



## AN OBJECT ORIENTED APPROACH TO BEAM ANALYSIS

M. S. Islam, G. M. Khalil and N. A. Nahian

Department of Naval Architecture and Marine Engineering,  
Bangladesh University of Engineering and Technology, Dhaka-1000  
E-mail: shahid113@gmail.com, nishat.wms@gmail.com

### ABSTRACT

*An Object-Oriented Programming (OOP) procedure is presented for solving structural mechanics 'beam' problems by using the deflection, bending moment and shear force equations. The program is developed for a single span and for cantilever and simply supported beam. A beam is divided into one hundred segments, i.e., one hundred and one nodes and deflection, bending moment and shear force are calculated on each node for applied loads and moments.*

**Key words:** Beam analysis, C++, Object oriented programming, shear force, bending moment, deflection.

### 1. INTRODUCTION

'Beam Analysis' is a very important and fundamental topic of the subject 'Mechanics of Materials' studied by engineering students. The behaviors of a beam are represented by bending moment, shear force and deflection curves. A beam can have supports of fixed, hinged, simple or of combined type and is classified accordingly. The topic 'Beam Analysis' informs us how a beam behaves under different loading conditions and supports. The main objective of this research is the development of a reliable and user friendly program in the field of beam analysis. By using this program, an engineer or an engineering student alike can easily generate his own answers to basic questions concerning the behavior of beams. This program will help them understand the subject better and verify/correct their approach to the problems. The program is developed in C++ language. The method uses object oriented approach and superposition technique. The program takes the value and direction of loads and moments applied to the particular type of beam as an input. The total length of the beam is divided into one hundred parts by one hundred and one nodes. The program will give the nodal values of deflection, shear force and bending moment using corresponding equations given in section 2 of this paper. These nodal values can be used to draw accurate curves.

### 2. RELEVANT EQUATIONS FOR DEVELOPING THE PROGRAM

The deflection equations<sup>1</sup> for the different loading conditions are listed below. The shear force and bending equations are the second and third derivatives respectively of the deflection equations. The

derivations of these equations (deflection, shear force and bending moment) are available in Pytel and Singer (2005). Downward load is considered as negative and upward load as positive. Clockwise moment is considered negative while the anti-clockwise moment is taken as positive. The fundamental properties of a beam are  $E$  (modulus of elasticity),  $I$  (moment of inertia),  $L$  (length of the beam),  $W$  (point load) and  $w$  (uniformly distributed load per unit length).

#### 2.1 Cantilever beam

Loading condition 1: Point load at any point on the beam (Fig. 1)

Where  $a$  is the position of load from the fixed end,  $b$  is the position of load from the free end and  $x$  is the variable distance from the fixed end.

When the variable distance  $x$  is less than or equal to  $a$ , the deflection equation is

$$y = \frac{Wx^2(3a-x)}{6EI} \quad (1)$$

The bending moment equation is the second derivative of the deflection equation (1), which is

$$EI \frac{d^2y}{dx^2} = -W(a-x) \quad (2)$$

Finally the shear force equation is the third derivative of the deflection equation (1) and the first derivative of the equation (2),

$$EI \frac{d^3y}{dx^3} = -W \quad (3)$$

When the variable distance  $x$  is greater than  $a$ , the deflection equation is

$$y = \frac{Wa^2(3x - a)}{6EI} \quad (4)$$

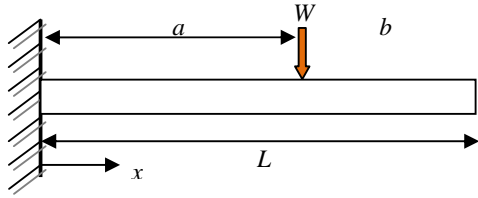


Fig. 1: Point load at any point on the beam

Loading condition 2: Uniformly distributed load (Fig. 2)

A uniformly distributed load of magnitude  $w$  per unit length is applied to a cantilever beam of length  $L$  where the variable distance is  $x$ . The deflection equation is

$$y = \frac{wx^2(6L^2 - 4Lx + x^2)}{24EI} \quad (5)$$

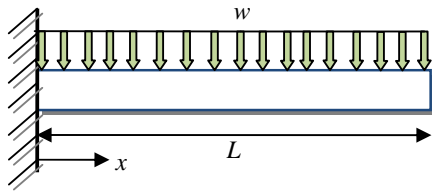


Fig. 2: Partially uniformly distributed load

Loading condition 3: Partially uniformly distributed load (Fig. 3)

Where  $p$  is the distance of the starting position of the uniformly distributed from the fixed end (in Figure 2,  $p$  is zero),  $q$  is the distance of the uniformly distributed load from the free end,  $a$  is equal to  $L - q$  and  $b$  is equal to  $L - p$ .

When  $p = 0$  and  $x \leq a$ , the deflection equation is

$$y = \frac{wx^2(6a^2 - 4ax + x^2)}{24EI} \quad (6)$$

When  $p = 0$  and  $x > a$ , the deflection equation is

$$y = \frac{pa^4}{8EI} + \frac{pa^3(x - a)}{6EI} \quad (7)$$

When  $q = 0$  and  $x \leq p$ , the deflection equation

$$y = \frac{wx^2(6L^2 - 4Lx + x^2)}{24EI} - \frac{wx^2(6p^2 - 4px + x^2)}{24EI} \quad (8)$$

When  $x > p$  and  $q = 0$ , the deflection equation

$$y = \frac{wx^2(6L^2 - 4Lx + x^2)}{24EI} - \frac{wa^4}{8EI} - \frac{wa^3(x - p)}{6EI} \quad (9)$$

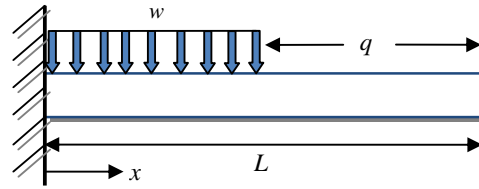


Fig. 3: Uniformly distributed load

Loading condition 4: A gradually varying load (Fig. 4)

A gradually varying load of highest magnitude  $w$  is subjected to a beam of length  $L$  where the variable distance is  $x$ . The deflection equation is

$$y = \frac{w(10L^3 - 10L^2x + 5Lx^2 - x^3)}{120EIL} \quad (10)$$

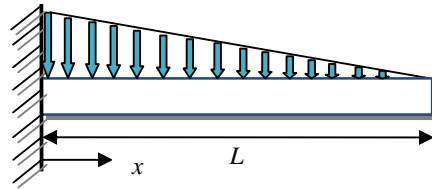


Fig. 4: Gradually varying distributed load

## 2.2 Simply supported beam:

Loading condition 1: Point load at any point on the beam (Fig. 5)

$$y = \frac{Pbx}{6EIL}(L^2 - x^2 - b^2) \quad \text{for } 0 < x < b \quad (11)$$

$$y = \frac{Pb}{6EIL} \left[ \frac{L}{b}(x - a)^3 + (L^2 - b^2)x - x^3 \right] \quad \text{for } a < x < L$$

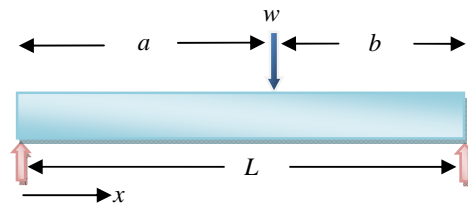


Fig. 5: Point load at any point on the beam

Loading condition 2: Uniformly distributed load over the length of the beam

$$y = \frac{wx}{24EI}(L^3 - 2Lx^2 + x^3) \quad (12)$$

Loading condition 3: A gradually varying load (Fig. 6)

$$y = \frac{wx}{360EIL} (7L^4 - 10L^2x^2 + 3x^4) \quad (13)$$

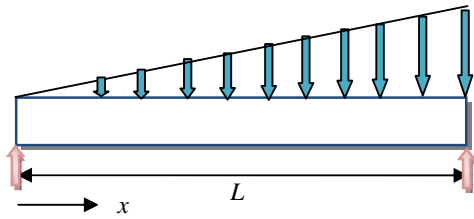


Figure 6: A gradually varying load

### 3. THE PROGRAM

The programming language used is C++, an extension of C. It was developed by Bjarne Stroustrup in the early 1980s at Bell Laboratories (Deitel, 2005). The reason for using this language is that it provides capabilities for object-oriented programming.

As seen in section 2, for each of the two types of beams (cantilever and simply supported), a number of loading conditions are present. Procedural program does not work very well for large and complex problems. Though it can not be claimed that the presented program is such a complicated one, but object oriented programs are easier to understand, correct and modify (Deitel, 2005). It results in programs that are clearer, more reliable and more easily maintained. Instead of viewing a program as a series of steps to be carried out, it views it as a group of objects that have certain properties and can take certain actions.

The developed program is a multifile program. The source file `main_beam.cpp` contains the function `main`. This function makes an object `beam` of class `Beam`. Its next task is to call a member function of class `Beam` named `add_node()`. Then according to the type of beam and presence of point load, uniformly distributed load, partially distributed load, gradually varying load and moment, it calls different member functions of class `Beam` like `add_pointload_cantibeam()` to find out the effect due to the presence of corresponding loads or moments. Finally to add up all the effects on the beam, ie, to apply the method of superposition, the function `summation_beam()` is called.

In the header file named `class_beam.h`, all the classes used in the program are created. These classes are `class pointload`, `class unidl` (uniformly distributed load), `class gvd1` (gradually varying distributed load), `class partidl` (partially distributed load), `class moment`, `class Node` and `class Beam`. The first five classes mentioned above contain member data about the positions and values of loads

and moments. The class `Node` has member data about deflection shear force and bending moment due to different types of loads and finally total deflection, total shear force and total moment to sum up all the individual effects. The class `Beam` has member data as `typebeam` (value 1 means cantilever beam and 2 means simply supported beam), `E` (Young's modulus of elasticity), `I` (moment of inertia) and length. The beam is divided into 100 segments, i.e., contains 101 nodes. Then for applied loads (maximum four point loads) and moments (maximum four), appropriate member functions are created to evaluate deflections, shear force and bending moments at each node. Then a member function named `void summation_beam()` is created to add up the individual deflection, shear force and bending moment components to give total values at each node.

In short, all the member functions are declared in corresponding classes in `class_beam.h` file. These functions are defined in either `canti_beam.cpp` or in `simply_beam.cpp` and are called in `main_beam.cpp`.

### 4. RESULTS AND DISCUSSION

To investigate the quality of the output generated by the program, a problem (A cantilever beam of 3 m length is carrying a load of 1 kN (downward) at the free end, 2 kN (downward) at a distance 1 m from the free end, 3 kN (downward) at a distance 2 m from the free end and finally a partially uniformly distributed load of 2 N/mm as shown in Figure 7. The modulus of elasticity of the beam is  $200 \times 10^3 \text{ N/mm}^2$  and moment of inertia is  $48.6 \times 10^6 \text{ mm}^4$ ) is selected. Then the generated output is compared with the outputs from a professional FEM software and hand calculation.

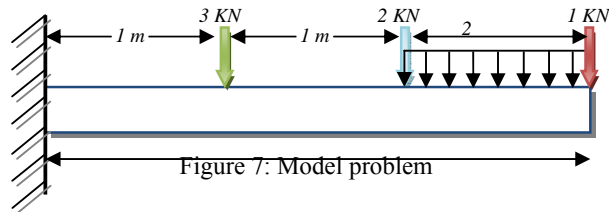


Figure 7: Model problem

Node	Bending moment in (N-mm)	Shear force (N)	Deflection (mm)
0	1.50E+07	8000	0
1	1.48E+07	8000	-0.000690741
2	1.45E+07	8000	-2.75E-03
---	---	---	---
49	4.65E+06	5000	-1.24E+00
50	4.50E+06	5000	-1.28E+00
51	4.35E+06	5000	-1.32E+00
---	---	---	---
98	6.36E+04	1120	-3.59431
99	30900	1060	-3.64472
100	0	1000	-3.69513

Table 1. Nodal values for the developed software

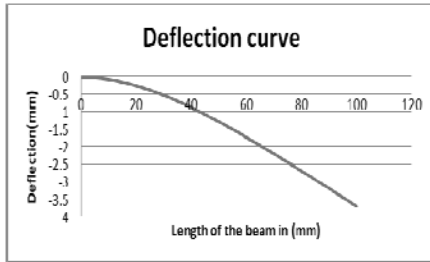


Figure 8: Deflection Curve

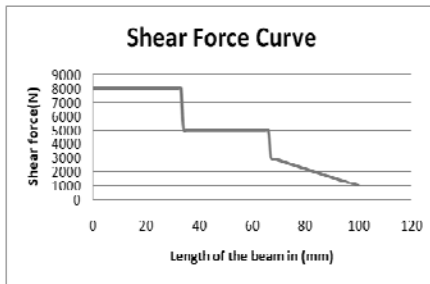


Figure 9: Shear Force Curve

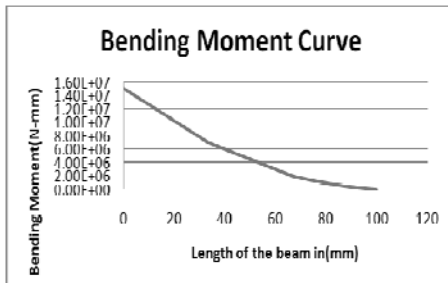


Figure 10: Bending Moment Curve

**5. COMPARISON**

Developed software output is compared with the finite element software and hand calculation below.

	Developed software	Finite element software	Hand calculation
BM(max) N-mm	1.50E+07	15000127	1.50E+07
SF(max) N	8000	8000.06	8000
DEF(max) mm	-3.69513	-----	-3.69513

Table 2: Table of Comparison

The curves drawn from the results of the program

are shown in Figures 8-10. Demo version of the professional software BEAM (based on finite element method), used for comparison (Table 2), gives only bending moment and shear force data. For that reason the space for deflection in the table of comparison is kept empty. It is clear from the table that the developed program gives accurate results.

Though the problem discussed above has only point loads but the program is capable of performing the analysis of beams subjected to a combination of point loads, uniformly distributed loads, gradually varying loads, non-uniformly distributed loads and moments. At present both the cantilever beam and simply supported beam can be analyzed. The program is capable of analyzing beam of only one span.

**6. CONCLUSION**

An Object-Oriented Program is developed for solving structural mechanics ‘beam’ problems by using the deflection, bending moment and shear force equations. The developed program generates output without any error. At present no graphics is associated with the program. The generated results are transferred to Microsoft Excel to create the curves. Future goal is to create graphical input and output system.

**REFERENCES**

1. A. Pytel and F. L. Singer (2005), *Strength of Materials*, Fourth Edition, Harper Collins publishers.
2. H. M. Deitel (2005), *C++, How to Program*, Fifth Edition, Prentice Hall.