Design of a Bolted Lap Joint as per IS 800:2007

Abhinash Roy

April 6, 2025

1 Introduction

In structural engineering, bolted lap joints are commonly used to connect two plates in various load-bearing applications. The design of these joints must adhere to relevant standards and ensure that the connection is both efficient and safe. This report details the design of a bolted lap joint as per IS 800:2007, which provides guidelines for the general construction of steel structures.

2 Problem Statement

The goal of this project is to design a bolted lap joint that connects two plates of specific thicknesses and width, subjected to a known tensile force. The design should be performed in accordance with IS 800:2007 standards, focusing on selecting appropriate bolt diameters, bolt grades, and plate grades, while ensuring that the joint is both efficient and meets safety requirements.

The design problem is defined with the following parameters:

- **Plate Width** (w): Width of the plates in mm.
- **Plate Thicknesses** (t_1, t_2): Thickness of the two plates in mm.
- **Tensile Force** (P): Applied tensile force in kN.

2.1 Objective

Develop an algorithm to design a bolted lap joint that meets the following requirements:

- Select a bolt diameter d from a list of available bolts: $\{10, 12, 16, 20, 24\}$ mm.
- Choose a bolt grade GB from a list of available grades: $\{3.6, 4.6, 4.8, 5.6, 5.8\}$.
- Choose a steel plate grade GP from a list of available grades: {"E250", "E275", "E300", "E350", "E410"}.
- Calculate the yield strength and ultimate strength of the plate and bolt based on their grades.
- Find the most efficient connection with the least number of bolts, ensuring more than two bolts are used.
- Calculate pitch, gauge, end, and edge distances.
- Ensure the utilization ratio is close to 1.
- The length of the connection should be minimal.
- Detail distances should be in round figures as far as possible.
- The strength of the connection should exceed the tensile strength of the plate.
- Ensure the design complies with IS 800:2007 standards.

3 Methodology

The design process involves several steps to ensure the lap joint meets all requirements:

1. **Input Parameters**: Receive the values for the tensile force, plate width, and thicknesses.
2. **Material Selection**: Choose appropriate initial bolt and plate grades, and determine their mechanical properties.
3. **Bolt Strength Calculation**: Compute the ultimate tensile strength and yield strength of the bolts.
4. **Design Calculation**: Determine the number of bolts required and the corresponding distances based on standard practices.

5. **Check Compliance**: Ensure that the designed joint meets all IS 800:2007 requirements and the utilization ratio is close to 1.

6. **Optimize Design**: Select the design with the minimal number of bolts while ensuring efficiency and safety.

4 Design Calculations

The calculations are performed using the following equations and considerations:

4.1 Input Parameters:

- Plate thickness t (in mm)
- Plate width w (in mm)
- Tensile force P (in kN)

Initial choices

- Bolt diameter d from dlist
- Bolt grade GB from GBlist
- Plate grade GB from GPlist

4.2 Bolt Strength

The strength of the bolt is calculated based on the chosen grade. For a given bolt grade, the ultimate tensile strength (f_u) and yield strength (f_y) are calculated as follows:

$$f_u = 100 \times \text{Grade}$$
$$f_y = (\text{Grade} - \text{int}(\text{Grade})) \times f_u$$

4.3 Shear Strength of the Bolt

The shear strength of a bolt is calculated using the formula:

$$V_b = \text{Shear Capacity}(f_y, A_b, A_b, 0, 0, \text{Field})$$

4.4 Number of Bolts

- Calculate the number of bolts required to carry the tensile force P.
- Ensure the number of bolts is more than 2.

The required number of bolts is determined by:

$$N_b = \lceil \frac{P}{V_b \times 0.75} \rceil$$

4.5 Design Distances

- Determine pitch, gauge, end, and edge distances.
- Ensure distances are in round figures.

The pitch (p), gauge (g), end distance (e), and edge distance (e) are calculated as follows:

4.6 Length of the Connection

- Minimize the length of the connection while satisfying strength and detailing requirements.

The length of the connection is given by:

$$\text{Length of Connection} = w + 2 \times e$$

4.7 Bearing Strength

The bearing strength of the bolt is calculated as:

$$V_{dpb} = \text{Bearing Capacity}(f_u, t_1 + t_2, d, e, p, \text{Standard}, \text{Field})$$

4.8 Efficiency of Connection

- Check that the connection strength is greater than the tensile strength of the plate.
- Ensure compliance with IS800:2007 standards.

The efficiency of the connection is evaluated as:

$$\text{Utilization Ratio} = \frac{P_N}{N_b \times V_b \times 0.75}$$

5 Expected Outcomes

The final design should include:

- **Bolt Diameter** (d)
- **Bolt Grade** (GB)
- **Number of Bolts** (N_b)
- **Pitch Distance** (p)
- **Gauge Distance** (g)
- **End Distance** (e)
- **Edge Distance** (e')
- **Number of Rows of Bolts**
- **Number of Columns of Bolts** (N_b columns)
- **Diameter of Hole** (d_h)
- **Strength of Connection**
- **Yield Strength of Plates 1 and 2** (f_y)
- **Length of Connection**
- **Efficiency of Connection** (Utilization Ratio)

6 Flowchart

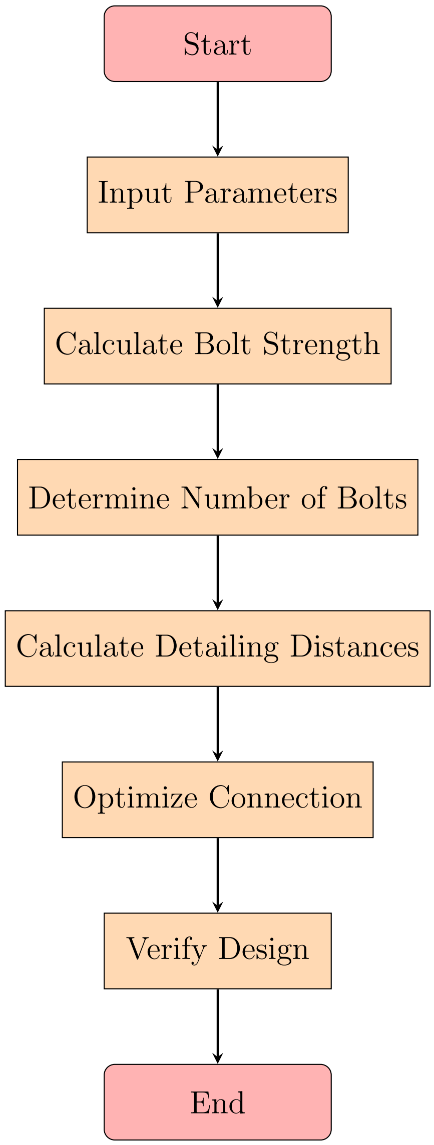

Figure 1: Flowchart of the Design Procedure

7 Solution in Python

```
import math
def design_lap_joint(P, w, t1, t2):
    """
    Design a bolted lap joint connecting two plates.
```

```
    :param P: Tensile force in kN
    :param w: Width of the plates in mm
    :param t1: Thickness of plate 1 in mm
    :param t2: Thickness of plate 2 in mm
    :return: Dictionary of design parameters and results
    """

    # Convert tensile force to Newtons
    P_N = P * 1000

    # Available data
    d_list = [10, 12, 16, 20, 24]  # Bolt diameters in mm
    GB_list = [3.6, 4.6, 4.8, 5.6, 5.8]  # Bolt grades
    GP_list = ["E250", "E275", "E300", "E350", "E410"]  # Plate grades

    # Define a mapping from plate grade to yield and ultimate strength
    plate_grades = {
        "E250": (250, 410),
        "E275": (275, 440),
        "E300": (300, 470),
        "E350": (350, 510),
        "E410": (410, 550)
    }

    # Select the best plate grade based on the given thicknesses
    plate_grade = GP_list[-1]  # Choose the highest grade for the design
    fy_plate, fu_plate = plate_grades[plate_grade]  # Get the yield and ultimate strengths for the chosen grade

    # Initialize variables to store the best design
    best_design = None
    min_length = float('inf')

    for d in d_list:
        for GB in GB_list:
            # Calculate the bolt strength
            bolt_fu, bolt_fy = calculate_bolt_strength(GB)

            # Calculate the shear strength of one bolt
            A_bolt = math.pi * (d / 2) ** 2  # Cross-sectional area of the bolt
            V_b = 0.6 * bolt_fy * A_bolt / 1.25  # Shear capacity (as per IS 800:2007)

            # Calculate the required number of bolts
```

```
        N_b = math.ceil(P_N / (V_b * 0.75))  # Using a safety factor of 1.33

        if N_b <= 2:
            continue  # Skip if the number of bolts is less than 3

        # Calculate distances
        e = d + 5  # End distance (typically 5 mm larger than bolt diameter)
        p = d + 10  # Pitch distance (typically 10 mm larger than bolt diameter)
        g = w / 2  # Gauge distance (for simplicity, use half of the plate width)

        # Calculate the length of the connection
        length_of_connection = w + 2 * e

        # Calculate the bearing strength of the bolt
        V_dpb = 2.5 * d * (t1 + t2) * fu_plate / 1.25  # Bearing capacity (as per IS 800:2007)

        # Calculate the efficiency of the connection
        Utilization_ratio = P_N / (N_b * V_b * 0.75)  # Using a safety factor of 1.33

        # Check if this design is better
        if Utilization_ratio <= 1 and length_of_connection < min_length:
            min_length = length_of_connection
            best_design = {
                "bolt_diameter": d,
                "bolt_grade": GB,
                "number_of_bolts": N_b,
                "pitch_distance": p,
                "gauge_distance": g,
                "end_distance": e,
                "edge_distance": e,
                "number_of_rows": 1,  # Simple design assumption, can be improved
                "number_of_columns": N_b,  # One column for simplicity
                "hole_diameter": d + 2,  # Diameter of hole is slightly larger than the bolt
                "strength_of_connection": N_b * V_b * 0.75,  # Strength based on shear capacity
                "yield_strength_plate_1": fy_plate,
                "yield_strength_plate_2": fy_plate,
                "length_of_connection":
```

```
length_of_connection,
                    "efficiency_of_connection": Utilization_ratio
                }

    if best_design is None:
        raise ValueError("No suitable design found that meets the requirements.")

    return best_design


def calculate_bolt_strength(bolt_grade):
    """
    Calculate the ultimate tensile strength and yield strength of the bolt based on its grade.
    :param bolt_grade: Bolt grade (e.g., 4.6, 5.6)
    :return: List containing [ultimate tensile strength, yield strength] of the bolt
    """
    bolt_fu = float(int(bolt_grade) * 100)  # Ultimate tensile strength (MPa)
    bolt_fy = float((bolt_grade - int(bolt_grade)) * bolt_fu)  # Yield strength (MPa)
    return [bolt_fu, bolt_fy]


def analyze_beam(L, W1, W2, x):
    """
    Analyze shear force and bending moment for a simply supported beam under moving loads W1 and W2.
    :param L: Total length of the beam (m)
    :param W1: Point load 1 (kN)
    :param W2: Point load 2 (kN)
    :param x: Distance between W1 and W2 (m)
    :return: Dictionary of results
    """
    dx = 0.1  # Step size for moving load simulation
    positions = [i for i in frange(0, L - x, dx)]  # Positions of W1

    max_reaction_A = 0
    max_reaction_B = 0
    BM_01 = 0
    SF_01 = 0
    SF_max = float('-inf')
    BM_max = float('-inf')
    SF_max_location = 0
    BM_max_location = 0
```

```python

    for p in positions:
        if p + x > L:
            continue

        # Reactions at supports
        RA = ((L - p) * W1 + (L - (p + x)) * W2) / L
        RB = (p * W1 + (p + x) * W2) / L

        max_reaction_A = max(max_reaction_A, RA)
        max_reaction_B = max(max_reaction_B, RB)

        if p == 0:
            BM_01 = W1 * (L - p) * p / L

        if abs((p + x / 2) - L / 2) < dx:
            SF_01 = RA

        z = L / 2
        BM = 0
        for load, load_pos in [(W1, p), (W2, p + x)]:
            if load_pos <= z:
                BM += load * (L - load_pos) * load_pos / L
        if BM > BM_max:
            BM_max = BM
            BM_max_location = z

        SF = RA - (W1 if p <= z else 0) - (W2 if p + x <= z else 0)
        if abs((p + x / 2) - z) < dx and SF > SF_max:
            SF_max = SF
            SF_max_location = z

    return {
        "Max Reaction at A": round(max_reaction_A, 2),
        "Max Reaction at B": round(max_reaction_B, 2),
        "BM_01": round(BM_01, 2),
        "SF_01": round(SF_01, 2),
        "SF_max": round(SF_max, 2),
        "SF_max_location": round(SF_max_location, 2),
        "BM_max": round(BM_max, 2),
        "BM_max_location": round(BM_max_location, 2),
    }

def frange(start, stop, step):
    """Floating point range generator"""
    while start <= stop:
        yield round(start, 4)
        start += step
```

```python


# Example usage
if __name__ == "__main__":
    P = 100  # Tensile force in kN
    w = 150  # Width of the plates in mm
    t1 = 10  # Thickness of plate 1 in mm
    t2 = 12  # Thickness of plate 2 in mm

    design = design_lap_joint(P, w, t1, t2)
    for key, value in design.items():
        print(f"{key}: {value}")

    L = 10  # Beam length in meters
    W1 = 5  # Load W1 in kN
    W2 = 3  # Load W2 in kN
    x = 2   # Distance between W1 and W2 in meters

    results = analyze_beam(L, W1, W2, x)
    for key, value in results.items():
        print(f"{key}: {value}")
```

8 Conclusion

This report outlines the design of a bolted lap joint for two plates subjected to a tensile force, as per IS 800:2007. The design process includes material selection, strength calculations, and optimization to meet safety and efficiency standards. The designed joint will be evaluated based on various parameters to ensure it fulfills all design requirements.

9 References

1. IS 800:2007: General Construction in Steel – Code of Practice
2. IS 2062:2011: Steel for General Structural Purposes
3. *IS 800:2007 Code of Practice for General Construction in Steel – Code of Practice*
4. *IS 2062:2011 Steel for General Structural Purposes*