and H is the overburden depth.
  - **Formula 2 :**
  $$S_p = K\left(\frac{w}{h}\right)^{\alpha}(h)^{\beta}$$
  In this example, $K = 260$ psi (converted to MPa), $\alpha = 0.46$, and $\beta = 0.66$.
- **Stress Calculation Using Tributary Area Theory:** The pillar stress is calculated by considering the overburden pressure and the tributary area:
  $$\sigma_{\text{pillar}} = 0.025 \times H \times \frac{(\text{centre-to-centre distance})}{(\text{pillar area})}$$
  For a square pillar, the pillar area is w^2; for a rectangular pillar, it is $w \times l$. The centre-to-centre distance is computed using the gallery width (w_g) as an offset.

- **Factor of Safety (FoS):** The Factor of Safety is determined by the ratio:

$$\text{FoS} = \frac{S_p}{\sigma_{\text{pillar}}}$$

3 Methodology

The script follows these steps:

1. **User Input for Pillar Geometry:** The user is asked whether the pillar is *square* or *rectangular*. Based on the input, the script collects the necessary dimensions:
   - *Square:* Pillar width (w).
   - *Rectangular:* Pillar width (w) and length (l).

   The gallery width (w_g) is also input, which is used to calculate the centre-to-centre spacing.
2. **Strength Formula Selection:** The user chooses between the two strength formulas:
   - *Formula 1:* Requires σ_c (UCS) and geometry.
   - *Formula 2:* Uses fixed constants for Indian coal
3. **Input Collection for Common Variables:** Pillar height (h) and overburden depth (H) are entered.
4. **Calculations:**
   - *Pillar Area:* Determined based on the geometry (square or rectangular).
   - *Centre-to-Centre Distance:* Calculated as:

   $$\text{square: } (w + \frac{w_g}{2})^2, \quad \text{rectangular: } (w + \frac{w_g}{2})(l + \frac{w_g}{2}).$$

   - *Pillar Strength (S_p):* Computed using the chosen strength formula.
   - *Pillar Stress (σ_{pillar}):* Using tributary area theory:

   $$\sigma_{\text{pillar}} = 0.025 \times H \times \left(\frac{\text{centre-to-centre}}{\text{pillar area}} \right).$$

   - *Factor of Safety:*

   $$\text{FoS} = \frac{S_p}{\sigma_{\text{pillar}}}.$$
5. **Output:** The script prints the computed pillar strength, stress, and Factor of Safety.

4 Final Python Code

Below is the final Python script. To run it, save the code (e.g., `pillar_fos.py`) and execute with:

```
python pillar_fos.py
```

```python
def calculate_pillar_strength_and_FOS():
    """
    This script asks:
      1. Which pillar strength formula to use.
      2. Whether the pillar is square or rectangular.
    Then it collects the necessary inputs:
      - For both formulas: pillar height (h), overburden depth (H), and center-to-center pillar spacing (s).
      - For Formula #1: UCS (sigma_c) and pillar effective width (w).
      - For Formula #2: pillar effective width (w) is used in the formula.
      - For pillar geometry: if square, pillar width (w) is used; if rectangular, both width (w) and length (l) are input.
    It then computes:
      1. Pillar Strength (S_p) using the chosen formula.
      2. Pillar Stress using Tributary Area Theory.
      3. Factor of Safety = S_p / (pillar stress).
    """

    # First, ask the user about the pillar shape.
    print("Is the pillar square or rectangular?")
    pillar_shape = input("Enter 'square' or 'rectangular': ").strip().lower()
    w_g = float(input("Enter the width of gallery in meters: "))

    if pillar_shape == "square":
        # For a square pillar, ask for the width.
        w = float(input("Enter the pillar width (w) in meters: "))
        pillar_area = w ** 2
        centre_to_centre = (w + w_g / 2) ** 2
    elif pillar_shape == "rectangular":
        # For a rectangular pillar, ask for both width and length.
        w = float(input("Enter the pillar effective width (w) in meters (used in strength formula): "))
        l = float(input("Enter the pillar length (l) in meters: "))
        pillar_area = w * l
        centre_to_centre = (w + w_g / 2) * (l + w_g / 2)
    else:
        print("Invalid pillar shape. Please run the program again and enter 'square' or 'rectangular'.")
        return

    # Next, choose the strength formula.
```

```
    print("\nChoose the formula for pillar strength calculation:")
    print("1) Formula 1:")
    print("   S_p = 0.27 * sigma_c * h^(-0.36) + (1 + H/250)*((w/h) - 1), [MPa]")
    print("2) Formula 2:")
    print("   S_p = K * (w/h)^alpha * (h)^beta,  [MPa after conversion]")
    formula_choice = input("Enter '1' or '2': ").strip()

    # Common inputs for both formulas.
    h = float(input("Enter pillar height (h) in meters: "))
    H = float(input("Enter overburden depth (H) in meters: "))

    if formula_choice == '1':
        # ------------------------------
        # FORMULA #1
        # S_p = 0.27 * sigma_c * h^(-0.36) + (1 + H/250) * ((w/h) - 1)
        # ------------------------------
        sigma_c = float(input("Enter UCS (sigma_c) in MPa: "))

        part1 = 0.27 * sigma_c * (h ** -0.36)
        part2 = (1 + (H / 250.0)) * ((w / h) - 1.0)
        S_p = part1 + part2  # Pillar strength in MPa

    elif formula_choice == '2':
        # ------------------------------
        # FORMULA #2 (Indian coal example)
        # S_p = K * (w/h)^alpha * (h)^beta
        # Using Jharia coalfield constants: K=260 psi, alpha=0.46, beta=0.66
        # Convert K from psi to MPa: 1 psi  0.00689476 MPa
        # ------------------------------
        K_psi = 260.0
        K_MPa = K_psi * 0.00689476  # ~1.7926 MPa
        alpha = 0.46
        beta = 0.66

        S_p = K_MPa * ((w / h) ** alpha) * (h ** beta)
    else:
        print("Invalid formula choice! Please run again and enter '1' or '2'.")
        return

    # Compute stress using Tributary Area Theory:
    # Overburden stress gradient = 0.025 MPa/m is assumed.
    # Pillar stress = (0.025 * H) * (tributary area / pillar cross-sectional area)
    stress_pillar = 0.025 * H * (centre_to_centre / pillar_area)  # MPa

    # Factor of Safety (FoS)
```

```python
    fos = S_p / stress_pillar if stress_pillar != 0 else 0

    # Print results
    print("\n--- RESULTS ---")
    print(f"Pillar Strength (S_p):  {S_p:.4f} MPa")
    print(f"Pillar Stress:          {stress_pillar:.4f} MPa")
    print(f"Factor of Safety (FoS): {fos:.4f}")


if __name__ == "__main__":
    calculate_pillar_strength_and_FOS()
```

5 Discussion

Input Validation and Flexibility: The script first validates the pillar shape and then dynamically adjusts input requirements. This ensures that the code can handle both square and rectangular pillars.

Tributary Area Calculation: The pillar stress is computed as:

$$\sigma_{\text{pillar}} = 0.025 \times H \times \frac{(\text{centre-to-centre})}{(\text{pillar area})}.$$

In this code, the centre-to-centre distance accounts for the gallery width.

Strength Formulas: Two formulas are provided to cater to different scenarios. Formula 1 is more generic and relies on the UCS and geometry, while Formula 2 uses empirically derived constants applicable to certain coalfields.

Safety Considerations: By computing the Factor of Safety, the script helps in evaluating whether the designed pillar is adequate to support the expected loads. A FoS greater than 1 generally indicates a safe design.

6 Conclusion

This Python script is a practical tool for calculating pillar strength and Factor of Safety in mining applications. It combines engineering formulas with tributary area theory to provide a comprehensive analysis of pillar stability. By allowing the user to choose between different formulas and pillar geometries, the tool offers flexibility and can be adapted for various site-specific conditions.

Github Repository: Github