

# Boundary Condition Application Using Finite Element Method

Paulo Henrique Brito de Souza

December 5, 2023

## **Contents**

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Code</b>	<b>3</b>

## 1 Introduction

This article will explore the application of the finite element method (FEM) to solve boundary value problems involving beam deflection. The article will compare the FEM solution with the analytical solution and discuss the advantages and limitations of each method. It will be presented an example of a beam subjected to a concentrated load at the free end, and show how to obtain the deflection distribution using both FEM and analytical solutions.

## 2 Code

This is a MATLAB function named ‘KFTransform’. It takes three inputs: ‘Neu’, ‘Dir’, and ‘K’. The function initializes a zero matrix ‘k’ of the same size as ‘Dir’ and ‘K’. It then enters a loop over the size of ‘Dir’. If the ‘Dir’th index of ‘Neu’ is not zero, it throws an error. Otherwise, it modifies the ‘k’ and ‘K’ matrices. Finally, it calculates ‘Neumod’ by subtracting the product of ‘k’ and the second column of ‘Dir’ from ‘Neu’, and assigns ‘K’ to ‘Kmod’. The function returns ‘Kmod’ and ‘Neumod’.

Here’s a brief explanation of what each line does:

- ‘k = zeros(size(Dir, 1), size(K, 2));’: Initializes ‘k’ as a zero matrix with the same number of rows as ‘Dir’ and the same number of columns as ‘K’.
- ‘for i = 1:size(Dir, 1)’: Starts a loop that iterates over each row of ‘Dir’.
- ‘if Neu(Dir(i, 1)) = 0’: Checks if the ‘Dir(i, 1)’th element of ‘Neu’ is not zero.
  - ‘error(‘Make sure that there are not two types of BC at the same node.’)’: If the above condition is true, it throws an error.
- ‘k(i, :) = K(Dir(i, 1), :);’: Assigns the ‘Dir(i, 1)’th row of ‘K’ to the ‘i’th row of ‘k’.
- ‘K(Dir(i, 1),:) = 0;’: Sets the ‘Dir(i, 1)’th row of ‘K’ to zero.
- ‘K(:, Dir(i, 1)) = 0;’: Sets the ‘Dir(i, 1)’th column of ‘K’ to zero.
- ‘K(Dir(i, 1), Dir(i, 1)) = 1;’: Sets the ‘Dir(i, 1)’th diagonal element of ‘K’ to one.
- ‘Neumod = (Neu - k'\*Dir(:, 2));’: Calculates ‘Neumod’ by subtracting the product of the transpose of ‘k’ and the second column of ‘Dir’ from ‘Neu’.
- ‘Kmod = K;’: Assigns ‘K’ to ‘Kmod’.