Development of Computer Aided Interaction Diagram for Bi-axially Loaded Column

Md. Alvee Islam Navid[1], Syed Jamal Uddin Ahmed[2], Md. Mahbubul Islam[3*]

[1]Department of Civil Engineering, University of Asia Pacific (UAP), Dhaka-1207, **BANGLADESH**
[2]Assistant Professor, Department of Civil Engineering, University of Asia Pacific, Dhaka-1207, **BANGLADESH**
[3]Chief Scientific Officer, Agronomy Division, Bangladesh Jute Research Institute, Dhaka-1207, **BANGLADESH**

[*]Corresponding Contact:
Email: mahbub_agronomy@yahoo.com
Cell Phone: +88 01685 821219

ABSTRACT

A research work was undertaken at The Department of Civil Engineering of University of Asia Pacific (UAP), Dhaka, Bangladesh during April to November 2015. Biaxial bending means the column is carrying bending by one or both axis with axial load and with calculations it is possible to put those unique values into a pattern to make an interaction diagram with balanced failure zone, tension failure zone and finally compression failure zone of a short or slender column. By using programming it is possible to make the calculations in seconds. The method is to make functions and calling them to solve certain specific values to generate the diagram pattern. The outcome was diagram data generating application having the ability to combine programming and "Civil Logic". This is made for students and Civil Engineers who want to make interaction diagrams for designing a short, square and even slender column with ease.

Key words

Computer, Interaction Diagram, Bi-axially, Loaded Column

6/30/2015	**Source of Support:** None, **No Conflict of Interest:** Declared

INTRODUCTION

Interaction diagrams helps in designing a short columns and square columns for biaxial bending as well as for axial load. Biaxial bending mean the cross-section is under bending from one or both x-axis and y-axis simultaneously. Axial loading is loading along the normal line of an axis. Interaction diagram is used to find tension failure range, compression failure range and finally balanced failure region. Tensions failures occur while the eccentricities are large, Compressions failures occur for the small eccentricities. Balanced failure mode happens to produce failure for the concrete reaching its limit strain.

Many countries such as India, Pakistan, Malaysia and Netherlands have done extensive research and development of interaction diagram making software but they used none windows based application to do so. Interaction diagrams is introduced by "Universal Modeling Language" (UML), where it stands as simply as a sequence of work done by a series of different objects. But in Civil Engineering; for any eccentricity, there is a unique pair of Load, Pn and Moment, Mnand plotting them to their corresponding different eccentricities it will result in an interaction diagram.

The purpose of interaction diagram is vast but some are more important, practical design of a column, constructing strength interaction diagram, finding failure load and failure moment, finding the tension and compression failure region of the columns.

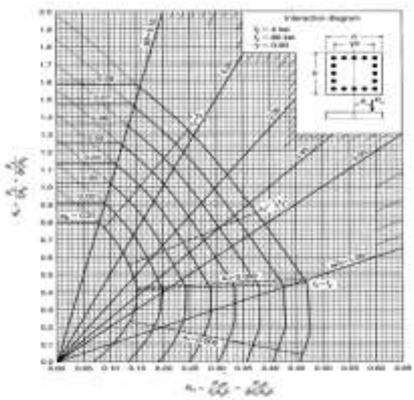


Figure 1: Interaction diagram (Nilson et al., 2004)

Different types of results showed for comparing the Load-Moment interaction diagrams for steel columns submitted to buckling according to various standards and codes. The purpose of the research was to check by means of buckling tests for steel columns submitted to eccentric loading, and also to compare the results under Eurocode 3 and other national standards. By using Numerical simulations of different profiles with Finelg software the tests were done for 13 steel columns (6).

A technique for seismic strengthening of concrete columns is presented by using straps constructed from high=strength fiber woven to form a flexible fabric like material. This gave increased ductility and shown increased in strength to the tested columns. Two types of straps with different fiber composites were used one was E-glass straps and the other one was Carbon fiber straps. Tests were done for circular and rectangular columns with three conditions unconfined or original states, with E-glass straps and finally Carbon straps.

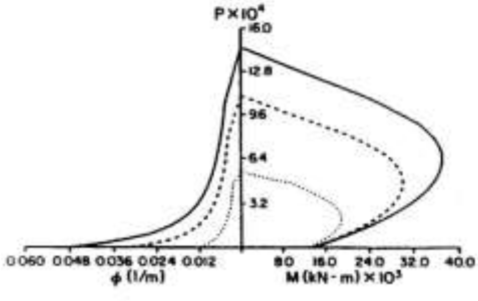


Figure 1a: Rectangular column interaction diagram (Saadatmanesh, 2015)

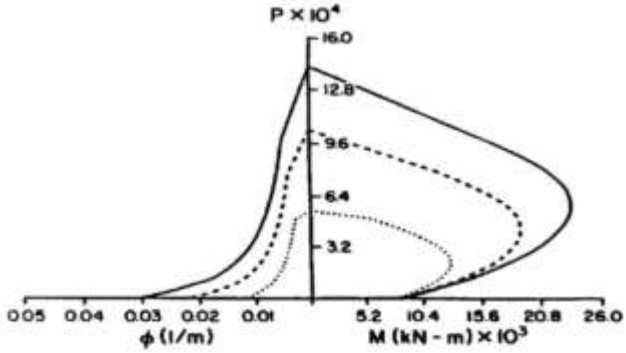


Figure 1b: Circular column interaction diagram (Saadatmanesh, 2015)

Method of using fiber model that employs computer graphics as a computational tool for the integration of normal stresses over the sections area. Many things such as geometrical definition of the failure surface in written at broad perspective. Both uniaxial bending for zero ("0") degree and right angle ("90") was done here.

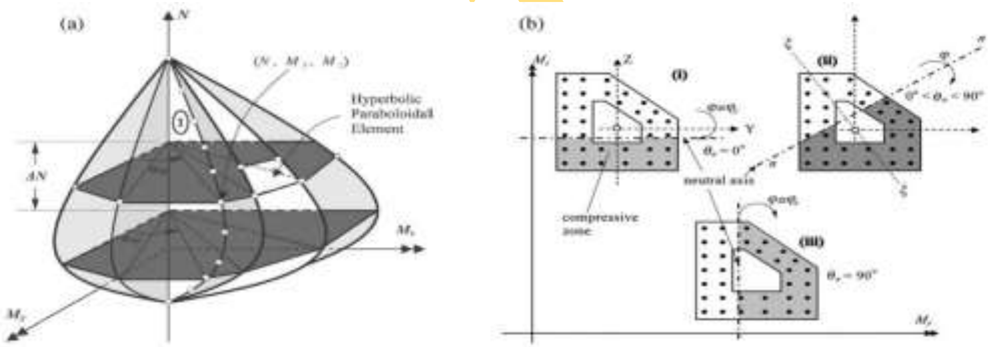


Figure 1c: Modeling of a surface (Sfakianakis, 2015)

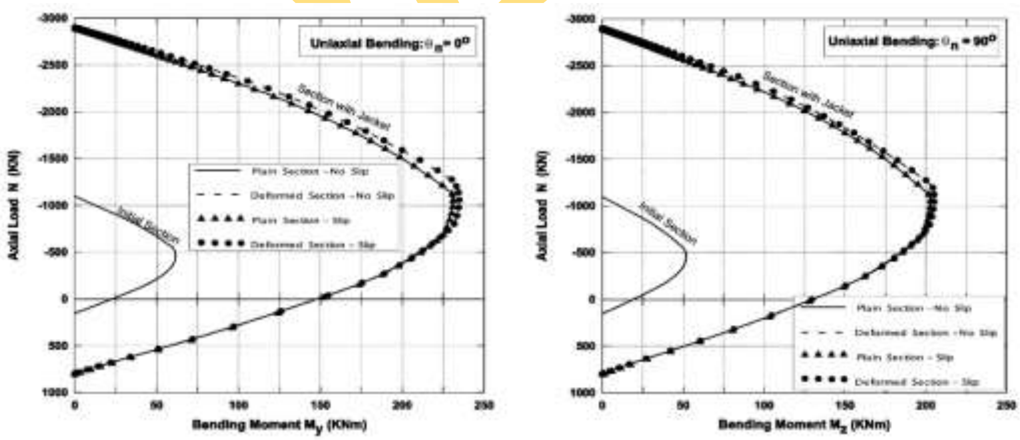


Figure 1d: Simulation and interaction diagram generation (Sfakianakis, 2015)

By using fiber model algorithm which allows for the efficient analysis of arbitrary composite sections under biaxial bending and axial load. The geometry of the cross section is defined by multi-nested curvilinear polygons.

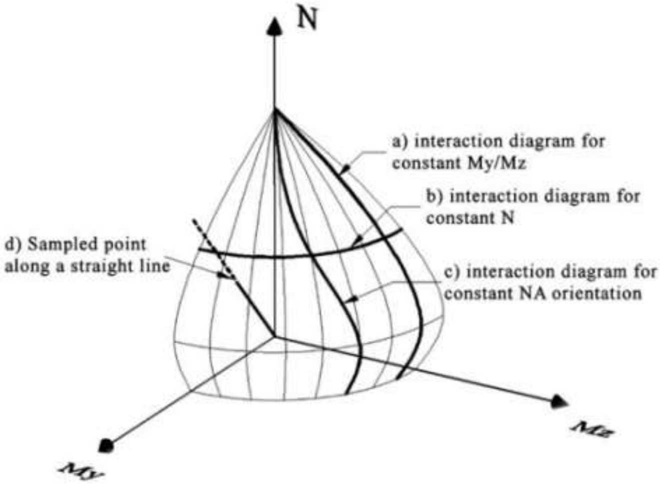


Figure 1e: Surface modeling (Charlampakis and Koumousis, 2015)

MATHCAD provide efficient learning environment for reinforced concrete design. This software contains powerful presentation capabilities, which includes use of charts, graphic objects, and animation effects. With MATCHCAD trend analyses, trial-and-error analyses, and optimization are possible. MATHCAD contains greater degree reliability and presentation quality. It saved time by freeing the time from tedious computation and transcription (7). Different cross-sections of different columns were used to make interaction diagrams. From two dimensions calculation to three dimensional surface modeling is spoken at abroad. Using RC-BIAX software is used to make the diagrams (15).

Interaction diagram plays an important role for the designing parts of civil engineering. The traditional method of explaining and doing consumes time. The modern way of programming gives the operator to make, to understand every part of a problem and solving that problem with ease of technology at hand. The significance of this study shows with common and primitive way and also by using less than half the understanding of a programming language to make a computer based application for everyone to use and also to excel in various parts of civil engineering. Therefore the research work was done with the things in mind and also using programming the theme to make a computer application which implements “Civil Logic”(an application with the ability of taking user inputs, having the ability to generate data from user input for column height and width, having the functions for Civil Engineering and also for designing aid for columns and generating the values of loads, moments and eccentricities for generating an interaction diagram) to solve problem as accurately possible.

METHODOLOGY

The research work was undertaken at The Department of Civil Engineering of University of Asia Pacific (UAP), Dhaka, Bangladesh during April to November 2015. To start with the breakdown of example 8.1 from Design of Concrete Structures (14th edition) page number-262, Chapter-8. From there the formulas were set to their order and also the Pseudo Code. After the written parts, the setup for the entire program was thoroughly explained on how to done things and also to use the appropriate functions at the best possible way.

EXAMPLE 8.1 **Column strength interaction diagram.** A 12 × 20 in. column is reinforced with four No. 9 (No. 29) bars of area 1.0 in^2 each, one in each corner as shown in Fig. 8.11*a*. The concrete cylinder strength is f'_c = 4000 psi and the steel yield strength is 60 ksi. Determine (*a*) the load P_b, moment M_b, and corresponding eccentricity e_b for balanced failure; (*b*) the load and moment for a representative point in the tension failure region of the interaction curve; (*c*) the load and moment for a representative point in the compression failure region; (*d*) the axial load strength for zero eccentricity. Then (*e*) sketch the strength interaction diagram for this column. Finally, (*f*) design the transverse reinforcement, based on ACI Code provisions.

SOLUTION.

(*a*) The neutral axis for the balanced failure condition is easily found from Eq. (8.15) with $\epsilon_u = 0.003$ and $\epsilon_y = 60/29{,}000 = 0.0021$

$$c_b = 17.5 \times \frac{0.003}{0.0051} = 10.3 \text{ in.}$$

giving a stress-block depth $a = 0.85 \times 10.3 = 8.76$ in. For the balanced failure condition, by definition, $f_s = f_y$. The compressive steel stress is found from Eq. (8.12):

$$f'_s = 0.003 \times 29{,}000\,\frac{10.3 - 2.5}{10.3} = 65.9 \text{ ksi} \quad \text{but} \quad \le 60 \text{ ksi}$$

confirming that the compression steel, too, is at the yield. The concrete compressive resultant is

$$C = 0.85 \times 4 \times 8.76 \times 12 = 357 \text{ kips}$$

The balanced load P_b is then found from Eq. (8.7) to be

$$P_b = 357 + 2.0 \times 60 - 2.0 \times 60 = 357 \text{ kips}$$

and the balanced moment from Eq. (8.8) is

$$M_b = 357(10 - 4.38) + 2.0 \times 60(10 - 2.5) + 2.0 \times 60(17.5 - 10)$$
$$= 3806 \text{ in-kips} = 317 \text{ ft-kips}$$

The corresponding eccentricity of load is $e_b = 10.66$ in.

(*b*) Any choice of c *smaller than* $c_b = 10.3$ in. will give a point in the tension failure region of the interaction curve, with eccentricity larger than e_b. For example, choose $c = 5.0$ in. By definition, $f_s = f_y$. The compressive steel stress is found to be

$$f'_s = 0.003 \times 29{,}000\,\frac{5.0 - 2.5}{5.0} = 43.5 \text{ ksi}$$

With the stress-block depth $a = 0.85 \times 5.0 = 4.25$, the compressive resultant is $C = 0.85 \times 4 \times 4.25 \times 12 = 173$ kips. Then from Eq. (8.7), the thrust is

$$P_n = 173 + 2.0 \times 43.5 - 2.0 \times 60 = 140 \text{ kips}$$

and the moment capacity from Eq. (8.8) is

$$M_n = 173(10 - 2.12) + 2.0 \times 43.5(10 - 2.5) + 2.0 \times 60(17.5 - 10)$$
$$= 2916 \text{ in-kips} = 243 \text{ ft-kips}$$

giving eccentricity $e = 2916/140 = 20.83$ in., well above the balanced value.

(*c*) Now selecting a c value *larger than* c_b to demonstrate a compression failure point on the interaction curve, choose $c = 18.0$ in., for which $a = 0.85 \times 18.0 = 15.3$ in. The compressive concrete resultant is $C = 0.85 \times 4 \times 15.3 \times 12 = 624$ kips. From Eq. (8.10) the stress in the steel at the left side of the column is

$$f_s = 0.003 \times 29{,}000\,\frac{17.5 - 18.0}{18.0} = -2 \text{ ksi}$$

Note that the negative value of f_s indicates correctly that A_s is in compression if c is greater than d, as in the present case. The compressive steel stress is found from Eq. (8.12) to be

$$f'_s = 0.003 \times 29{,}000\,\frac{18.0 - 2.5}{18.0} = 75 \text{ ksi} \quad \text{but} \quad \le 60 \text{ ksi}$$

Then the column capacity is

$$P_n = 624 + 2.0 \times 60 + 2.0 \times 2 = 748 \text{ kips}$$
$$M_n = 624(10 - 7.65) + 2.0 \times 60(10 - 2.5) - 2.0 \times 2(17.5 - 10)$$
$$= 2336 \text{ in-kips} = 195 \text{ ft-kips}$$

giving eccentricity $e = 2336/748 = 3.12$ in.

(*d*) The axial strength of the column if concentrically loaded corresponds to $c = \infty$ and $e = 0$. For this case,

$$P_n = 0.85 \times 4 \times 12 \times 20 + 4.0 \times 60 = 1056 \text{ kips}$$

Note that, for this as well as the preceding calculations, subtraction of the concrete displaced by the steel has been neglected. For comparison, if the deduction were made in the last calculation,

$$P_n = 0.85 \times 4(12 \times 20 - 4) + (4.0 \times 60) = 1042 \text{ kips}$$

The error in neglecting this deduction is only 1 percent in this case; the difference generally can be neglected, except perhaps for columns with reinforcement ratios close to the maximum of 8 percent. In the case of design aids, however, such as those presented in Refs. 8.2 and 8.7 and discussed in Section 8.10, the deduction is usually included for all reinforcement ratios.

(*e*) From the calculations just completed, plus similar repetitive calculations that will not be given here, the strength interaction curve of Fig. 8.11*d* is constructed. Note the characteristic shape, described earlier, the location of the balanced failure point as well as the "small eccentricity" and "large eccentricity" points just found, and the axial load capacity.

In the process of developing a strength interaction curve, it is possible to select the values of steel strain ϵ_s, as done in step *a*, for use in steps *b* and *c*. Selecting ϵ_s uniquely establishes the neutral axis depth c, as shown by Eqs. (8.9) and (8.15), and is useful in determining M_n and P_n for values of steel strain that correspond to changes in the strength reduction factor ϕ, as will be discussed in Section 8.9.

(*f*) The design of the column ties will be carried out following the ACI Code restrictions. For the minimum permitted tie diameter of $\frac{3}{8}$ in., used with No. 9 (No. 29) longitudinal bars having a diameter of 1.128 in a column the least dimension of which is 12 in., the tie spacing is not to exceed

$$48 \times \frac{3}{8} = 18 \text{ in.}$$
$$16 \times 1.128 = 18.05 \text{ in.}$$
$$b = 12 \text{ in.}$$

The last restriction controls in this case, and No. 3 (No. 10) ties will be used at 12 in. spacing, detailed as shown in Fig. 8.11*a*. Note that the permitted spacing as controlled by the first and second criteria, 18 in., must be reduced because of the 12 in. column dimension.

Figure 2: Example 8.1 (Nilson et al., 2004)

It was to broken down the entire math into small pieces so that it becomes easier to read and also to set the functions for making the calculations.

$$\epsilon_s = \epsilon_u \frac{d - c}{c}$$

$$f_s = \epsilon_u E_s \frac{d - c}{c} \le f_y$$

$$\epsilon'_s = \epsilon_u \frac{c - d'}{c}$$

$$f'_s = \epsilon_u E_s \frac{c - d'}{c} \le f_y$$

$$a = \beta_1 c \le h$$

$$C = 0.85 f'_c ab$$

$$c = c_b = d \frac{\epsilon_u}{\epsilon_u + \epsilon_y}$$

$$a = a_b = \beta_1 c_b$$

$$P_n = 0.85 f'_c ab + A'_s f'_s - A_s f_s$$

$$M_n = P_n e = 0.85 f'_c ab\left(\frac{h}{2} - \frac{a}{2}\right) + A'_s f'_s\left(\frac{h}{2} - d'\right) + A_s f_s\left(d - \frac{h}{2}\right)$$

Figure 3: Breakdown Method (Nilson et al., 2004)

Table 1: Functions declarations

Formulas	Functions
$c_b = d \times \frac{\varepsilon_u}{\varepsilon_u + \varepsilon_y}$	PrivateFunction balanceFailureC(ByVal x AsDouble, ByVal y AsDouble,ByVal a AsDouble) AsDouble Dim answer AsDouble = Nothing answer = (x * y) / (y + a) Return answer EndFunction
$a = \beta_1 * c_b$	PrivateFunction sTressBlockDepth(ByVal x AsDouble, ByVal y AsDouble) AsDouble Dim answer AsDouble = Nothing answer = x * y Return answer EndFunction
$f_s = \varepsilon_u * E * \frac{d - c_b}{c_b}$	PrivateFunction f_sub_s(ByVal x AsDouble, ByVal y AsDouble, ByVal z AsDouble, ByVal a AsDouble) AsDouble Dim answer AsDouble = 0 answer = (x * y * (z - a)) / a Return answer EndFunction
$f'_s = \varepsilon_u * E * \frac{c_b - d'}{c_b}$	PrivateFunction fPrimeS(ByVal x AsDouble, ByVal y AsDouble, ByVal a AsDouble, ByVal b AsDouble) AsDouble Dim answer AsDouble = Nothing answer = ((x * y) / a) * (a - b) Return answer EndFunction
$e = \frac{M_n}{P_n}$	PrivateFunction eccentricityE(ByVal x AsDouble, ByVal y AsDouble) AsDouble Dim answer AsDouble = Nothing answer = x / y Return answer EndFunction
$C = .85 * f'_c * a * b$	PrivateFunction conCRETECompResultant(ByVal x AsDouble, ByVal y AsDouble, ByVal a AsDouble) AsDouble Dim answer AsDouble = Nothing answer = 0.85 * x * y * a Return answer EndFunction
$P_n = C + A'_s * f'_s - A_s * f_s$	PrivateFunction balanceLoadPb(ByVal x AsDouble, ByVal y AsDouble, ByVal z AsDouble,ByVal a AsDouble, ByVal b AsDouble) AsDouble Dim answer AsDouble = Nothing answer = x + (y * z) - (a * b) Return answer EndFunction

$M_n = P_n * e = .85 * f'_c * a * b * \left(\frac{h-a}{2}\right) + A'_s * f'_s \left(\frac{h}{2} - d'\right) + A_s * f_s\left(d - \frac{h}{2}\right)$	PrivateFunction balancedMomentMb(ByVal x AsDouble, ByVal y AsDouble, ByVal z AsDouble, ByVal a AsDouble, ByVal m AsDouble, ByVal n AsDouble,ByVal o AsDouble, ByVal p AsDouble, ByVal q AsDouble, ByVal r AsDouble) AsDouble Dim answer AsDouble = Nothing answer = (0.85 * x * y * z * (0.5 * (a - y))) + (m * n * ((0.5 * a) - q)) + (o * p * (r - (0.5 * a))) Return answer EndFunction
$\rho = \frac{A_s}{b * d}$	PrivateFunction RohNormal(ByVal x AsDouble, ByVal y AsDouble,ByVal a AsDouble) AsDouble Dim answer AsDouble = Nothing answer = x / (y * a) Return answer EndFunction
$\rho_{max} = .85 * \beta_1 * \frac{f'_c}{f_y} * \frac{\varepsilon_u}{\varepsilon_u + \varepsilon_y}$	PrivateFunction rohMax(ByVal x AsDouble, ByVal y AsDouble, ByVal z AsDouble,ByVal a AsDouble, ByVal b AsDouble) AsDouble Dim answer AsDouble = Nothing answer = 0.85 * (x * y * a) / (z * (a + b)) Return answer EndFunction

A Flow Chart was used to saw the path of the program. The functions parts that were used to make this program shown below

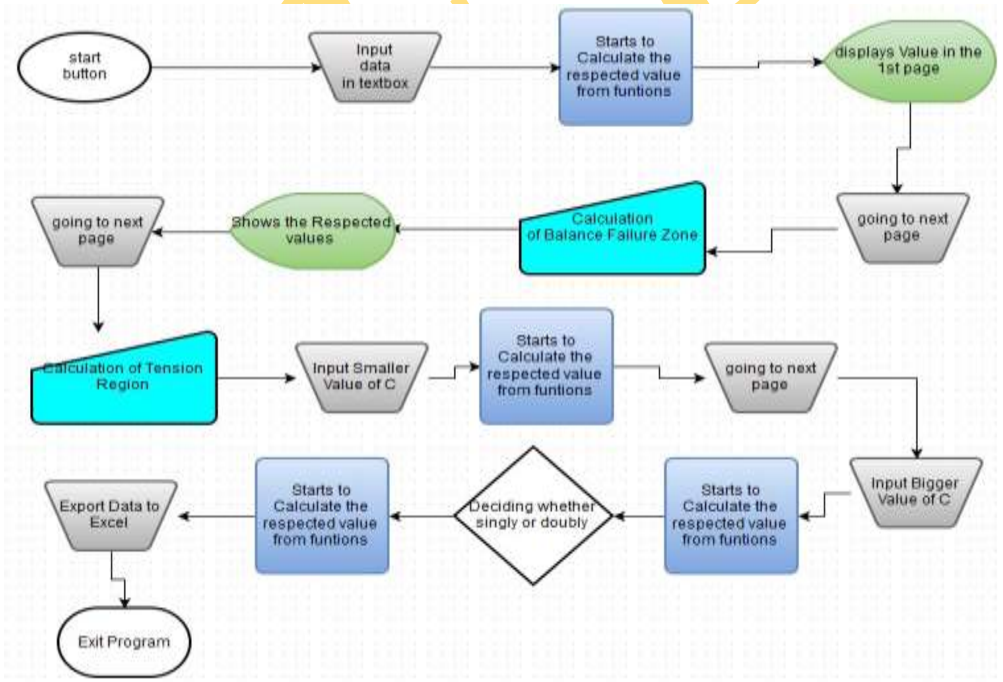


Figure 4: Flow Chart

For each of the variables taken to have a name. So, to give them a proper name, used Camel Case for better use and also to identify the variables easily.

Table 2: Camel cased variables

Variables	Declarations	Variables	Declarations
f_y	Dim**stlYield**AsDouble = 0	E	Dim**valueEs**AsDouble = 0
f_c'	Dim**concreteC**AsDouble = 0	ρ	Dim**roh_Normal**AsDouble = 0
ε_u	Dim**efsilonU**AsDouble = 0	ρ_{max}	Dim**rohMaximum**AsDouble = 0
ε_y	Dim**efsilonY**AsDouble = 0	d	Dim**eff_depht**AsDouble = 0
β_1	Dim**valueBeta**AsDouble = 0	E	Dim**value_Es**AsDouble = 0
A_s	Dim**asubS**AsDouble = 0	ρ_{max}	Dim**roHmaxCalc**AsDouble = 0
A_s'	Dim**aPrimeS**AsDouble = 0	d'	Dim**dprimeS**AsDouble = 2.5
$Width\ , b$	Dim**colmnXsecB**AsDouble = 0	$Height\ , h$	Dim**colmnXsecH**AsDouble = 0

The entire program was written in plain English. Not using any kind of technical terms to set up the entire programs outlook. This was mainly used for initial analysis and also for references for developing any software because this had the flexibility of changing at any time. This was also used for making logic patterns, design patterns. If anything was out of place this could be used for fixing and even correcting certain errors, bugs and design flaws. The Flow Chart shown above for guideline.

Main Page

Declare the variables fy,f'c,E,efsilonU,efsilonU, width(b),height(h),Beta.

Input steel bar NO.#, Area of steel and given steel bar in the cross section
Calculates
d,d', Asteel,
Initial assumption Doubly reinforced
Calculates the entire page

Balanced page

Calculates the Neutral Axis "c"
Gets values from the first page linked-in
Calculates f's, CapC, BalancedLoad, BalancedMoment

Tension Page

Neutral axis [(manual / auto), smallValue]
Calculates the Neutral Axis "c"
Gets values from the first page linked-in
Calculates f's, CapC, Load, Moment

Compression Page

Neutral axis [(manual / auto), smallValue]
Calculates the Neutral Axis "c"
Gets values from the first page linked-in
Calculates f's, CapC, Load, Moment

Concentric Page

Take As
Calculate Roh, Rohmax
If Roh > Rohmax doubly
Else Roh < Rohmax singly

Configure Clear button

Figure 5: Pseudo Code Example

Showing the basics on the left with a dummy program and the developed program on the right. The main source code will be given on the appendix.

Interface:

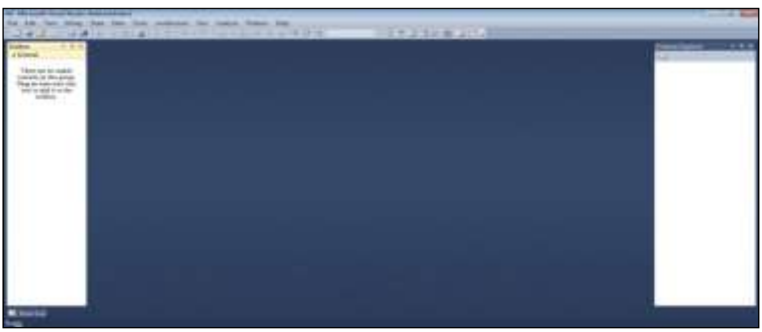


Figure 6. Visual Studio Interface

At the top there was a menu bar and was a Tab called File.
The following sequence was:

File → New Project→ (select) Windows Forms Application → (input) Name (of project) → (Press) Ok

Layout:

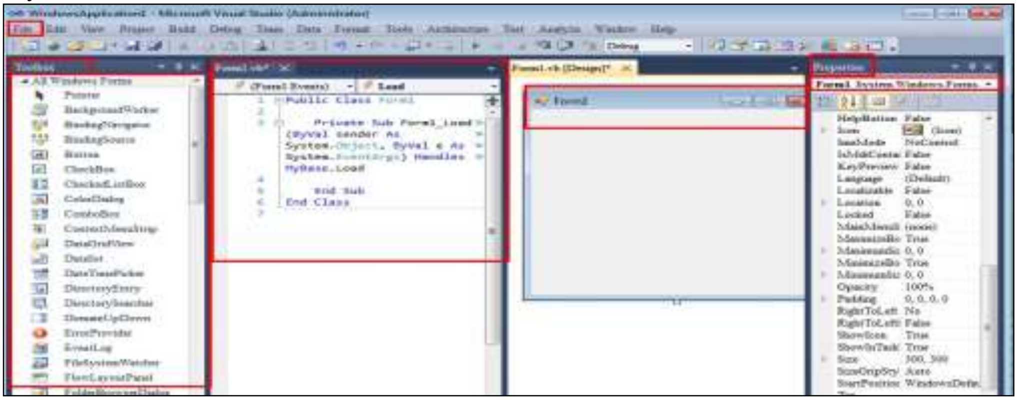


Figure 7. Visual Studio Layout

- Form.vb was created when the Form 1 was double clicked
- Form.vb was where the coding was done
- Toolbox was where all the tools such as labels, textboxes etc. was kept.
- Properties were where all the properties were stored.
- From the toolbox drag and drop of buttons, textboxes, graphs etc. was possible
- Each object holds its special name and properties which was shown in the Properties section

Dummy Program:

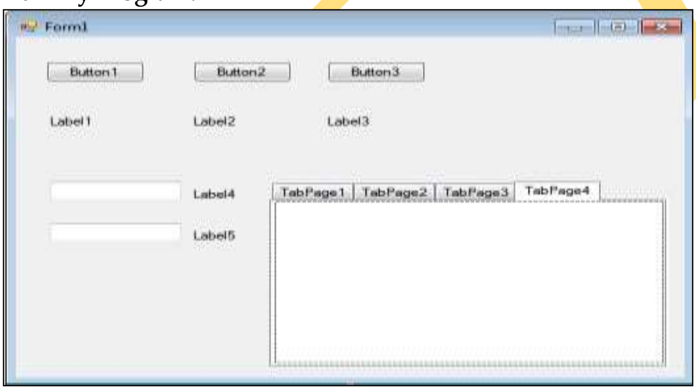


Figure 8: Dummy Program

This was shown as an example to show the readers how to make a program in visual studio environment

Developed Program:

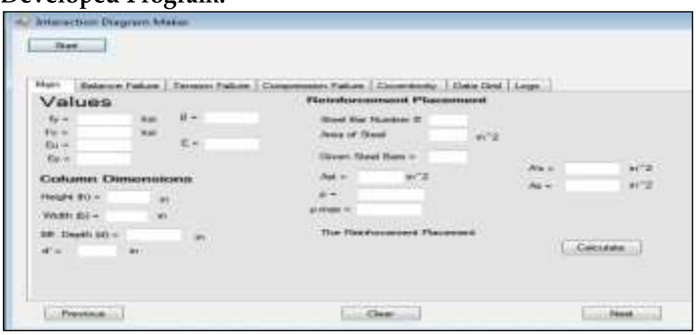


Figure 9: Developed Program

This was the final developed software for doing the interaction diagram. As it was seen everything to labels, textboxes, buttons were done accordingly

Functions were needed because to use the same equations with different parameters to save time and efficiency of the software both on a command-line interface (CLI) and graphical user interface (GUI). Each programming language had its own rules on how to declare the functions.

GLOBAL PRIVATE FUNCTION DECLARATION

The pseudo code shown below-

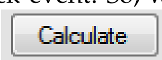
```
Private Function [EqnName] (ByVal <parameters>As DataType) As DataType
        Dim ReturnValueName As dataType = 0
        ReturnValueName = <Parameters in Equation Format>
        Return [ReturnValueName]
End Function
```

Example:

```
PrivateFunction balancedMomentMb(ByVal x AsDouble, ByVal y AsDouble, ByVal z AsDouble,ByVal a AsDouble, ByVal m AsDouble, ByVal n AsDouble, ByVal o AsDouble, ByVal p AsDouble, ByVal q AsDouble, ByVal r AsDouble) AsDouble
Dim answer AsDouble = Nothing
answer = (0.85 * x * y * z * (0.5 * (a - y))) + (m * n * ((0.5 * a) - q)) + (o * p * (r - (0.5 * a)))
Return answer
EndFunction
```

Private means the variables that are declared inside the function are only accessed by the only this function. Nothing can access the variables unless the Private modifiers are changed to Public but if that is done then all the parameters that are used multiple times cannot be used.

BUTTON FUNCTION DECLARATION

Every Button is a private function to be used with its own set of variables and it is declared as a click event. So, when the button is clicked the calculation will be executed will be executed. 


```
PrivateSub btnMainPageButton_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnMainPageButton.Click

'Reinforcement Placement
Dim barNumber AsDouble = Nothing
Dim barArea AsDouble = Nothing
Dim givenBar AsDouble = 0
Dim areaOfSteel AsDouble = 0

'Calculation(Rein. Placement)
barNumber = txBXStlBarNo.Text
barArea = txBXAreaSteel.Text
givenBar = txBXBars.Text
areaOfSteel = areaForAll(givenBar, barArea)
txBXArea.Text = areaOfSteel
'Values
'Calcutions
eff_depht = colmnXsecH - 2.5
txBxColEff_depth.Text = eff_depht
roh_Normal= RohNormal(areaOfSteel, colmnXsecB, eff_depht)
txBXroH.Text = roh_Normal.ToString(".0000000")
roHmaxCalc = rohMax(valueBeta, concreteC, stlYield, efsilonU, efsilonY)
txBXrohMax.Text = roHmaxCalc.ToString(".0000000")
txBxColDb_dPrime.Text = dprimeS
If roh_Normal > roHmaxCalc Then
        lblReinPlacement.Text = dobuli
ElseIf roh_Normal < roHmaxCalc Then
        lblReinPlacement.Text = sigeli
EndIf
If roh_Normal > roHmaxCalc Then
```

```
Dim stlYield AsDouble = 0
Dim concreteC AsDouble = 0
Dim efsilonU AsDouble = 0
Dim efsilonY AsDouble = 0
Dim valueBeta AsDouble = 0
Dim valueEs AsDouble = 0
Dim colmnXsecH AsDouble = 0
Dim colmnXsecB AsDouble = 0
Dim roh_Normal AsDouble = 0
Dim rohMaximum AsDouble = 0
Dim eff_depht AsDouble = 0
Dim value_Es AsDouble = 0
Dim roHmaxCalc AsDouble = 0
Dim dobuli AsString = "The Reinforcement is Doubly "
Dim sigeli AsString = "The Reinforcement is Singly"
Dim dprimeS AsDouble = 2.5
Dim aPrimeS AsDouble = 0
Dim asubS AsDouble = 0

stlYield = txBXfy.Text
concreteC = txBXfprimeC.Text
efsilonU = txBXefcilonU.Text
efsilonY = txBXefcilonY.Text
valueBeta = txBXBeta.Text
colmnXsecH = txBxColHeight.Text
colmnXsecB = txBxColWidthB.Text
areaOfSteel = txBXArea.Text
value_Es = txBXVlaue_Es.Text
lblReinPlacement.Text = sigeli
asubS = areaOfSteel
txBXDbAprimeS.Text = 0
txBXDbAs.Text = asubS

ElseIf roh_Normal < roHmaxCalc Then
lblReinPlacement.Text = dobuli
txBxColDb_dPrime.Text = dprimeS
aPrimeS = areaOfSteel / 2
asubS = areaOfSteel / 2
txBXDbAprimeS.Text = aPrimeS
txBXDbAs.Text = asubS
EndIf
EndSub
```

Method of Using the Software

Step 1

User had to push the Start button to initiate the program

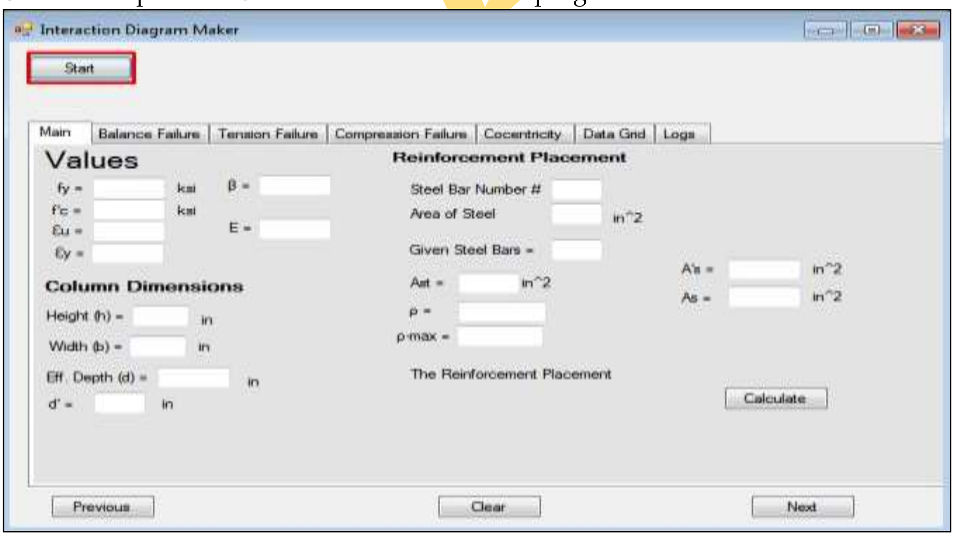


Figure 10: Starting the program

Step 2

Now the user has to input the values shown in the Figure. To initiate the calculation user had to push the calculate button.

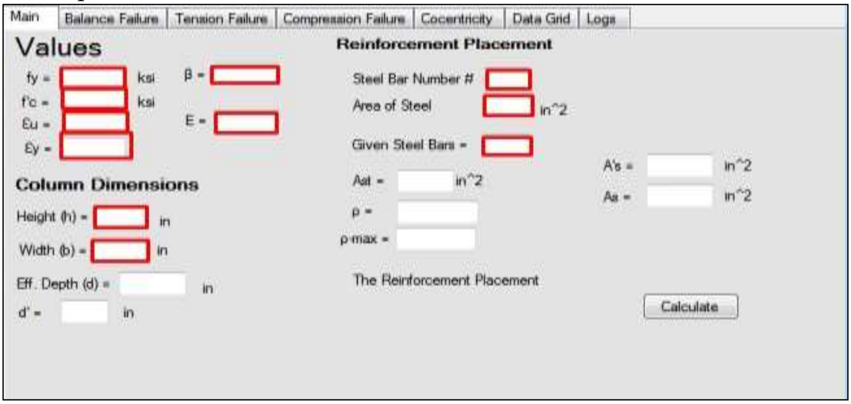


Figure 11: Inputting required data

Step 3

$input\ the\ values\ for\ f'_c, \beta, \varepsilon_y, \varepsilon_{u,} f_{y,} E,$
$Steel\ bar\ Number, Area\ of\ steel, Given\ steel\ bar, width(b), height(h)$

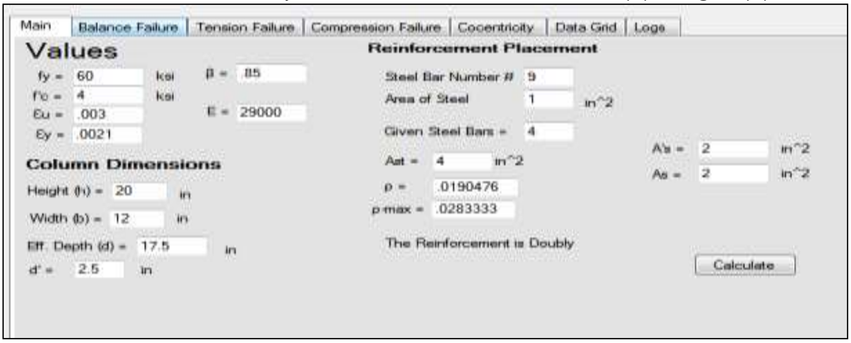


Figure 12: First calculations

Step 4

To Press the calculate button

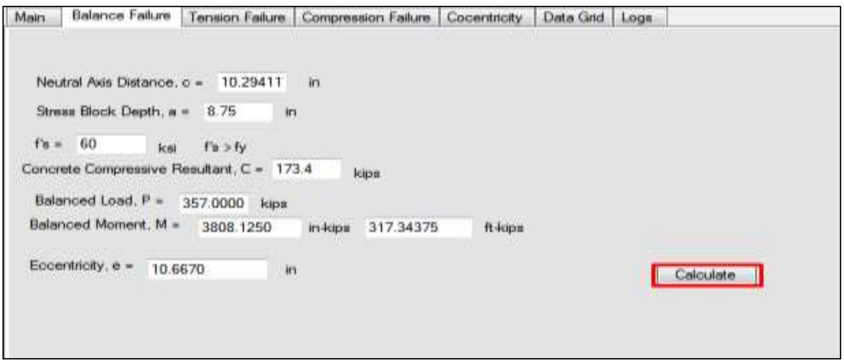


Figure 13: Second calculations

Step 5

Input either the first or the auto radio button for user customized input of small value c. To Press the calculate button.

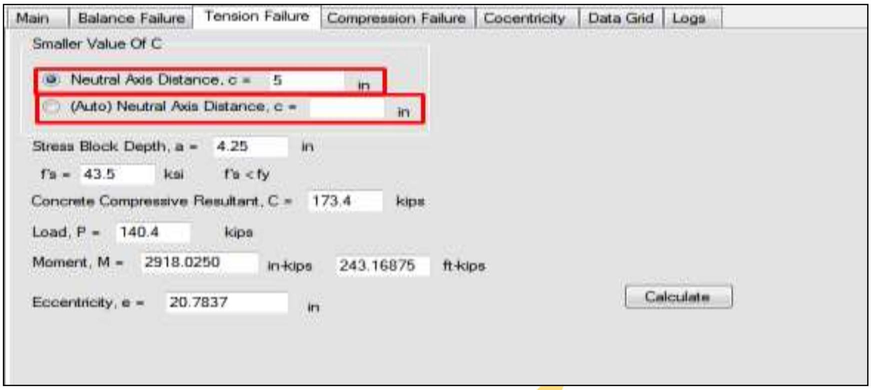


Figure 14: User value input; small

Step 6

Input either the first or the auto radio button for user customized input of big value c. To Press the calculate button.

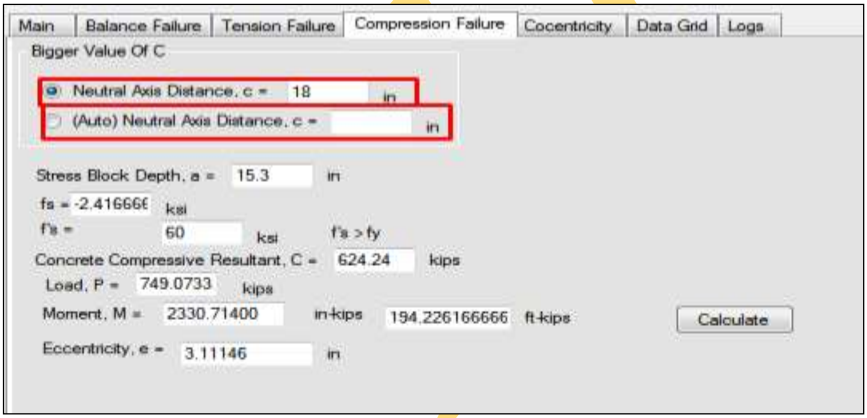


Figure 15: User value input; big

Step 7

To Press the calculate button for final data generation

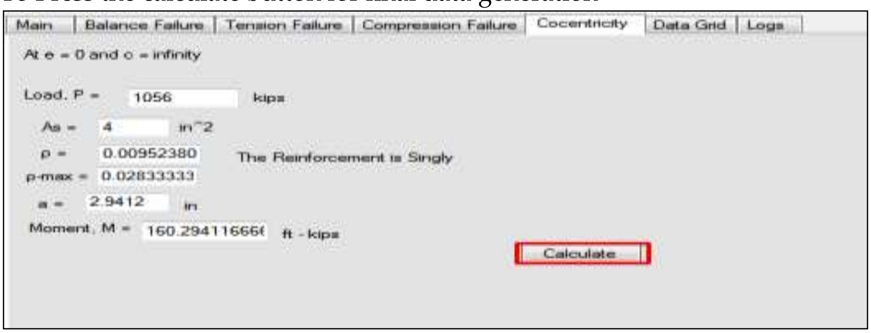


Figure 16. Final calculations

Step 8

To take value as shown value and put it in an Excel work sheet with x-axis as Moment and y-axis as load. The Biggest to smallest values according to load and corresponding moments.

Table 3: Value sorting

Value	Moment	Load
Biggest	0	1056
Bigger	195	748
Big	317	357
Small	243	140
Smallest	140	0

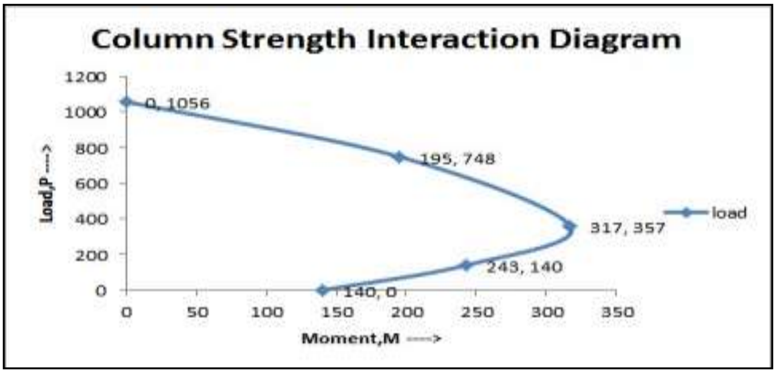


Figure 17: Excel generated interaction diagram

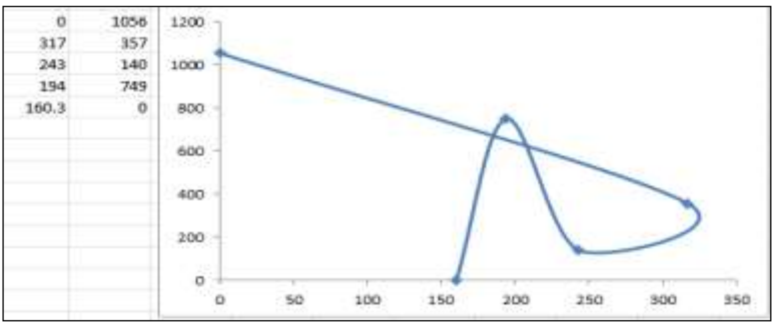


Figure 18: Excel generated interaction diagram (without value sorting)

RESULTS AND ANALYSIS

The data taken from the application and also compare the data with the reference math.

Reference Problem

EXAMPLE 8.1 **Column strength interaction diagram.** A 12 × 20 in. column is reinforced with four No. 9 (No. 29) bars of area 1.0 in^2 each, one in each corner as shown in Fig. 8.11*a*. The concrete cylinder strength is $f_c' = 4000$ psi and the steel yield strength is 60 ksi. Determine (*a*) the load P_b, moment M_b, and corresponding eccentricity e_b for balanced failure; (*b*) the load and moment for a representative point in the tension failure region of the interaction curve; (*c*) the load and moment for a representative point in the compression failure region; (*d*) the axial load strength for zero eccentricity. Then (*e*) sketch the strength interaction diagram for this column. Finally, (*f*) design the transverse reinforcement, based on ACI Code provisions.

SOLUTION.

(a) The neutral axis for the balanced failure condition is easily found from Eq. $\epsilon_u = 0.003$ and $\epsilon_y = 60/29{,}000 = 0.0021$

$$c_b = 17.5 \times \frac{0.003}{0.0051} = 10.3 \text{ in}$$

giving a stress-block depth $a = 0.85 \times 10.3 = 8.76$ in. For the balanced failur by definition, $f_s = f_y$. The compressive steel stress is found from Eq. (8.12):

$$f'_s = 0.003 \times 29{,}000 \frac{10.3 - 2.5}{10.3} = 65.9 \text{ ksi} \quad \text{but} \quad \leq 60 \text{ ksi}$$

confirming that the compression steel, too, is at the yield. The concrete compressiv

$$C = 0.85 \times 4 \times 8.76 \times 12 = 357 \text{ kips}$$

The balanced load P_b is then found from Eq. (8.7) to be

$$P_b = 357 + 2.0 \times 60 - 2.0 \times 60 = 357 \text{ kips}$$

and the balanced moment from Eq. (8.8) is

$$M_b = 357(10 - 4.38) + 2.0 \times 60(10 - 2.5) + 2.0 \times 60(17.5 - 10$$
$$= 3806 \text{ in-kips} = 317 \text{ ft-kips}$$

The corresponding eccentricity of load is $e_b = 10.66$ in.

(b) Any choice of c *smaller than* $c_b = 10.3$ in. will give a point in the tension fa of the interaction curve, with eccentricity larger than e_b. For example, choose By definition, $f_s = f_y$. The compressive steel stress is found to be

$$f'_s = 0.003 \times 29{,}000 \frac{5.0 - 2.5}{5.0} = 43.5 \text{ ksi}$$

With the stress-block depth $a = 0.85 \times 5.0 = 4.25$, the compressive resultant is $4 \times 4.25 \times 12 = 173$ kips. Then from Eq. (8.7), the thrust is

$$P_n = 173 + 2.0 \times 43.5 - 2.0 \times 60 = 140 \text{ kips}$$

and the moment capacity from Eq. (8.8) is

$$M_n = 173(10 - 2.12) + 2.0 \times 43.5(10 - 2.5) + 2.0 \times 60(17.5 - 1$$
$$= 2916 \text{ in-kips} = 243 \text{ ft-kips}$$

giving eccentricity $e = 2916/140 = 20.83$ in., well above the balanced value.

(c) Now selecting a c value *larger than* c_b to demonstrate a compression failure point on interaction curve, choose $c = 18.0$ in., for which $a = 0.85 \times 18.0 = 15.3$ in. The c pressive concrete resultant is $C = 0.85 \times 4 \times 15.3 \times 12 = 624$ kips. From Eq. (8.10) stress in the steel at the left side of the column is

$$f_s = 0.003 \times 29{,}000 \frac{17.5 - 18.0}{18.0} = -2 \text{ ksi}$$

Note that the negative value of f_s indicates correctly that A_s is in compression if c is gre than d, as in the present case. The compressive steel stress is found from Eq. (8.12) t

$$f'_s = 0.003 \times 29{,}000 \frac{18.0 - 2.5}{18.0} = 75 \text{ ksi} \quad \text{but} \quad \leq 60 \text{ ksi}$$

Then the column capacity is

$$P_n = 624 + 2.0 \times 60 + 2.0 \times 2 = 748 \text{ kips}$$
$$M_n = 624(10 - 7.65) + 2.0 \times 60(10 - 2.5) - 2.0 \times 2(17.5 - 10)$$
$$= 2336 \text{ in-kips} = 195 \text{ ft-kips}$$

giving eccentricity $e = 2336/748 = 3.12$ in.

(d) The axial strength of the column if concentrically loaded corresponds to $c = \infty$ For this case,

$$P_n = 0.85 \times 4 \times 12 \times 20 + 4.0 \times 60 = 1056 \text{ kips}$$

Note that, for this as well as the preceding calculations, subtraction of th displaced by the steel has been neglected. For comparison, if the deduction we the last calculation,

$$P_n = 0.85 \times 4(12 \times 20 - 4) + (4.0 \times 60) = 1042 \text{ kips}$$

Figure 19: Example 8.1; Reference math

Application data

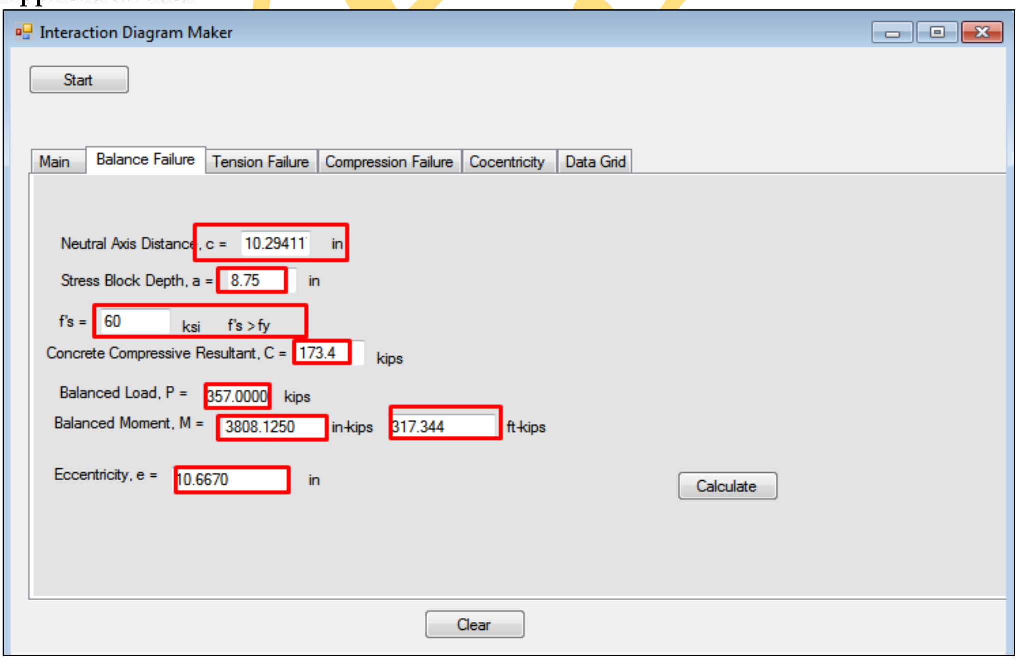


Figure 20: Balance failure data generation

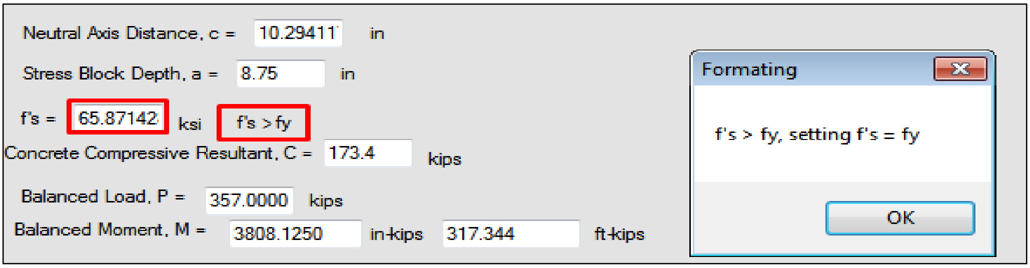


Figure 21: Transformation of data; balance failure

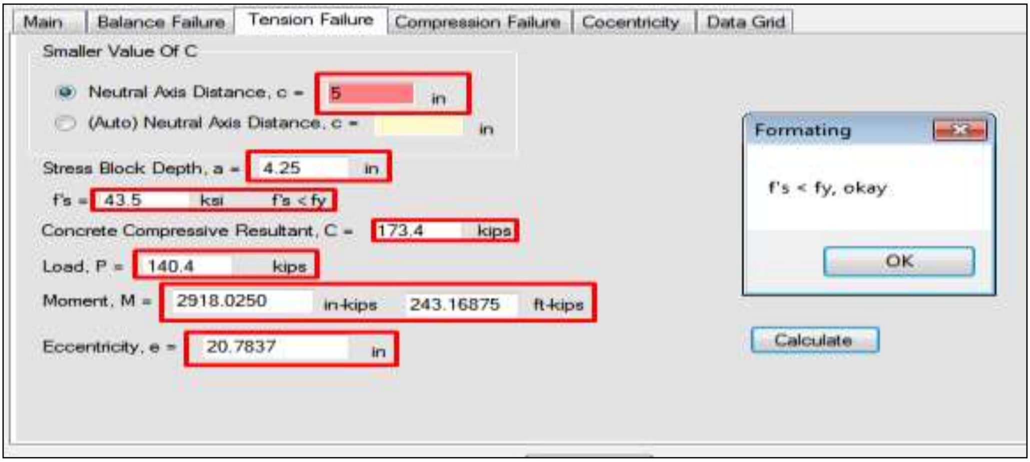


Figure 22: User small value input

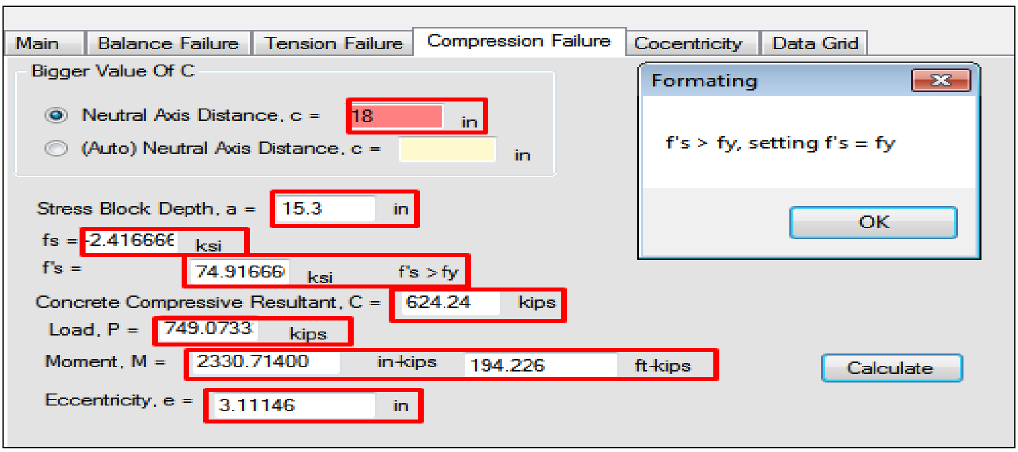


Figure 23: Data transformation compression failure

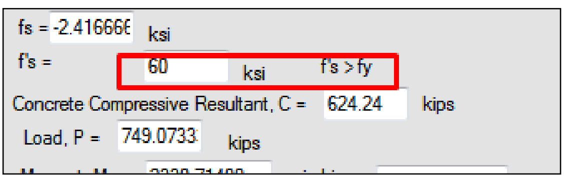


Figure 24: Data alteration

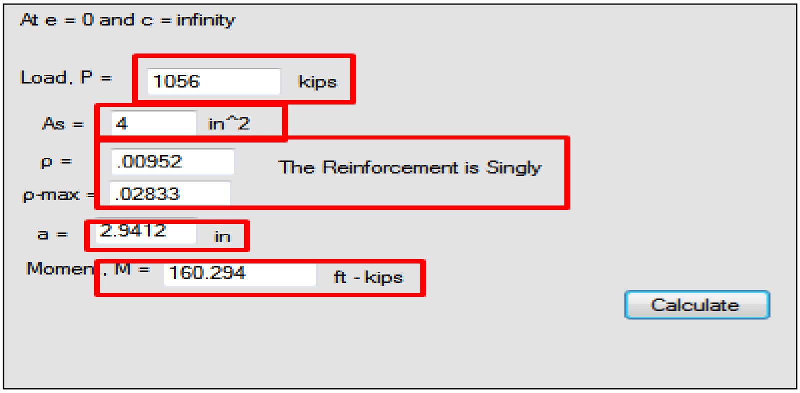


Figure 25: Concentric data generation

The comparative analysis was shown below for better showing the results and the accuracy of the application made.

Table 4: Data comparison

Variable/ Character	Reference Math Value	Application Generated Value
$Balacne, c_b$	10.3	10.29411
$Balacne, f_s'$	65.9 > 60	65.4712
$Balacne, C$	357	173.4
$Balacne, P_n$	357	357.000
$Balacne, M_n$	317	317.344
$Balacne, e_b$	10.66	10.667
$Balance, a$	8.76	8.75
$smallValue, a$	4.25	4.25
$smallValue, c$	10.3 ($\frac{10.33}{2}$ = 5.15) < 5, taken	5 (input), 5.15 (auto)
$smallValue, f_s'$	43.5	43.5
$smallValue, C$	173	173.4
$smallValue, P_n$	140	140.4
$smallValue, M_n$	243	243.16875
$smallValue, e_b$	20.83	20.7837
$bigValue, a$	15.3	15.3
$bigValue, c$	$10.3(10.33 * 2 = 20.66) > 18$,taken	18 (input), 20.66 (auto)
$bigValue, f_s'$	75 > 60	74.916666 > 60
$bigValue, C$	624	624.24
$bigValue, P_n$	748	749.0733
$bigValue, M_n$	195	194.226
$bigValue, e_b$	3.12	3.11146
$bigValue, f_s$	"-ve" 2	"-ve" 2.416666
$concenValue, P_n$	1056	1056
$concenValue, M_n$	140	160.294
$concenValue, e$	0	0
$concenValue, c$	∞	∞
$concenValue, a$	12	2.9412

From the Table 4 it was visible that the accuracy was very close for the application data to meet the data of the reference math.

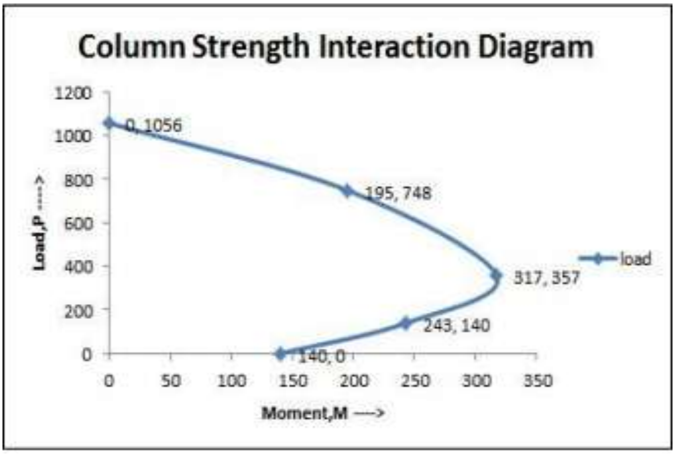


Figure 26: Eexcel diagram generation

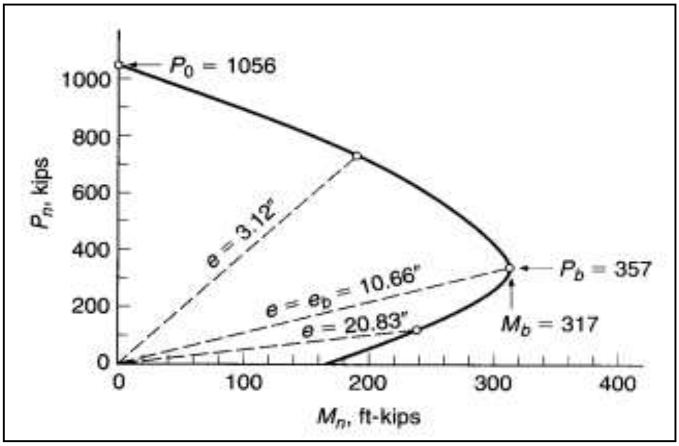


Figure 27: Example 8.1 Interaction diagram (Nilson et al., 2004)

From the results it was observed that the program had the ability to do all the calculation and also gave near accurate results from the reference problem. The program was versatile by changing any values and had the ability to calculate data within milliseconds.

momnet	load
0	852
162.97	547.7
242.537	263
170.67	60.65
137	0

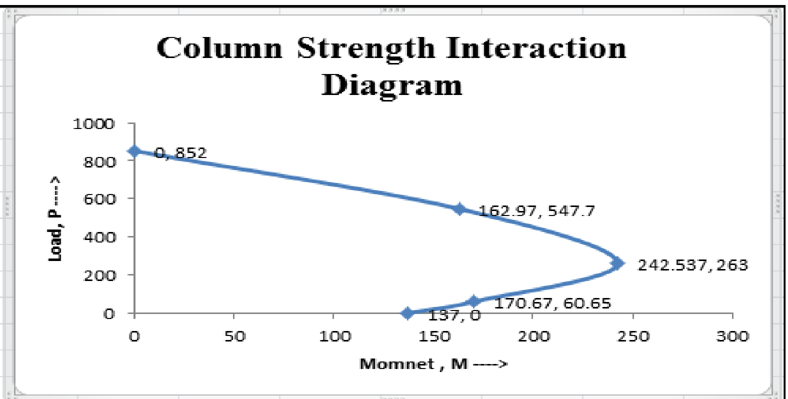


Figure 28: 18x10 column

By also changing column properties it also to possible to generate the values for the interaction diagrams. Figure 27 showed that the interaction diagram for a column having width 10 inch and height 18 inch. The next interaction diagram was where the concrete strength and also the yielding strength both were changed to 14 ksi and 80 ksi for a cross-section of 22x16 column.

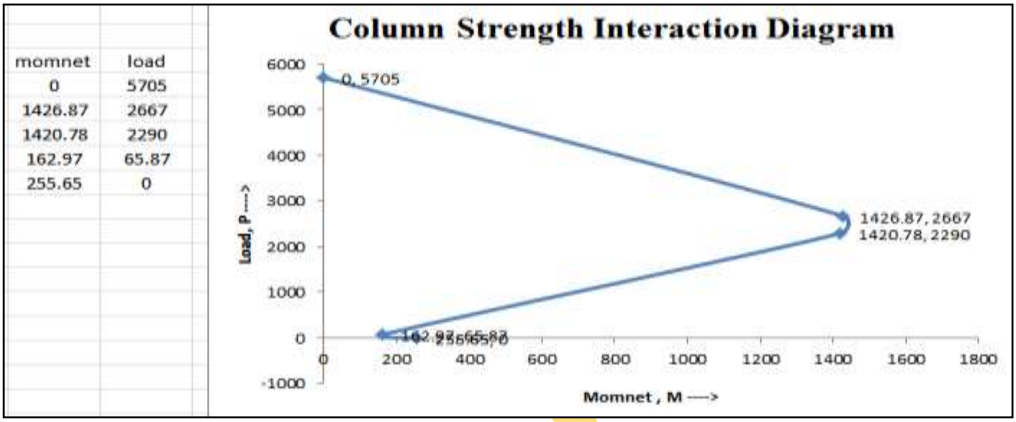


Figure 29: Column Strength Property Alteration

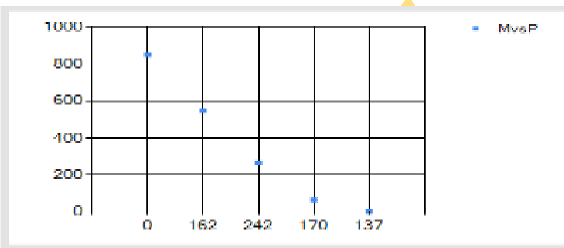


Figure 30: Application generated diagram

Where it would take pages upon pages to do these iterations, by using this program it was possible to do in the matter of seconds. Nothing in this world was without flaw. This program had some flaws that was unable to fix. The main flaws were, Making the program without implementing Gama , γ and fixing its depth, Making the program with only the generation of data not having the ability to make graph on its own window. The graph making calculations and method has to be dynamic in a sense to make the diagram possible. Even giving various combinations of parameters and procedures it was not possible to make the diagram within this program. Not showing the values of R_n, K_n for using the interaction diagram as a design aid for steel requirement. It was not impossible to fix these flaws, just by implementing a higher framework for declaring functions and also by using an updated library to make the dynamic generation of graphs would be possible.

Conclusion

The program was made for Civil Engineers and Civil Engineering students and also to help by giving the ability of calculating accurately and swiftly. The future of this program is vast. By developing the program the it would be available for multi-platform use by coding it with different languages, be available on handheld devices for quick check for a column and by giving it the ability to show a columns cross-section and reinforcement placement whether doubly and singly as well as the spacing between the reinforcements for the column.

REFERENCE

Al-Ansari S., (2005) "MATHCAD: Teaching and Learning Tool for Reinforced Concrete Design". Website: http://goo.gl/jX1sxl. Last visited on 2015.11.03.
Charlampakis E., Koumousis K., "Ultimate Strength Analysis of Composite Sections under Biaxial Bending and Axial Load". Website:https://goo.gl.St7Ra. Last visited on 2015.11.03.
Nilson H. N., Darwin D. and Dolan W. (2004) "Design of Concrete Structures, 14th edition".
Saadatmanesh H., Li W., "Strength and Ductility Of Concrete Columns Externally Reinforced with Fiber Composite Straps "Website:http://goo.gl/2jyZXP. Last visited on 2015.11.03.
Sfakianakis G., "Biaxial Bending with Axial Force of Reinforced, Composite and Repaired Concrete Sections of Arbitrary Shape by Fiber Model and Computer Graphics". Website: http://goo.gl/Jq78i4. Last visited on 2015.11.03.
Wai F.,"Analysis and Design of Reinforced Columns under Biaxial Loading". Website: http://goo.gl/kZ4qhS. Last visited on 2015.11.03.
Carpani C. Phd. (2006/7) "Civil Engineering Design: Design of Reinforced Concrete Columns". Website:http://goo.gl/YnM3AX. Last visited on 2015.11.03.
www.tutorialspoint.com (Last visited on 2015.11.03)
www.thenewboston.com (Last visited on 2015.11.03).
www.webcoachbd.com (Last visited on 2015.11.03).
www.w3school.com (Last visited on 2015.11.03).

--0--

Online Archive Link: https://abc.us.org/ojs/index.php/ei/issue/archive