Reliability-based Design Optimization of a Nonlinear Elastic Plastic Thin-Walled T-Section Beam

by

Mazen A. Ba-abbad

Dissertation submitted to the Faculty of Virginia Polytechnic Institute and State University in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in

Aerospace Engineering

Rakesh K. Kapania, Chair
Eric R. Johnson
Michael J. Allen
Robert L. West
Surot Thangjitham

March 2004

Blacksburg, Virginia

Keywords: Reliability-Based Optimization, Finite Element, Nonlinear Analysis, T-Beam, Elastic-plastic, Combined Loads, Flexural-Torsional Buckling, Reliability-Based Factor of Safety

Copyright © 2004, Mazen A. Ba-abbad
Reliability-based Design Optimization of a Nonlinear Elastic Plastic Thin-Walled T-Section Beam

Mazen A. Ba-abbad
Virginia Polytechnic Institute and State University
Blacksburg, VA 24061-0203

(ABSTRACT)

A two part study is performed to investigate the application of reliability-based design optimization (RBDO) approach to design elastic-plastic stiffener beams with T-section. The objectives of this study are to evaluate the benefits of reliability-based optimization over deterministic optimization, and to illustrate through a practical design example some of the difficulties that a design engineer may encounter while performing reliability-based optimization. Other objectives are to search for a computationally economic RBDO method and to utilize that method to perform RBDO to design an elastic-plastic T-stiffener under combined loads and with flexural-torsional buckling and local buckling failure modes. First, a nonlinear elastic-plastic T-beam was modeled using a simple 6 degree-of-freedom non-linear beam element. To address the problems of RBDO, such as the high non-linearity and derivative discontinuity of the reliability function, and to illustrate a situation where RBDO fails to produce a significant improvement over the deterministic optimization, a graphical method was developed. The method started by obtaining a deterministic optimum design that has the lowest possible weight for a prescribed safety factor (SF), and based on that design, the method obtains an improved optimum design that has either a higher reliability or a lower weight or cost for the same level of reliability as the deterministic design. Three failure modes were
considered for an elastic-plastic beam of T cross-section under combined axial and bending loads. The failure modes are based on the total plastic failure in a beam section, buckling, and maximum allowable deflection. The results of the first part show that it is possible to get improved optimum designs (more reliable or lighter weight) using reliability-based optimization as compared to the design given by deterministic optimization. Also, the results show that the reliability function can be highly non-linear with respect to the design variables and with discontinuous derivatives. Subsequently, a more elaborate 14-degrees-of-freedom beam element was developed and used to model the global failure modes, which include the flexural-torsional and the out-of-plane buckling modes, in addition to local buckling modes. For this subsequent study, four failure modes were specified for an elastic-plastic beam of T-cross-section under combined axial, bending, torsional and shear loads. These failure modes were based on the maximum allowable in-plane, out-of-plane and axial rotational deflections, in addition, to the web-tripping local buckling. Finally, the beam was optimized using the sequential optimization with reliability-based factors of safety (SORFS) RBDO technique, which was computationally very economic as compared to the widely used nested optimization loop techniques. At the same time, the SOPSF was successful in obtaining superior designs than the deterministic optimum designs (either up to 12% weight savings for the same level of safety, or up to six digits improvement in the reliability for the same weight for a design with Safety Factor 2.50).
Acknowledgements

The author wishes to express his gratitude for the guidance and advice of his advisers Dr. Rakesh K. Kapania and Dr. E. Nikolaidis, and of all the other respected committee members: Dr. Eric Johnson, Dr. Michael Allen, Dr. Robert West and Dr. Surot Thangjitham. Also, the author would like to express his gratitude to ONR STTR and Adoptech for funding this research, Dr. Zafer Gurdal, Mr. Mostafa Abdulla and Dr. Scott Ragon for their critique and advice. Finally, the author would like to express his deep appreciation and gratitude to his parents, family and friends for their support and help.
# Table of Contents

List of Figures........................................................................................................................................ix  
List of Tables .........................................................................................................................................xv  

Chapter 1. Introduction ..........................................................................................................................1  
1.1 Uncertainties in Engineering Design..............................................................................................1  
1.2 Designing with Uncertainty: Deterministic Design and  
    Reliability-Based Design..................................................................................................................2  
1.3 Reliability-Based Design Optimization (RBDO)............................................................................4  
   1.3.1 Concepts and History.................................................................................................................4  
   1.3.2 Application in Structural Design Optimization.................................................................6  
   1.3.3 Finite Element Analysis and Reliability..................................................................................7  
1.4 Scope of the Present Work..............................................................................................................7  

Chapter 2. Reliability Based Structural Optimization:  
    Background and Literature Review..................................................................................................9  
2.1 Reliability Calculation Methods....................................................................................................9  
   2.1.1 Simulation Based Reliability Methods: Monte-Carlo Simulation........................................11  
      2.1.1.1 The Direct Monte-Carlo Simulation..............................................................................12  
      2.1.1.2 Monte-Carlo Simulation with Importance Sampling................................................13  
      2.1.1.3 Other Monte-Carlo Simulation Sampling Methods.....................................................15  
   2.1.2 Analytical Reliability Approximation Methods.......................................................................16  
      2.1.2.1 The First Order Reliability Methods (FORM).................................................................16  
         2.1.2.1.1 The Cornell Reliability index....................................................................................16  
         2.1.2.1.2 The Hasofer and Lind Reliability Index.................................................................19  
      2.1.2.2 Other Analytical Reliability Calculation Methods......................................................22  
2.2 Reliability of Structural Systems and Structural Components with  
    Multiple Failure Modes..................................................................................................................23  
2.3 Approximations of Performance Functions....................................................................................24  
2.4 Reliability Based Structural Optimization......................................................................................27  
   2.4.1 Optimization Methods..............................................................................................................27  
   2.4.2 Standard Mathematical Statements of the Two RBDO
Problems and their Classical Solutions…………………………………..30

2.4.3 Reliability-Based Design Optimization
Methods……………………………………………………………………….34

2.4.4 A Modified Sequential Optimization with Reliability-Based Factors of Safety (SORFS) RBDO Approach…………………..40

Chapter 3. Nonlinear Finite Element Model of Six Degrees of Freedom
Elastic-Plastic Beam……………………………………………………….43
3.1 Introduction…………………………………………………………………….43
3.2 Six Degrees of Freedom Elastic-Plastic Beam Element……………………….44

Chapter 4. RBDO Exploratory Example: Elastic-Plastic T-Beam…………………..54
4.1 Failure Modes………………………………………………………………….54
4.2 Deterministic Structural Optimization…………………………………………63
4.3 Reliability-Based Structural Optimization……………………………………..64
4.4 Numerical Application…………………………………………………………67
4.5 Discussion of Reliability-Based Optimization Results………………………...82
4.6 Summary and Conclusions…………………………………………………….85

Chapter 5. Application of the Sequential Optimization with
Probabilistic Safety Factors RBDO to the Exploratory
Elastic-Plastic T-Beam Example……………………………………………..86
5.1 Introduction…………………………………………………………………………..86
5.2 The Optimization Problem………………………………………………………….86
  5.2.1 Problem Statement…………………………………………………………...86
5.3 The Sequential Optimization with Probabilistic Safety Factors Approach………….88
5.4 Reliability-Based Optimization Results for the Exploratory
  Elastic-Plastic T-Beam Example using the SOPSF Approach…………………..89
5.5 Discussion of the Results…………………………………………………………….93

Chapter 6. Nonlinear Finite Element Model of 14 Degrees of Freedom
Elastic-Plastic Beam……………………………………………………….94
6.1 The element loads………………………………………………………………..94
6.2 Cross-Section Deformation Modeling………………………………………….95
6.3 Deformation field approximation……………………………………………….102
Appendix A: FORTRAN 90 Code Listing for the 6 DOF
Nonlinear Elastic-Plastic T-Beam..............................................191
Appendix B: Backward Euler Method to Obtain the Plastic Stresses.........202
Appendix C: FORTRAN 90 Code Listing for the 6 DOF Nonlinear
Elastic-Plastic T-Beam.................................................................208
Vita..............................................................................................232
List of Figures

Figure 2.1 Graphical Representation of the Stress-Strength Interference.................10

Figure 2.2 The Original Probability Distribution Function $f_X(\bar{X})$ and Importance
Sampling Function $h_i(\bar{v})$ in $x_1$ and $x_2$ Space.............................................14

Figure 2.3 The Probability Distribution Function $f_U(\bar{U})$ and the
Performance Function in the standard $u_1$ and $u_2$ Space...........................................20

Figure 2.4 Flow Chart of the Sequential Safety-Factor Based RBDO Methods..........39

Figure 2.5 Flow Chart of the Modified Sequential Safety-Factor Based RBDO
Methods......................................................................................................................42

Figure 3.1 The Example Beam under the Applied Loads.................................46

Figure 3.2 Six Degrees of Freedom Beam Element.................................................46

Figure 3.3 The Added Moment due to the Axial Load............................................49

Figure 3.4 Newton-Raphson Iterations Combined with Incremental Predictor........51

Figure 3.5 Plastic Zones Developed in a T-Beam Cross-section for
Increasing Values of the Axial and Bending Loads.................................................52

Figure 3.6 Mid-Span Deflection of a Simply Supported Beam of Rectangular
Cross-section under Lateral Load (see Yang and Saigal (1984)).........................54

Figure 4.1 The State of Stress for a T-Beam under the Combination
of Axial Load and Bending Moment that Causes the Whole
Cross-Section to Deform Plastically.................................................................55

Fig 4.2.a The Safe Design Zone (under the yield surface) that Combines the Axial,
Bending and Torsional Shear Loads....................................................................62

Fig 4.2.b The Projection of the Design Point on the M-P Plane.........................63

Figure 4.3.a Geometry and Dimensions of the T beam......................................69

Figure 4.3.b The Loads Applied to the Beam .......................................................69
Fig 4.4.a Load-Deflection Curve for a T Beam, Used here under Axial and Lateral Loads.................................70

Fig 4.4.b Location of the Failure Point on the Axial Load-Bending Moment Interaction Curve ...............................................................71

Figure 4.5 Comparison Between the Results of Non-Linear FEM and the Linear Elastic Formula (Eq. 4.26)..........................................................72

Figure 4.6 The Boundary Performance Functions G1(X) = 0, G2(X) = 0 and G3(X) = 0, Plotted on the Random Loads Plane (the loads are normalized by their mean values).................................74

Figure 4.7.a Effects of Varying the Design Variables on the Weight and Probability of Failure of the Deterministic Optimum (SF 1.50)..............76

Fig 4.7.b Searching for the Optimum Design that has the Lowest Probability of Failure (Pf)...............................................................77

Figure 4.8.a Effects of Varying the Design Variables on the Weight and Probability of Failure of the Deterministic Optimum (SF 1.75)..............78

Fig 4.8.b Searching for the Optimum Design that has the Lowest Probability of Failure (Pf)...............................................................78

Figure 4.9.a Effects of Varying the Design Variables on the Weight and Probability of Failure of the Deterministic Optimum (SF 2.00)..............80

Fig 4.9.b Searching for the Optimum Design that has the Lowest Probability of Failure (Pf)...............................................................80

Figure 4.10.a Effects of Varying the Design Variables on the Weight and Probability of Failure of the Deterministic Optimum (SF 2.25)..............81

Fig 4.10.b Searching for the Optimum Design that has the Lowest Probability of Failure (Pf)...............................................................81

Figure 4.11.a Effects of Varying the Design Variables on the Weight and Probability of Failure of the Deterministic Optimum (SF 2.50)..............82

Fig 4.11.b Searching for the Optimum Design that has the Lowest Probability of Failure (Pf)...............................................................82

Figure 4.12.a Comparing the Reliability of the Deterministic and the Reliability-Based Optimum Designs.................................................84
Figure 4.12.b Weight Saving with Reliability-Based Design over the Deterministic Design.................................................................84

Figure 5.1 The Beam Loads and Dimensions.........................................................87

Figure 5.2 Comparing the Reliability of the Deterministic and the Reliability-Based Optimum Designs.......................................................90

Figure 5.3 Weight Saving of the Reliability-Based Design over the Deterministic Design.................................................................91

Figure 6.1 A Prismatic Beam of Arbitrary Open Cross-Section under the Action of a System of Loads.................................................................94

Figure 6.2 Beam Element Deformations under a System of Loads.........................95

Figure 6.3 Displacements of a Point P in the Beam Cross-Section.........................96

Figure 6.4 Axial, Tangential and Normal Components of Displacements at a Point in Beam Wall.................................................................98

Figure 6.5 Shear Strain Resulting from Axial and Tangential Components of Displacement.................................................................99

Figure 6.6 The Longitudinal Strain......................................................................102

Figure 6.7 The Beam Element Generalized Displacements and Forces................105

Figure 6.8 Resistance to Torsion Provided by an I-Beam......................................114

Figure 6.9 Resistance to Torsion Provided by a T-Beam......................................114

Figure 6.10 Flow Chart for the Stress Calculation Algorithm..............................124

Figure 6.11 Flow Chart of the Stress Integration Procedure...............................126

Figure 6.12 Simply Supported Rectangular Beam under Uniform Lateral Load $q_o$ (v the central in-plane deflection).................................127

Figure 6.13 The Central Deflection of a Simply Supported Rectangular Elastic-Plastic Beam........................................................................128

Figure 6.14 The Cantilever Beam Tested by Woolcock and Trahair (1974) and analyzed by Ronagh and Bradford (1999).................................129
Tip Load (Nonlinear Elastic Analysis).................................................................146

Figure 7.9 Stress Contours for a Nonlinear Elastic-plastic Analysis of a Cantilever T-beam under In-plane Tip Load (localized deformations have developed).................................147

Figure 7.10 Web Local Buckling Failure in a Cantilever T-section Beam under Vertical Tip Load.................................................................148

Figure 7.11 Local Buckling of the Web and Flange of a Cantilever Beam of T-section under Axial Load (ABAQUS continuum element model linear buckling analysis).................................149

Figure 7.12 Local Buckling of the Web and Flange of a Cantilever Beam of T-section under Uniform Lateral Pressure Load Applied at its Flange (ABAQUS continuum element model linear buckling analysis).................................................................150

Figure 7.13 Cantilever Beam of T-section under Torque Load at its Free End (ABAQUS shell model).................................................................151

Figure 7.14 Comparison between the Probabilities of Failure for Each Failure Mode for the Improved Reliability Optimum Design and the Deterministic Optimum Design SF 1.50 in the Safety Index (β) Space........................................164

Figure 7.15 Comparison between the Probabilities of Failure for Each Failure Mode for the Improved Reliability Optimum Design and the Deterministic Optimum Design SF 1.75 in the Safety Index (β) Space........................................166

Figure 7.16 Comparison between the Probabilities of Failure for Each Failure Mode for the Improved Reliability Optimum Design and the Deterministic Optimum Design SF 2.00 in the Safety Index (β) Space........................................168

Figure 7.17 Comparison between the Probabilities of Failure for Each Failure Mode for the Improved Reliability Optimum Design and the Deterministic Optimum Design SF 2.25 in the Safety Index (β) Space........................................170

Figure 7.18 Comparison between the Probabilities of Failure for Each Failure Mode for the Improved Reliability Optimum Design and the Deterministic Optimum Design SF 2.50 in the Safety Index (β) Space........................................172

Figure 7.19 Weight Saving of the Reliability-Based
Optimization over the Deterministic Optimization for Different Safety Levels………………………………………………………………173

Figure 7.20 Comparing the Reliability of the Deterministic and the Reliability-Based Optimum Designs……………………………………173
List of Tables

Table 4.1 Deterministic Optimum Designs for Various Safety Factors

Table 4.2 Probabilities of Failure for the Three Performance Functions ($G_1(X)$, $G_2(X)$, and $G_3(X)$) Using Monte-Carlo Simulation with Importance Sampling (IS) and by Using H-L Method for the Deterministic Optimum Design with SF of 1.50

Table 4.3 Probabilities of Failure for the Deterministic Optimum Designs (SF of 1.5, 1.75, 2.00, 2.25, and 2.50)

Table 5.1 Convergence of RBDO of the T-Stiffener Example (finding the least weight for a certain level of safety) for SF 2.50 using the SORFS Method

Table 5.2 Convergence of RBDO of the T-Stiffener Example (finding the highest reliability for a certain weight) for SF 2.50 using the SORFS Method

Table 7.1 Deterministic Optimum Designs for Various Safety Factors

Table 7.2 the MPP loads for each failure mode presented as a multiple of the nominal loads

Table 7.3 Results of the RBDO for SF 1.50

Table 7.4 Results of the RBDO for SF 1.75

Table 7.5 Results of the RBDO for SF 2.00

Table 7.6 Results of the RBDO for SF 2.25

Table 7.7 Results of the RBDO for SF 2.50
Chapter 1: Introduction

1.1 Uncertainties in Engineering Design

In engineering design, there are a number of uncertainties that result from the variability of applied loads and material properties, in addition to that resulting from the design modeling. Also, during manufacturing, a number of uncertainties arise from the manufacturing processes and from the material selection. The problem of uncertainty in estimating the loads can be clearly noticed in the aerodynamic and hydrodynamic loads that are considered in the design of aircrafts and ships. As these loads are not known exactly, aerospace and naval designers have to consider some kind of statistical representation of these loads. Also, it is known that the material properties documented in the material specification handbooks and manuals are not the exact properties of the actual product. Since these documented material properties are taken from the averages of the measured experimental data. Also, the experiments were performed in laboratory conditions that may differ from the situation in hand.

In addition, some material defects, such as micro-cracks and voids, can seriously weaken the material and may be very difficult to detect. Likewise, in some applications, the material may experience environmental deterioration (such as corrosion and abrasion), that may not be quantified with certainty. Moreover, there are uncertainties in manufacturing the components of engineering systems despite the quality control measures that are applied. The reason is that inspection for quality is performed most of the time on some random samples of the components and not on all the components; otherwise, the cost will be prohibitively high. Finally, in modeling the engineering systems a number of idealizations may be made to simplify the analysis, which will result in reducing the accuracy in representing the real system. However,
these idealized engineering system designs are rarely produced with the exact dimensions specifications, since tolerances must be given for material processing and fabrication, which adds to the level of uncertainty in the design.

1.2 Designing with Uncertainty: Deterministic Design and Reliability-Based Design

A designer must deal with the existence of uncertainties in the product in a manner that will make the design perform as expected in a safe and reliable way. For that purpose, two methods have been developed over the years to quantify the uncertainties in an engineering product and their effects on the design level of safety. The first approach is the deterministic design, in which it is assumed that all the information about the design is known and a conservative assumption is made to compensate for any unaccounted factors. Accordingly, a factor of safety or a load factor is assigned to either the material strength or the applied loads and it is the representation to the design level of safety. The second approach is the reliability-based design, in which it is assumed that the information about the design is known to be within certain bounds and have known distributions of probability.

Accordingly, the probability of survival of the design is calculated and can be considered to represent the level of safety of the design. The deterministic approach considers average values for the loads and material properties. It does not consider quantitatively the frequency of occurrence of some particular values of the loads and material properties during the life-span of the product. Instead, a subjective value is assigned to the factor of safety or the load factor on a qualitative basis. Also, the deterministic approach becomes highly subjective in the sense that
factor of safety is usually estimated and modified according to the cost and the consequences of failure of a particular component. In addition, the deterministic approach does not use a quantitative method for combining variable loads that are applied to the design (see Mori et al, 2003.) Thus, the level of safety of an engineering design that has multiple components that differ in their required level of safety will not be consistent, because the overall level of safety of a system cannot be determined from adding the safety factors of the individual components. As a result, an overestimation or underestimation of the design level of safety will be made. Finally, it must be noted that the deterministic approach does not identify the parts or regions of the design that may fail much earlier than others, and also, does not identify the more critical loads or design variables since all the uncertainties are covered by one factor throughout the design (Long and Narciso, 1999).

On the other hand, the reliability of an engineering design is calculated using statistical analysis and probability principles to the samples of the expected service loads and the properties of the material used in the design. The uncertainties are modeled by randomly distributed variables, in which the frequency of occurrence of each possible value of the variable is considered. Specifically, the most repeated values of a random variable are associated with the highest values in the probability distribution function. However, it must be noted that there are some experience-based assumptions that are made in determining the type and shape of the probability distribution of each random variable, since it is impossible to perform experiments that cover all the possible values of a random variable (Hess et al, 1994; Sundararajan, 1995; White et al, 1995; and Melchers, 2001). Nevertheless, the accuracy of the model used to represent the actual data increases as the volume of the available statistical data increases. For
example, flight loads spectra can now be generated with considerable accuracy (Herb, 1995). Also, by tightening the quality measures and by using more accurate models, a better assessment of the design safety may be achieved (Der Kiureghian, 1989.) Once the uncertainties are modeled, a consistent level of safety can be obtained for the engineering design. In particular, the failure of an engineering system that has multiple components with different safety levels may be defined by combining the reliabilities of the components as in the following. In the case of independent failures, the reliabilities can be combined in serial (failure of one component makes the whole system fail) and/or in parallel (failure of all components makes the whole system fail.) Also, in the case of correlated failures, joint probabilities of failure can be calculated for the design (Ang and Tang, 1975; and Melchers, 2001). Hence, the uncertainties present in the design can be quantified and calculations of the safety level can be performed in a more coherent manner.

Finally, it is appropriate to mention some other methods of design under uncertainty that do not consider explicit safety measures, but instead calculate estimates of the uncertain design variables, such as the fuzzy sets (membership functions), interval methods (see Muhanna, and Mullen, 2001; and Rao and Berke, 1997), evidence theory (Bae et al, 2003) and possibility theory (Moeller et al, 1999).

1.3 Reliability-Based Design Optimization (RBDO)

1.3.1 Concepts and History

Given that one is able to determine the reliability of a certain engineering design, it may be prudent to perform optimization to search for the best design (the safest, least cost, or least weight … etc.) while satisfying certain restrictions. In fact two types of reliability-based optimization
problems are considered a) finding the most reliable design within certain design (or cost) constraints, and b) finding the best design that has a constraint on the minimum value of reliability. However, despite that the RBDO has a more consistent description of the safety of designs; the field of design optimization is still dominated by deterministic methods.

One of the reasons that has slowed the acceptance of the RBDO by industry is that its computational cost is much higher as compared to the cost of employing a deterministic approach to the design of various components. In the case of the deterministic design, the analysis calculations are performed for only one set of parameters to yield the design response to the applied loads, and then this process is repeated for a different set of variables in the optimization iterations. In reliability-based design, on the other hand, the analysis calculations are repeated several times just to calculate the reliability of a design, and then this process is repeated again for optimization. Hence, there is a need to perform reliability-based optimization, by using only a limited number of analyses. A second reason that may have stalled the use of RBDO, is that the reliability-based optimization calculations may not converge all the time. Frangopol (1995) has pointed out that RBDO problems were solved with varying degrees of success (this reference also contains lists of references that cover the development of the reliability-based design from the 1973 to 1993). Also, this problem was clearly addressed by Royset et. al.(2001), who explained that the reason for this failure is that the reliability function may be highly non-linear and/or non smooth (have discontinuous derivatives), and thus, proposed a procedure to avoid this difficulty. Their procedure requires reformulating the reliability-optimization problem in such a way that will make it solvable by semi-infinite optimization methods, in which the search for the optimum solution considers a finite domain (e.g. a circle or a sphere) and then uses a semi-infinite algorithm to search for the optimum
solution within this finite domain (see Polak, 1997). Still, Royset et al. concluded that the success of their proposed method depends on how the reliability calculations modify the optimization sub-problem.

Wang et al. (1995) tried to overcome the reliability function non-linearity and derivative discontinuity problem by using a piece-wise approximation of the reliability function and they proposed approximating the design constraints using multivariate splines and then using these splines to perform the reliability-based optimization. However, the spacing between the nodes of the splines needs to be optimized to yield the maximum computational economy. Earlier attempts did not address the issue of non-smooth objective and constraint functions and used optimization methods that are either based on the derivatives of the reliability function (Nikolaidis et al., 1988; Li and Yang, 1994; Nikolaidis and Stroud, 1996; Rajagopalan and Grandhi, 1996; Pu et al., 1997; Nakib, 1997) or even based on random search of the design space (Natarajan and Santhakumar, 1995; and Cheng et al., 1998). Thus, to perform RBDO, a method that can be both computationally economical and handle highly non-linear functions that have derivative discontinuities should be used.

1.3.2 Application in Structural Design Optimization

Today’s structural designer has to consider a variety of structural demands in addition to insuring the safety of the structure. Among these demands that concern the aerospace and ship industries are weight and cost saving, the efficient use of materials (especially composite materials), and the efficiency in energy consumption. However, saving structural weight, and at the same time, maintaining an acceptable level of safety is critical in many aerospace and naval applications. Thus,
RBDO can be applied to aerospace and naval structures, provided that it can produce better designs (safer or lighter designs) than those obtained by the deterministic design optimization that is currently in use. The structural optimization can be carried out to find the optimum dimensions (sizing), configuration (e.g. the orientation of truss members, or fibers in composite materials), or topology (e.g. the number of stiffeners in a stiffened panel, or number of spars and ribs in a wing), see Frangopol (1995).

1.3.3 Finite Element Analysis and Reliability

The use of finite element analysis (FEA) has become common for analyzing complex structural designs that have no closed-form solutions. FEA has also become very popular in analyzing structural elements under complex loads. However, to use the FEA in calculating the reliability of a design, it has to be formulated based on the nature of uncertainties. For instance, in the case of uncertainty in the applied loads the usual deterministic FEA can be employed and the results are then processed to calculate the reliability. On the other hand, if the material properties have uncertainties in their spatial distribution (as the case of non homogenous distribution of strength or modulus of elasticity) special formulations for FEA must be employed (Liu et al, 1995; Elishakoff et al 1995, and 1997; Melchers, 2001; and Kapania and Goyal 2002)

1.4 Scope of the Present Work

The aim of this work is to explore the potential of the RBDO in providing superior structural designs when compared with the present deterministic design and optimization methods. Also, to obtain a better understanding to the problems that hinder the total acceptance of RBDO in the industry. Particularly, the aspects of computational economy (i.e. the method is performed with as
minimum code runs as possible), robustness (i.e. the optimization process can be carried with
minimal designer interference), and the possible optimization limitations (i.e. when is it not possible
to get a design that is better than the deterministic optimum design). Also, this work aims to
compare the state-of the art methods that are used to perform RBDO and tries to point out the most
promising method out of them. For this purpose, a practical example (a beam of T-section) was
devised. First, the beam example will be used to perform some exploratory study of the RBDO
process, which addresses the above mentioned aspects. Then, one of the state of the art RBDO
techniques will be modified and applied to the exploratory beam example. After that, more involved
failure modes will be added to the beam. Finally, the updated beam model will be used to compare
results obtained using one of the standard RBDO methods and one of most promising state-of the art
RBDO methods.

As a preparation for the exploratory and advanced RBDO applications, a background and
literature review of RBDO will be presented in Chapter 2. Next, an elastic-plastic six degrees of
freedom nonlinear beam finite element, which was used in the exploratory study, is presented in
Chapter 3. Then, the RBDO exploratory study is presented in Chapter 4. Chapter 5 contains an
application of the sequential optimization with reliability-based factors of safety (using the
coordinates of the most probable failure points as an approximate safety factors) RBDO (see Chen et
al. (1997); Wu et al. (2001); Du and Chen (2002); and Qu and Haftka (2003) technique to the same
exploratory beam example of Chapter 4. Then in Chapter 6, we will develop an elastic-plastic,
fourteen degrees of freedom, nonlinear beam finite element that considers the torsional effects on
beam failure. Chapter 7 presents reliability-based optimization of the updated beam model. Finally,
Chapter 8 presents some concluding remarks and suggestions for future research.
Chapter 2

Reliability Based Structural Optimization:

Background and Literature Review

The process of reliability-based optimization is concerned with using a computer-based model to calculate the reliability of the design, and employing a suitable optimization method to obtain an optimum design (the safest with the least cost or weight). However, as we are going to see later, it generally requires an extensive computing effort to calculate the reliability of one design; this step is repeated several times to obtain an optimum design. Also, the success and efficiency of the RBDO process depends on the way the problem is formulated and suitability of the RBDO algorithms to the nature of the design. Thus, the focus is to seek methods that are both computationally efficient and have a wide range of applicability. In the following sections these issues will be introduced with a focus on the methods that were devised for this work. Discussing the computer-based models will be delayed until the next chapter.

2.1 Reliability Calculation Methods

Since the goal is to determine the level of safety of a design in its service environment, it would be very expensive to apply actual loads to the design and calculate the chance of failure. Instead, computer based analytical models are used to simulate the behavior of the design under different conditions, and then, the results are used to calculate the reliability of the design. The number of simulation runs is related to the number of uncertainties present in the design parameters or loads. Accordingly, the design is evaluated every time with a different set of values of the random variables. In turn, the values of the random variables that enter every evaluation run are
selected according to their respective probability distribution. After that, statistical methods are used to evaluate the output and predict the reliability of the design.

For example, if we consider a simple design such as a rod that has a strength $R$ that may vary from one rod to the other (depending on some unknown manufacturing conditions). However, it was known that from previous samples, a statistical distribution of the rod’s strength was obtained. Then the strength of the rod can be represented by a randomly distributed variable with a known probability density function (PDF). Now, if we have a stress $S$ that acts on the rod that has a magnitude that varies randomly, we know that the event of failure occurs when:

$$R - S < 0$$  \hspace{1cm} (2.1)

Now, to calculate the reliability of the rod we need to calculate the probability of occurrence of the failure event, which can occur at any point inside the failure domain $\Omega$ that is represented graphically by the shaded area in Figure 2.1.

![Figure 2.1 Graphical Representation of the Stress-Strength Interference](image)

So, in the case that the rod’s strength and the applied stress values are statistically independent, the probability of failure $P_f$ for this simple example can be calculated from the following integral:
\[ P_f = \iint_{\Omega} f(r)f(s) \, dr \, ds \quad (2.2) \]

However, despite that the integral of equation 2.2 can have a direct analytical solution for some special cases, or can be performed numerically for some other cases (Sundararajan and Witt, 1995), in most real life situations the integral can not be evaluated directly. In many cases the failure domain \( \Omega \) may not have an analytical expression and the problem gets more complicated as the number of random variables increases. Thus, other methods such as the simulation-based reliability methods (Monte-Carlo Simulation methods), or the analytical reliability approximation methods (e.g. the first and second order reliability methods, and the advanced mean value method) must be employed, and are presented in the following sub-sections.

### 2.1.1 Simulation Based Reliability Methods: Monte-Carlo Simulation

To perform Monte-Carlo Simulation (MCS) to calculate reliability we need an analytical or an approximate model of the physical system that can be defined in the computer environment. Then, the analysis needs to be evaluated numerous times. In each analysis run, a different set of values of the design random variables will be used. These random variables are selected in each evaluation according to their respective probability distribution, and they are generated using computer built-in standard random number generators. These, computer built-in functions generate the random variables in two steps. The first step generates random numbers that are uniformly distributed between 0 and 1, which is done either by using the bits and binary digits of the computer or by using some special mathematical formulas. Once the uniformly-distributed numbers are generated, the random variables can be produced by either the inverse transformation method or some other statistical methods (Ayyub and Mccun 1995, and Fishman 1996).
Now, to evaluate the reliability of the design (i.e. $1 - P_f$) from equation 2.2 using Monte-Carlo Simulation MCS, researchers have developed a number of methods, such as the direct Monte-Carlo Simulation MCS, or the MCS with variance reduction techniques. We will present both of these approaches next.

### 2.1.1.1 The Direct Monte-Carlo Simulation

The direct MCS is the simplest, and at the same time, the most computationally expensive approach to perform a probabilistic analysis of a component or a system. It calculates the probability of failure $P_f$ as the ratio of the failure trials (e.g. when the load exceeds the resistance of the structure) to the total number of trials i.e.,

$$P_f = \frac{N_f}{N_{total}} \quad (2.3)$$

The accuracy in estimating $P_f$ is represented by the Coefficient Of Variation (COV) of $P_f$, which can be calculated by assuming each simulation cycle to constitute a Bernoulli trial and can be obtained from:

$$COV(P_f) = \sqrt{\frac{(1 - P_f)P_f}{N_{total} \cdot P_f^2}} \quad (2.4)$$

(Ayyub and Mccun 1995; Melchers 2001). It is apparent that as the number of trials increases, the accuracy in calculating $P_f$ increases. However, this would become prohibitively expensive as the required level of safety increases. Therefore, some other techniques must be used to reduce the required number of trials. These techniques include the variance reduction methods (e.g. Importance Sampling, and Latin Hypercube Sampling (Olsson and Sandberg 2002; Olsson et al. 2003)) and
other methods that depend on the physical nature of the design such as the proposed method of Dasgupta (2000).

2.1.1.2 Monte-Carlo Simulation with Importance Sampling

The importance sampling method (ISM) is one of the popular methods that are applied to reduce the number of MCS simulation runs required to calculate the probability of failure by reducing the variance of the calculated reliability value, and thus, improving its convergence. This method modifies the sampling process in the direct MCS by using modified probability distribution functions for the random variables, which is called the sampling density function or the importance function. Then, the computer generates the values of the random variables \( \tilde{v} \) according to the defined importance sampling function \( h_v(\tilde{v}) \) and the analysis is performed according to these values.

From the analysis, we calculate the design performance function \( G(\tilde{X}) \) (where \( \tilde{X} \) is the vector of random design variables), which defines the safe \( G(\tilde{X}) > 0 \) and the failure \( G(\tilde{X}) \leq 0 \) zones in the design variables space. Finally, the probability of failure is calculated from:

\[
P_f \approx \frac{1}{N} \sum_{j=1}^{N} I[G(\tilde{v}_j) \leq 0] \frac{f_x(\tilde{v}_j)}{h_v(\tilde{v}_j)}
\]  \hspace{2cm} (2.5)

where \( N \) is the total number of trials, \( I[\ ] \) is an indicator that is assigned a value of 1 for the failure event of the \( j^{th} \) trial (when \( G(\tilde{v}_j) \leq 0 \) ) and 0 for the event of safe design (when \( G(\tilde{v}_j) > 0 \) ), and \( f_x(\tilde{v}_j) \) is the original joint probability distribution function evaluated at the importance sampling values \( \tilde{v}_j \). Then the error in estimating the probability of failure is obtained from (see Melchers, 2001)
\[ COV(P_f) = \frac{J_1 - J_2^2}{N} \quad (2.6.a) \]

where

\[
J_1 = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \left[ I[G(\mathbf{X}_j) \leq 0] \frac{f_X(\mathbf{X}_j)}{h_r(\mathbf{X}_j)} \right]^2 h_r(\mathbf{X}_j) d\mathbf{X} \quad (2.6.b)
\]

\[
J_2 = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \left[ I[G(\mathbf{V}_j) \leq 0] \frac{f_X(\mathbf{V}_j)}{h_r(\mathbf{V}_j)} \right] h_r(\mathbf{V}_j) d\mathbf{V} \quad (2.6.c)
\]

From equations 2.6, we can clearly see that the choice of the importance sampling function is important to the convergence of the calculation of the probability of failure. In fact, some choices may result in slower convergence than the direct MCS. Therefore, Melchers (2001) suggests use of a sampling probability distribution function \( h_r(\mathbf{V}) \) that has its mean value centered at the most probable failure point; \( \mathbf{X}^* \) as shown in Fig 2.2 for a design space that has two random variables \( x_1 \) and \( x_2 \).

Figure 2.2 The Original Probability Distribution Function \( f_X(\mathbf{X}) \) and the Importance Sampling Function \( h_r(\mathbf{V}) \) with mean value at MPP in \( x_1 \) and \( x_2 \) Space.
Yet, it is not easy in most cases to determine the most probable failure point a priori. However, it may be found by some numerical maximization techniques. However, as we will see in section 2.1.2, we may use a suitable FORM method to determine the most probable failure point.

2.1.1.3 Other Monte-Carlo Simulation Sampling Methods

As mentioned earlier, researchers have developed other methods to reduce the required number of trial points. For instance, the Latin Hyper Cube Sampling (LHS) method has some popularity (Ayyub and McCun 1995, Olsson and Sandberg 2002; Olsson et al. 2003). However, as pointed out by Olsson et al. (2003), its efficiency over the direct MCS is realized in the case that the probability of failure is dominated by a single random variable. However, they propose using LHS with the importance sampling method to increase its efficiency. Finally, other sampling techniques exist, but they did not get much popularity among researchers and can be found in the literature. For example, researchers have used Stratified sampling, adaptive sampling, and directional sampling techniques see Ayyub and McCun (1995), Moony (1997), and Melchers (2001) for a review of some of these methods.

2.1.2 Analytical Reliability Approximation Methods:

To calculate the reliability of a design we may need to use a suitable way to approximate the integral in equation 2.1. However, depending on the types of the functions involved and the shape of the failure region $\Omega$, we may be able to approximate the reliability by using first or second order Taylor series approximation to the limit state function (or the performance function) $G(X)$. In the following sub-sections we are going to present one of the widely used methods and refer to other existing methods.
2.1.2.1 The First Order Reliability Methods (FORM)

This method is a development of the fast probability integration method FIP (see Haskin et al, 1996.), and the name first order reliability method comes from approximating the performance function $G(X)$ by a first order Taylor series. Also, when we are only considering the first two moments of the random variables (for normally distributed random variables, the first moment is mean value, and the second is the variance), and ignoring the higher moments (i.e. skewness, flatness …etc.), then these methods are called the first-order second-moment methods (FOSM). However, before we present some of the FORM methods, it is appropriate to define the Cornell safety (or reliability) index ($\beta$).

2.1.2.1.1 The Cornell Reliability index

The Cornell reliability index was the first analytical approximation method to calculate the probability of failure, and it had paved the way for other methods that have a wider domain of application (Ang and Tang, 1975). To introduce it in a simple way, let’s recall the simple example of the rod under load presented in Sec. 2.1, where we had the following simple limit state function

$$G(\bar{X}) = R - S \quad (2.7)$$

Assuming that $R$ and $S$ are statistically independent and normally distributed random variables, we may define a new random variable $Z$ with the following properties (Ang and Tang, 1975):

$$Z = R - S$$
$$\mu_Z = \mu_R - \mu_S$$
$$\sigma^2_Z = \sigma^2_R + \sigma^2_S \quad (2.8)$$
where $\mu_z$ and $\sigma_z$ are the mean value and the standard deviation of the random variable $Z$ respectively. Then the probability of failure can be calculated from

$$P_f = P[Z < 0] = \Phi\left(-\frac{\mu_z}{\sigma_z}\right) = \Phi(-\beta) \quad (2.9)$$

where $\Phi(\cdot)$ is the cumulative distribution function for a standard normal variable, and $\beta$ is the safety index. The same concept can be generalized to the case of more than two random variables and to the case of nonlinear performance function and this can be done by Taylor series expansion of the performance function around the mean values of the random variables as in the following:

$$Z = g(\bar{X}) + \sum_{i=1}^{n} \frac{\partial g}{\partial x_i}(x_i - \bar{x}_i) + \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{\partial^2 g}{\partial x_i \partial x_j}(x_i - \bar{x}_i)(x_j - \bar{x}_j) + \cdots \quad (2.10)$$

where $g(\bar{X})$ is the performance function evaluated at the mean values of the random variables, and $\bar{x}_i$ is the mean value of the random variable $x_i$. Then, if we truncate the series at the linear terms, the first approximate mean value and the variance of $Z$ will be given by

$$\mu_Z \approx g(\bar{X})$$

$$\sigma_Z \approx \left(\sum_{i=1}^{n} \sum_{j=1}^{n} \frac{\partial g}{\partial x_i} \frac{\partial g}{\partial x_j} \text{Cov}(x_i, x_j)\right)^{\frac{1}{2}} \quad (2.11)$$

where $\text{Cov}(x_i, x_j)$ is the coefficient of variation for the random variables $x_i$ and $x_j$. 


Also, a better estimation of the mean value of $Z$ can be obtained from considering the square term in the Taylor series

$$\mu_Z \approx g(\bar{X}) + \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{\partial g}{\partial x_i \partial x_j} Cov(x_i, x_j)$$  \hspace{1cm} (2.12)$$

However, the second order variance requires obtaining the higher moments of the random variables, which may not be available in practical situations (Ang and Tang, 1975)

Finally, the safety index $\beta$, and the probability of failure can be determined from (Eq. 2.9) as before. Also, since the limit state function is linearized around the mean value, the safety index method is also known as the mean value first-order second-moment (MVFOSM). However, it is important to know that the estimation of the probability of failure using the safety index can only give accurate values for special cases, particularly, when the performance function is a simple addition or multiplication of the statistically independent random variables (see Chang et. al., 1998). Also, it gives different reliability values for the same design problem if the formulation of the performance function is changed to an equivalent formulation (i.e. this reliability calculation method lacks invariance). Thus, there was a need to develop some improved methods that avoid this problem (some of them will be presented shortly). Yet, these improved methods are based on the safety index idea, which from equation 2.9 we can see that it gives a qualitative measure of safety, in the sense that larger values of $\beta$ means safer design, and vice versa.
2.1.2.1.2 The Hasofer and Lind Reliability Index

The Hasofer and Lind (H-L) reliability index is one of the most widely used reliability calculation methods (Madsen et al., 1986; Nikolaidis and Burdisso, 1987; Yang, 1989; Enevoldsen, and Sorensen, 1994; Haldar and Mahadevan, 1995; Nikolaidis and Stroud, 1996; Barakat, et al., 1999; Der Kiureghian, 2000, Melchers, 2001; Stroud, et al., 2002). It is an improvement over the Cornell’s safety index and it avoids its lack of invariance problem. Also, it can be used for explicit and implicit performance functions. The H-L method first transforms the random variables into a standardized form (i.e. zero mean value and unit standard deviation) as in the following:

\[ U_i = \frac{X_i - \mu_X}{\sigma_{X_i}} \]  \hspace{1cm} (2.13)

Also, the performance function is transformed accordingly by using the random variables transformation in Eq. 2.11. However, this transformation is only applied directly to normally distributed random variables, in the case of other distributions the random variables are transformed to equivalent normal variables by using some appropriate transformations (see Haldar and Mahadevan, 1995; Melchers, 2001). Then the H-L safety index \( \beta_{HL} \) is defined as the minimum distance from the origin of the axis in the reduced coordinate system to the limit state surface, which becomes an optimization problem of the form:
Hence, we are essentially searching for the most probable failure point (MPFP) in the standardized normal variables space (see Figure 2.3 for the case of two random variables).

\[ \beta_{HL} = \min \left( \sum_{i=1}^{n} u_i^2 \right)^{\frac{1}{2}} = \min \left( \mu^T \bar{u} \right)^{\frac{1}{2}} \]

\[ \text{s.t. } g(\bar{u}) = 0 \]  

(2.14)

The H-L reliability index gives an exact estimation for the design reliability for linear performance functions, and an acceptable approximation for most of the nonlinear performance functions as long as the radius of curvature of the performance function is large compared to \( \beta_{HL} \). Consequently, for those cases quadratic approximations for the failure surface at the most probable failure point may be appropriate, which is the basis of the second-order reliability methods (SORM), (see Madsen et
Thus, it may be worthwhile to check for the suitability of the H-L method by comparing its results with some of the more direct methods such as the Monte-Carlo simulation (Olsson et al., 2003; Stroud et al., 2002). However, it should be noted that for designs of high reliability, it is often sufficient to calculate the reliability within a factor of 2 to 5 on the probability of failure. Also, more importantly, the probability density function for each random variable PDF decays very quickly with the distance $r$ from the origin (with a rate of $\exp(-\frac{r^2}{2})$). Hence, the major contribution comes from the most probable failure point and the points closest to the origin (Madsen et al., 1986), and thus, higher accuracy in approximating the true performance function may not be required.

2.1.2.2 Other Analytical Reliability Calculation Methods

As was mentioned in the previous sub-section, the second-order reliability methods SORM may be used to calculate the reliability of the design in the cases where the first order methods are not suitable. The basic idea for SORM is to approximate the failure surface around the MPFP by calculating the local curvatures, which requires calculating the second derivative of the failure surface at MPFP for each random variable (see Madsen et al., 1986; Haldar and Mahadevan, 1995; Melchers, 2001). Also, another method that employs a higher order approximation of the performance function is the advanced mean value method (AMV). The advanced mean value method calculates the cumulative distribution function of the design probability to provide more information about the reliability of the design (Wu et al., 1990). It uses perturbed values around the mean values of the random variables, and then expands the performance function using linear Taylor series. Next, MPFP is determined and the performance function value is determined at MPFP. Finally, the values obtained from the previous iteration are combined to obtain a second-order
approximation of the failure probability or the cumulative distribution function of the random structural response. Although, the AMV can calculate the reliability of the design (Chandu and Grandhi, 1995), it needs extra calculations to provide more information than the FORM (that is concerned only with the probability of failure). Hence, the AMV is used mainly to calculate the probability of occurrence of certain values of structural responses (Riha et al., 1992; Wirsching, 1995; Rajagopalan and Grandhi, 1996; Pepin et al., 2002).

2.2 Reliability of Structural Systems and Structural Components with Multiple Failure Modes

For a structure with multiple components or for a component with multiple failure modes, there are some simple combinations of failure events such as in series, parallel, \( k \)-out-of-\( n \), or their combinations. For other combinations, such as progressive failure situations (cracks, fatigue …etc.), failure event logical trees may be constructed (see Karamchandani, et al.a, 1992; Karamchandani, et al.b, 1992; and Xiao and Mahadevan, 1994). In any case, the total probability of structural failure may be expressed as

\[
P(F) = P(F_1) + P(F_2) + P(F_3) + \cdots \\
- P(F_1 \cap F_2) - P(F_2 \cap F_3) - P(F_1 \cap F_3) - \cdots \\
- P(F_1 \cap F_2 \cap F_3) - \cdots
\]  

(2.15)

where, \( P(F_i) \) is probability of the failure event \( F_i \).

However, the statistical correlation (i.e. the intersection) between failure events is generally very difficult to quantify, and hence, the effect of multiple failure events is usually quantified by bounds on the reliability. For the case of series combination of failure events for a structure or for a
component, the first-order series bound is among the widely used. This failure bound assumes statistically independent failure modes, and it is defined as:

\[
\sum_{i=1}^{m} P(F_i) \leq P(F) \leq 1 - \prod_{i=1}^{m} [1 - P(F_i)]
\] (2.16)

On the other hand, higher order failure bounds take into account the correlations of failure modes, and hence may result in a reduced value of the probability of failure. Nevertheless, improved accuracy for failure bounds can be obtained by using Monte-Carlo simulation even by using FORM. However, it may be expensive to use MCS and the use of FORM is limited for the case linear performance functions, since nonlinear functions may cause failure of the optimization process (see Madsen et al., 1986; Enevoldsen, and Sorensen, 1994; Moses, 1995; Melchers, 2001). It is appropriate to note that there are other methods that combine multiple failure modes into a global failure event, and assign weighted influence factors to each failure mode (see Hong-Zong and Der Kiureghian, 1989). Also, for some structures the actual failure modes may not be known a priori, despite the fact that the possible failure modes may be known. For this situation, failure mode identification must be carried out before combining the resulting failure modes (see Zimmermann, et. al., 1991; Melchers, 2001).

### 2.3 Approximations of Performance Functions

In structural analysis, sometimes the performance function \( G(X) \) can be obtained as a closed-form equation, which will allow direct application of standard optimization techniques (see next subsection for a short review of some of the commonly used optimization methods for reliability calculation and reliability-based optimization) needed to calculate the design reliability using FORM (see Nikolaidis and Burdisso, 1988; Yang, 1989; Yang et al., 1990; Zimmermann, 1991;
Karamchandani and Cornell, 1992; Karamchandani et al., 1992; Li and Yang, 1994; Nikolaidis and Stroud; 1996; Nakib, 1997) Also, in some other situations the performance function of the design may not be explicit, but may have just one random variable and may be approximated linearly or quadratically during optimization (see Hisada et al, 1983; Liu and Der Kiureghian®, 1991; Enevoldsen and Soresn, 1993; Enevoldsen and Soresn, 1994; Natarajan and Santhakumar, 1995; Pu et al., 1997).

Yet, it was realized that in many other structural designs the performance function can not be represented by closed-form equations nor has more than one random variable. This situation occurs mostly when failure is simulated point by point in finite element analysis. In those applications, a high non-linearity and possible discontinuities may occur especially in the case of sudden failure (e.g. buckling or crack propagation) and in the case of multiple failure modes. However, it may become computationally expensive to make a point-by-point discovery of the entire failure domain. Also, to ensure convergence of the optimization process that the FORM methods use to calculate reliability, a differentiable (or if possible) a closed-form of the performance function should be obtained. Therefore, it would be beneficial to approximate the performance function rather than having it in point-by-point format. For this purpose, researchers have used different methods to approximate the structural response. For instance, some researchers have developed analytical gradients of the performance function with respect to the random variables by reformulating the finite element model. This was done to facilitate the optimization process needed to perform FORM and SORM, since there were some difficulties in numerical calculation of these gradients (see Lin and Der Kiureghian, 1989; Liu and Der Kiureghian®, 1991). Yet, in general, it may not be easy to reformulate all other finite element models especially those that are a part of a general
purpose code. However, sensitivity derivatives can be obtained from the stiffness matrix of the finite element model, and thus, may have wide range of application were developed. For instance, Santos et al. (1995) suggested using a continuum sensitivity method to calculate the needed gradients for the FORM optimization, and their work was later adapted by Kleiber et al (1999). Despite that, it would be preferable to find a way to approximate the performance function for the case of multiple failure modes, which normally results in derivative discontinuity in the performance function approximation (see Wang et al; 1995; Royset et. al., 2001; Ba-abbad et al., 2002).

Therefore, other methods have been employed. These methods range from linear and quadratic response surface (RS) methods (see Myers, 1971), that was employed by many researchers, to higher order approximations that take into account possible derivative discontinuity. For instance, Shüeller et al., (1991) used the response surface technique to approximate the performance function and then used MCS to calculate $P_f$ from the integral of Eq. 2.1. Also, Tandjiria et al., (2000), Krishnamurthy and Romero (2002), and Gayton et al., (2003) have used the response surface techniques to facilitate calculating the reliability of a design by approximating the performance function and then using FORM or SORM to calculate the reliability. In addition, Stroud et al., (2002) devised RS to calculate the structural response and direct MCS to calculate the probability of failure.

Multi-variable higher order approximations were proposed by Wang and Grandhi (1995), who have used multivariate splines to approximate the performance function, and then used FORM to calculate $P_f$. A subsequent work, Grandhi and Wang (1998) developed a two point adaptive nonlinear approximation (TANA2) to approximate the performance function and also used FORM to
calculate the design reliability. Moreover, Penmetsa and Grandhi (2003) developed a method that is based on the TANA2 developed by Grandhi and Wang (1998) to approximate the performance function and then adapted the Fast Fourier Transform (FFT) to estimate the probability of failure. However, it should be noted that the use of RS and similar methods gets to be computationally very expensive as the number of random variables increases.

Finally other interpolation methods were suggested, for instance, Papadrakakis and Lagaros, (2002), have used neural network to approximate the performance function and then used MCS to calculate the probability of failure.

2.4 Reliability Based Structural Optimization

Before presenting the methods used to perform the reliability based optimization it would be appropriate to present a quick review of some of the optimization methods that are widely used in reliability calculations (for FORM and SORM) and in performing the RBDO. Also, it would be appropriate to present the standard mathematical formulations of the two problems mentioned in Sec. 1.3.1.

2.4.1 Optimization Methods

The general optimization problem is defined as to find the minimum (or the maximum) of the objective or the cost function \( F(\vec{X}) \) (\( \vec{X} \) is the vector of design variables), and usually, this is performed within a specified set of constraints. These constraints are divided into: the inequality constraints \( g(\vec{X}) \) that define the feasible and the unfeasible domains, the equality constraints \( h(\vec{X}) \) that give some relations that the design parameters must satisfy, and the side constraints that constrain the range of variation of the design variables. The mathematical representation of the general optimization problem is:
\[
\begin{align*}
\text{Min} & \quad F(\tilde{X}) \\
\text{such that:} & \quad g(\tilde{X}) \leq 0 \\
& \quad h(\tilde{X}) = 0 \\
& \quad l_i \leq X_i \leq u_i
\end{align*}
\] (2.17)

where \( l \) is the lower limit and \( u \) is the upper limit of the design variable \( X_i \). The classical closed form solution of this problem is obtained by minimizing the Lagrangian of the optimization problem

\[
\text{Min: } \mathcal{L}(\tilde{X}, \lambda_1, \lambda_2) = F(\tilde{X}) + \lambda_1 (g(\tilde{X}))^2 + \lambda_2 (h(\tilde{X}))^2
\]

\[
\frac{\partial}{\partial X_i} \mathcal{L}(\tilde{X}, \lambda_1, \lambda_2) = 0
\]

\[
\frac{\partial}{\partial \lambda_1} \mathcal{L}(\tilde{X}, \lambda_1, \lambda_2) = 0
\] (2.18)

\[
\frac{\partial}{\partial \lambda_2} \mathcal{L}(\tilde{X}, \lambda_1, \lambda_2) = 0
\]

\[
\lambda_i \geq 0
\]

where \( \lambda_1 \) and \( \lambda_2 \) are the Lagrange multipliers. The analytical solution must satisfy the Kuhn-Tucker optimality conditions (see Vanderplaats, 1999). However, the Lagrange multipliers approach is only used if the objective function and the constraints can be represented by closed form equations, which is rarely the case in most practical situations. Thus, most optimization problems are solved numerically using iterative algorithms. In particular, most of the optimization methods start from an initial \( \tilde{X}_{i-1} \) point and then perform calculations to determine the direction \( \tilde{S}_i \) and the distance \( \alpha_i \) to the next point, i.e.
\[ \bar{X}_i = \bar{X}_{i-1} + \alpha_i \bar{S}_i \]  

(2.19)

The direction of the search \( \bar{S}_i \) is along the direction that has the most impact in reducing (or increasing) the objective function, while the magnitude \( \alpha_i \) is chosen such that no constraints would be violated. The optimizer searches the design space until it reaches an optimum solution, at which no further reduction in the value of the objective function is possible without violating the constraints. However, except in special cases, there is generally no guarantee that the optimum solution obtained is unique (i.e. the global minimum), that is why the optimum solution is perturbed and the optimization is started from different starting points to check for possible other optima. The optimization methods that are applied to non linear problems can be classified according to their use of the objective functions and its derivatives (in Taylor series expansion) into three groups.

i. The zero-order methods, in which no gradients of the objective function is considered (e.g. random search, genetic algorithms ... etc.) See Kloda et al, 2003 for a review of these methods

ii. The first-order methods, in which the first gradient of the objective function is considered, on which most of popular optimization methods are based (e.g. steepest decent method, sequential linear and quadratic programming methods, the feasible directions method, the conjugate gradient method ...etc.)

iii. The second-order methods, in which the second order Taylor series terms, are considered (e.g. the Newton method).

Also, the optimization methods can be classified based on the way the constraints are considered into direct methods (that use the constraints explicitly), and indirect methods (that augment the
constraints in the form of added penalty functions with the objective function). For other methods and more details see Haftka and Gurdal, 1992, Polak (1997), Vanderplaats (1999).

2.4.2 Standard Mathematical Statements of the Two RBDO Problems and their Classical Solutions

1. The first RBDO problem is to find the design with the least cost (or least weight), with constraints on the acceptable level of safety and some other deterministic constraints. This RBDO problem consists of two optimization problems (when using the analytical reliability approximation methods):

   i. Optimization problem under reliability constraints

   \[
   \begin{align*}
   \text{Min} & : F(\bar{x}) \\
   \text{s.t.} & \quad g_k(\bar{x}) \leq 0 \\
   & \quad \beta(\bar{x}, \bar{u}) \geq \beta_t
   \end{align*}
   \]  
   \(2.20\)

   where \( F(\bar{x}) \) is the objective function, \( \bar{x} \) is the vector of deterministic variables, \( g_k(\bar{x}) \leq 0 \) are the deterministic constraints that do not affect the safety of the design (e.g. packaging clearance), \( \beta(\bar{x}, \bar{u}) \) is the reliability index of the design, \( \bar{u} \) is the vector of normalized random variables (see Eq. 2.11), and \( \beta_t \) is the target reliability index (the minimum acceptable reliability).

   ii. Calculation of the reliability index \( \beta(\bar{x}, \bar{u}) \) of the design (using FORM)

   \[
   \begin{align*}
   \text{Min} & : d(\bar{u}) = (\bar{u}^T \cdot \bar{u})^{\frac{1}{2}} \\
   \text{s.t.} & \quad H(\bar{x}, \bar{u}) \leq 0
   \end{align*}
   \]  
   \(2.21\)
where \( d(\bar{u}) \) is the distance in the normalized random space, and \( H(\bar{x}, \bar{u}) \) is the transformed performance function (see Eq. 2.13).

Hence, the classical solution for these problems consists of minimizing the two Lagrangians

\[
\text{Min: } \mathcal{L}_1(\bar{x}, \bar{u}, \bar{\lambda}) = f(\bar{x}) + \lambda_\beta ([\beta_1 - \beta(\bar{x}, \bar{u})])^2 + \sum_k \lambda_k (g_k(\bar{x}))^2 \tag{2.22}
\]

\[
\text{Min: } \mathcal{L}_2(\bar{x}, \bar{u}, \lambda_H) = d(\bar{u}) + \lambda_H H(\bar{x}, \bar{u})
\]

where \( \lambda_i \) are the Lagrange multipliers. The analytical optimum solutions of these optimization problems are given by:

\[
\frac{\partial}{\partial x_i} \mathcal{L}_1(\bar{x}, \bar{u}, \bar{\lambda}) = \frac{\partial f}{\partial x_i} - \lambda_\beta \frac{\partial \beta}{\partial x_i} + \sum_k \lambda_k \frac{\partial g_k}{\partial x_i} = 0 \tag{2.23.a}
\]

\[
\frac{\partial}{\partial \lambda_\beta} \mathcal{L}_1(\bar{x}, \bar{u}, \bar{\lambda}) = \beta_1 - \beta(\bar{x}, \bar{u}) = 0
\]

\[
\frac{\partial}{\partial \lambda_k} \mathcal{L}_1(\bar{x}, \bar{u}, \bar{\lambda}) = g_k(\bar{x}) = 0
\]

\[
\frac{\partial}{\partial u_j} \mathcal{L}_2(\bar{x}, \bar{u}, \lambda_H) = \frac{\partial d}{\partial u_j} + \lambda_H \frac{\partial H}{\partial u_j} = 0 \tag{2.23.b}
\]

\[
\frac{\partial}{\partial \lambda_H} \mathcal{L}_2(\bar{x}, \bar{u}, \lambda_H) = H(\bar{x}, \bar{u}) = 0
\]
2. The second RBDO problem is to find the design with the highest reliability, with constraints on the acceptable cost (or weight). Again, this RBDO problem consists of two optimization problems (when using analytical methods to calculate the reliability):

i. Optimization problem under cost constraints

\[ \text{Max : } \beta(\bar{x}, \bar{u}) \]
\[ \text{s.t. } g_k(\bar{x}) \leq 0 \]
\[ f(\bar{x}) \leq f_t \]  \hspace{1cm} (2.24)

where \( f_t \) is the target cost function

ii. Calculation of the reliability index \( \beta(\bar{x}, \bar{u}) \) of the design (using FORM)

\[ \text{Min : } d(\bar{u}) = (\bar{u}^T \cdot \bar{u})^{\frac{1}{2}} \]
\[ \text{s.t. } H(\bar{x}, \bar{u}) \leq 0 \]  \hspace{1cm} (2.24)

where \( d(\bar{u}) \) is the distance in the normalized random space, and \( H(\bar{x}, \bar{u}) \) is the transformed (see Eq. 2.13) performance function.

Hence, the classical solution for these problems consists of minimizing the two Lagrangians

\[ \text{Min : } \]  
\[ \mathcal{L}_1(\bar{x}, \bar{u}, \lambda) = \beta(\bar{x}, \bar{u}) + \lambda_f \left( [f_t - f(\bar{x})] \right)^2 + \sum_k \lambda_k \left( g_k(\bar{x}) \right)^2 \]  \hspace{1cm} (2.25)

\[ \text{Min : } \mathcal{L}_2(\bar{x}, \bar{u}, \lambda_H) = d(\bar{u}) + \lambda_H \left( H(\bar{x}, \bar{u}) \right)^2 \]

where \( \lambda \) are the Lagrange multipliers. The analytical optimum solutions of these optimization problems are given by:
2.4.3 Reliability-Based Design Optimization Methods

Structural RBDO methods can be classified, according to the way reliability and optimization calculations are performed, in three categories:

a. Nested double optimization loops, in which an inner optimization loop is used to calculate the reliability of the design using FORM (or SORM) and an outer optimization loop is used to obtain an optimum reliability-based design (the one that has the highest reliability for a specified weight (or a
specified cost), or the one that has the least weight (or the least cost), for a specified value of reliability).

b. Single outer optimization loop, in which the reliability is calculated by MCS and this outer optimization loop, is used to obtain a reliability-based optimum design.

c. Combination and reformulation of the RBDO problem, in which the objective function is a multi-objective function of reliability and cost function (see Madsen and Hansen, 1991). Also, some researchers have developed a combined optimization space to perform RBDO formulation, in which they combine the design and reliability variables in a hybrid optimization space (Kharmanda et al, 2002), or in a design potential space (a concept similar to the hybrid optimization space) as proposed by Tu et al, 1999, 2001, and Choi et al, 2001.

d. Sequential Optimization with Probabilistic Safety Factors (SORFS) RBDO, in which the optimization and the reliability calculation are performed in sequence after approximating the probabilistic constraints with modified safety factors.

Since the goal is to find the most computationally efficient and robust RBDO method, it would be appropriate to compare all these different methods to find the most computationally efficient and the most robust method. Now considering the nested optimization problem format, we find that it is the most widely used. For example, Nikolaidis and Burdisso (1988) used nested gradient-based optimization loops to perform RBDO for a simple structure that was modeled by
closed form equations. Yang (1989) and Yang et al. (1990) used gradient-based nested RBDO to optimize aircraft wings that were modeled by closed-form equations. Also, Enevoldsen and Soresnsen (1993) used gradient-based nested RBDO to optimize a series system of parallel systems that also was modeled by closed form equations. Then in a later work, Enevoldsen and Soresnsen (1994) formulated the nested RBDO method on the basis of the classical decision theory and they presented some solution strategies for component optimization and for structural system optimization. In addition, Li and Yang (1994) have linearized the reliability index and used nested linear programming optimization loops to perform RBDO for a truss. Moreover, Natarajan and Santhakumar (1995) used nested RBDO approach to optimize transmission line towers, which employed gradient-based optimization algorithm to find the reliability, and random search optimization algorithm to find an optimum design for the transmission tower. Nikolaidis and Stroud (1996) used nested optimization loops with gradient based penalty function optimization techniques to perform RBDO for 10-bar truss. Rajagopalan and Grandhi (1996) have developed reliability based structural optimization (RELOPT) code that performs nested gradient-based optimization to obtain the optimum design. Then, RELOPT was used by Barakat et al (1999) to perform RBDO of laterally loaded piles. Nakib (1997) has used the nested gradient-based optimization to perform RBDO of truss bridges. Also, Pu et al (1997) used nested gradient-based optimization to perform RBDO for the frame of a small twin hull ship.

Some researchers suggested approximating the performance functions first before carrying on the optimization, especially when the analysis is performed using implicit models such as the finite element. For instance, Wang et al (1995) have developed multivariate splines to approximate the design constraints and then they used nested sequential quadratic programming to carry out the
optimization for different structural problems. Then, Grandhi and Wang (1998) used the multivariate splines (which they called: two-point adaptive nonlinear approximation (TANA2)) to approximate the structural response and then used double optimization loops to perform RBDO for a frame and a turbine blade. Moreover, some researchers suggested approximating the derivatives of the performance function always near the reliability constraint, instead of other points in the design space, to accelerate the convergence. For example, Tu et al (2001) proposed the design potential method and reported to get around 64% reductions in the number of evaluations (compared to RBDO method that takes the derivatives of the performance function at other points).

In addition, some researchers have used other methods of optimization to avoid calculating the gradients of the reliability function. For example, Thampan and Krishnamoorthy (2001) employed another form of the nested optimization approach which uses genetic optimization algorithm to optimize the configuration of truss structures and gradient-based optimization to calculate the reliability of the design. Also, Ba-abbad et al (2002) employed nested optimization which uses graphical optimization to optimize the design of an elastic-plastic T-beam and used gradient-based methods to calculate the design reliability (see chapter 4). Finally, some researchers have used variable complexity approach to carry on RBDO. For example, Burton and Hajela (2002) devised a nested optimization approach that uses variable complexity RBDO (optimization starts by deterministic optimization, followed by FORM, then, the active constraints are calculated by SORM), and they devised a neural network to approximate error between FORM and SORM.

Hence, the nested optimization approach can be used to perform RBDO for a variety of RBDO problems provided that the gradients needed for optimization are calculated using an efficient and robust method (see Enevoldsen and Sorensen, 1994, Santos et al., 1995, Kleiber et al., 1999).
Otherwise, the nested optimization loop methods may not converge or may not be successful in obtaining an optimum design.

On the other hand, the single optimization loop methods that use MCS may not experience difficulties in calculating the reliability of the design, but may experience convergence problems that may result from the high non-linearity and/or derivative discontinuity of the reliability function. However, to overcome the reliability function discontinuity researchers have used non-gradient based methods, such as the genetic algorithms, or reformulated the problem to be semi-infinite optimization. For example, Papadrakakis and Lagaros (2002) used genetic optimization algorithm to perform the optimization of a 3-D frame and its reliability was calculated using MCS with importance sampling. Also, Royset et al, 2001 proposed reformulating the RBDO to an outer deterministic semi-infinite optimization loop and a separate inner loop to calculate the reliability of the design by any reliability method (for semi-infinite optimization algorithms, see Polak, 1997). The efficiency and robustness of the semi-infinite algorithm used by Royset et al (2001) was not compared to other widely used optimization methods, and thus, Royset et al concluded that the success of their proposed method depends on how the reliability calculations modify the optimization sub-problem. Also, more importantly, if the MCS was used to calculate the reliability instead of the analytical reliability methods (such as the FORM/SORM, and AMV methods), it is generally much more computationally expensive than these methods.

Alternatively, some researchers have reformulated the RBDO techniques by combining the two optimization loops to reduce the overall number of the performance function evaluations. Nevertheless, these attempts sometimes were not successful in achieving their goals, or may work
only for special cases. For example, Madesen and Hansen (1991) have found that a multi-objective function RBDO formulation may need 50% more iterations than a nested RBDO formulation for the same problem. Also, the hybrid optimization space approach proposed by Kharamanda et al (2002) which was reported to have a five fold reductions in the number of evaluations compared to a nested RBDO formulation, was in fact creating a large number of local minima. This is because the hybrid space combines the optimization search for minimum cost that satisfies the reliability constraint with the reliability calculation optimization search to locate the most probable point (MPP) in FORM by multiplying the two objective functions together. However, this combination may terminate the reliability calculation search prematurely and give an overestimation of the reliability.

Finally, despite the fact that there is rarely an explicit relation between the safety factors and the reliability of a design (Elishakoff and Starnes (1999)), researchers have found that they may be able to derive approximate probabilistic safety factors from the reliability calculations and use decoupled sequential loops for optimization and reliability calculations. This is the main idea behind the Sequential Optimization with Probabilistic Safety-Factor (SORFS) RBDO methods. For example, Chen et al. (1997); Wu et al. (2001); Du and Chen (2002) have used the FORM or SORM to obtain the MPFP coordinates, and then used these coordinates to update the probabilistic constraints in the design optimization loop. Also, Qu and Haftka (2003) have used probabilistic safety factors PSF that are calculated from MCS in the design optimization loop. In this way, the optimum design search can be performed deterministically, instead of using the actual reliability constraints and searching in the probability space. Thus, the RBDO can be performed in sequential optimization loops instead of nested optimization loops. A substantial reduction in the number of performance function evaluation may be achieved.
In this scheme (please see Figure 2.3), the first optimization loop searches deterministically for an optimum design with the mean values of the random variables implemented in the reliability constraints, and then, the second loop calculates the reliability of the design and finds its MPP. In the next iteration, the deterministic optimization loop searches for a new optimum design by using the MPP(s) $\tilde{X}_k^*$ from the last reliability calculation loop to modify the new reliability constraints, and then, the reliability loop calculates the reliability of this new optimum design. The reliability of the design is improved by shifting the performance functions for the deterministic loop by an amount that equals the difference between the current safety index $\beta^k$ and the target safety index $\beta_T$ (Wu et. al. (2001); Du and Chen (2002)), or by scaling up the design variables based on the value of a calculated probabilistic safety factor (Qu and Haftka (2003)). The procedure is terminated when the target reliability is reached and no improvement in the design is possible. Figure 2.4 shows the flow chart for such procedure.
Figure 2.4 Flow Chart of the Sequential Safety-Factor Based RBDO Methods
However, these procedures vary in the way of handling multiple constraints, and accordingly in the way the design improves during optimization. For example, Wu et. al., (2001) consider only the active constraint in each optimization cycle, which may cause the design to oscillate between competing constraints. Alternatively, Du and Chen (2002), have shifted the constraints towards the target reliability value $\beta_T$ using an inverse MPP search. Yet, for an RBDO problem with multiple failure modes, the system reliability may not be optimum. Finally, Qu and Haftka (2003), are suggesting handling multiple constraints using a probabilistic safety factor that combines all the critical constraints, and they calculated this safety factor from Monte-Carlo simulation of a response surface. Also, similar to Du and Chen (2002), in their procedure the system reliability was not optimized. Accordingly, there is a need to develop an RBDO method that will optimize the system reliability.

### 2.4.4 A Modified Sequential Optimization with Reliability-Based Factors of Safety (SORFS) RBDO Approach

In this RBDO problem formulation, we are considering the first RBDO problem (safest design for the least cost)

\[
\text{max} : \text{cost} \quad \text{subject to} \quad P_s = P_1 + \ldots + P_m \leq P_{s,\text{max}}
\]

where $d_1, \ldots, d_p$ are the design variables (which may contain the mean values of the random variables), $F(\bar{d}, \bar{X})$ is the objective function (cost or weight), $P_1, \ldots, P_m$ are the failure probabilities of the $m$ failure modes, and $P_{s,\text{max}}$ is the maximum allowable system failure probability. The constraint on system reliability can be written in terms of the safety indexes of the modes as follows:
\[ \Phi(-\beta_1) + \ldots + \Phi(-\beta_m) \leq P_{s,max} \]  
(2.28)

where \( \beta_1^*, \ldots, \beta_m^* \) are the safety indices for the system failure modes.

We perform deterministic optimization to find a good initial design and then perform reliability analysis of the deterministic optimum. Let \( \bar{X}_1^*, \ldots, \bar{X}_m^* \) be the most probable failure points for the \( m \) failure modes and \( \beta_1^*, \ldots, \beta_m^* \) the corresponding safety indices. Let \( \bar{Z}_1^*, \ldots, \bar{Z}_m^* \) be the values of the reduced random variables at the most probable point for each failure mode where \( \bar{X}_i^* = T(\bar{Z}_i^*) \) and \( T \) is the transformation from the space of the reduced random variables to the space of the random variables. This is a vector with \( n \) functions, where \( n \) is the number of random variables.

Now we perform deterministic optimization considering the target safety indices of the \( m \) failure modes, \( \beta_{T1}, \ldots, \beta_{Tm} \) as design variables. The optimizer will seek the both the optimum values of the design variables and the optimum target values of the safety indices to minimize the weight, as in the following.

\[
\begin{align*}
\text{Find} & : \bar{d}, \bar{\beta}_r \\
\text{Min} & : F(\bar{d}, \bar{X}) \\
\text{s.t.} & : G_i(\bar{d}, T(\bar{Z}_i)) \geq 0 \\
& \quad \ldots \\
& : G_m(\bar{d}, T(\bar{Z}_m)) \geq 0
\end{align*}
\]
(2.29)
where \( \tilde{Z}_i \) are the projected most probable points \( \tilde{Z}_i = \frac{\beta_{T_i}}{\beta_i} \tilde{Z}_i^* \), \( i = 1, \ldots, m \)

so that:

\[
P_1 + \ldots + P_m = \Phi(-\beta_{T_1}) + \ldots + \Phi(-\beta_{T_m}) \leq P_{s,max}
\]

and \( \beta_{T_i} > 0 \).

In Eq. 2.28, we can use the values of the design variables and the safety indices of the deterministic optimum as initial guesses. Once we found the optimum, we perform a new reliability analysis using a performance measure approach (that is we fix the value of the safety index and find the minimum value of the performance function) and formulate and solve the optimization problem (Eq. 2.28) again. We repeat the process until convergence. Figure 2.5 illustrates the approach.

---

**Figure 2.5 Flow Chart of the Modified Sequential Safety-Factor Based RBDO Method**
The second RBDO problem follows the same steps, but with interchanging the objective function and constraints, so Eq. 2.27 becomes

\[
\begin{align*}
\text{Find: } & \tilde{d} \\
\text{Min: } & P_{\text{sys}}(\tilde{d}, \tilde{X}) = P_1 + \ldots + P_m \\ 
\text{s.t. } & \text{Weight}(\tilde{d}, \tilde{X}) \leq \text{Weight}_{\text{max}} 
\end{align*}
\]  

(2.30)

We will present two examples for applying this modified RBDO technique, the first (Chapter 5) will involve application to an exploratory elastic-plastic T-beam example, and then we will investigate the application to a more elaborate elastic-plastic T-beam model.
Chapter 3

Nonlinear Finite Element Model of Six Degrees of Freedom Elastic-Plastic Beam

3.1 Introduction

In this part of our work, we are exploring the different issues that may face the design engineer when performing RBDO for practical structural design situations. For this purpose, we selected a representative structural member that exhibits nonlinear elastic-plastic behavior, which is known from the literature to cause the most problems for RBDO. In particular, we chose an example of T-stiffener beam that is acted upon by combined axial, bending and torsional shear loads. However, to reduce the computational costs at this investigative part of the study, we chose to develop a simple six degrees of freedom beam element that has nonlinear behavior.

The concept of finite element modeling in structures stems from the idea of discretizing the structure into a number of elements that have simpler shapes (or simpler mechanical behavior) onto which the structural response is approximated. These simple elements include beam, plate, shell, and solid elements. Hence, complex structures and complex mechanical behavior can be greatly simplified. For this purpose, the applied system of loads is replaced by a statically equivalent forces and moments applied to the nodes. In turn, the nodes are the points in the element where the interpolation functions are specified, and where the structural response is obtained. However, the real structural response is continuous between the joined portions. Therefore, continuity between adjacent elements is ensured at the connecting nodes by enforcing continuity of displacements. Generally, there are two sources for nonlinearity in a finite element problem:
a. Material nonlinearity, in which the material behavior does not follow a linear elastic path (e.g. plastic material behavior)

b. Geometric nonlinearity, in which the geometry of the problem affects the magnitude and/or the direction of the applied loads (e.g. large deflections of the structure may cause the loads to change direction or magnitude). Also, the strain-displacement relations may develop non-linearity in the model by having coupled and/or higher order terms (see Ch. 5.)

For both these cases, the structural behavior is assumed to be linear within small iterative steps, and thus, iterative solution procedures are employed. For more information and details about finite element methods and procedures see Yang (1986); Crisfield (1991); Bathe (1996); Belytschko et al (2000); and Doyle (2001).

3.2 Six Degrees of Freedom Elastic-Plastic Beam Element

This nonlinear beam element calculates the in-plane and axial deformations of an elastic-plastic beam of the T-section subject to combined axial and lateral loads (see Figure 3.1). For this purpose, three degrees of freedom were assigned to each of the two nodes of the element. The first degree of freedom is the axial displacement \( u \) that corresponds to the axial compressive load \( F_x \). Also, the second degree of freedom is the in-plane deflection of the beam \( w \) that corresponds to the shear load \( F_z \). Finally, the third degree of freedom is the slope of the beam in-plane deflection \( \theta = -\frac{\partial w}{\partial x} \) that corresponds to the moment \( M_y \). Figure 3.2 illustrates these six degrees of freedom.
Figure 3.1 The Example Beam under the Applied Loads

Figure 3.2 Six Degrees of Freedom Beam Element
Since, in most of the cases the variation in axial compression along the beam element is much smaller than the variation in the in-plane deflections; the interpolation functions that are used to approximate deformation are linear for the axial degree of freedom and cubic for the others.

The displacements interpolation functions are:

\[
\begin{align*}
    u(x) &= \begin{bmatrix} N_1 & N_4 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \\
    w(x) &= \begin{bmatrix} N_2 & N_3 & N_5 & N_6 \end{bmatrix} \begin{bmatrix} w_1 \\ \theta_1 \\ w_2 \\ \theta_2 \end{bmatrix}
\end{align*}
\] (3.1)

(note that \( \theta = -\frac{\partial w}{\partial x} \))

where,

\[
N_1 = 1 - \zeta; \quad N_4 = \zeta
\]

\[
N_2 = 1 + 2\zeta^2 - 3\zeta^2
\]

\[
N_3 = -\zeta L(1 - \zeta)^2
\]

\[
N_5 = 3\zeta^2 - 2\zeta^3
\]

\[
N_6 = -\zeta^2 L(\zeta - 1)
\]

and

\[
\zeta = \frac{x}{L}
\] (3.1.b)

where \( L \) is the element length.

The linear strain due to bending at any point \((x, z)\) in the element is given by
\[ \varepsilon_x = \frac{\partial u}{\partial x} - z \frac{\partial^2 w}{\partial x^2} \quad (3.2) \]

which can be rewritten after approximation by the interpolation functions as

\[ \varepsilon_x = \begin{bmatrix} \frac{\partial N_1}{\partial x} - z \frac{\partial^2 N_2}{\partial x^2} & \frac{\partial N_2}{\partial x} - z \frac{\partial^2 N_3}{\partial x^2} & \frac{\partial N_3}{\partial x} - z \frac{\partial^2 N_4}{\partial x^2} & \frac{\partial N_4}{\partial x} - z \frac{\partial^2 N_5}{\partial x^2} & \frac{\partial N_5}{\partial x} - z \frac{\partial^2 N_6}{\partial x^2} \end{bmatrix} \begin{bmatrix} u_1 \\ w_1 \\ \theta_1 \\ u_2 \\ w_2 \\ \theta_2 \end{bmatrix} \quad (3.3) \]

Also, it is customary to write the strains in the following format

\[ \varepsilon_x = [B] \{q\} \quad (3.4) \]

where, \([B]\) is vector that contains the derivatives of the interpolation functions and \(\{q\}\) is the vector of nodal displacements (see Yang and Saigal, 1984).

Now, the elastic strain energy of the beam element is given by:

\[ U = \frac{1}{2} \iiint \sigma \cdot \varepsilon_x \, dV = \frac{1}{2} \iiint \varepsilon_x \cdot E(x,z) \cdot \varepsilon_x \, dV \quad (3.5) \]

which we can rewrite using the strain approximation

\[ U = \frac{1}{2} \iiint [q] [B]^T E(x,y)[B] \{q\} \, dV \quad (3.6) \]

Then, by using Castigliano’s theorem we can get the standard element stiffness matrix (see Yang, 1986) as the following

\[ [k_e] = \iiint [B]^T E(x,y)[B] \, dV \quad (3.7) \]
Now, to consider the geometric nonlinearity the effect of the axial force $P$ that generates an added moment to the beam that increases as the in-plane deflection increases (see Figure 3.3) is considered.

The part of the strain energy that represents the shorting of the beam due to the applied axial force is given by

$$V = -\frac{1}{2} \int P \left( \frac{dw}{dx} \right)^2 dx$$

(3.8)

Considering an axial force that is constant along the beam element and differentiating with respect to the six degrees of freedom (after substituting with the deformation interpolation functions), we get the following matrix of the geometric effects (see Shames and Dym; 1985 and Yang, 1986):

$$\begin{bmatrix}
0 & 12/L & & & & \\
0 & 1 & 4L/3 & & & \\
0 & 0 & 0 & 0 & & \\
0 & -12/L & -1 & 12/L & & \\
0 & 1 & -L/3 & 0 & -1 & 4L/3 \\
0 & 1 & 0 & 0 & & \\
\end{bmatrix} \begin{bmatrix}
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
\end{bmatrix} = \frac{P}{10}$$

(3.9)
It is clear that for a given value of $P$ this matrix depends only on the length of the element, is a geometric parameter, and hence, this matrix is called the geometric stiffness matrix. Also, since there is nonlinearity the two stiffness matrices represent the relation of the incremental loads to incremental displacements, and therefore, constitute the tangent stiffness matrix

$$[k_t] = [k_e] + [k_s] \quad (3.10)$$

Accordingly, the stiffness equation can be written as

$$\{P\} = [k_t] \{q\} \quad (3.11)$$

where $\{P\}$ is the vector of nodal forces.

In the case that the beam element displacements are large, a transformation matrix $[T]$ can be used to transform the loads, the stiffness matrices, and the displacements from the local to the global coordinates (through angle of rotation $\phi$) at the element level (see Yang, 1986).

$$\{P^g\} = [T]^T \{P\}$$

$$[k_t^g] = [T]^T [k_t] [T] \quad (3.12)$$

$$\{q^g\} = [T]^T \{q\}$$

where

$$[T] = \begin{bmatrix}
\cos \phi & \sin \phi & 0 & 0 & 0 & 0 \\
-\sin \phi & \cos \phi & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & \cos \phi & \sin \phi & 0 \\
0 & 0 & 0 & -\sin \phi & \cos \phi & 0 \\
0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix} \quad (3.13)$$
Finally, the element loads vector, the stiffness matrices, and displacements vector are assembled into their global counterparts.

For computational efficiency, the solution procedure is based on the Newton-Raphson iterative method applied within a predictor/corrector approach (see Figure 3.4). This way, a quadratic convergence rate is achieved in determining the equilibrium path (see Crisfield, 1991).

The full Newton-Raphson method requires that the internal forces must be determined after each iteration and balanced with the external applied loads. Also, since the beam may yield and deform plastically at some portions of the cross-section, the modulus of elasticity contained in the linear stiffness matrix would change. Therefore, numerical integration must be performed to obtain the above mentioned quantities. However, since the beam element is assumed to deform according to the Euler-Bernoulli beam model (plane sections remain plane after deformation) we can determine the stress distribution over the cross-section analytically. The analytic integration is more accurate and more efficient than a point by point quadrature. It uses the value of the strain along the depth of
the beam (as obtained from Eq. 3.3) to determine the location of the yield zones on the cross-section, and from this information the stress distribution over the cross-section is obtained (see Figure 3.5)

![Figure 3.5 Plastic Zones Developed in a T-Beam Cross-section for Increasing Values of the Axial and Bending Loads](image)

**Step Control**

Since some equilibrium path non-linearity due to the onset of buckling or plasticity is going to be encountered, it is recommended that some form of displacement control be used (see Thomas and Gallagher, 1975; Batoz and Dhatt, 1979; and Crisfield, 1991). Accordingly, displacement control of a specific displacement variable is used to progress. In particular, following the work of Batoz and Dhatt (1979), one would prescribe the value of the dominant displacement component \( \delta q \) to a certain value (say a fraction of the ratio of the beam’s depth) at the beginning of the
iterations (i.e. \(i = 0\)). Then, the tangent stiffness matrix, displacements, the residual loads, and the applied external loads are reorganized as the following:

\[
\begin{bmatrix}
K_{11} & K_{12} \\
K_{21} & K_{22}
\end{bmatrix}
\begin{bmatrix}
\delta q_1 \\
\delta q_2
\end{bmatrix}
= \begin{bmatrix}
\delta R_1 \\
\delta R_2
\end{bmatrix} + \delta \lambda \begin{bmatrix}
P_1 \\
P_2
\end{bmatrix}
\tag{3.14}
\]

where \(\delta q_2 = \delta q_2\), \([K_{11}]\) is the tangent stiffness matrix (of order \(n \times n\)) with the \(q\)th row and column are suppressed during processing (i.e. factorization or inversion), \([K_{12}]\) and \([K_{21}]\) are the \(q\)th column and row of the tangent stiffness matrix without \(K_{22} = K_{qq}\) (of dimension \(n - 1\)).

Also, \(\{\delta q\}, \{\delta R\}, \) and \(\{P\}\) are the displacement, the residual load, and the applied load vectors that are organized accordingly.

Taking the second row equation in Eq. 3.14,

\[
K_{21} \delta q_1 + K_{22} \delta q_2 = \delta R_2 + \delta \lambda \ P_2
\tag{3.15}
\]

and splitting the displacement vector into two components; one associated with the external loads vector \(\{P\}\) and the other with the unbalanced loads \(\{\delta R\}\) in the following form

\[
\{\delta q\} = \{\delta q^a\} + \delta \lambda \{\delta q^b\}
\tag{3.16}
\]

Equation 3.14 can be rewritten as (note that \(\delta q^b_q = 0\))

\[
K_{21} (\delta q^a_1 + \lambda \delta q^b_1) + K_{22} \delta q_2 = \delta R_2 + \delta \lambda \ P_2
\tag{3.17}
\]

Finally, solving for the applied load fraction \(\delta \lambda\)

\[
\delta \lambda = \frac{K_{21} \delta q^a_1 - \delta R_2 + K_{22} \delta q_2}{P_2 - K_{21} \delta q^a_1}
\tag{3.18}
\]

the desired iterative or incremental step size is obtained as:

\[
\lambda_{i+1} = \lambda_i + \delta \lambda_i
\tag{3.19}
\]
\[
\delta q_i = \delta \lambda_i^\alpha + \delta \lambda_i^\beta \delta q_i^\beta
\]
\[(3.20)\]

\[q_{i+1} = q_i + \delta q_i\]
\[(3.21)\]

It is important to note that the stiffness matrix \([K_i]\) and the residual forces vector \(\delta R\) are calculated for the initial step after updating the displacement with the prescribed iterative displacement component (i.e. \(q_o = q_o + \delta q_q\)). This method was implemented in a FORTRAN-90 computer program that is documented in Appendix A. Also, the method gave exact match to the results of a simply-supported elastic-plastic beam analyzed by Yang and Saigal (1984), as shown in Figure 3.6.

---

Figure 3.6 Mid-Span Deflection of a Simply Supported Beam of Rectangular Cross-section under Lateral Load (see Yang and Saigal (1984))
Chapter 4

RBDO Exploratory Example: Elastic-Plastic T-beam

4.1 Failure Modes

*Failure under axial and bending stresses*

For a linear-elastic material, the axial stress can be linearly combined with the flexural stresses until the extreme fibers yield. However, after yielding the failure stresses do not combine linearly and they can be determined from the equilibrium of forces and moments over the cross-section. Let us consider the state of stress at a beam of T cross-section under the combination of axial load and bending moment that causes the whole cross-section to deform plastically as shown in Figure 4.1

![Figure 4.1 The State of Stress for a T-Beam under the Combination of Axial Load and Bending Moment that Causes the Whole Cross-Section to Deform Plastically](image-url)
From the equilibrium of forces in the axial direction (noting that the axial compressive load $P$ is assumed to be acting at the centroid of the cross-section):

$$ P = \sigma_y (A_c - A_t) \quad (4.1) $$

which becomes after simplifying

$$ P = \sigma_y (2A_c - A) \quad (4.2) $$

where $P$ is the applied axial load, $\sigma_y$ is the yield stress of the material, $A_c$ is the area under compression, $A_t$ is the area under tension, and $A$ is the area of the cross-section. Normalizing the axial load by the maximum axial load on the cross-section that causes the fully plastic deformation $P_y = \sigma_y A$, the normalized axial load becomes:

$$ R_p = \frac{2A_c}{A} - 1 \quad (4.3) $$

From equation (4.3) solving for $A_c$ we get

$$ A_c = A\left(\frac{R_p + 1}{2}\right) \quad (4.4) $$

From the force equilibrium and Figure 4.1, it can be seen that the axial load at which the flange becomes entirely plastic under compression is an important transition load because the equilibrium equations will change when the net compressive force is distributed entirely on the flange (or portions of it), and when the net compressive force is distributed over the flange and part of the web. We will designate the first situation, case (a), and the second situation, case (b). Hence, from equilibrium of forces we can get the axial load that causes the flange to become fully plastic under compression. This load becomes after normalization by the maximum axial load and simplification becomes:
\[ R_{pc} = \frac{2 t_f b}{t_f b + h t_w} - 1 \]  \hfill (4.5)

Also, from the equilibrium of the moment we get:

\[ M = \sigma_y (A_c y_1 + A_t y_2) \]  \hfill (4.6)

where \( y_1 \) and \( y_2 \) are the distances between the centroid of the entire cross-section and the centroids of the compression and tension zones, respectively. From the geometry of case (a), we can determine the following relations:

\[ a_c = \frac{A_c}{b} \]  \hfill (4.7)

\[ C_c = \frac{a_c}{2} \]  \hfill (4.8)

\[ C_t = \frac{h t_w \frac{h}{2} + (t_f - a_c) b (h + \frac{t_f - a_c}{2})}{h t_w + (t_f - a_c) b} \]  \hfill (4.9)

Similarly, for case (b):

\[ a_c = \frac{A_c - t_f b}{t_w} + t_f \]  \hfill (4.10)

\[ C_c = \frac{t_f b \frac{t_f}{2} + (a_c - t_f) t_w \frac{a_c + t_f}{2}}{t_f b + (a_c - t_f) t_w} \]  \hfill (4.11)
where $C_c$ and $C_t$ are the centroids of the compression and tension zones respectively measured from the bottom edge of the web. For both cases, the expressions for distances $y_1$ and $y_2$ are given by the following relations:

$$y_1 = h + t_f - \bar{Y} - C_c$$

(4.13)

$$y_2 = \bar{Y} - C_t$$

(4.14)

where $\bar{Y}$ gives the location of the centroid of the entire cross-section measured from the bottom edge of the web.

We can normalize the moment by the maximum elastic moment, $M_e$, that the cross-section can experience before any plastic deformation in the extreme fiber sets in.

This moment is given as:

$$M_e = \frac{\sigma_y I}{\bar{Y}}$$

(4.15)

Then the normalized moment becomes:

$$R_m = \frac{A_c \bar{Y}}{I} y_1 + \frac{(A - A_c) \bar{Y}}{I} y_2$$

(4.16)
We can now substitute the value of $A_c$ from Eq. (4.4) into Eq. (4.16) and obtain two expressions that relate $R_p$ with $R_m$ and the cross-section dimensions for cases (a) and (b). These two expressions will be of the form

$$R_m = f_a(R_p)$$  \hspace{1cm} (4.17.a)  \\
$$R_m = f_b(R_p)$$  \hspace{1cm} (4.17.b)

The first expression is valid for axial loads in the range from zero to $R_{pc}$ (Eq. 4.5); the second expression is valid for loads higher than $R_{pc}$. The obtained expressions were applied to the example presented by Smith and Sidebottom (1965) and gave identical results.

The first expression is valid for axial loads in the range from zero to $R_{pc}$ (Eq. 4.5); the second expression is valid for loads higher than $R_{pc}$. The obtained expressions were applied to the example presented by Smith and Sidebottom (1965) and gave identical results. Thus, we have obtained the interaction curve between the axial force and bending moment and now we will consider the effects of the shear loads

*The effect of transverse shear stress:*

Transverse shear stresses due to a bending load have relatively small contribution to the failure of the cross-section compared to the bending or torsional shear stresses for long and slender beams (having height to length ratio $> 0.3$). However, we can include their contribution to failure in case we found that they reduced significantly the yield strength of the material. In that case, the new failure stress becomes
\[
\sigma_f = \sqrt{\sigma_y^2 - 3\tau^2} \quad (4.18)
\]

where \(\sigma_f\) is the failure stress of the material, \(\sigma_y\) is its yield strength, and \(\tau\) the average shear stress on the beam cross-section.

However, it is important to note the limitation on this reduction factor as it is only valid for shear stresses up to 30% of the axial stresses due to flexure loads. (see Hughes, 1983).

*The effects of torsional shear stresses:*

Finally, we include the effect of the torsional shear stress on the total plastic failure of the beam and construct the interaction surface. These torsional effects include the thickness shear stresses (St. Venant’s shear stress) and the warping axial stresses (contour stresses). However, the warping axial stresses have secondary effect in a thin walled section and can be ignored (see Megson, 1999).

Now, we need to determine the value of the maximum torque that will deform the entire cross-section plastically. We will apply the Nadai sand heap analogy to the cross-section and obtain the following expression for the maximum shear stress that the cross-section can withstand with no other stresses present (see Nadai, 1950):

\[
T_P = \tau_y\left(\frac{-f^3}{6} - \frac{t_w^3}{12} + \frac{t_f^2}{2}b + \frac{t_w^2}{2}h + \frac{t_w^2}{6}t_f\right) \quad (4.19)
\]

where
\[ \tau_y = \frac{\sigma_y}{\sqrt{3}} \text{ (Von Mises)} \quad (4.20) \]

We can normalize the plastic torque by the maximum elastic torque, that is the torque after which the extreme fiber deforms plastically.

\[ T_e = \frac{\tau_y J}{t_w} = \frac{\tau_y (h t_w^3 + b t_f^3)}{3 t_w} \quad (4.21) \]

Finally, we obtain the normalized maximum plastic torque for this cross-section

\[ R_{T_o} = \frac{T_p}{T_e} = \frac{t_w^2 [2 t_f^2 + t_w^3 - 6 t_f^2 b - 6 t_w^2 h - 2 t_w^2 t_f]}{-4 [h t_w^3 + b t_f^3]} \quad (4.22) \]

So far, we have developed an interaction curve between the axial force and bending moment and we have derived an expression for the maximum torsional shear stress for the cross-section. We still need to relate all these quantities together in one constraint. However, we do not need a rigorous mathematical formulation to relate these loads to the limit combined load; but instead, we can use a suitable empirical formulation. This empirical formulation must satisfy the convexity condition (Drucker's postulate) and must not assume that the material will bear stresses that are higher than the allowable Von Mises stresses at any point in the cross-section (see Chakrabarty, 1987; Gebbeken, 1998; and Drucker, 2001)

With this in mind, let us assume that the interaction of the torsional load with both the axial load and the bending moment can be represented by a ruled surface as shown in Figure 4.2.a.
This surface is constructed by extending straight lines from the maximum normalized torsional load to the normalized interaction curve between the bending and the axial load. This way, the convexity of the failure surface is assured (since we are using straight lines). Also, no concavity is allowed to occur at any portion of the surface (otherwise the plastic work will be negative), which means that we are at the lower bound of any possible interaction curve between the torsional shear and the other loads. Therefore, at no point in the cross-section the stress exceeds the yield strength of the material, and that fulfils the second condition that the yield surface must satisfy.

Now, we need to obtain the factor of safety against failure determined by this surface, which tells us how far or near we are from the constraint corresponding to failure of a beam section. Let us assume the projection of any point inside (or outside) the failure surface onto the $M-P$ plane to be denoted by $P_I$, and let us draw a vector, $V_I$, from the origin to the projected point (see Figure 4.2.b).
Then let us extend the vector $V_1$ until it intersects the $M$-$P$ interaction curve and name the new vector $V_2$. Now, let $R_{PM}$ be the ratio of the lengths:

$$R_{PM} = \frac{\|V_1\|}{\|V_2\|} \quad (4.23)$$

which will allow us to obtain the safety factor $\gamma$ after solving for it from the failure surface equation below:

$$\gamma \ R_{PM} + \gamma \ R_T = 1 \quad (4.24)$$

*Fig 4.2.b The Projection of the Design Point on the M-P Plane*

**Beam Plastic Deflection**

Here, we are setting a limit for the maximum deflection of the beam; (in our case 10% of the total depth of the beam). Moreover, we will assume that the applied loads on the beam do not change with the magnitude of the deflection. Consequently, we do not expect any change in the load
distribution over the beam. Although we could have used a general-purpose finite element code to calculate the deflection, it was more economical to write a non-linear beam finite element code for this problem (see Chapter 3.)

**Out-of-Plane Buckling**

The out-of-plane buckling (buckling in the plane of the flange) was considered to prevent the optimizer from obtaining narrow beams that can buckle out of plane. We used Euler’s beam buckling formula for this purpose.

### 4.2 Deterministic Structural Optimization

To understand the advantages and disadvantages of the reliability-based optimization, we will compare it with a number of deterministic optimum designs with different safety factors. The intent here is to compare the deterministic optimum designs’ weight and safety level to those of reliability-based optimum designs.

**The Deterministic Optimization Problem Formulation**

\[
\text{find } \bar{d}_{\text{opt}} \\
\text{min } W(\bar{d}) \\
s.t. : G_{i}(\bar{d}, \mu_{x} \times SF) \geq 0 \quad (4.25) \\
\bar{d} \geq \bar{d}^{L}
\]

where \( \bar{d}_{\text{opt}} \) is the vector of design variables (height of the web, the width of the flange, and their thicknesses as shown in Figure 3), \( W(\bar{d}) \) is the weight of the beam, \( G_{i}(\bar{d}, \mu_{x} \times SF) \) are the constraints that depend on and mean values of the random loads \( \mu_{x} \) and the safety factor \( SF \), and \( \bar{d}^{L} \) are the lower limits of the design variables. The constraints are:
1. A strength constraint that prevents failure by total plastic deformation at any cross-section of the beam: the beam can sustain the applied loads without total plastic deformation at any cross-section ($\gamma \geq 1.0$ in equation 4.24), which is specified by the performance function $G_1(\tilde{d}, \tilde{X})$.

2. A constraint about out-of-plane stability that prevents failure by loss of out-of-plane stability: the beam would not buckle out-of-plane, which specifies the performance function $G_2(\tilde{d}, \tilde{X})$.

3. A design constraint that prevents failure by excessive in-plane deflection: the in-plane deflection of the beam will not exceed 10% of its total depth, which specifies the performance function $G_3(\tilde{d}, \tilde{X})$.

Also, the lower limits on the design variables are manufacturing constraints that prevent the thickness from being less than 5 mm for any portion of the beam.

For the optimization process, we have used the Modified Method of Feasible Directions and the Sequential Quadratic Programming optimization algorithms of Visual Doc software (Vanderplaats, 1999; and Ghosh et al, 2000). Both methods converged to the same optimum design in the examples considered.

### 4.3 Reliability-Based Structural Optimization

Two formulations of the reliability-based optimization problem will be considered, *design for maximum reliability* and *design for minimum weight*. Design for maximum reliability is used if the objective is to find the design with highest reliability (probability of survival) whose weight does not exceed a maximum acceptable limit. In this formulation, objective function is the beam reliability. On the other hand, design for minimum weight is used when the objective is to find the lightest beam design whose reliability is no less than a minimum acceptable value. In this formulation,
objective function will be the beam weight. We will use both approaches and compare the obtained designs to the deterministic optimum design. The beam reliability is a function of the failure modes considered earlier, that are the constraints in the deterministic optimization problem. It is important to note that manufacturing constraints are considered as side constraints and will not affect the calculation of the design reliability.

Formulations of Reliability-Based Optimization Problems

a. Design for minimum weight

\[
\begin{align*}
\text{find} & : \tilde{d}_{\text{opt}} \\
\text{min} & : W(\tilde{d}) \\
\text{s.t.} & : R(\tilde{d}, \tilde{X}) \geq R_d^* \\
& : \tilde{d} \geq \tilde{d}^l
\end{align*}
\]

(4.26)

where \( R_d^* \) is the reliability of the corresponding deterministic optimum design against all the failure modes.

b. Design for maximum reliability:

\[
\begin{align*}
\text{find} & : \tilde{d}_{\text{opt}} \\
\text{max} & : R(\tilde{d}, \tilde{X}) \\
\text{s.t.} & : W(\tilde{d}) \leq W_d^* \\
& : \tilde{d} \geq \tilde{d}^l
\end{align*}
\]

(4.27)

where \( \tilde{d}_{\text{opt}} \) is the vector of optimal design variables, \( R(\tilde{d}, \tilde{X}) \) is the system reliability of the beam, which is the probability of survival of the beam under the applied loads (calculated against the previously determined failure modes \( G_i(\tilde{d}, \tilde{X}) \)), \( W(\tilde{d}) \) is the weight of the beam, and \( W_d^* \) is the weight of the corresponding deterministic optimum design.
Methods Used to Calculate the Reliability of the Beam

To carry out the reliability-based optimization we need to calculate the reliability of different beam designs under variable loads and/or material properties. Thus, for this application it would be suitable to use a very economical method to calculate the reliability of a design, such as the First Order Second Moment Methods (FOSM), and in the case of normally distributed variables we can directly use Hasofer-Lind (H-L) method (otherwise we may need to transform the random variables into equivalent normal variables, see Melchers, 2001). The H-L method requires transforming the random variables and limit-state functions into a new space (space of reduced variables) with independent, normal random variables with zero mean and unit standard deviation. Then, we need to find the most probable failure point, which is the closest failure point to the origin in the space of reduced variables.

Finding the most probable failure point for non-linear state functions is an optimization problem. Thus, it would be worthy to check the accuracy of the results by comparing them to those by another more direct method. We chose Monte-Carlo Simulation with Importance Sampling, since using the Direct Monte-Carlo Simulation needs around $10^7$ sample points to for a reliability of 0.999 and a radius of confidence of 1%. Also, to simplify the reliability calculations we will calculate the probability of failure for each failure mode separately and then use the summation of the probabilities of failure for all the failure modes as an upper bound on the system probability of failure\textsuperscript{21}. Once the beam reliability at different design points is calculated, we can use a suitable optimization scheme to find the reliability-based optimum design.

Reliability-Based Optimization Method

As was mentioned earlier, we needed to devise a simple method that demonstrates the features of RBDO and the different considerations that the designer must consider in the RBDO, without being
computationally expensive. However, although the weight of the beam is a simple function of the design variables, the reliability of the beam is an implicit function of the design variables and may not always be differentiable. Hence, one may not expect that we can use one of the standard optimization algorithms to obtain the reliability-based optimum designs. In light of this, one may think of other optimization tools that will be both economical and can easily overcome the discontinuity in the derivatives of the constraint and/or the objective functions.

Thus, we’ve turned to the sensitivity analysis of the beam in the sense that the effect of each design variable on both the objective function and the constraint is studied separately. This analysis was used to identify two groups of important variables: those that significantly affect the weight and those that affect significantly the reliability. This allows us to perturb the design from the deterministic optimum to either increase the reliability of the design with the least increase in the cost, or to reduce the cost with the minimal reduction in reliability. To achieve this, we can vary one design variable at a time and study its effect on both the reliability and the weight of the beam. To save the unnecessary effort, this analysis may be performed in the neighborhood of the optimum deterministic design with the goal of improving its reliability or reduce its cost. Using this information the optimization problem is then changed into a simple one-dimensional search problem that can be solved graphically.

4.4 Numerical Application

We chose a beam of T cross-section, which is commonly used as a stiffener in aerospace, ship, and automotive industries. The geometry and dimensions of the beam are shown in Figure 4.3.a. The loads applied to the beam are an axial compressive load applied to the beam’s centroid and a uniform lateral distributed load with an eccentricity of 4mm, as shown in Figure 4.3.b. Both the axial load $P$ and the distributed lateral load $Q_o$ are random, having normal probability
distribution with mean values 225 kN and 38 kN/m, respectively and coefficients of variation given as 25% and 35%, respectively. The material is assumed to behave as elastic-plastic with yield strength of 400 MPa and Modulus of Elasticity of 70 GPa.

Figure 4.3.a Geometry and Dimensions of the T beam

Figure 4.3.b The Loads Applied to the Beam
Combined Load and Deflection Analysis

The following should be considered in failure analysis of the beam in this example:

i. The applied bending moment ($M_{ext}$) will be a function of the applied loads and the resulting deflection will have the following form:

$$M_{ext} = P \delta - Q_o \frac{z^2}{2} + \frac{Q_o L}{2} z$$  \hspace{1cm} (4.28)

where $\delta$ is the beam deflection in the y-direction and $L$ is the length of the beam.

ii. If any portion of the beam cross-section yields, then the beam bending stiffness is reduced, and thus, the beam may be considered to have failed because of excessive deflection and lose its capacity to bear more load even before reaching the yield surface (see Figure 4.4. a. and 4.4.b.).

Fig 4.4.a Load-Deflection Curve for a T Beam, Used here under Axial and Lateral Loads
iii. This situation may not occur for other boundary conditions, such as fixed end beams, since in that case the beam may continue carrying the load until the yield surface has been reached. Thus, we have modified our constraint formulation accordingly and replaced the limit combined load constraint with the maximum load fraction that the beam can experience before going into bifurcation buckling. However, to avoid getting singularities in the stiffness matrices we limited iterations up to 95% of stiffness reduction of the beam. That is the load at which the slope of the load-deflection curve is less than 5% of its initial value (i.e. when the beam loses 95% of its stiffness). Also, this observed loss of stiffness occurs much earlier than that predicted by the linear elastic beam-column equation given below and the maximum deflection at obtained at the mid-span ($\delta_{\text{max}}$) is greater (see Figure 4.5).

$$\delta_{\text{max}} = \frac{Q_o}{EI} \left[ \sec\left(\frac{kL}{2}\right) - 1 \right] - \frac{Q_o}{8EI} \frac{L^2}{k^2} \quad (4.29)$$
\[ k = \sqrt{\frac{P}{E I}} \]

where \( I \) is the cross-section moment of inertia and \( E \) is the material modulus of elasticity (see Chen and Lui, 1987). These results emphasize the importance of considering the inelastic behavior of the beam.

iv. The shear loads vary linearly along the beam and vanish at the mid-span.

**Deterministic Optimum Designs**

We have first performed deterministic optimization for safety factors (SF) of 1.50, 1.75, 2.00, 2.25, and 2.50. The results are shown in Table 4.1. Also, to check if the obtained designs from optimization were global optima, we perturbed the values of the design variables of the designs and we have arrived at the same optimum design every time.

---

**Figure 4.5 Comparison Between the Results of Non-Linear FEM and the Linear Elastic Formula (Eq. 26)**
Table 4.1. Deterministic Optimum Designs for Various Safety Factors

<table>
<thead>
<tr>
<th>Factor of Safety</th>
<th>Mass (kg)</th>
<th>h (mm)</th>
<th>$t_w$ (mm)</th>
<th>b (mm)</th>
<th>$t_f$ (mm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.50</td>
<td>10.82</td>
<td>244</td>
<td>5</td>
<td>162</td>
<td>5</td>
</tr>
<tr>
<td>1.75</td>
<td>11.69</td>
<td>269</td>
<td>5</td>
<td>170</td>
<td>5</td>
</tr>
<tr>
<td>2.00</td>
<td>12.53</td>
<td>292</td>
<td>5</td>
<td>178</td>
<td>5</td>
</tr>
<tr>
<td>2.25</td>
<td>13.34</td>
<td>317</td>
<td>5</td>
<td>185</td>
<td>5</td>
</tr>
<tr>
<td>2.50</td>
<td>14.17</td>
<td>341</td>
<td>5</td>
<td>191</td>
<td>5</td>
</tr>
</tbody>
</table>

Now, before calculating the respective probabilities of failure of these deterministic optimum designs, we may need to check on the suitability of Hasofer-Lind (H-L) method used to calculate the probabilities of failure in our application. For this purpose, we chose the design with the lowest safety level (highest probability of failure), which is the deterministic optimum design for SF=1.50. We have obtained the probabilities of failure for the three performance functions using Monte-Carlo Simulation with Importance Sampling and by using H-L method. They were in reasonable agreement (see Table 4.2). However, for such large probabilities of failure (a typical acceptable probability of failure is less than $1 \times 10^{-7}$) we may expect the agreement between FORM and Monte-Carlo simulation not to be exact especially for non-linear constraints. Since, FORM works best for linear constraints and non-linear constraints at small probability of failure (see Madsen et al. (1986); Haldar and Mahadevan, (1995); Melchers, (2001))

Table 4.2. Probabilities of Failure for the Three Performance Functions ($G_1(X)$, $G_2(X)$, and $G_3(X)$) Using Monte-Carlo Simulation with Importance Sampling (IS) and by Using H-L Method for the Deterministic Optimum Design with SF of 1.50.

<table>
<thead>
<tr>
<th></th>
<th>$G_1$</th>
<th>$G_2$</th>
<th>$G_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>IS</td>
<td>H-L</td>
<td>IS</td>
</tr>
<tr>
<td>$P_f$</td>
<td>1.0E-2</td>
<td>1.10E-2</td>
<td>2.1E-2</td>
</tr>
</tbody>
</table>
Also, we have plotted the three limit state functions \( G_i(X) = 0 \) on the load space to see their degree of non-linearity and to check for the existence of local minima (see Figure 4.6).

![Figure 4.6 The Boundary Performance Functions \( G_1(X) = 0, G_2(X) = 0 \) and \( G_3(X) = 0 \), Plotted on the Random Loads Plane (the loads are normalized by their mean values).](image)

These limit state functions did not show a high level of non-linearity; nor did they have a local minimum. These trends show that it would be adequate to use H-L method to calculate the reliability of the T beam. However, it is interesting to note that the deflection limit state function \( G_3(X) = 0 \) was not monotonic, which is due to the fact that certain load ratios (axial/lateral) produce more deflection before the buckling failure than the others. Finally, we obtained the probabilities of failure for the other deterministic optimum designs (SF of 1.75, 2.00, 2.25, and 2.50) for the three failure modes and they are listed (along with the results of SF 1.50) in Table 4.3.
Table 4.3. Probabilities of Failure for the Deterministic Optimum Designs (SF of 1.5, 1.75, 2.00, 2.25, and 2.50)

<table>
<thead>
<tr>
<th>Factor of Safety</th>
<th>Failure Modes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$G_1$</td>
</tr>
<tr>
<td>1.50</td>
<td>1.10E-2</td>
</tr>
<tr>
<td>1.75</td>
<td>2.03E-4</td>
</tr>
<tr>
<td>2.00</td>
<td>1.07E-6</td>
</tr>
<tr>
<td>2.25</td>
<td>7.4E-10</td>
</tr>
<tr>
<td>2.50</td>
<td>1.5E-13</td>
</tr>
</tbody>
</table>

Reliability-Based Optimum Designs

We have performed sensitivity analysis around each deterministic optimum design. Specifically, we have varied each design variable around the deterministic optimum and plotted the beam probability of failure against the percent increase in weight. This allowed us to determine which design variables we should increase to get the maximum increase in reliability (see Figs. 4.7.a – 4.11. a). We used this information to solve both reliability-based structural optimization problems a. and b. (design for maximum reliability and design for minimum weight, respectively). Initially, we planned to vary each design variable around ±10 % of its value, but we later modified this range during calculations according to the contribution of each design variable to the reliability and weight of the design. Also, since the thicknesses of the flange and the web were constrained to be no less than 5 mm we studied the sensitivity for values above that limit up to 15%. In addition, we used quadratic interpolation between the points. Following are the results of reliability-based optimization starting from the corresponding deterministic design for safety factors of 1.50, 1.75, 2.00 and 2.25.

• **Deterministic optimum design with SF 1.50:** Fig 4.7.a shows the reliability of the deterministic optimum design as a function of the weight when the each of the four dimensions of the cross
section is changed -- one at a time -- while the other three dimensions are fixed at their respective values corresponding to the deterministic optimum design. The flange width, $b$, is the most important design variable for -4% to 10% change in the design weight, while the web thickness, $t_w$, has the least effect within that range (see Fig 4.7.a). However, since the web thickness was at its lower limit we chose the web height, $h$, to be the variable to be reduced. Now, for the design for maximum reliability (Prob. a.), we can improve the reliability of the design for the same weight by increasing the flange width and reducing the web height in a way that the weight remains the same. However, the improvement was insignificant, but at least it achieved a more equal balance between different failure modes (see Figure 4.7.b.). Consequently, for the second optimization problem (Prob. b.), it was unlikely to obtain a design with a weight less than the weight of the deterministic optimum design.

![Figure 4.7.a Effects of Varying the Design Variables on the Weight and Probability of Failure of the Deterministic Optimum (SF 1.50)](image-url)
Deterministic optimum design with SF 1.75: The flange width, $b$, is the most important design variable for -4% to 10% change in the design weight, while the web height $h$ has the least effect within that range (see Fig 4.8.a). For the first reliability-optimization problem (Prob. a.) we have improved the reliability of the design for the same weight by increasing the flange width and reducing the web height (see Fig 4.8.b.). The situation was favorable for the second optimization problem (Prob. b.) and we were able to save about 1.5% of the weight without reducing reliability.

Fig 4.7.b Searching for the Optimum Design that has the Lowest Probability of Failure ($P_f$)
Figure 4.8.a Effects of Varying the Design Variables on the Weight and Probability of Failure of the Deterministic Optimum (SF 1.75)

Fig 4.8.b Searching for the Optimum Design that has the Lowest Probability of Failure ($P_f$)
Deterministic optimum designs with SF 2.00, 2.25, and 2.50: These three designs showed similar trends as in the deterministic design with SF 1.75 (see Figs. 9.a to 11.a) and we were able to improve their respective values of the reliability from \((1.0 - 3.0 \times 10^{-7})\) to \((1.0 - 5.0 \times 10^{-9})\), from \((1.0 - 3.1 \times 10^{-5})\) to \((1.0 - 4.1 \times 10^{-6})\), and from \((1.0 - 2.0 \times 10^{-9})\) to \((1.0 - 2.0 \times 10^{-12})\) respectively. Also, we were able to save weight by 1.27%, 3.6%, and 5.2% respectively. Figures 4.12.a and 4.12.b summarize these findings.
Figure 4.9.a Effects of Varying the Design Variables on the Weight and Probability of Failure of the Deterministic Optimum (SF 2.00)

Fig 4.9.b Searching for the Optimum Design that has the Lowest Probability of Failure ($P_f$)
Figure 4.10.a Effects of Varying the Design Variables on the Weight and Probability of Failure of the Deterministic Optimum (SF 2.25)

Fig 4.10.b Searching for the Optimum Design that has the Lowest Probability of Failure ($P_f$)
Figure 4.11.a Effects of Varying the Design Variables on the Weight and Probability of Failure of the Deterministic Optimum (SF 2.50)

Searching for the Optimum Design with the Lowest Pf (SF 2.50)

Fig 4.11.b Searching for the Optimum Design that has the Lowest Probability of Failure ($P_f$)
4.5 Discussion of Reliability-Based Optimization Results

From Table 4.3, we observe that the reliability of the beam designed deterministically for a factor of safety of 1.50 is rather low (<0.97). This shows that the factor of safety may not be an adequate method to describe the level of safety of a design. Figs 4.7.a to 4.11.a provide an insight into the behavior of the reliability and the weight of the beam as functions of each design variable. In addition, looking at Figs 4.7 to 4.11, one may notice the discontinuity of the beam reliability function and how the contribution of each design variable to the reliability of the beam may differ before and after this discontinuity. Moreover, looking at Figs 4.7.b-4.11.b we can see that the location of the deterministic optimum designs in the space of the reliability-design variables is located near just one failure mode. This is because the deterministic optimizer stops as soon as it reaches any of the constraints and no further improvement in the objective function is possible. Thus, the risk of failure may not be evenly distributed among the failure modes and further improvement of the design (by making it more reliable or obtain one with less weight) may be possible by redistributing the material to more equal risks of failure among the three failure modes.

Finally, as we can see from Figs. 4.12.a. and 4.12.b. the benefits of the reliability-based optimization over the deterministic optimization increase for higher safety levels and the increase in benefits is monotonic as found in an earlier study on the design of aircraft wings subject to gust loads\textsuperscript{35}. This is because as the safety level of the design increases more material may be used, and thus, a better distribution of the material can yield higher benefits.
Figure 4.12.a Comparing the Reliability of the Deterministic and the Reliability-Based Optimum Designs

Figure 4.12.b Weight Saving with Reliability-Based Design over the Deterministic Design
4.6 Summary and Conclusions

An elastic-plastic beam design of T cross-section under combined loads was optimized. Three failure modes were considered: failure due to yielding of the entire beam cross section under the combined action of axial and lateral loads, buckling in the plane of the flange, and excessive deflection in the plane of the web. A load interaction surface was constructed to determine the limit load that combines axial, bending, and shear loads. Also, a non-linear finite element procedure was used to determine the deflection of the beam in the plastic range. Next, deterministic optimization was carried out to find the lightest beam design that can support the applied loads for a given safety factor. The deterministic optimization was repeated to obtain optimum designs for a number of different safety factors. Then, the probability of failure was calculated for each deterministic optimum design using a) Monte-Carlo simulation with importance sampling and b) first-order second-moment methods. Finally, reliability-based optimization was performed to find a) the lightest design that has same reliability as the deterministic design and b) the most reliable design that has same weight as the deterministic design.

The results show that the reliability-based optimization can obtain designs with lower probability of failure than the designs obtained by the deterministic optimization. Consequently, this advantage can be used to obtain safer designs that have the same weight as the deterministic optimum designs, or designs that have less weight and the same probability of failure as the deterministic ones. Also, these results show we were able to deal with the problem of high non-linearity and derivative discontinuity of the reliability function in such a way that allowed us to obtain an improved design over the deterministic design in most cases.
Chapter 5

Application of the Sequential Optimization with Probabilistic Safety Factors (SORFS) RBDO to the Exploratory Elastic-Plastic T-Beam Example

5.1 Introduction

Before considering a more involved example, it would be appropriate to use the exploratory elastic-plastic T-beam example that we have considered in Chapters 3 and 4 to investigate the SORFS RBDO technique (see Sec 2.4.4). In particular, we will compare the results of the graphical method used earlier (see Ch. 4) to the results of SORFS RBDO method. Also, we will examine the ability of the SORFS method to overcome the problems of high computational costs and the high non-linearity and derivative discontinuity of the reliability function. Finally, we will study the convergence of the SORFS RBDO method.

5.2 The Optimization Problem

5.2.1 Problem Statement

The following is a brief presentation of the beam optimization problem that we have considered in Chapter 4. The beam loads and dimensions are shown in Figure 5.1. It is required to find the beam design that has the minimum weight and satisfies the following constraints:

1. The beam can sustain the applied load without failure \((G_1)\)
2. The beam must not experience out-of-plane buckling \((G_2)\)
3. The beam must not have an in-plane deformation that exceed 10% of its total depth \((G_3)\)
4. Design Constraint: *no thickness less than 5 mm*

The Design Variables are: the web depth $h$, web thickness $tw$, flange width $b$ and flange thickness $tf$.

$P_o = 225 \text{ kN}$

$COV = 25\%$

**Figure 5.1 The Beam Loads and Dimensions**

Also, it is required to obtain three optimum designs:

a. A deterministic optimum design that is obtained by using deterministic safety
factor.

b. A reliability-based optimum design that has the same level of safety of the deterministic optimum design but with the least possible weight.

c. A reliability-based optimum design that has the same weight as the deterministic optimum design but with the highest reliability.

5.3 The Sequential Optimization with Reliability-Based Factors of Safety Approach

To compare the results of the SORFS technique with the graphical approach that we have used before, we will perform the optimization for the five levels of safety considered in Chapter 4 (safety factors of 1.50, 1.75, 2.00, 2.25 and 2.50). Accordingly we will follow the following steps to obtain the two reliability-based optimum designs for each factor of safety (see Ch.2 for more details):

1. Perform deterministic optimization for the given deterministic safety factor

2. Calculate the reliability of the deterministic optimum design using FORM (or any other reliability calculation method), and we set the maximum allowable probability of failure to equal the calculated reliability of this deterministic optimum design. Also, we identify the active and the near active constraints (having probability of failure that is larger than 10% of the active constraints.).

3. Perform semi-probabilistic optimization using the values of the safety indices $\beta_\gamma$ of the deterministic optimum design as initial guesses (see Eq. 2.28 and 2.29), and fixing the coordinates of the most probable failure points (MPP) $\bar{X}$ (see Eq. 2.29)
4. Perform performance measure analysis (PMA) to find corrected most probable failure points (MPP) $\bar{X}$ that correspond to the optimum safety indices $\beta_r$ obtained in step 3, and use them as the lower bounds on the $\beta_r$ for the next optimization iteration.

5. Calculate the reliability of the new optimum design

Repeat steps 3-5 until getting a design with the least weigh that has a reliability that is equal or better than the reliability of the corresponding optimum design is obtained.

Finally, to obtain a reliability-based optimum design that has the highest probability for a given weight we use the same steps above, but with the system reliability as the objective function and the weight as a constraint. Dissertation

5.4 Reliability-Based Optimization Results for the Exploratory Elastic-Plastic T-Beam Example using the SORFS Approach

For this example we found that the constraints $G_1$ and $G_2$ were competing constraints (see Figure 4.6.) Since $G_2$ depends only on the value of the axial load, and to improve the design against the corresponding out-of-plane failure, the flange width has to be increased. On the other hand, $G_1$ depends more on the flexural strength of the beam, which requires deeper web more than a wider flange. The results were in excellent agreement with the results that we have obtained earlier in Chapter 4 (compare Figures 5.2 and 5.3 to Figures 4.12.a and 4.12.b.) In addition, we were able to achieve an average reduction of calculations of about 80% compared to the graphical method used in Chapter 4.
Figure 5.2 Comparing the Reliability of the Deterministic and the Reliability-Based Optimum Designs
Finally, to check the convergence of the method, we have performed convergence analysis for both the RBDO problems for SF 2.50, and the results are shown in Table 5.1 and Table 5.2. Similar convergence results were found for SF 2.25.
Table 5.1 Convergence of RBDO T-Stiffener Example (finding the least weight for a certain level of safety) for SF 2.50 using the SORFS Method

<table>
<thead>
<tr>
<th>Deterministic Opt</th>
<th>RBDO Cycle 1</th>
<th>Cycle 2</th>
<th>Cycle 3</th>
<th>Cycle 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>h=0.341</td>
<td>h = 0.2960257</td>
<td>h = 0.3146491</td>
<td>h = 0.3151761</td>
<td>h = 0.3151761</td>
</tr>
<tr>
<td>t_w=0.005</td>
<td>t_w = 0.00500</td>
<td>t_w = 0.005</td>
<td>t_w = 0.005</td>
<td>t_w = 0.00500</td>
</tr>
<tr>
<td>b=0.191</td>
<td>b = 0.1928092</td>
<td>b = 0.191884</td>
<td>b = 0.1950536</td>
<td>b = 0.1950536</td>
</tr>
<tr>
<td>t_f=0.005</td>
<td>t_f = 0.005</td>
<td>t_f = 0.005</td>
<td>t_f = 0.00500</td>
<td>t_f = 0.00500</td>
</tr>
<tr>
<td>Weight= 14.1512</td>
<td>Weight= 13.47389</td>
<td>Weight= 13.572</td>
<td>No change</td>
<td></td>
</tr>
<tr>
<td>h = 0.3260836</td>
<td>h = 0.3001136</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t_w = 0.0050</td>
<td>t_w = 0.00500</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>b = 0.2057630</td>
<td>b = 0.2003197</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t_f = 0.00500</td>
<td>t_f = 0.00500</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Weight= 14.15</td>
<td>Weight= 14.15</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G_1 Xp = 2.354154</td>
<td>G_1 Xp = 2.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Xq = 2.7081620</td>
<td>Xq = 2.1550</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \beta_T1 ) = 7.2909976</td>
<td>( \beta_T1 ) = 6.707436</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( P_{r} ) = 1.549889E-013</td>
<td>( P_{r} ) = 7.0023E-012</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G_1 Xp = 2.1550</td>
<td>G_1 Xp = 2.19834</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Xq = 2.758314</td>
<td>Xq = 2.758314</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \beta_T1 ) = 6.943662</td>
<td>( \beta_T1 ) = 6.943662</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( P_{r} ) = 7.0023E-012</td>
<td>( P_{r} ) = 1.92237E-012</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PMAX Xp = 2.1550</td>
<td>PMAX Xq = 2.934</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G_2 Xp = 2.472443881</td>
<td>G_2 Xp = 3.0903</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Xq = 1.00000000</td>
<td>Xq = 1.00000000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \beta_T1 ) = 5.88978</td>
<td>( \beta_T1 ) = 5.88978</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( P_{r} ) = 1.940E-009</td>
<td>( P_{r} ) = 7.0023E-012</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G_3 (Out of range)</td>
<td>G_3 (Out of range)</td>
<td>G_3 (Out of range)</td>
<td>G_3 (Out of range)</td>
<td></td>
</tr>
</tbody>
</table>

Table 5.2 Convergence of T-Stiffener Example RBDO (finding the highest reliability for a certain weight) for SF 2.50 using the SORFS Method

<table>
<thead>
<tr>
<th>Deterministic Opt</th>
<th>RBDO Cycle 1</th>
<th>Cycle 2</th>
<th>Cycle 3</th>
<th>Cycle 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>h=0.341</td>
<td>h = 0.3260836</td>
<td>h = 0.3316407</td>
<td>h = 0.3316407</td>
<td>h = 0.3316407</td>
</tr>
<tr>
<td>t_w=0.005</td>
<td>t_w = 0.00500</td>
<td>t_w = 0.00500</td>
<td>t_w = 0.00500</td>
<td>t_w = 0.00500</td>
</tr>
<tr>
<td>b=0.191</td>
<td>b = 0.2003197</td>
<td>b = 0.2003197</td>
<td>b = 0.2003197</td>
<td>b = 0.2003197</td>
</tr>
<tr>
<td>t_f=0.005</td>
<td>t_f = 0.00500</td>
<td>t_f = 0.00500</td>
<td>t_f = 0.00500</td>
<td>t_f = 0.00500</td>
</tr>
<tr>
<td>G_1 Xp = 2.354154</td>
<td>G_1 Xp = 2.1550</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Xq = 2.7081620</td>
<td>Xq = 2.1550</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \beta_T1 ) = 7.2909976</td>
<td>( \beta_T1 ) = 6.707436</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( P_{r} ) = 1.549889E-013</td>
<td>( P_{r} ) = 7.0023E-012</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G_1 Xp = 2.1550</td>
<td>G_1 Xp = 2.19834</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Xq = 2.758314</td>
<td>Xq = 2.758314</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \beta_T1 ) = 6.943662</td>
<td>( \beta_T1 ) = 6.943662</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( P_{r} ) = 7.0023E-012</td>
<td>( P_{r} ) = 1.92237E-012</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PMAX Xp = 2.1550</td>
<td>PMAX Xq = 2.934</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G_2 Xp = 2.472443881</td>
<td>G_2 Xp = 3.0903</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Xq = 1.00000000</td>
<td>Xq = 1.00000000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \beta_T1 ) = 5.88978</td>
<td>( \beta_T1 ) = 5.88978</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( P_{r} ) = 1.940E-009</td>
<td>( P_{r} ) = 7.0023E-012</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G_3 (Out of range)</td>
<td>G_3 (Out of range)</td>
<td>G_3 (Out of range)</td>
<td>G_3 (Out of range)</td>
<td></td>
</tr>
</tbody>
</table>
5.5 Discussion of the Results

From Figures 5.2 and 5.3, we may realize that we were generally able to obtain improved optimum designs (that has less weight than the corresponding deterministic optimum design for the same reliability or that has higher reliability for the same weight of the deterministic optimum design). However, for the same reasons that were explained in Chapter 4, we could not get a substantial improvement for the deterministic optimum design with safety factor 1.50. Yet, the optimization process did not encounter any difficulty that is related to the reliability function, since the optimization is performed deterministically without using an actual reliability objective function or constraints in the optimization process. Moreover, using the SORFS RBDO method we were able to achieve large reductions in calculations (an average of 80% reduction). Finally, the convergence analysis as presented in Tables 5.1 and 5.2 indicates that the method converged after the second semi-probabilistic optimization cycle.
Chapter 6

Nonlinear Finite Element Model of 14 Degrees of Freedom

Elastic-Plastic Beam

6.1 The element loads

We consider a beam of arbitrary open constant cross-section under the action of a system of loads (see Figure 6.1). These loads that are applied externally to the beam surfaces may include an axial load $N$, shear forces in the $X$ and $Y$ directions ($V_x$ and $V_y$), moments around the $X$ and $Y$ axis ($M_x$ and $M_y$), torsion around the shear center $T$, and distributed lateral loads in the $X$ and $Y$ direction ($w_x$ and $w_y$).

![Figure 6.1 A Prismatic Beam of Arbitrary Open Cross-Section under the Action of a System of Loads](image)
6.2 Cross-Section Deformation Modeling

In the following we are going to model the cross-section deformations in a way similar to that of Trahair (1993). Let’s define the beam element translations in the X, Y, Z directions (the origin is at the centroid) by \( u, v, w \) respectively. We will follow the Euler-Bernoulli beam assumptions, and hence, the rotations around the X and Y axes are given by (for small rotations around the X and Y axes): \[
\frac{dv}{dz} \equiv v' \quad \text{and} \quad \frac{du}{dz} \equiv u',
\] where a prime indicates derivatives with respect to the beam axis, and the counter clock-wise is taken to be the positive direction. Also, the rate of change of twist of the beam around the shear center will be denoted by \( \phi' \) (see Figure 6.2.), which is also positive if in the counter clock-wise.

![Figure 6.2 Beam Element Deformations under a System of Loads](image)

For a long prismatic beam under the given load the cross-section experiences shear center displacements \( u_S, v_S, w_S \) parallel to the original X, Y, Z axes that displaces \( S_o \) to \( S_l \). Also, a twist rotation \( \phi \) around the displaced shear center \( S_l \). Therefore a point \( P_o \) in the cross-section will
experience a displacement to \( P_1 \) (parallel to \( S_o-S_i \)) and will rotate around \( S_i \) to \( P_2 \) (see Figure 6.3 (a)-(c)).

Figure 6.3. Displacements of a Point P in the Beam Cross-Section
We are assuming that the cross-section deformations will be moderately small, and that the rotations will be moderately large. Therefore, we are using small angle approximation ($\sin \phi \approx \phi$, $\cos \phi \approx 1$) we can write the displacements of point P$_2$ as

\[
u_{p_2} = u_s - (y - y_o)\phi \quad (6.1)
\]

\[
u_{p_2} = v_s + (x - x_o)\phi \quad (6.2)
\]

The rotation of the line $S_1P_1$ in the X-Z plane is denoted by $u'$ and its corresponding deflection in that plane makes a displacement in the axial direction $Z$ of:

\[
w_{p_2}' = -u'(x - x_o) - (y - y_o)\phi \quad (6.3)
\]

Similarly, the rotation of the line $S_1P_1$ in the Y-Z plane is denoted by $v'$, and its corresponding deflection in that plane makes a displacement in the axial direction $Z$ of:

\[
w_{p_2}' = -v'(y - y_o) + (x - x_o)\phi \quad (6.4)
\]

Now, we need to consider the axial St.Venant warping displacement of point $P_2$ due to twist around the shear center. For that we need to consider the two types of warping that an open-section thin-walled beam may experience; the primary warping along the mid-thickness of the section and the secondary warping across the thickness (see Figure 6.4). To determine the primary warping of a thin-walled beam, let’s look at the components of displacement of a point in a thin walled open section beam.
where \( v_t \) is the tangential component of displacement at a point

\( v_n \) is the normal component of displacement at a point

\( w \) is the axial component of displacement at a point

\( \phi \) is the rotation angle of the cross-section

\( sc \) is the shear center of the cross-section

\( \rho_{sc} \) is the normal distance between the shear center and the tangent to the mid-thickness of the cross-section at a point

\( s \) is a dimension that is measured around the cross-section of the beam tangent to the mid-thickness

Now, let’s look at the shear strains at an element \( \delta z \times \delta s \) of the beam wall as shown in Figure 6.5.
We can express the shear strain as:

$$\gamma_{zs} = \frac{\partial w}{\partial s} + \frac{\partial v_t}{\partial z}$$  \hspace{1cm} (6.5)$$

Also, we can express the tangent component of displacement as

$$v_t = (\rho_{sc} + n) \phi$$  \hspace{1cm} (6.6)$$

Then, we can write

$$\gamma_{zi} = \frac{\partial w}{\partial s} + (\rho_{sc} + n)\phi'$$  \hspace{1cm} (6.7)$$

The secondary warping distribution is assumed to be that of a narrow rectangular strip, so that

$$w_\omega = [ns - \omega(s)]\phi'$$  \hspace{1cm} (6.8)$$

where $\omega(s)$ is the Vlasov’s sectorial coordinate.
The shear strain is given by

$$\gamma_{sz} = \left[2n + \rho_{sc} - \frac{\partial \omega(s)}{\partial s}\right] \phi'$$  \hspace{1cm} (6.9)

For open cross-section the shear flow at the mid-thickness (at \(n = 0\)) is identically zero

$$\int_{-\ell/2}^{\ell/2} G \gamma_{sz} ds = 0$$ \hspace{1cm} (6.10)

which gives us at the mid-thickness

$$\frac{\partial \omega(s)}{\partial s} = \rho_{sc}$$ \hspace{1cm} (6.11)

Let’s define the warping function

$$\omega(s) = \int_{0}^{s} \rho_{sc} ds$$ \hspace{1cm} (6.12)

where the limits of integration are chosen such that

$$\int \omega(s) t ds = 0$$ \hspace{1cm} (6.13)

Then the shear strain reduces to

$$\gamma'_{n} = 2n \phi'$$ \hspace{1cm} (6.14)

Also, we can write the axial displacement as

$$w_{n} = w_{s} - u'\{(x - x_{o}) - (y - y_{o})\phi\} - v'\{(y - y_{o}) + (x - x_{o})\phi\} + (n s - \omega) \phi'$$ \hspace{1cm} (6.15)

So at the centroid \(C\) (where \(x = 0\) and \(y = 0\))

$$w_{c} = w_{s} + x_{o}(u' + v'\phi) + y_{o}(v' - u'\phi) + (n_{c}s_{c} - \omega_{c}) \phi'$$ \hspace{1cm} (6.16)

where \(\omega_{c} = \int_{0}^{c} \rho_{sc} ds\), and \(n_{c}\) and \(s_{c}\) are the \(n\) and \(s\) coordinates of the centroid.
Let,

$$w = w_c + (n_c s_c - \omega_c) \phi'$$  \hspace{1cm} (6.17)

Then,

$$w_s = w - x_o (u' + \phi') - y_o (v' - \phi')$$  \hspace{1cm} (6.18)

and

$$w_p = w - xu' - yv' + (n s - \omega) \phi' - xv' \phi + yu' \phi$$  \hspace{1cm} (6.19)

Now, we can determine the longitudinal and shear strains. The longitudinal strain at point P of the cross-section is obtained as the following (see Figure 6.6.)

$$(1 + \varepsilon_p) \delta_Z = \sqrt{(\delta_Z + \delta_Z w_p')^2 + (u_p' \delta_Z)^2 + (v_p' \delta_Z)^2}$$  \hspace{1cm} (6.20)

Using the binomial theorem (assuming $w_p'^2 << v'$ and $u'$)

$$\varepsilon_p \approx w_p' + \frac{1}{2} (u_p'^2 + v_p'^2)$$  \hspace{1cm} (6.21)

or,

$$\varepsilon_p = w' - xu'' - yv'' + (n s - \omega) \phi'' - xv' \phi - xv' \phi'$$
$$+ y u'' \phi + y u' \phi' + \frac{1}{2} (u' - (y - y_o) \phi')^2$$
$$+ \frac{1}{2} (v' + (x - x_o) \phi')^2$$  \hspace{1cm} (6.22)

After rearranging

$$\varepsilon_p = w' + \frac{1}{2} (u'^2 + v'^2) - x(u'' + v'' \phi) - y(v'' - u'' \phi) + (2n s - \omega) \phi''$$
$$- x_o (v' \phi') + y_o (u' \phi') + \frac{1}{2} ((x - x_o)^2 + \frac{1}{2} (y - y_o)^2) \phi'^2$$  \hspace{1cm} (6.23)
6.3 Deformation field approximation

We will use Hermite polynomials to approximate the element deformations along the element. We will need a first order polynomial to approximate the axial deformation $w$, and cubic polynomials to approximate the lateral $(u, v)$ and twist ($\phi$) deformations. Thus, for the axial deformation, we can write

$$w = (1 \frac{z}{L}) w_o + (\frac{z}{L}) w_L = \left\langle N_1, N_2 \right\rangle \begin{bmatrix} w_o \\ w_L \end{bmatrix}$$

(6.24)

where $w_o$ and $w_L$ are the axial displacements at the element ends (at $z = 0$ and $z = L$).

Similarly, for the lateral deformations, we can write

$$u = (1 - \frac{z^2}{L^2} + \frac{2z^3}{L^3}) u_o + (z - \frac{2z^2}{L} + \frac{z^3}{L^2}) u'_o + (\frac{3z^2}{L^2} - \frac{2z^3}{L^3}) u_L + (-\frac{z^2}{L} + \frac{z^3}{L^2}) u'_L$$

$$= \left\langle H_1, H_2, H_3, H_4 \right\rangle \begin{bmatrix} u_o \\ u'_o \\ u_L \\ u'_L \end{bmatrix}$$

(6.25)

where $u_o$ and $u'_o$ are the lateral displacement and its rate of change along the $z$-direction respectively, and $u_L$ and $u'_L$ are those at the other end. Also, we can write
\[ v = \{ H \} \begin{bmatrix} v_o \\ v'_o \\ v_L \\ v'_L \end{bmatrix} \]  \hspace{1cm} (6.26)

and

\[ \phi = \{ H \} \begin{bmatrix} \phi_o \\ \phi'_o \\ \phi_L \\ \phi'_L \end{bmatrix} \]  \hspace{1cm} (6.27)

Finally, we can write the deformations vector in the form of

\[ \begin{bmatrix} w \\ u \\ v \\ \phi \end{bmatrix} = [T_c] \{ q_1, q_2, \ldots, q_{14} \}^T \]  \hspace{1cm} (6.28)

where

- \( q_1 = w_o \) and \( q_8 = w_L \) are the axial displacements that correspond to axial loads \( N_o \) and \( N_L \), respectively.

- \( q_2 = u_o \) and \( q_9 = u_L \) are the transverse displacements that correspond to shear loads \( V_{xo} \) and \( V_{xL} \), respectively.

- \( q_3 = v_o \) and \( q_{10} = v_L \) are the transverse displacements that correspond to shear loads \( V_{yo} \) and \( V_{yL} \), respectively.

- \( q_4 = \phi_o \) and \( q_{11} = \phi_L \) are the twist rotations that correspond to the uniform torsion \( T_o \) and \( T_L \), respectively.
\( q_5 = u'_o \) and \( q_{12} = u'_L \) are the beam end rotations that correspond to the end moments \( M_{yo} \) and \( M_{yL} \), respectively.

\( q_6 = v'_o \) and \( q_{13} = v'_L \) are the beam end rotations that correspond to the element end moments \( M_{xo} \) and \( M_{xL} \), respectively.

\( q_7 = \phi'_o \) and \( q_{14} = \phi'_L \) are the rates of change of twists that correspond to the ends bi-moments \( B_o \) and \( B_L \), respectively. And

\[
[T_c] = \begin{bmatrix}
T_e \\
T_o \\
T_i \\
T_s
\end{bmatrix} = \begin{bmatrix}
N_1 & 0 & 0 & 0 & 0 & 0 & N_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & H_1 & 0 & 0 & H_2 & 0 & 0 & 0 & H_3 & 0 & 0 & H_4 & 0 & 0 \\
0 & 0 & H_1 & 0 & 0 & H_2 & 0 & 0 & 0 & H_3 & 0 & 0 & H_4 & 0 \\
0 & 0 & 0 & H_1 & 0 & 0 & H_2 & 0 & 0 & 0 & H_3 & 0 & 0 & H_4
\end{bmatrix}
\]

(6.29)

See Figure 6.7 for the beam element generalized displacements and forces.
6.4 Element Stiffness Matrix

The strain energy is expressed as:

\[ U = \frac{1}{2} \iiint_V (\sigma_p \epsilon_p + \tau_p \gamma_p) \, dV = \frac{1}{2} \iiint_V (E \epsilon_p^2 + G \gamma_p^2) \, dV \]  
(6.30)

\[ U = \frac{1}{2} \iiint_V \begin{bmatrix} \epsilon_p \\ \gamma_p \end{bmatrix}^T \begin{bmatrix} E & 0 \\ 0 & G \end{bmatrix} \begin{bmatrix} \epsilon_p \\ \gamma_p \end{bmatrix} \, dV \]  
(6.31)
or symbolically

\[ U = \frac{1}{2} \iiint_V \{ \varepsilon \}^T [D] \{ \varepsilon \} \, dV \]  \hspace{1cm} (6.32)

where \( \{ \varepsilon \}^T = \{ \varepsilon_p \} \).

We can write

\[ \{ \varepsilon \} = [G(x,y)] [\Gamma(z)] \]  \hspace{1cm} (6.33)

Where

\[
[G(x,y)] = \begin{bmatrix}
1 & -x & -y & (n s - \omega) & \frac{1}{2}((x-x_o)^2 + (y-y_o)^2) & 0 \\
0 & 0 & 0 & 0 & 0 & 2n
\end{bmatrix}
\]  \hspace{1cm} (6.34)

And

\[
[\Gamma(z)]^T = \begin{bmatrix}
w' + \frac{1}{2}(u'^2 + v'^2) - x_o v' \phi' + y_o u' \phi' & (u'' + v'' \phi) & (v'' - u'' \phi) & \phi'' & \phi'^2 & \phi'
\end{bmatrix}
\]  \hspace{1cm} (6.35)

Now, we can express the energy equation as

\[ U = \frac{1}{2} \int_0^l \int_A [\Gamma]^T [G]^T [D] [G] [\Gamma] \, dA \, dz \]  \hspace{1cm} (6.36)

which can be written as

\[ U = \frac{1}{2} \int_0^l \int_A [\Gamma]^T [G]^T [D] [G] dA \, \{ \Gamma \} dz \]  \hspace{1cm} (6.37)

Let’s define
\[
[C] = \iint_A [G]^T [D][G] dA
\]  

(6.38)

where

\[ C_{1,1} = EA \]
\[ C_{1,2} = -E \int_A x \, dA \]
\[ C_{1,3} = -E \int_A y \, dA \]
\[ C_{1,4} = E \int_A (ns - \omega) \, dA \]
\[ C_{1,5} = \frac{1}{2} E \int_A (x - x_o)^2 + (y - y_o)^2 \, dA = \frac{1}{2} EI_{p0} \]
\[ C_{1,6} = 0 \]

\[ C_{2,2} = EI_y \]
\[ C_{2,3} = E \int_A xy \, dA = EI_{xy} \]
\[ C_{2,4} = -E \int_A x (ns - \omega) \, dA \]
\[ C_{2,5} = -\frac{1}{2} E \int_A x ((x - x_o)^2 + (y - y_o)^2) \, dA \]
\[ C_{2,6} = 0 \]

\[ C_{3,3} = EI_x \]
\[ C_{3,4} = -E \int_A y (ns - \omega) \, dA \]
\[ C_{3,5} = -\frac{1}{2} E \int_A y ((x - x_o)^2 + (y - y_o)^2) \, dA \]
\[ C_{3,6} = 0 \]

\[ C_{4,4} = E \int_A (ns - \omega)^2 \, dA \]
\[ C_{4,5} = \frac{1}{2} E \int_A (ns - \omega) ((x - x_o)^2 + (y - y_o)^2) \, dA \]
\[ C_{4,6} = 0 \]

\[ C_{5,5} = \frac{1}{4} E \int_A ((x - x_o)^2 + (y - y_o)^2)^2 \, dA = I_{p^2} \]
\[ C_{5,0} = 0 \]
\[ C_{6,0} = G \int_A 4n^2 \, dA \]

so we can write

\[ U = \frac{1}{2} \int_0^l \{ \Gamma \}^T [C(x, y)] \{ \Gamma \} \, dz \quad (6.39) \]

Taking the first variation of the energy

\[ \delta U = \int_0^l \{ \delta \Gamma \}^T [C(x, y)] \{ \Gamma \} \, dz \quad (6.40) \]

where

\[ \{ \delta \Gamma \} = \begin{bmatrix}
\delta w' + u' \delta u' + v' \delta v' - x_o (\delta v' \phi' + v' \delta \phi') + y_o (\delta u' \phi' + u' \delta \phi') \\
\delta u'' + \delta v'' \phi + v'' \delta \phi \\
\delta v'' - \delta u'' \phi - u'' \delta \phi \\
\delta \phi'' \\
2 \phi' \delta \phi' \\
\delta \phi'
\end{bmatrix} \quad (6.41) \]

Now, we will substitute in \{ \delta \Gamma \} the four deformations \{w, u, v, \phi\} and write \{ \delta \Gamma \} as a multiplication of two matrices

\[ \{ \delta \Gamma \} = [A] \cdot \{ \delta d \} \quad (6.42) \]
where

\[
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 \\
u' + y_o \phi' & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & \phi & 0 & 0 & 0 \\
v' - x_o \phi' & 0 & 0 & 0 & 0 & 0 \\
0 & \phi & 1 & 0 & 0 & 0 \\
0 & v'' - u'' & 0 & 0 & 0 & 0 \\
y_o u' - x_o v' & 0 & 0 & 0 & 2 \phi' & 1 \\
0 & 0 & 0 & 1 & 0 & 0
\end{bmatrix}
\]

and

\[
\{\delta d\}_T = \{\delta w' \ \delta u' \ \delta u'' \ \delta v' \ \delta v'' \ \delta \phi \ \delta \phi' \ \delta \phi''\} 
\] 

(6.44)

Then, we can use the interpolation functions and express \{\delta d\} in terms of the element deformations

\[
\{\delta d\} = [Z_c] \{\delta q\} 
\] 

(6.45)

Where

\[
\{\delta q\} = \delta\{q\} 
\] 

(6.46)

And from equation 6.29.

\[
[Z_c] = \begin{bmatrix}
\langle T'w \rangle \\
\langle T'u \rangle \\
\langle T''t \rangle \\
\langle T'v \rangle \\
\langle T''t \rangle \\
\langle T_\phi \rangle \\
\langle T'_\phi \rangle \\
\langle T''_\phi \rangle \\
\langle T'_\phi \rangle \\
\langle T''_\phi \rangle \\
\langle T''_\phi \rangle \\
\end{bmatrix} 
\] 

(6.47)

For convenience, we will define the matrix [B]
\[ [B] = [A] \cdot [Z_r] \quad (6.48) \]

Hence, we can write
\[ \{ \delta \Gamma \} = [B] \cdot \{ \delta q \} \quad (6.49) \]

So, at the equilibrium path
\[
\delta \{ U - V \} = \{ \delta q \} \int_0^l [B]^T [C(x, y)] \{ \Gamma \} \, dz - \{ \delta q \} \{ F_{in} \} = 0 \quad (6.50)
\]

where \{ F_{in} \} is the internal force vector

To obtain the stiffness matrices, we can take the first variation of the internal force vector, which will be:
\[
\{ \delta F_{in} \} = \int_0^l [\delta B]^T [C(x, y)] \{ \delta \Gamma \} \, dz + \int_0^l [B]^T [C(x, y)] \{ \delta \Gamma \} \, dz = 0 \quad (6.51)
\]

or
\[
\{ \delta F_{in} \} = \int_0^l [\delta B]^T [C(x, y)] \{ \Gamma \} \, dz + \int_0^l [B]^T [C(x, y)] [B] \, dz \{ \delta q \} \quad (6.52)
\]

Taking the variation of the \([A]\) matrix
We can write

$$\mathbf{[\delta B]} = \mathbf{[\delta A][Z_c]}$$

(6.54)

However, we need to multiply the matrices in the first term of equilibrium equation to be able to extract the stiffness matrix.

Let’s define

$$\{\Omega\} = \mathbf{[C(x,y)]}\mathbf{[\Gamma]}$$

(6.55)

and let’s define the vector

$$\{S\} = \mathbf{[\delta A]^T}\{\Omega\}$$

(6.56)

$$\{S\} = \begin{bmatrix}
0 \\
(\delta u' + y_o \delta \phi') \Omega_1 \\
-\delta \phi \Omega_3 \\
(\delta v' - x_o \delta \phi') \Omega_1 \\
\delta \phi \Omega_2 \\
(\delta v'' \Omega_2) - (\delta u'' \Omega_3) \\
(y_o \delta u' - x_o \delta v') \Omega_1 + 2 \delta \phi' \Omega_5 \\
0
\end{bmatrix}$$

(6.57)

that can be split into
\[ \{S\} = [S_q] \{\delta d\} \quad (6.58) \]

Where

\[
[S_q] = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \Omega_1 & 0 & 0 & 0 & 0 & y_p \Omega_1 & 0 \\
0 & 0 & 0 & 0 & 0 & -\Omega_3 & 0 & 0 \\
0 & 0 & 0 & \Omega_1 & 0 & 0 & -x_p \Omega_1 & 0 \\
0 & 0 & 0 & 0 & \Omega_2 & 0 & 0 & 0 \\
0 & 0 & -\Omega_3 & 0 & \Omega_2 & 0 & 0 & 0 \\
0 & y_p \Omega_1 & 0 & -x_p \Omega_1 & 0 & 0 & 2\Omega_5 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix} \quad (6.59)
\]

Substituting for \(\{\delta d\}\) by its interpolation functions (Eq. 6.42) in the \(\{\delta F_{in}\}\) expression (equation 6.52), we get

\[
\{\delta F_{in}\} = \left\{ \int_0^l [Z_c]^T [S_q] [Z_c] \, dz + \int_0^l [B]^T [C(x, y)] [B] \, dz \right\} \{\delta q\} \quad (6.60)
\]

Now, we define the element tangent stiffness matrix as

\[
[K_t] = \int_0^l [Z_c]^T [S_q] [Z_c] \, dz + \int_0^l [B]^T [C(x, y)] [B] \, dz \quad (6.61)
\]

in which,

\[
[K_\sigma] = \int_0^l [Z_c]^T [S_q] [Z_c] \, dz \quad (6.62)
\]

is the element stability matrix (that is known as the geometric stiffness matrix), and

\[
[K_e] = \int_0^l [B][C(x, y)][B]^T \, dz \quad (6.63)
\]
is the element stiffness matrix (that contains linear and non-linear terms).

### 6.5 Application to an elastic-plastic T beam under combined loads

For a T section with the origin at the centroid we have

\[ C_{1,2} = C_{1,3} = C_{1,4} = C_{2,3} = 0 \]  \hspace{1cm} (6.64)

In addition, since we have a mono-symmetric T section, we have \( x_o = 0 \). Also, since the shear center of the T section is at the intersection of its branches, then \( \rho_{sc} = 0 \). This means that \( \omega = 0 \), and thus, the primary axial warping rigidity is reduced. However, the resistance to torsion is also provided by the resistance to the secondary warping mechanism that acts across the thickness, which is resisted by the rigidities in \([C]\)

\[ G \int_A n^2 \, dA = GJ \]  \hspace{1cm} (6.65)

and

\[ E \int_A n^2 s^2 \, dA \]  \hspace{1cm} (6.66)

Also, the torsion is resisted by the other cross-rigidities that resist the twist \( \phi' \) combined with the other lateral displacements \( u' \) and \( v' \). To have a better view, we can compare the difference between the resistances to torsion for an I-section and T-section (see Figs. 6.8 and 6.9.) In the case of an I-beam \( \omega \neq 0 \), and thus both the primary and secondary warping mechanisms can resist torsion if the beam was constrained against warping. On the other hand, for a T-section and under the same constraint conditions only the secondary warping mechanism is resisting torsion along with the other cross rigidities in \([C]\).
Primary warping: torsion is resisted by bi-moment acting on the flanges (in the presence

Secondary warping: torsion is resisted by warping through the thickness (its effect is much

Figure 6.8. Resistance to Torsion Provided by an I-Beam

Figure 6.9 Resistance to Torsion Provided by a T-Beam
6.6 Element Integrations

Since the axial deformation is assumed to change linearly along the beam element, we will use Gauss quadrature of two points for the integration along the element. However, the integrations over the cross-section to calculate the internal forces and the element rigidities need special considerations. First, we need to integrate over the cross-section using a suitable number of integration points. For that, we will use Lobatto quadrature since it provides integration points on the edges of the cross-section where we expect the stresses to be the highest. Second, we need to devise a method to determine the state of stress on the cross-section from which we can calculate the internal forces and the rigidities. This method must take into account three dimensional state of stress that is present in the cross-section (since $\sigma_z$, $\tau_{xz}$, $\tau_{yz} \neq 0$.) Also, it must be known that as the loads are increased, some portions of the cross-section may develop plastic regions which may grow plastically or unload elastically. This means that we must be able to determine when the stress at an integration point reaches the yield surface, and to be able to return to the yield surface if we have departed from it during iteration. Also, we need to control the iteration step size to limit the divergence errors. Finally, we will use a consistent tangent modular matrix to improve the convergence of the method. We will use the generalized Von Mises yield criterion.

$$\sigma_y = \sqrt[6]{(\sigma_x - \sigma_y)^2 + (\sigma_y - \sigma_z)^2 + (\sigma_z - \sigma_x)^2 + 6(\tau_{xy}^2 + \tau_{yz}^2 + \tau_{xz}^2)}$$ (6.67)

Also, we will devise a backward Euler return procedure to ensure that our stresses do not depart from the yield surface. Moreover, we will use the strains to update the stresses as in the following module:

Solution Method
We have used the full Newton-Raphson method for better convergence, and since the tangent stiffness matrix during iterations does not relate to an equilibrium state, then we may not be able to use the line search technique to optimize the iterations step length. Instead, we will use the full amount of displacements during iterations.

*Step control*

Since we are going to encounter some path non-linearity due to the onset of buckling or plasticity, it is recommended that we use some form of displacement control. Hence, the same step control method used in Ch.3 will be used here. However, since the equilibrium curve has a varying slope, we have implemented a scheme over the displacement increments to reduce the step size when there is a large variation in the equilibrium curve slope.

### 6.7 Post-Buckling Analysis

After the onset of buckling, the beam may experience either a limit-point type of buckling in which case the equilibrium path is unique and may raise or drop; or it may experience type of bifurcation buckling in which case more than one equilibrium path could exist. In our study, we may expect to have a bifurcation buckling, since the beam in our problem has multiple failure modes. Hence, the stability of the equilibrium path is what decides the post-buckling behavior of the beam. However, the analysis of the second variation of the potential energy gives us information about the critical points and the states of stability along the different equilibrium paths. Particularly, we know that when the beam is critically stable, the tangent stiffness matrix is singular, which means that at least one eigenvalue of the stiffness matrix is zero.
The eigenvector that corresponds to the vanishing eigenvalue gives direction of deformation of the post-buckling state of the beam (see Thomas and Gallagher, (1975); De Borst, (1987); Riks et al., (1996); Ronagh and Bradford (1999); Riks, (2000); and Bazant and Cedolin (2003))

Based on the above, researchers have proposed a number of ways to switch to the stable equilibrium branch after the onset of buckling. For instance, Ronagh and Bradford (1999) employed a method in which they constrain the solution from converging to the primary equilibrium path by superimposing a displacement vector that does not contain the primary path displacements over the displacements that exist at the initiation of buckling. In that method they augmented the total energy potential equation with a Lagrangian multiplier constraint that used a matrix to assign a zero value to the component of displacement in the direction of the primary solution; otherwise it gives a non-zero value. This value depends on the angle between the primary solution vector and the new vector, and is proportional to the Eigenvector at the onset of buckling (Eigenvector that corresponds to the vanishing eigenvalue at the critical point.) Finally, iterations are performed to converge to the displacement vector that satisfies the constraints. However, since we are using displacement control, we can use a more direct method that was proposed by De Borst (1987), which is similar to the one used by Riks et al (1996) and Riks (2000), in which they used displacement control and modified the post-buckling displacement $\{\delta q_{pg}\}$ vector by using a weighted and normalized buckling eigenvector $\{\nu\}$ in the following form:

$$
\{\delta q_{pg}\} = \frac{\{\delta q\}^T \{\nu\} - \{\delta q\}^T \{\delta q\} \{\nu\}}{\sqrt{\{\delta q\}^T \{\delta q\} - \{\delta q\}^T \{\nu\} \{\nu\}^T}} \quad (6.68)
$$

It must be noted that the tangent stiffness matrix that we are using is composed of two parts; the element stiffness matrix $[K_e]$ that came from the first variation of the strain energy (see equations
and the element stability matrix \([K_\sigma]\) that we obtained by considering the second variation of the strain energy (see equations 6.52 and 6.62). Thus, we may linearize the Eigenvalue problem near the critical point and consider the tangent stiffness matrix to be linear with respect to the applied loads fraction \(\lambda\). This is expressed in the form:

\[
[K_e](\delta \vec{y}) = ([K_e] + \delta \lambda [K_\sigma])\{\delta \vec{y}\} = \{0\}
\]  
(6.69)

Then, we can multiply equation 6.69 by the inverse of the element stiffness matrix \([K_e]^{-1}\) to get

\[
([I] + \delta \lambda [K_e]^{-1}[K_\sigma])\{\delta \vec{y}\} = \{0\}
\]  
(6.70)

where \([I]\) is the identity matrix. Also, we can put the Eigenvalue problem in the form

\[
[K_e]^{-1}[K_\sigma] = -\frac{1}{\delta \lambda}[I]
\]  
(6.71)

Finally, we can determine the critical Eigenvector \(\nu\) to be the one corresponding to the maximum Eigenvalue of \([K_e]^{-1}[K_\sigma]\) (see Ronagh and Bradford (1999); Bazant and Cedolin (2003))

### 6.8 Accounting for Plastic Deformations

**Assumptions and Considerations**

Since the beam may experience considerable plastic deformations before its total collapse, we must account for loads that exceed the beam elastic limit and cause the beam to deform plastically. For this purpose, we must consider the state of the stress at each integration point and update the displacements accordingly. However, elastic unloading may occur during deformation.
due to possible occurrences of different buckling modes, which may add some complication to the convergence of the problem. Since this elastic unloading may cause some portions of the beam to unload elastically, and thus, would not follow the same tangent stiffness matrix as the other portions of the beam that are under increasing plastic loads (Crisfield, 1991, Bathe 1996, Belytschko et al, 2000).

Also, we must consider the new relations that will exist between the stresses and the strains at the onset of plastic yielding. In particular, we must obtain the stiffness matrix that relates the applied external loads to the beam deformation after the onset of plasticity, and we must be able to calculate the internal forces (that must be in equilibrium with the external forces) from integration of the stresses over the beam’s cross-section. However, the state of stress in our case only considers the axial and the through-thickness shear stresses (see Eq. 6.30). This is because the contour shear stresses are contributing to the axial stress at every point through the warping function (see Eq. 6.9 and 6.15).

Therefore, we can use the following Von Mises flow role

\[ f = \left( \sigma_p^2 + 3 \tau_p^2 \right)^{\frac{1}{2}} - \sigma_y \]  \hspace{1cm} (6.72)

where \( f \) is the yield function, \( \sigma_p \) the axial stress at a point \( P \) of the cross-section, \( \tau_p \) the shear stress at the point \( P \), and \( \sigma_y \) is the yield strength of the beam material.

*The tangent modular elastic-plastic matrix*

As was mentioned before, we need to determine the stresses at each integration point of the element (span-wise and at the cross-section) given the strains.
Now, to derive the modular matrix we will adapt the method that can be found in Crisfield (1991), Bathe (1996), and Belytschko et al, (2000), which requires obtaining the gradient of the yield function with respect to each of the stresses:

\[
\begin{bmatrix}
\frac{\partial f}{\partial \sigma_p} \\
\frac{\partial f}{\partial \tau_p}
\end{bmatrix} = \begin{bmatrix}
\sigma_p \\
\sqrt{\sigma_p^2 + 3\tau_p^2}
\end{bmatrix}
\]

(6.73)

The relation between plastic stress rates and plastic strain rates for isotropic elastic material (ignoring the dynamic effects of the strain rates) is given by:

\[
\begin{bmatrix}
\delta \sigma_p \\
\delta \tau_p
\end{bmatrix} = \begin{bmatrix}
C_e \\
C_e
\end{bmatrix} \cdot \begin{bmatrix}
\delta \varepsilon_e \\
\delta \gamma_e
\end{bmatrix} = \begin{bmatrix}
\delta \varepsilon_t - \delta \varepsilon_p \\
\delta \gamma_t - \delta \gamma_p
\end{bmatrix}
\]

(6.74)

where

\[
[C_e] = \begin{bmatrix}
E & 0 \\
0 & G
\end{bmatrix}
\]

(6.75)

\(\delta \varepsilon_e\) and \(\delta \gamma_e\) are the variations in the elastic longitudinal and shear strain rates respectively,
\(\delta \varepsilon_t\) and \(\delta \gamma_t\) are the total variations in longitudinal and shear strain rates respectively, and
\(\delta \varepsilon_p\) and \(\delta \gamma_p\) are the variations in the plastic longitudinal and shear strain rates respectively.

The variations in the plastic strain rates are given by:

\[
\begin{bmatrix}
\delta \varepsilon_p \\
\delta \gamma_p
\end{bmatrix} = \delta \lambda_p \begin{bmatrix}
\frac{\partial f}{\partial \sigma} \\
\frac{\partial f}{\partial \tau}
\end{bmatrix}
\]

(6.76)
where $\delta \lambda_p$ is a plastic strain rate multiplier, which we need to determine. Now, when the plastic flow occurs, the rate of change of stress will be tangent to the yield surface and orthogonal to the gradient of the yield function at that point:

$$
\left[ \frac{\partial f}{\partial \sigma} \right] \cdot \left[ \begin{array}{c} \delta \sigma_p \\ \delta \tau_p \end{array} \right] = 0
$$

(6.77)

Hence, we can find the plastic strain rate multiplier $\dot{\lambda}_p$ by solving the following equation:

$$
\left[ \frac{\partial f}{\partial \sigma} \right] \cdot \left[ \begin{array}{c} \delta \varepsilon_t \\ \delta \gamma_t \end{array} \right] - \delta \lambda_p \left[ \begin{array}{c} \frac{\partial f}{\partial \sigma} \\ \frac{\partial f}{\partial \tau} \end{array} \right] = 0
$$

(6.78)

which yields (for elastic-plastic behavior)

$$
\delta \lambda_p = \frac{E \delta \varepsilon_t, \sigma_p + 3G \delta \gamma_t, \tau_p}{\sqrt{\sigma_p^2 + 3\tau_p^2}} \cdot \frac{\sigma_p^2 + 3\tau_p^2}{E \sigma_p^2 + 9G \tau_p^2}
$$

(6.79)

The plastic strain rate multiplier $\delta \lambda_p$ determines the state of stress at each integration point since positive sign means plastic deformation and negative sign means elastic deformation (a value of zero is not likely to occur because of the numerical errors in computations). Substituting Eq. 6.79 for the plastic strains of Eq 6.76 and then in Eq. 6.74, we get the required tangent modular elastic-plastic matrix that relates the change in the stress vector to the change in the strain vector as:

$$
\left[ \begin{array}{c} \delta \sigma_p \\ \delta \tau_p \end{array} \right] = \left[ C_m \right] \left[ \begin{array}{c} \delta \varepsilon_t \\ \delta \gamma_t \end{array} \right]
$$

(6.80)
\[
[C_m] = \frac{1}{E\sigma_p^2 + 9G\tau_p^2} \begin{bmatrix}
9EG\tau_p^3 & -3EG\tau_p\sigma_p \\
-3EG\tau_p\sigma_p & EG\sigma_p^2
\end{bmatrix}
\]

where \([C_m]\) is the tangent modular matrix that relates the rates of stress to the rates of strain in the plastic region.

However, for the case of elastic deformation the stress and strain rates are related through the elastic matrix \([C_e]\) (see Eq. 6.78.)

**Stress Integration and Updating**

So far, we have obtained the elastic-plastic relation between the rates of stresses and the rates of strains at each integration point. Yet, we still need to integrate these rates (assuming the rates to be derivatives in pseudo time) to obtain the values of stresses and strains at each integration point.

One simple way of carrying on the rates integration is to use the forward Euler method at each integration point. However, the forward Euler method will cause an unsafe drift from the yield surface at each integration point, which can be corrected by employing a suitable algorithm to return to the yield surface or an algorithm to scale the step size. Nevertheless, a backward Euler method can be used in predictor/corrector scheme without the need to scale the step size or to use any technique to return to the yield surface (see Crisfield, 1991, Bathe 1996, and Belytschko et al, 2000).

**Stress integration using backward Euler method**

We start by incrementing the strains at every integration point by \(\Delta \varepsilon\),

\[
\tilde{\varepsilon}_{n+1} = \tilde{\varepsilon}_n + \Delta \tilde{\varepsilon}
\]  \hspace{1cm} (6.81)
where, \( n \) denotes the previous converged step, and \( n + 1 \) is the new predictor increment. However, we will increment the stress by an elastic predictor \( \tilde{\sigma}_{n+1}^{\text{trial}} \) and our aim is to find the plastic corrector component of the trial stress \( \Delta \tilde{\sigma}^{(k)} \)

\[
\tilde{\sigma}_{n+1}^{(k)} = \tilde{\sigma}_{n+1}^{\text{trial}} - \Delta \tilde{\sigma}^{(k)}
\]  

(6.82)

where, \( (k) \) denotes the iteration number. However, it is important to note that during the elastic predictor step the plastic strain is unchanged and during the plastic corrector steps the total strain (elastic + plastic) is fixed. Accordingly, the plastic component of the trial stress is given by

\[
\Delta \tilde{\sigma}^{(k)} = [C_e] \cdot \Delta \tilde{\epsilon}^P_{n+1} = [C_e] \cdot (\tilde{\epsilon}^P_{n+1} - \tilde{\epsilon}^P_n) = \Delta \dot{\lambda}_{p_{n+1}} [C_e] \cdot \left( \frac{\partial f}{\partial \tilde{\sigma}_{n+1}} \right)
\]  

(6.83)

where \([C_e] = \begin{bmatrix} E & 0 \\ 0 & G \end{bmatrix}\) is the matrix of elastic axial and shear modulus, \( \tilde{\epsilon}^P_{n+1} \) is the plastic strain, \( \Delta \dot{\lambda}_{p_{n+1}} \) is the plastic strain rate at increment \( n + 1 \), and \( \left( \frac{\partial f}{\partial \tilde{\sigma}_{n+1}} \right) \) is the gradient of the flow rule (Von Mises). Accordingly, it is required to calculate the plastic strain rate at increment \( n + 1 \), and this is performed using Newton’s method (see details at appendix B, which is adapted from Belytschko et al, 2000). Now, we can obtain the stresses according to the following algorithm shown in Figure 6.10
At the end of the last incremental step \((n)\): set
\[
\begin{align*}
    k &= 0 \\
    \Delta \sigma^{(0)} &= 0 \\
    \Delta \lambda p^{(0)} &= 0
\end{align*}
\]

Increment the strains and calculate the trial stresses
\[
\begin{align*}
    \tilde{\varepsilon}_{n+1} &= \tilde{\varepsilon}_n + \Delta \tilde{\varepsilon} \\
    \bar{\sigma}^{(0)}_{(n+1)} &= \bar{C} \cdot (\tilde{\varepsilon}_{n+1} - \bar{\varepsilon}^{(0)}_n)
\end{align*}
\]

Check convergence
\[
\begin{align*}
f(\bar{\sigma}^{(k)}_{(n+1)}) &\leq Tol_1 \\
\text{and} \\
\| \tilde{\varepsilon}^{(p)}_{(n+1)} \| - \| \tilde{\varepsilon}^{(p)}_{(n)} + \Delta \lambda p \left( \frac{\partial f}{\partial \sigma} \right)_{(n+1)} \| &\leq Tol_2
\end{align*}
\]

Calculate
\[
\begin{align*}
\delta \lambda^{(k)} \\
\text{and} \\
\Delta \bar{\sigma}^{(k)}
\end{align*}
\]

Update
\[
\begin{align*}
\tilde{\varepsilon}^{(k+1)} &= \tilde{\varepsilon}^{(k)} - \bar{C}^{-1} \cdot \Delta \bar{\sigma}^{(k)} \\
\Delta \lambda^{(k+1)} &= \Delta \lambda^{(k)} + \delta \lambda^{(k)} \\
\bar{\sigma}^{(k+1)} &= \bar{\sigma}^{(k)} + \Delta \bar{\sigma}^{(k)}
\end{align*}
\]

Set
\[
k + 1 \rightarrow k
\]

Figure 6.10. Flow Chart for the Stress Calculation Algorithm
Plasticity consideration for the stiffness matrix and the internal forces

Having calculated the stresses at the plastic range, we need to determine the tangent stiffness matrix that relates the displacements to the unbalanced forces, and we need to determine the internal forces on each element. Accordingly, we will integrate the stresses over the beam element cross-section and obtain the axial, bending, shear, and bi-moment loads vector \( \{ \Omega \} \) (see Eq. 6.55). Then, we update the initial stress matrix \([K_{\sigma}]\) (see Eq. 6.56 - 6.62), and the internal element forces \( F_{int} \)

\[
\{F_{int}\} = \int [B]^T \cdot \{\Omega\} \, dV
\]  

(6.84)

where \([B]\) is a matrix that relates the element strains to the generalized displacements as defined in Eq. 6.48. Finally, we will substitute the tangent modular matrix \([C_{int}]\) (Eq. 6.80) in the place of the matrix \([D]\) (Eq. 6.32) and update the element stiffness matrix \([K_{r}]\) accordingly. Figure 6.11 shows the flowchart of the procedure.
Increment the displacements (predictor loop)

\[ \{q^{(k)}\} = \{q^{(k-1)}\} + \{\Delta q\} : K \text{ increment} \]

Calculate the incremental strains at each Gauss integration point (2 points span-wise)

\[ [\Delta \Gamma]^{(k)}_{i} = [\Gamma]^{(k)}_{i} - [\Gamma]^{(k-1)}_{i} : i \text{ Gauss point} \]

Obtain the incremental strains at each integration point \((i, j)\) (Eq. 5.33)

\[ [\Delta \epsilon^{(k)}_{ij}] = [\Delta \Gamma(z_{i})] \cdot [G(x_{j}, y_{j})] : j \text{ Lobatto Point} \]

Obtain the updated stresses at each integration point using backward Euler

\[ \left[ \{\sigma^{(k)}_{ij}\} = \{\sigma^{(k-1)}_{ij}\} + \{\Delta \sigma^{(k)}_{ij}\} \right] \]

Obtain the tangent modular matrix at each integration point

\[
[C_{\text{tm}}] = \frac{1}{E \sigma_{p}^{2} + 9G \tau_{p}^{2}} \begin{bmatrix}
9E G \tau_{p}^{2} & -3E G \tau_{p} \sigma_{p} \\
-3E G \tau_{p} \sigma_{p} & E G \sigma_{p}^{2}
\end{bmatrix}
\]

Obtain the stresses at Gauss integration points

\[ \left[ \{\Omega^{(k)}_{ij}\} = \iint [G(x_{j}, y_{j})] \cdot \{\sigma^{(k)}_{ij}\} dA \right] \]

Update the matrices

\[
[K_{s}]^{(k)} = \int_{0}^{1} [B] \cdot \int \{G\} \cdot [C_{\text{tm}}] \cdot [G]^T dA \cdot [B]^T dz
\]

\[
[F_{\text{ex}}]^{(k)} = \int_{0}^{1} [B] \cdot \{\Omega^{(k)}_{ij}\} dz
\]

\[
[K_{s}]^{(k)} = \int_{0}^{1} [Z_{c}] \cdot [S_{c}] \cdot [Z_{c}]^T dz
\]

Solve \( \{\Delta q\} = -[K_{s}]^{-1} \cdot (\{F_{\text{ex}}\} - \{F_{\text{int}}\}) \)

Use the above matrices in the displacement corrector (iterative loop)

Figure 6.11 Flow Chart of the Stress Integration Procedure
6.9 Code Comparisons and Verifications

6.9.1 Comparisons with Published Data

The results of the code were compared and verified against other published works, as in the following four examples.

*Example 1: In-plane plastic bending under uniform lateral load*

A simply supported rectangular beam under lateral load was investigated using the present finite element model (see Figure 6.12), and compared to the example found in Yang and Saigal (1982) (see Ch. 3.) In the analysis, 6 elements were used and the displacement was controlled from the mid-span in plane displacement degree of freedom $q_{10}$ (see Figure 6.7). The results were in excellent agreement with the results presented by Yang and Saigal (see Figure 6.13)

![Figure 6.12 Simply Supported Rectangular Beam under Uniform Lateral Load $q_o$ (v the central in-plane deflection)](image-url)
Example 2: Determining the Critical Flexural-Torsional Buckling Load for a Cantilever Beam with a Tip Load.

The critical flexural-torsional buckling load for a cantilever beam with a tip load (see Figure 6.14) was investigated using the present finite element model, and compared to the example found in Ronagh and Bradford (1999), and Woolcock and Trahair (1974). In this analysis, 6 elements were used and the displacement was controlled from the beam tip in-plane displacement degree of freedom $q_{10}$ (see Figure 6.7.) Also, the critical load was obtained by determining the load fraction at which the stiffness matrix becomes singular. The results were in good agreement with the results presented (the predicted critical load was 1.92% less than the critical load obtained by Woolcock and Trahair (1974), and 3.38% less than the critical load predicted by Ronagh and Bradford (1999)).
Example 3: Determining the non-Linear Torsional response of a Cantilever Beam with a Tip Torque Load.

The non-linear response of a cantilever beam with a tip torque load (see Figure 6.15.a) was investigated using the present finite element model, and compared to the example found in Ronagh and Bradford (1999), which was experimentally tested by Tso and Ghobarah (1971). In this analysis, 6 elements were used and the displacement was controlled using the beam tip axial force.

\[ E = 9300 \text{ kip/ in}^2 \]
\[ \nu = 0.25 \]

Figure 6.14 The Cantilever Beam Tested by Woolcock and Trahair (1974) and analyzed by Ronagh and Bradford (1999).

\[ P_{Cr} = 7.28 \text{ lb} \] (Woolcock and Trahair, 1974)
\[ P_{Cr} = 7.39 \text{ lb} \] (Ronagh and Bradford, 1999)
\[ P_{Cr} = 7.14 \text{ lb} \] (Present Code)
rotational degree of freedom $q_{11}$ (see Figure 6.7.) The results were in excellent agreement with the results predicted by Ronagh and Bradford (1999) as in Figure 6.15 b.

$$T = \gamma \frac{EI_\omega}{L^3}$$

a) I-Beam Subject to End Torque

$E = 57.3664$ GPa
$\nu = 0.375$

(b) Non-Linear Torque-Twist Response (the results of Ronagh and Bradford finite element are overlapped by the current code)

Figure 6.15 The Response of Cantilever Beam Tested by Tso and Ghobarah (1971) and analyzed by Ronagh and Bradford (1999) (six elements were used, and step control from the tip axial rotation DOF $q_{11}$)
Example 4: Determining the Response of a Beam under Axial, Bi-axial Bending and Torsional loads

The non-linear response of a simply supported beam under axial, bi-axial bending and torque loads (see Figure 6.16) was investigated using the present finite element model, and compared to the example found in El-Khenfas and Nethercot (1989). In this analysis, 12 elements were used and the displacement was controlled from the mid-span out-of-plane displacement degree of freedom $q_9$ (see Figure 6.7). The results were in excellent agreement with the results predicted by El-Khenfas and Nethercot (1989) (see Figure 6.17).
\[ \sigma_y = 28 \text{ksi} \]
\[ E = 20 \times 10^6 \text{ psi} \]
\[ G = 11.2 \times 10^6 \text{ psi} \]

Figure 6.16 A Simply Supported Beam under Bi-axial Bending and Torque Loads
6.9.2 Comparisons with ABAQUS Beam Element

Example: Non-linear Elastic-Plastic Cantilever Beam with T-section Section under Axial, Lateral and Torsional Loads

A non-linear elastic-plastic cantilever beam under axial, lateral and torsional loads (see Figure 6.18) was analyzed using ABAQUS 14 DOF three nodes quadratic open section shear flexible beam element B32OS (for more information about this element refer to ABAQUS Standard User’s Manual 2002), and the results were compared to the present code (see Figure 6.19).
Figure 6.18 Cantilever Beam with T Section under Axial, Lateral and Torsional Loads

Lateral Load Eccentricity = 8 mm

\[ E = 70.0 \text{ GPa} \]
\[ \nu = 0.3 \]
\[ \sigma_y = 400 \text{ MPa} \]
This study vs Abaqus Beam Element in terms of lateral and out-of-plane deformations.
It can be seen from the figures that the ABAQUS solution is more flexible when compared to the code and this is may be due to the difference between the Euler-Bernoulli beam model that is used in this formulation and the shear flexible Timoshenko beam model of the B32OS ABAQUS element. Also, this difference may be due to the fact that the ABAQUS beam models have no through thickness integration points and considers a constant torsional rigidity of the cross-section (ABAQUS Standard User’s Manual 2002). Thus, the ABAQUS beam model does not consider material non-linearity under pure torsional loads as shown in Figure 6.20.

Figure 6.19 Comparison between the Results of the Developed Beam Element and ABAQUS B32OS Beam Element, for Non-linear Elastic-Plastic Solution, for the Case of Cantilever Beam under Combined Axial, Flexural and Torsional Loads (36 elements were used, and the displacement was controlled mid-span in plane displacement degree of freedom $q_{10}$.)
Figure 6.20 Comparison between the Results of the Developed Beam Element and ABAQUS B32OS Beam Element, for Non-linear Elastic-Plastic Solution for the Case of Cantilever Beam under Pure Torque in the Tip of the Beam (36 elements were used and the displacement was controlled from the tip axial rotation degree of freedom $q_{11}$

![Graph comparing results](image-url)
Chapter 7

Reliability Based Optimization of T-Beam under Random Axial, Flexural and Torsional Loads with Torsional and Local Buckling

Failure Modes

This is the same beam example that was considered in Chapter 4, however, this time torsional and local buckling effects will be considered, also a more economic RBDO method will be used.

7.1 Failure Modes

Even though, more degrees of freedom were added to the beam model, but still there is no guarantee that the actual beam will behave according to the predictions of the more advanced model. For example, the 14 degrees-of-freedom beam model can not predict local buckling modes and localized failures that may occur. Therefore, higher order elements (such as a shell or a continuum element) must be used to explore the existence of local failures. For this purpose, examples of cantilever beams under simple in-plane tip load are analyzed using shell and continuum elements and presented in the following sub-section.

7.1.1 Examples

Example 1: Continuum Element Analysis of Non-linear Elastic-Plastic Cantilever Beam with Narrow Rectangular Section under in-plane Tip Load

To get familiar with the continuum (3-D) element analysis and to further test the capabilities of the FE code developed here, a non-linear elastic-plastic cantilever beam with narrow rectangular
section under tip load (see Figure 7.1) was analyzed using the ABAQUS continuum 20 nodes quadratic continuum hexahedral (brick) element C3D20R. First, a linear buckling analysis was performed for the ABAQUS continuum model to determine the buckling mode shapes. In the analysis the tip load (1.0 lb) was distributed over the four corners of the rectangular cross-section and a small out-of-plane load (0.001 lb) was added to trigger the buckling instability. The linear buckling analysis revealed no local buckling modes for the first three buckling modes (see Figure 7.2 (a) and (b)). Next, a nonlinear elastic-plastic analysis was performed for the continuum model to investigate the collapse of the beam. During the analysis, localized effects (stress hot spots) have occurred at the corners of the fixed end of the beam (see Figure 7.3.) However, since these hot spots were restricted to the fixed end of the beam, they had only a limited influence on the equilibrium path. Accordingly, there was a good agreement between the analysis results of the beam model and the continuum model (see Figure 7.4) and the experimental results found in Woolcock and Trahair (1974)

Figure 7.1 Cantilever Beam with Narrow Rectangular Section under Tip load

\[
\begin{align*}
E &= 9300 \text{ kip/in}^2 \\
\nu &= 0.3 \\
\sigma_y &= 23 \text{ kip/in}^2 \\
P &= 1 \text{ lb} \\
80.0'' \\
2.0'' \\
0.124''
\end{align*}
\]
Figure 7.2 Buckling Mode Shapes of a Narrow Rectangular Cantilever Beam under Tip Load (the second mode shape is the same as the first, but with an opposite sign). ABAQUS quadratic continuum hexahedral (brick) element C3D20R
Figure 7.3 The Deformed Shape of a Non-linear Elastic-Plastic Cantilever Beam under Tip Load, ABAQUS quadratic continuum hexahedral (brick) element C3D20R.
Example 2: Continuum Element Analysis of a Non-linear Elastic-Plastic Cantilever Beam having T-Section and under in-plane Tip Load

ABAQUS continuum element model was used to investigate the failure modes of a cantilever beam having T-section and subject to an in-plane tip load (see Figure 7.5.) The aim is to determine what types of failure modes are present in such a situation. A linear elastic buckling analysis was carried out first to determine the critical buckling loads and the corresponding mode shapes (as
shown in Figs. 7.6 (a) and (b)). The buckling analysis revealed that local buckling modes are present.

Figure 7.5 Cantilever Beam with T Section under Tip load

\[ E = 70.0 \text{ GPa} \]
\[ \nu = 0.3 \]
\[ \sigma_y = 400 \text{ MPa} \]
Figure 7.6 Buckling Mode Shapes of a Cantilever Beam with T-Section under in-plane Tip Load (see Figure 6.5)
Then, a nonlinear elastic analysis was performed and localized stress concentration spots started developing (see Figure 7.7). These localized stresses have developed a localized buckling deformation (as shown in Figure 7.8). Similar results were obtained for nonlinear elastic-plastic analysis, but with a slight localized deformation in addition to the local buckling. Theses localized deformations were more pronounced as the web depth is increased (as shown in Figure 7.9).

Figure 7.7 Nonlinear Elastic Analysis of a Continuum Element Model for a Cantilever Beam under Vertical Tip Load, Revealing Local Stress Concentration Spots
Figure 7.8 Stress Contours of a Cantilever T-beam under In-plane Tip Load (Nonlinear Elastic Analysis)
Figure 7.9 Stress Contours for a Nonlinear Elastic-plastic Analysis of a Cantilever T-beam under In-plane Tip Load (localized deformations have developed)

Also, the same beam (cantilever beam with vertical tip load) was modeled and analyzed by a shell model and the beam experienced local web buckling, which is shown in Figure 7.10.
Figure 7.10 Local Deformation of the Web of a Cantilever T-section Beam under Vertical Tip Load (see Figure 6.5)

Thus, one may conclude that local buckling and localized effects can occur for a cantilever elastic-plastic beam under rather simple loading conditions (such as the vertical tip load), and therefore, we must constraint the loads from exceeding the values of the local buckling loads. In the next section local buckling constraints will be provided to constraint any local buckling behavior of the beam.
7.2 Local Buckling Consideration

As was demonstrated in the previous section, a beam with a T-section is susceptible to local buckling failures that may have different equilibrium paths than the global buckling or the plastic hinge failures. The main contributors to the local buckling failures of the web and flange plates are the axial, shear and bending loads as shown in Figure 7.11 and Figure 7.12. The pure torsional load may not induce local buckling as can be seen from Figure 7.13, which shows an ABAQUS shell model of a cantilever T-beam under pure torque applied at its free end. Figure 7.13 shows that the beam has developed a significant plastic zone without producing any local buckling due to torque.

Figure 7.11 Local Buckling of the Web and Flange of a Cantilever Beam of T-section under Axial Load (ABAQUS continuum element model linear buckling analysis)
Figure 7.12 Local Buckling of the Web and Flange of a Cantilever Beam of T-section under Uniform Lateral Pressure Load Applied at its Flange (ABAQUS continuum element model linear buckling analysis)
Accordingly, to prevent the beam web and flange from deforming in local buckling modes, local buckling constraints are specified. Despite the fact that local buckling behavior of columns can be found in many structural stability textbooks (see for example Gerard, (1962); Bruhn (1973)) local buckling of beams under combined loads is treated as buckling of plates, that uses either plate theory solutions (and empirical formulas for some of the boundary conditions) such as Hughes (1988); Megson (1999) and Paik and Thayamballi (2003), modeled using semi-analytical models (see Byklum and Amdahl (2002)) or finite strip methods (Stewart and Sivakumaran (1997); Azhari et al, (2004)) or shell elements (Shidharan et al, 1994). Given the computational expense of the other

Figure 7.13 Cantilever Beam of T-section under Torque Load at its Free End (ABAQUS shell model)
methods, using the formulae derived from the plate theory to be the local buckling constraints would be sufficient.

**Web Plate Local Buckling**

If the beam web plate was modeled as fixed from the flange side and free at the other edges, and for axial, shear and bending loads, we can get use the load interaction formula found in Hughes (1988), which is given below

\[
\frac{0.625(1 + 0.6/\alpha)R_c}{[1 - R_h^3]} + R_s^2 = 1
\]

Where:

\[
R_c = \frac{\sigma_a}{(\sigma_a)_{cr}}; (\sigma_a)_{cr} = 3.25 \left(\frac{\pi}{h}\right)^2 \frac{D}{t_w}
\]

\[
R_h = \frac{\sigma_h}{(\sigma_h)_{cr}}; (\sigma_h)_{cr} = 41.8 \left(\frac{\pi}{h}\right)^2 \frac{D}{t_w}
\]

\[
R_s = \frac{\tau}{\tau_{cr}}; \tau_{cr} = \left[5.35 + 4 \left(\frac{L}{h}\right)^2\right] \left(\frac{\pi}{h}\right)^2 \frac{D}{t_w}
\]

\[
\alpha = \frac{L}{h}; D = \frac{E t_w^3}{12(1 - \nu^2)}
\]

Where \(L\) is the beam length, \(h\) is the web depth, \(E\) is the modulus of elasticity, \(t_w\) is the web thickness and \(\nu\) is the Poisson’s ratio.

One may notice that the length of the beam \(L\) is much larger than the web depth \(h\), and hence, the critical shear stress would be larger than the critical axial load by about two orders of magnitude. In
addition, for a long beam the shear stress is usually much smaller than the bending stresses. Thus, the shear effects on local buckling of the web plate may be ignored. Similarly, from Eq. 7.1 the critical bending load coefficient is about ten times larger than that of the critical axial load, and since we are calculating the stresses at any point of the cross-section the axial and bending stresses will be added together at each point. Accordingly, the most important critical load would be the axial load.

**Web Tripping**

Another important local failure mode is the beam web tripping under axial or flexural loads, where the web rotates locally and distorts the flange through the attachment. In fact, this may be the type of failure that is shown in 7.11 and Figure 7.12. The following formula given in Paik and Thayamballi (2003), gives the critical axial stress after which the web may fail by tripping:

\[ \left( \sigma_{a,T} \right)_{cr} = \frac{G J}{I_p} - \frac{q L^2}{12 I_p} \left( \frac{S}{4\pi^2} \right) \]

Where

\[ J = \frac{h t_w^3}{3} \]

\[ I_p = \frac{h^3 t_w}{3} + \frac{h t_w^3}{12} \]

\[ I = \frac{b t_f^3}{12} + b t_f (h + \frac{t_f}{2} - \bar{Y})^2 + \frac{h^3 t_w}{12} + h t_w (\bar{Y} - \frac{h}{2})^2 \]

\[ S = h^3 t_w \left( \frac{h}{4} - \frac{1}{3} \bar{Y} \right) \]

Where \( \bar{Y} \) is the centroid of the cross-section, \( h \) is the web depth, \( t_w \) is the web thickness and \( t_f \) is the flange thickness.
**Flange Local Buckling**

In the case of a mild compression load, which is will below the buckling load and the ultimate axial load, the axial load only magnifies the deformation that is caused by the lateral load. However, when the axial load becomes significant, the lateral load effects become secondary in local buckling and the critical load is based on the axial load alone. In this case the flange plate will be treated as two plates that are simply supported from three edges while the fourth is treated as rigid because of symmetry. Thus, we would have the following relation Hughes (1988)

\[
R_c = \frac{\sigma_a}{(\sigma_a)_{cr}} ; \quad (\sigma_a)_{cr} = 6.25 \left( \frac{\pi^2}{b} \right)^2 D \frac{t_f}{D} \quad (7.3)
\]

\[
D = \frac{E t_f^3}{12(1-\nu^2)}
\]

In all the local buckling calculations, the critical local buckling stress may have a much higher value than the yield strength of the material, which will give unrealistic critical values. For moderately high values of the critical buckling stress (for \(\sigma_{cr} \leq 4 \sigma_y\)), an empirical scaling factor that is given by the Johnson-Ostenfeld formula (see Paik and Thayamballi (2003))

\[
\sigma_f = \begin{cases} 
\sigma_{cr} & \text{for } \sigma_{cr} \leq 0.5 \sigma_y \\
\sigma_{cr} \left(1 - \frac{\sigma_{cr}}{4\sigma_y}\right) & \text{for } \sigma_{cr} > 0.5 \sigma_y 
\end{cases} \quad (7.4)
\]

However, in the case that \(\sigma_{cr} > 4 \sigma_y\), another empirical formula may be used (see Hughes, 2001)
\[ \sigma_f = \eta \sigma_{cr} \]

Where \( \eta = \frac{1}{\Delta} - \frac{1}{4\Delta^2} \)

\( \Delta = \frac{\sigma_{cr}}{\sigma_y} \) \hfill (7.5)

### 7.3 Specifying the Constraints

In the optimization of the beam, the following constraints are specified:

- The maximum load that the beam can withstand

\[ G_\lambda(\bar{d}, \bar{X}) = \lambda - 1.0 \] \hfill (7.6)

Where \( G_i(\bar{d}, \bar{X}) \) is the performance function, which depends on the design variables \( \bar{d} \) and the mean values of the random variables \( \bar{X} \), and \( \lambda \) is the amount of the applied load that the beam can withstand before failure.

- The mid-span in-plane deformation must not exceed 50.0 mm (design consideration)

\[ G_{\delta_y}(\bar{d}, \bar{X}) = \left( \frac{0.05}{\delta_y^*} \right)^2 - 1.0 \] \hfill (7.7)

Where \( \delta_y^* \) is the mid-span in-plane deformation at the failure load (for \( \lambda < 1.0 \)), or at the nominal load (for \( \lambda \geq 1.0 \))

- The mid-span axial rotation must not exceed 0.1 rad (design consideration)

\[ G_{\phi_z}(\bar{d}, \bar{X}) = \left( \frac{0.1}{\phi_z^*} \right)^2 - 1.0 \] \hfill (7.8)

Where \( \phi_z^* \) is the mid-span axial rotation at the failure load (for \( \lambda < 1.0 \)), or at the nominal load (for \( \lambda \geq 1.0 \)).
The mid-span out-of-plane deformation must not exceed 5.0 mm (design consideration)

\[
G_{\delta} \left( \delta, \bar{X} \right) = \left( \frac{0.005}{\delta_x} \right)^2 - 1.0 \tag{7.9}
\]

Where \( \delta_x \) is the mid-span out-of-plane deformation at the failure load (for \( \lambda < 1.0 \)), or at the nominal load (for \( \lambda \geq 1.0 \)).

- The web must not fail by local compressive buckling when the beam is loaded with the full nominal load (i.e. \( \lambda = 1.0 \)).

\[
G_{wc} \left( \delta, \bar{X} \right) = \lambda - \frac{\sigma_x}{\sigma_{wc}} \tag{7.10}
\]

Where \( \sigma_x \) is the maximum stress at the beam at the time of failure, and \( \sigma_{wc} \) is the web local compressive buckling critical stress.

- The web must not fail by tripping when the beam is loaded with the full nominal load.

\[
G_{wt} \left( \delta, \bar{X} \right) = \lambda - \frac{\sigma_x}{\sigma_{wt}} \tag{7.11}
\]

Where \( \sigma_{wt} \) is the web tripping critical stress.

- The flange must not fail by local compressive buckling when the beam is loaded with the full nominal load.

\[
G_{fc} \left( \delta, \bar{X} \right) = \lambda - \frac{\sigma_x}{\sigma_{fc}} \tag{7.12}
\]

Where \( \sigma_{fc} \) is the flange local compressive buckling critical stress.
However, it is important to note that the constraints $G_\lambda$, $G_{wc}$, $G_{wt}$ and $G_{fc}$ are related, since they only differ by a constant term. This is because the local buckling constraints are composed of the maximum load fraction of the applied load $\lambda$, and the ratio of the maximum stress at the beam at the time of failure $\sigma_\lambda$ to the critical local buckling stresses $\sigma_{cr}^{wc}$, $\sigma_{cr}^{wt}$ and $\sigma_{cr}^{fc}$. The critical local buckling stresses are constants for a given beam geometry and applied loads. Thus, we may only consider the constraint that is the most critical among these inter-related constraints and ignore the others. Also, for the beam under consideration, the most critical constraint was found to be the web tripping failure $G_{wt}(\tilde{d}, \tilde{\mu})$ for all the loads and beam dimension considered for optimization (as presented next). Thus, we are going to consider only the following constraints

$$G_1 = G_{\delta_y}(\tilde{d}, \tilde{X}) = \left(\frac{0.05}{\delta_y}\right)^2 - 1.0$$

$$G_2 = G_{\phi_y}(\tilde{d}, \tilde{X}) = \left(\frac{0.1}{\phi_\delta}\right)^2 - 1.0$$

(7.13)

$$G_3 = G_{\delta_x}(\tilde{d}, \tilde{X}) = \left(\frac{0.005}{\delta_\sigma}\right)^2 - 1.0$$

$$G_4 = G_{\sigma_{cr}^{wt}}(\tilde{d}, \tilde{X}) = \lambda - \frac{\sigma_{cr}^{wt}}{\sigma_{cr}}$$

### 7.4 Deterministic Optimization for Safety Factors 1.50, 1.75, 2.00, 2.25 and 2.50

To compare between the optimum designs provided by the deterministic and the reliability-based optimization, and to provide a feasible starting point for the reliability-based optimization, a number
of deterministic optimum designs with different safety factors would be obtained. The intent here is to compare the deterministic optimum designs' weight and safety level to those of reliability-based optimum designs.

The Deterministic Optimization Problem Formulation

\[
\min : W(\tilde{d}) \\
\text{s.t.} : \quad G_i(\tilde{d}, \bar{\mu}_X \times SF) \geq 0.0 \quad (7.14)
\]

where \( W(\tilde{d}) \) is the weight of the beam, \( \tilde{d} \) is the vector of design variables (height of the web, the width of the flange, and their thicknesses), \( G_i(\tilde{d}, \bar{\mu}_X \times SF) \) are the constraints that were specified earlier and \( \bar{\mu}_X \times SF \) are the nominal (mean) values of the random loads multiplied by the safety factor SF.

For the optimization process, we have used the Modified Method of Feasible Directions and the Sequential Quadratic Programming optimization algorithms of Visual Doc software (Vanderplaats, 1999; and Ghosh et al, 2000). The five deterministic optimum beam designs (SF 1.50, 1.75, 2.00, 2.25 and 2.50) are presented in Table 7.1, and in all the optimum design the constraint \( G_i \) was the most critical

<table>
<thead>
<tr>
<th>Factor of Safety</th>
<th>Mass (kg)</th>
<th>h (mm)</th>
<th>( t_w ) (mm)</th>
<th>b (mm)</th>
<th>( t_f ) (mm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.50</td>
<td>20.47</td>
<td>125.88</td>
<td>19.74</td>
<td>184.07</td>
<td>7.41</td>
</tr>
<tr>
<td>1.75</td>
<td>22.87</td>
<td>129.37</td>
<td>21.73</td>
<td>188.83</td>
<td>7.88</td>
</tr>
<tr>
<td>2.00</td>
<td>25.21</td>
<td>130.55</td>
<td>23.75</td>
<td>190.035</td>
<td>8.62</td>
</tr>
<tr>
<td>2.25</td>
<td>27.55</td>
<td>137.56</td>
<td>25.71</td>
<td>189.96</td>
<td>8.85</td>
</tr>
<tr>
<td>2.50</td>
<td>29.83</td>
<td>133.44</td>
<td>27.53</td>
<td>189.42</td>
<td>10.21</td>
</tr>
</tbody>
</table>
7.5 Reliability Calculation

The average runtime for the FE code on a 2.0 GHz Pentium 4 machine is about 3 minutes per analysis. As a result, it will be very expensive to calculate reliability using any of the Mote-Carlo methods (see Chapter 2.) Hence, the analytical reliability calculation methods should be used. For this purpose, a First Order Reliability Method was used, and since the random variables are assumed to be normally distributed, the Hasofer-Lind method was directly used (see Chapter 2.)

Reliability of the T-beam with Multiple Failure Modes

Since the beam can fail in 4 different failure modes, the reliability of the beam must consider the effect of each of them. Yet, some simplifying assumptions may help reduce the reliability calculation. These assumptions are:

- A first-order series bound was assumed (see Ch. 2)
- Any two failure modes that are not conditionally dependant on each other (the occurrence of one failure mode depends on the occurrence of another, and vice versa) will be considered statistically independent. Hence, probability of failure will be given by (equation 2.16 is repeated here)

\[
\sum_{i=1}^{m} P(F_i) \leq P(F) \leq 1 - \prod_{i=1}^{m} [1 - P(F_i)]
\]  

(7.15)

7.5.1 Reliability of Deterministic Optimum Design with Safety Factor 1.50

The reliability of the deterministic optimum design was 0.9608 with \( G_1 \) and \( G_4 \) being the most critical failure modes. It was noted that \( G_1 \) has contributed 79.1% to the probability of failure and
\( G_4 \) has contributed 20.9 %. While the other failure modes \(( G_2 \text{ and } G_3 \) \) have contributed with much less (orders of magnitude are less those of the others) to the probability of failure (see Table 7.2). Also, in Table 7.2 the MPP load for each failure mode is presented as a multiple of the nominal loads.

Table 7.2 the MPP loads for each failure mode presented as a multiple of the nominal loads

<table>
<thead>
<tr>
<th>Applied Load</th>
<th>SF 1.5</th>
<th>( MPP_{G1} )</th>
<th>( MPP_{G2} )</th>
<th>( MPP_{G3} )</th>
<th>( MPP_{G4} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Axial Load (kN)</td>
<td>225×1.5</td>
<td>225×1.194</td>
<td>225×1.415</td>
<td>225×1.26</td>
<td>225×1.24</td>
</tr>
<tr>
<td>Lateral Load (kN/m)</td>
<td>38×1.5</td>
<td>38×1.594</td>
<td>38×2.05</td>
<td>38×2.2</td>
<td>38×1.75</td>
</tr>
<tr>
<td>Eccentricity (mm)</td>
<td>4×1.5</td>
<td>4×1.0</td>
<td>4×1.23</td>
<td>4×1.03</td>
<td>4×1.05</td>
</tr>
</tbody>
</table>

* This value is not an actual failure probability for the design, and it is calculated assuming that the safety factor can provide a MPP for the design that can be compared to the MPP of the individual failure modes.

These results show that the deterministic optimization may have produced an inefficient distribution of the material since the critical failure modes have unequal load multipliers compared to the uniform factor of 1.50 that is assumed by the safety factor. Also, these results show that it may be possible to obtain a better distribution of failure risks by redistributing the material in way that either saves weight for the same level of safety or improves the safety of the design by improving the critical failure modes at the expense of the less critical modes. This is shown next.

7.6 RBDO Applied to the Deterministic Optimum Designs with SF 1.50, 1.75, 2.00, 2.25 and 2.50

The RBDO will be performed for the two optimization problems

1. Design for maximum reliability:
\[
\text{find } \tilde{d} \\
\text{max : } R(\tilde{d}, \tilde{X}) \quad (7.16) \\
\text{s.t. : } W(\tilde{d}) \leq W_d^* 
\]

where \( R(\tilde{d}, \tilde{X}) \) is the system reliability of the beam, which is the probability of survival of the beam under the applied random loads \( \tilde{X} \) (calculated against the previously determined failure modes \( G_i(\tilde{d}, \tilde{X}) \)), \( W(\tilde{d}) \) is the weight of the beam, and \( W_d^* \) is the weight of the corresponding deterministic optimum design.

2. Design for minimum weight

\[
\text{find } \quad \tilde{d} \\
\text{min : } W(\tilde{d}) \quad (7.17) \\
\text{s.t. : } R(\tilde{d}, \tilde{X}) \geq R^*_d 
\]

where \( R^*_d \) is the reliability of the corresponding deterministic optimum design.

To perform RBDO for these designs, it would be computationally expensive to use nested optimization loops since one analysis run takes about 3 minutes. Hence, the SORFS RBDO methods will be used instead (see Ch. 2). However, since from table 7.2 we have the constraint \( G_i \) critical and \( G_4 \) close to being critical, we may need to modify the SORFS RBDO method to accommodate more than one critical constraint. In particular, we will adjust the constraint shift \( s \) to reflect the deficiency in reliability for the case that we have multiple critical failure modes. Therefore we will obtain the MPP loads for each failure mode and use them as probabilistic safety factors (PSF) after adjusting these loads such that they reflect the contribution of each critical failure mode to the probability of failure. In other words, we will use the MPP loads that relate to the most critical
failure mode (i.e. $G_i$) as the basic PSF and the loads that relate to the less critical failure modes (i.e. $G_d$) are added upon it accordingly. Then, the designs will be optimized using these updated PSF loads and their respective reliabilities will be checked. Yet, since there are no target reliability values, we will use scaling of the design variables that we have obtained for the minimum weight RBDO optimum design, and we can use more iterations to improve the designs as presented in the following.

7.6.1 RBDO for the Deterministic Optimum Design SF 1.50

For this design, using the SORFS RBDO method to obtain an optimum RBDO design with less weight, the first semi-probabilistic iteration (see Eq. 2.29) gave a design that was less safe than the original deterministic optimum ($P_f = 4.25 \times 10^{-2}$). A second iteration gave a design with a reliability that was slightly better than the deterministic optimum design ($P_f = 3.43 \times 10^{-2}$), and with 1.7% weight saving. Finally, the reliability-based optimum design with improved weight (see Eq. 2.30) was found to have a better reliability than the deterministic optimum design ($P_f = 1.72 \times 10^{-2}$).

However, no further significant improvement in the reliability was found, and hence, the reliability optimization has converged. Table 7.3 summarizes all the results for SF 1.50, and Figure 7.14 compares the respective probability of failure for each failure mode between the improved reliability optimum design and the deterministic optimum design in the safety index ($\beta$) space (see Ch 2.)
### Table 7.3 Results of the RBDO for the Deterministic Design with SF 1.50

<table>
<thead>
<tr>
<th>Failure Mode $P_f$</th>
<th>Deterministic SF 1.50</th>
<th>RB (reduced weight)</th>
<th>RB (improved Reliability)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$G_1$</td>
<td>3.10E-02</td>
<td>1.56E-02</td>
<td>7.42E-03</td>
</tr>
<tr>
<td>$G_2$</td>
<td>1.83E-05</td>
<td>2.59E-03</td>
<td>4.49E-04</td>
</tr>
<tr>
<td>$G_3$</td>
<td>1.09E-06</td>
<td>2.07E-05</td>
<td>9.40E-06</td>
</tr>
<tr>
<td>$G_4$</td>
<td>8.18E-03</td>
<td>1.61E-02</td>
<td>9.37E-03</td>
</tr>
<tr>
<td>$\Sigma P_f$</td>
<td>3.92E-02</td>
<td>3.43E-02</td>
<td>1.72E-02</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Design Variables</th>
<th>Deterministic SF 1.50</th>
<th>RB (reduced weight)</th>
<th>RB (improved Reliability)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$h$ (mm)</td>
<td>0.1258798</td>
<td>0.1257877</td>
<td>0.13102608</td>
</tr>
<tr>
<td>$t_w$ (mm)</td>
<td>0.01973912</td>
<td>0.0197247</td>
<td>0.01816369</td>
</tr>
<tr>
<td>$b$ (mm)</td>
<td>0.1840704</td>
<td>0.1839357</td>
<td>0.18329097</td>
</tr>
<tr>
<td>$t_f$ (mm)</td>
<td>0.00740926</td>
<td>0.0074038</td>
<td>0.00801275</td>
</tr>
<tr>
<td>Weight (kg)</td>
<td>20.474</td>
<td>20.1255</td>
<td>20.474</td>
</tr>
</tbody>
</table>
Figure 7.14 Comparison between the Probabilities of Failure for Each Failure Mode for the Improved Reliability Optimum Design and the Deterministic Optimum Design SF 1.50 in the Safety Index ($\beta$) Space.

Boundary Showing the Deterministic Safety Level SF = 1.50 On all Loads
7.6.2 RBDO for Deterministic Optimum Design SF 1.75, 2.00, 2.25 and 2.50

For these deterministic designs, the SORFS RBDO method was also used to perform the reliability-based optimization, and the optimization processes had similar trends to the deterministic optimum design with SF 1.50 considered earlier. Tables 7.4 - 7.7 summarize these results and Figure 7.15-7.18 compare the probabilities of failure for each failure mode between the improved reliability optimum designs and the deterministic optimum designs in the safety index ($\beta$) space. Finally, Figure 7.19 and Figure 7.20 present the improvements over the deterministic optimum designs that RBDO was able to achieve by either reducing the weight for the same level of safety (Figure 7.19) or improving the reliability for the same weight (Figure 7.20).

<table>
<thead>
<tr>
<th>Failure Mode $P_f$</th>
<th>Deterministic SF 1.75</th>
<th>RB (reduced weight)</th>
<th>RB (improved Reliability)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$G_1$</td>
<td>2.17E-03</td>
<td>1.75E-04</td>
<td>8.26E-05</td>
</tr>
<tr>
<td>$G_2$</td>
<td>2.32E-10</td>
<td>4.06E-06</td>
<td>2.06E-06</td>
</tr>
<tr>
<td>$G_3$</td>
<td>2.58E-12</td>
<td>6.49E-08</td>
<td>3.22E-08</td>
</tr>
<tr>
<td>$G_4$</td>
<td>1.24E-04</td>
<td>3.09E-04</td>
<td>1.68E-04</td>
</tr>
<tr>
<td>$\Sigma P_f$</td>
<td>2.30E-03</td>
<td>4.88E-04</td>
<td>2.52E-04</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Design Variables</th>
<th>Deterministic SF 1.75</th>
<th>RB (reduced weight)</th>
<th>RB (improved Reliability)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$h$ (mm)</td>
<td>0.1293719</td>
<td>0.1370959</td>
<td>0.1385811</td>
</tr>
<tr>
<td>$t_w$ (mm)</td>
<td>0.02172527</td>
<td>0.0199865</td>
<td>0.02016805</td>
</tr>
<tr>
<td>$b$ (mm)</td>
<td>0.1888285</td>
<td>0.1927857</td>
<td>0.1931157</td>
</tr>
<tr>
<td>$t_f$ (mm)</td>
<td>0.0078836</td>
<td>0.0077119</td>
<td>0.00779022</td>
</tr>
<tr>
<td>Weight (kg)</td>
<td>22.872</td>
<td>22.48663547</td>
<td>22.872</td>
</tr>
</tbody>
</table>
Figure 7.15 Comparison between the Probabilities of Failure for Each Failure Mode for the Improved Reliability Optimum Design and the Deterministic Optimum Design SF 1.75 in the Safety Index ($\beta$) Space.
### Table 7.5 Results of the RBDO for the Deterministic Design with SF 2.00

<table>
<thead>
<tr>
<th>Failure Mode</th>
<th>Deterministic SF 2.00</th>
<th>RB (reduced weight)</th>
<th>RB (improved Reliability)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$G_1$</td>
<td>1.23E-04</td>
<td>3.17E-05</td>
<td>5.99E-08</td>
</tr>
<tr>
<td>$G_2$</td>
<td>4.20E-15</td>
<td>2.46E-13</td>
<td>1.67E-09</td>
</tr>
<tr>
<td>$G_3$</td>
<td>5.61E-15</td>
<td>8.20E-10</td>
<td>1.35E-13</td>
</tr>
<tr>
<td>$G_4$</td>
<td>8.02E-07</td>
<td>2.04E-05</td>
<td>2.38E-07</td>
</tr>
<tr>
<td>$\Sigma P_f$</td>
<td>1.24E-04</td>
<td>5.21E-05</td>
<td>3.00E-07</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Design Variables</th>
<th>Deterministic SF 2.00</th>
<th>RB (reduced weight)</th>
<th>RB (improved Reliability)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$h$ (mm)</td>
<td>0.130551</td>
<td>0.137752</td>
<td>0.1426206</td>
</tr>
<tr>
<td>$t_w$ (mm)</td>
<td>0.02375239</td>
<td>0.0233886</td>
<td>0.0200452</td>
</tr>
<tr>
<td>$b$ (mm)</td>
<td>0.1900353</td>
<td>0.1980239</td>
<td>0.1917526</td>
</tr>
<tr>
<td>$t_f$ (mm)</td>
<td>0.00861739</td>
<td>0.0065148</td>
<td>0.0098006</td>
</tr>
<tr>
<td>Weight (kg)</td>
<td>25.209</td>
<td>24</td>
<td>25.209</td>
</tr>
</tbody>
</table>
Figure 7.16 Comparison between the Probabilities of Failure for Each Failure Mode for the Improved Reliability Optimum Design and the Deterministic Optimum Design SF 2.00 in the Safety Index ($\beta$) Space.
### Table 7.6 Results of the RBDO for Deterministic Design with SF 2.25

<table>
<thead>
<tr>
<th>Failure Mode</th>
<th>Deterministic SF 2.25</th>
<th>RB (reduced weight)</th>
<th>RB (improved Reliability)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$G_1$</td>
<td>2.31E-06</td>
<td>1.13E-06</td>
<td>3.22E-11</td>
</tr>
<tr>
<td>$G_2$</td>
<td>1.93E-21</td>
<td>1.90E-11</td>
<td>2.37E-13</td>
</tr>
<tr>
<td>$G_3$</td>
<td>7.34E-23</td>
<td>1.64E-11</td>
<td>1.76E-18</td>
</tr>
<tr>
<td>$G_4$</td>
<td>5.24E-10</td>
<td>1.71E-06</td>
<td>2.11E-10</td>
</tr>
<tr>
<td>$\Sigma P_f$</td>
<td>2.31E-06</td>
<td>2.84E-06</td>
<td>2.44E-10</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Design Variables</th>
<th>Deterministic SF 2.25</th>
<th>RB (reduced weight)</th>
<th>RB (improved Reliability)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$h$ (mm)</td>
<td>0.137055</td>
<td>0.1417367</td>
<td>0.1470413</td>
</tr>
<tr>
<td>$t_w$ (mm)</td>
<td>0.0256143</td>
<td>0.0214254</td>
<td>0.0214471</td>
</tr>
<tr>
<td>$b$ (mm)</td>
<td>0.1892625</td>
<td>0.1998203</td>
<td>0.1960113</td>
</tr>
<tr>
<td>$t_f$ (mm)</td>
<td>0.008814</td>
<td>0.0080286</td>
<td>0.0103303</td>
</tr>
<tr>
<td>Weight (kg)</td>
<td>27.55</td>
<td>24.69</td>
<td>27.75406</td>
</tr>
</tbody>
</table>
Figure 7.17 Comparison between the Probabilities of Failure for Each Failure Mode for the Improved Reliability Optimum Design and the Deterministic Optimum Design SF 2.25 in the Safety Index ($\beta$) Space.
### Table 7.7 Results of the RBDO for Deterministic Design with SF 2.50

<table>
<thead>
<tr>
<th>Failure Mode $P_f$</th>
<th>Deterministic SF 2.50</th>
<th>RB (reduced weight)</th>
<th>RB (improved Reliability)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$G_1$</td>
<td>1.97E-08</td>
<td>2.96E-09</td>
<td>2.58E-14</td>
</tr>
<tr>
<td>$G_2$</td>
<td>2.41E-30</td>
<td>7.48E-10</td>
<td>1.40E-18</td>
</tr>
<tr>
<td>$G_3$</td>
<td>6.01E-24</td>
<td>2.70E-17</td>
<td>6.29E-22</td>
</tr>
<tr>
<td>$G_4$</td>
<td>1.34E-13</td>
<td>1.10E-08</td>
<td>1.20E-14</td>
</tr>
<tr>
<td>$\Sigma P_f$</td>
<td>1.97E-08</td>
<td>1.47E-08</td>
<td>3.78E-14</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Design Variables</th>
<th>Deterministic SF 2.50</th>
<th>RB (reduced weight)</th>
<th>RB (improved Reliability)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$h$ (mm)</td>
<td>0.1334445</td>
<td>0.1453961</td>
<td>0.1476913</td>
</tr>
<tr>
<td>$t_w$ (mm)</td>
<td>0.0275258</td>
<td>0.0209613</td>
<td>0.0239623</td>
</tr>
<tr>
<td>$b$ (mm)</td>
<td>0.1894231</td>
<td>0.1984714</td>
<td>0.1908719</td>
</tr>
<tr>
<td>$t_f$ (mm)</td>
<td>0.0102119</td>
<td>0.0094911</td>
<td>0.0108385</td>
</tr>
<tr>
<td>Weight (kg)</td>
<td>29.832043</td>
<td>26.24</td>
<td>29.832043</td>
</tr>
</tbody>
</table>
Figure 7.18 Comparison between the Probabilities of Failure for Each Failure Mode for the Improved Reliability Optimum Design and the Deterministic Optimum Design SF 2.50 in the Safety Index ($\beta$) Space.
Figure 7.19 Weight Saving of the Reliability-Based Optimization over the Deterministic Optimization for Different Safety Levels

Figure 7.20 Comparing the Reliability of the Deterministic and the Reliability-Based Optimum Designs
7.7 Discussion of Reliability-Based Optimization Results

From Table 7.3, we observe that the reliability of the beam designed deterministically for a factor of safety of 1.50 is rather low (<0.97) compared to the expected level of safety of 1.27E-08, which is similar to the results found for the beam model considered in Ch. 4. In fact, looking at Table 7.3 and Figure 7.14, one may notice that the actual failure occurred at values of axial and torsional loads less than the safety factor of 1.50, while it took only a slight increase in the lateral load (from SF 1.50 to 1.594) to make one constraint \( G_i \) critical. This shows again that the factor of safety may not be an adequate method to describe the level of safety of a design, especially with loads that have high variability (COV 35%, 25% and 10%).

Also, Tables 7.4 to 7.7 and Figs 7.15 to 7.18 provide an insight into the process of RBDO as the risks of failure are redistributed to achieve a better level of safety for the same weight. Finally, as we can see from Figs. 7.19 and 7.20 the benefits of the reliability-based optimization over the deterministic optimization increase for higher safety levels. However, it may be noticed from Figure 7.19 and Figure 7.20 that the increase in the reliability seems to be improving very rapidly as the level of safety is increased, while the weight saving seems to stabilize after a sharp increase (comparing the weight saving before and after the SF 2.00 design).

This may be due to the fact that, as we approach the tail of the standardized normal distribution (\( \Phi \)), any increase in the safety index (\( \beta \)) affect the probability of failure much greater than at the lower \( \beta \) values (see Ch. 2.). This is why we can get very high safety gains for a rather constant increase in weight in an optimum design as we move from SF 1.50 to SF2.50. In addition, the material that can be removed to reduce the weight of the deterministic design for the same level of safety will be constant. Since, at high values of the safety index, a small reduction of the material
may cause large increase in the probability of failure; this limits the value of the weight savings as we increase the level of safety.
Chapter 8

Summary and Conclusions

8.1 Summary

Reliability-based design optimization (RBDO) was performed on a nonlinear elastic-plastic thin-walled T-section beam. In the first part of the study, the beam was modeled using a simple 6 degrees of freedom non-linear beam element. The objective was to demonstrate the benefits of reliability-based optimization over the deterministic optimization in such applications where the design requirements of the member tolerate some plastic behavior. Also, the aim was to address some of the difficulties that one may encounter while performing reliability-based optimization of elastic-plastic beams. For this purpose, a graphical method was used to illustrate the problems of high non-linearity and derivative discontinuity of the reliability function. The method started by obtaining a deterministic optimum design that has the lowest possible weight for a prescribed safety factor (SF), and based on that design, the method obtains an improved optimum design that has either a higher reliability or a lower weight or cost. In this application, three failure modes are considered for an elastic-plastic beam of T cross-section under combined axial, bending, and shear loads.

The failure modes are based on the beam total plastic failure in a section, buckling, and maximum allowable deflection. The results of the first part show that it is possible to get improved optimum designs (more reliable or lighter weight) using reliability-based optimization as compared to the design given by deterministic optimization. Also, the results show that the reliability function can be highly non-linear with respect to the design variables and with discontinuous derivatives. While in the second part of the study, a more elaborate 14 degrees of freedom beam element was
used to model the global failure modes, which include the flexural-torsional and the out-of-plane buckling modes. For this part, four failure modes were specified for an elastic-plastic beam of T-cross-section under combined axial, bending, torsional and shear loads. These failure modes were based on the maximum allowable in-plane, out-of-plane and axial rotational deflections, in addition, to the web-tripping local buckling failure. Finally, the beam was optimized using the sequential optimization with probabilistic safety factor (SORFS) RBDO technique, which was computationally very economic compared to the widely used nested optimization loops techniques. At the same time, the SORFS was successful in obtaining superior designs than the deterministic optimum designs (either weight savings for the same level of safety up to 12.5%, or up to seven digits improvement in the reliability for the same weight for a design with Safety Factor 2.50).

8.2 Conclusions

Deterministic structural designs with prescribed factors of safety may produce designs that may not be optimal in their level of safety or cost. The assigned factors of safety may ignore the chance of occurrence of certain load combinations that give the worst case scenario much earlier than expected. Also, for structural systems and structural components with multiple failure modes, the problem of assigning adequate safety factors becomes even more complicated. Therefore, using a more consistent measure of safety becomes a necessity. As a result, probabilistic methods have been considered as a substitute, as they can provide a consistent measure of safety of the structure. In addition to their ability to provide better optimum designs than the deterministic optimum designs. This is because the probabilistic optimum designs have better control of the material distribution vs.
the risks of failure, and therefore, can move the material from the over-designed portions to the under-designed ones.

However, the computational expense of most of the probabilistic optimization methods has limited the applicability of these methods and their acceptance by the industry as a complete substitute for deterministic optimum design. In addition, some difficulties in the calculations could occur because of the high non-linearity and the non-differentiability of the probability function, which may lead to the failure of the probabilistic optimization process.

Because of that, this study in its first part, has demonstrated computational expense and some of the RBDO difficulties. For this purpose, we have developed a simple graphical optimization method that has addressed the possible causes of RBDO difficulties using a practical example of a T-beam stiffener. Also, this study has presented in its second part an application of a very computationally economic reliability-based optimization procedure to a more elaborate beam model that considers torsional and local buckling failure modes. For this purpose, a 14 degrees of freedom non-linear elastic-plastic beam element was developed and verified against other published experimental and simulation results, in addition, to verification against ABAQUS B32OS beam element.

The results, from the second part of the study show that it is possible to perform a reliability-based design optimization of a nonlinear structure with only a moderate increase in computational expense over that for the deterministic optimum design by using the sequential optimization with
probabilistic safety factors technique. In fact, weight savings up to 12.5% and reliability improvement up to seven digits were possible using this technique.

8.3 Suggestions for Future Work

The Sequential Optimization with Probabilistic Safety Factors used in Ch. 6 needs to be further investigated by applying it to even more complex reliability optimization problems and using other probability distributions of the variations random variables. Also, it would be important to use the method for higher number of random variables and for an increased number of constraints. Finally, it would be beneficial to test the method for the case in which the design variables are also random, and hence, the cost function too becomes variable. Furthermore, applicability of the SORFS method for different methods of calculating the reliability of the design may be investigated, especially the very promising adaptive response surface techniques (Zou et al, (2003); Gupta and Manohar (2004)).

This study has considered that the loads are applied in a quasi-static manner to the beam, therefore it has ignored any dynamic effects. However, this approach may not reveal the true nature of the structure’s response to the actual loads that generally are dynamic. Therefore, the dynamic effect of these loads should be considered and possible dynamic failure modes (such as fatigue and unbounded vibrations) should be assigned. In addition, in this study, we have only considered a structural member for RBDO, however, in most real life situations more complex structures (e.g. frames, panels, bulkheads) need to be optimized under uncertainty. Moreover, the materials used in practice have variable properties that result from defects, fatigue or corrosion, which should be considered in modeling the material properties. Also, composite materials are nowadays replacing isotropic materials in increasing number of structural members, and their properties are more
uncertain than their isotropic counterparts. Finally, in this work we have used formulas to model the local buckling failure modes. Alternatively, better accuracy may be obtained by performing experimental validation, and then updating the models.
References


Bruhn, E. F., *Analysis and design of flight vehicle structures*, Tri-State Offset Co: Cincinnati, 1973


Appendix A: FORTRAN 90 Code Listing for the 6 DOF Nonlinear Elastic-Plastic T-Beam

Program FEM
Use Definitions

Implicit None
Integer, Parameter:: NE = 12
Double Precision:: q(1:7+7*NE)
Double Precision:: Kg(1:7+7*NE, 1:7+7*NE), Fg(1:7+7*NE)

Integer::NDOF, PDOF
Integer :: I, CDOF(7+7*Ne), J, K
DOUBLE PRECISION:: dq(1:7+7*Ne)
DOUBLE PRECISION:: Fex(1:7+7*Ne), P_load, q_load
DOUBLE PRECISION:: Tol, Lambda, dLambda, qP, Lambda_1, Lambda_2
DOUBLE PRECISION:: delta_1, D_m, INC, S, So

! <<<<Check for the case of pure axial load>>>>!!!

Open (Unit=1,File="Out.dat",Status="unknown",Access="append")

! Number of degrees of freedome NDOF
NDOF=7+7*Ne
!
! Number of elements Ne
! Global stiffness matrix Kg
! Nodal Force Matrix F=(Fx,Fz,M...NDOF)
! Nodal Displacements matrix W=(U,W,Theta...NDOF)
! Nodal Constraints CDOF
!
! Beam dimensions
! Beam cross-section dimensions
h = 0.20D0
b = 0.200D0
tw = 0.01D0
tf = 0.01D0

! Beam Length
L = 1.9D0
!

P_load = 225.0D3
q_load = 38.0D3

! Cross-section Properties
Area = b * tf + h * tw  ! Area
!
Ys = (b * tf * (h + 0.5D0 * tf) + h * tw * 0.5D0 * h) / Area  ! Centroid of Cross-section
!
!
! Material Properties
Em = 70.0D9
Sigma = 400.0D6
Nu = 0.3D0
strain_max = Sigma / Em
shear_st_max = strain_max / DSQRT(3.0D0)

! Element Length
LE = L / NE
NE1 = NE
!
! Section Area
Area = h * tw + b * tf
! Centroid
Ys = (h * (h + tf / 2.0D0) * tf + (h**2 * tw) / 2.0D0) / Area

! Shear Center (yo)
yo = h + 0.5D0 * tf - Ys

! Initializing matrices
! Fex(1:NDOF) = 0.0D0
Fg(1:NDOF) = 0.0D0
q(1:NDOF) = 0.0D0
CDOF(1:NDOF) = 0
dq = 0.0D0

! Initializing the tolerance
! dLambda = 1.0D0

! Specifying Constraints
!
CDOF(1) = 1 ! Simply Supported U_1
CDOF(2) = 1 ! Simply Supported W_1
CDOF(3) = 1 ! Simply Supported U_1
CDOF(4) = 1 ! Simply Supported W_1
CDOF(NDOF-4) = 1 ! Simply Supported W_NDOF
CDOF(NDOF-5) = 1 ! Simply Supported W_NDOF

!
! Specifying External Loads (+ ->-----<- +) (+ U +)
!
PDOF = 7 * NE / 2 + 3

Do J = 1, NDOF-7, 7
   Fex(J+2) = - q_load * LE ! Uniformly distributed load
End do

Fex(3) = - 0.5D0 * q_load * LE
Fex(6) = - q_load / 12.0D0 * LE**2 ! Moment at the left end of the first element
Fex(NDOF-1) = q_load / 12.0D0 * LE**2 ! Moment at the right end of the last element
Fex(NDOF-4) = - 0.5D0 * q_load * LE
Fex(NDOF-6) = - P_load ! Axial load at the end of the beam

!

! Initializing the tolerance
! dLambda = 1.0D0

!
!
Lambda = 0.0D0 ! External Loads fraction "Lambda"
I = 0 ! No of iterations to convergence "I"
!
! Using a fraction of maximum elastic displacement
Call Max_elastic_disp (P_load,q_load,D_m)

qP = 0.0D0 ! The initial displacement fraction "WP"
!
INC=-1.0D0*D_m/1000.0D0
K = 1

Do
Tol=1.0D0
Lambda_1=Lambda
I=0

do while(Tol > 1e-6)

   I = I + 1
! Calling the Global stifness matrix
   Call Assemble (q, Kg, Fg)
!
! Factoring the global stiffness matrix
   CALL Cholesky(NDOF, Kg, CDOF)
!
! Assigning [dq] to the unbalanced loads
   dq = Lambda * Fex - Fg  [dq] is an in/out variable!
!
   Fg = Fex  [Fg] is a dummy variable
!
! Elemenating the constraints from the solution
WHERE (CDOF .EQ. 1)
   dq = 0.0D0; Fg = 0.0D0
END WHERE
!
! Solving the system [KG].[W]=[Fex] to get [dq]
   CALL Solve(NDOF, kg, dq, CDOF)
!
! Solving the system [KG].[W]=[Fg] to get [Fg]
   CALL Solve(NDOF, kg, Fg, CDOF)

   S = 1.0D0/Fg(PDOF)
   dLambda = (qP - (dq(PDOF)+q(PDOF)))*S
!
! Updating Displacement
   q = q + dq + dLambda*Fg
!
! Updating the load fraction "Lambda"
   Lambda = Lambda + dLambda
!
! Checking Tolerance
   Tol=DABS(dLambda/(Lambda+1.0D-16))
!
! Limiting the number of iterations
   If (I > 500) then
      write(*,*) 'Have exceeded 500 Iterations !!!'
      End if

End do

IF(K == 1) So = S
K = K + 1
!
! Predictor Step
   q = q + S*INC*Fg
!
   Lambda_2=Lambda
!
! Checking for the mid-span displacement at the given load level
   If((Lambda >= 0.9) .and. (Lambda <= 1.1)) Delta_1=-q(PDOF)
!
   If((DABS(S/So) < 5.0D-2) .OR. (Lambda_2 < Lambda_1)) Exit
\[ S = S/So \]

Call TEELIN(Lambda,D)

write(*,'(3E16.4,I8)') Lambda, Wp, S, I

write(*,'(3E16.4,I8)') Lambda, Wp

write(*,'(3E16.4,I8)') Lambda, qp

End do

If (Lambda < 1.0) Delta_1 = -q(PDOF)

End Program FEM

! MODULE Definitions
SAVE
DOUBLE PRECISION :: h, tw, b, tf
    DOUBLE PRECISION :: L1, A, Ys, Is
    DOUBLE PRECISION :: Em, Sigma_y, Epsilon_y
END MODULE

SUBROUTINE Cholesky(N,A,CDOF)
IMPLICIT NONE
INTEGER, INTENT(IN) :: N
DOUBLE PRECISION, INTENT(INOUT) :: A(N,N)
INTEGER, INTENT(IN) :: CDOF(N)
DOUBLE PRECISION :: x, y
INTEGER :: i, j, k
!
! Calling Convention: (Note I is INTEGER D is DOUBLE PRECISION)
! CALL Cholesky(N,A,CDOF)
!
! N : D Size of Coefficients Matrix
! A : D Coefficients Matrix
! CDOF : I Vector of length N (1) for constrained DOF and 0 for free DOF
!
!
Do i=1,N
If(CDOF(i)) CYCLE
DO j=i,N
If(CDOF(j)) CYCLE
x=A(i,j)
    Do k=1,i-1
    If(CDOF(k)) CYCLE
    x=x-A(i,k)*A(j,k)
    END DO
    If(i==j) y=DSQRT(x)
    A(i,j)=x/y
    END DO
    END DO
END SUBROUTINE
SUBROUTINE Solve(N,A,b,CDOF)
IMPLICIT NONE
INTEGER, INTENT(IN) :: N
DOUBLE PRECISION, INTENT(IN) :: A(N,N)
INTEGER, INTENT(IN) :: CDOF(N)
DOUBLE PRECISION, INTENT(INOUT) :: b(N)
INTEGER :: i,j
!
! Forward Substitution
!
! Calling Convention:
!
! CALL Solve(N,A,b,CDOF)
!
! N : D Size of Coeffcients Matrix
! A : D Coeffcients Matrix
! b : D On input b contains the right hand side
! CDOF : I Vector of length N (1) for constrained DOF and 0 for free DOF
!
! DO i=1,N
! IF(CDOF(i)) CYCLE
! Do j=1,i-1
! IF(CDOF(j)) CYCLE
! b(i)=b(i)-A(i,j)*b(j)
! END DO
! b(i)=b(i)/A(i,i)
! END DO
!
! Backward Substitution
!
DO i=N,1,-1
! IF(CDOF(i)) CYCLE
! Do j=i+1,N
! IF(CDOF(i)) CYCLE
! b(i)=b(i)-A(j,i)*b(j)
! END DO
! b(i)=b(i)/A(i,i)
END DO

END SUBROUTINE

!
!
Subroutine Element(L,U,Kt_e,Fe)
Implicit None
Double Precision,Intent(IN)::L, U(1:6)
Double Precision,Intent(Out):: Kt_e(1:6,1:6),Fe(1:6)
Integer::I,J,K
Double Precision:: e,Kappa,P,M,EA,EX,EI
Double Precision::x(2),w(2),b(1:6),c(1:6)
!
This subroutine generates the element tangent stiffness matrix [Kte]
! and the element internal forces [Fe],
! given the element displacement vector [U] and using two point Gause
! integration.
Fe=0.0D0
Kt_e=0.0D0
!
Gause integration points and weights

x= (/ 0.21132486540518711775D0, 0.78867513459481288225D0 /)
w = (/ 0.5D0, 0.5D0 /)*L
Do I = 1, 2
! Calling Strain_Disp to get \[bu\], \[c\]
Call Strain_Disp(x(I), L, b, c)
!
! Strain at the neutral surface "e"
ev = DOT_PRODUCT(b, U)
!
! The curvature "Kappa"
Kappa = DOT_PRODUCT(c, U)
!
! Calling Moment_load to calculate the element internal loads
Call TeeSec(e, Kappa, P, M, EA, EX, EI)
!
Do J = 1, 6
! Element internal loads \[Fe\]
Fe(J) = Fe(J) + w(I) * ((P * b(J) + M * c(J))) !!!!!
!
Do K = 1, 6
! Element tangent stiffness matrix
Kt_e(J, K) = w(I) * (EA * b(J) * b(K) - EX * (c(K) * b(J) + c(J) * b(K)) +
+ EI * (c(J) * c(K))) + Kt_e(J, K) !!!!!
End do
End do

End Subroutine Element

!
!
Subroutine Geometric_stiff(P, L, Kgeo)
Implicit none
DOUBLE PRECISION, Intent (IN):: P, L
DOUBLE PRECISION, INTENT(OUT) :: Kgeo(6, 6)
DOUBLE PRECISION :: Kgeom
!
! The axial load "P" is +v for tensile loads
Kgeo = 0.0D0
!
Kgeom = P / (30.0D0* L)
!
kgeo (2, 2) = 36.0D0 * Kgeom
kgeo (2, 3) = 3.0D0 * L * Kgeom
kgeo (2, 5) = 36.0D0 * Kgeom
kgeo (2, 6) = 3.0D0 * L * Kgeom
!
kgeo (3, 2) = kgeo (2, 3)
kgeo (3, 3) = 4.0D0 * (L**2) * Kgeom
kgeo (3, 5) = 3.0D0 * L * Kgeom
kgeo (3, 6) = -1.0D0 * (L**2) * Kgeom
!
kgeo (5, 2) = kgeo (2, 5)
kgeo (5, 3) = kgeo (3, 5)
kgeo (5, 5) = 36.0D0 * Kgeom
kgeo (5, 6) = 3.0D0 * L * Kgeom
!
kgeo (6, 2) = kgeo (2, 6)
kgeo (6, 3) = kgeo (3, 6)
kgeo (6, 5) = kgeo (5, 6)
kgeo (6, 6) = 4.0D0 * (L**2) * Kgeom
!
End Subroutine Geometric_stiff

!
!
Subroutine Global_stiff(L, Ne, W, Kg, Fg)
Implicit None
Integer, Intent(In):: Ne
Double Precision, Intent(IN):: L, W(1:3 + 3* Ne)
Subroutine Global_stiff

Double Precision, Intent(OUT):: Kg(1:3+3*Ne,1:3+3*Ne), Fg(1:3+3*Ne)

Integer:: I,CNT

Double Precision:: Le,P,U(1:6),Fe(1:6)
Double Precision:: Kte(1:6,1:6),Kge(1:6,1:6)

! This subroutine assembles the Global stiffness matrices [Kg]
! and the internal nodal forces vector.
! Given the nodal displacement vector [W], element length "L",
! and number of elements "Ne",

Le = L/Ne

! Initializing [Kg]
Kg=0.0D0; Fg = 0.0D0

! Assembling
CNT=1
!
Do I=0,NE-1
! Assigning incremental displacements
U = W(CNT:CNT+5); CNT=CNT+3
!
! Calling the element tangent stiffness matrix [Kte]
Call Element(Le,U,Kte,Fe)
!
! Calling the element geometric stiffness matrix [Kge]
P = (Fe(4) - Fe(1))/2.0D0
Call Geometric_stiff(P,Le,Kge)
!
Kg(I*3+1:I*3+6,I*3+1:I*3+6) = Kg(I*3+1:I*3+6,I*3+1:I*3+6) + Kte + Kge
Fg(I*3+1:I*3+6) = Fg(I*3+1:I*3+6) + Fe + MATMUL(Kge,U)
End do
End subroutine Global_stiff

Subroutine Max_elastic_disp (P_load,q_load,D_m)
Use Definitions
Implicit None
Double Precision,Intent(IN):: P_load,q_load
Double Precision,Intent(OUT):: D_m
Double Precision:: P , Q , Pi, L , Sigma
Double Precision:: Lambda, Lambda_el_2

! This subroutine calculates the maximum elastic displacement "D_m"
! for a simply supported beam under axial load "P_load"
! and distributed lateral load "q_load"

P= P_load
If(P == 0.0D0) Goto 300
Q= q_load
Pi=3.14159265358979311D0
L=L1
Sigma = Sigma_y

! Lambda_el_1 = (Em*P**2*(8.0D0*Js*P + A*L**2*Q*Ys) + 8.0D0*A*L**2*P*Sigma + &
DSqrt(256.0D0*Em*Js*L**2*P**2*P**2*Sigma**2 + &
(Em*P**2*(8.0D0*Js*P + A*L**2*Q*Ys) + &
8.0D0*A*L**2*P*Sigma)**2))/ (-16.0D0*L**2*P**2)
\[ \Lambda_{el,2} = \frac{(Em*Pi^2*(8.0D0*Is*P - A*L^2*Q*Ys) - 8.0D0*A*L^2*P*Sigma + \sqrt{256.0D0*A*Em*Is*L^2*P^2*Pi^2*Sigma + (Em*Pi^2*(-8.0D0*Is*P + A*L^2*Q*Ys) + 8.0D0*A*L^2*P*Sigma)^2)}}{16.0D0*A*L^2*P**2} \]

\[ \Lambda = \Lambda_{el,2} \]

\[ D_m = \frac{(5.0D0*L^4*Pi^2*Q*\Lambda)}{384.0D0*(Em*Is*Pi^2 - L^2*P*\Lambda)} \]

\[ 300D_m = 5.0D0*Q*L^4/(384.0D0*Em*Is) \]

End Subroutine Max_elastic_disp

! Subroutine Strain_Disp(x,L,b,c)
Implicit none
Double precision, Intent(IN):: x, L
Double precision, Intent(OUT):: b(1:6), c(1:6)

! This subroutine generates the vectors \([b] \) "x dependant" and \([c] \) "x, Z dependant" to be used to obtain the strain and the tangent stiffness matrix \([Kte]\)

\[ b = 0.0D0 \]
\[ c = 0.0D0 \]
\[ b(1) = -1.0D0/L \]
\[ b(4) = 1.0D0/L \]
\[ c(2) = \frac{6*(1 - 2*x)}{L**2} \]
\[ c(3) = \frac{6*x - 4}{L} \]
\[ c(5) = \frac{-6*(1 - 2*x)}{L**2} \]
\[ c(6) = \frac{-2*(1 - 3*x)}{L} \]

End Subroutine Strain_Disp

! Subroutine TeeSec(e,Kappa,P,M,EA,EX,EI)
Use Definitions
Implicit None
DOUBLE PRECISION, INTENT(IN) :: e, Kappa
DOUBLE PRECISION, INTENT(OUT):: P, M
DOUBLE PRECISION, INTENT(OUT):: EA, EX, EI
Integer:: I, J, MINLOC_array(1), Locmin
DOUBLE PRECISION:: Strain
DOUBLE PRECISION:: Width
DOUBLE PRECISION:: Epsilon, Epsilon_1, Epsilon_2, slope
DOUBLE PRECISION:: Sigma, Sigma_e(1:6)
DOUBLE PRECISION:: Z, dh, Zp_c, Zp_t, Zo, Zf, Zt, Zb, Small
DOUBLE PRECISION:: Ze(1:6), V, ARM
Double Precision:: Z_sec(1:2)
Double Precision:: he, tfe, Yse, Ae, Fe, Me

! This subroutine calculates the moment "Me", the force "Fe",
! for each cross-section at each element span location x.
! By using function Strain(u1,u2,u3,u4,u5,u6,x,Z).

! <<<The Sign convention is + for compressive loads and +
! smiling moments!!>>>

198
Z=-Ys  ! Lower ply
! Calling the Strain function
Epsilon_2=STRAIN(e, Kappa, Z)  ! Lower ply strain
!
Z=h+tf-Ys  ! Upper ply
! Calling the Strain function
Epsilon_1=STRAIN(e, Kappa, Z)  ! Upper ply strain
!
! Checking for pure axial load
If (Epsilon_1 == Epsilon_2) then
  M=0.0D0
  If (Epsilon_1 < Epsilon_y) .and. (Epsilon_1 > - Epsilon_y) then
    Sigma=Epsilon * Em
    Else
      Sigma=SIGN(Sigma_y, Epsilon)
    End If
    P=Sigma*A
    EA=A*Em
    EX=0.0D0
    El=Is*Em
    Goto 200
  End if
!
! Calculating the slope of the strain distribution
slope=-1.0D0*(h+tf)/(Epsilon_2 - Epsilon_1)  ! Slope
!
! Checking the vertical dimension of the plastic tension zone
Zp_t=slope*(Epsilon_y - Epsilon_2) - Ys
!
! Checking if the tension yield zone is within boundaries of cross-section
If (Zp_t >= (h+tf-Ys)) Zp_t=h+tf-Ys
If (Zp_t <= -Ys) Zp_t=-Ys
!
! Checking the vertical dimension of the plastic compression zone
Zp_c=slope*(- Epsilon_y - Epsilon_2) - Ys
!
! Checking if the compression yield zone is within boundaries of cross-section
If (Zp_c >= (h+tf-Ys)) Zp_c=h+tf-Ys
If (Zp_c <= -Ys) Zp_c=-Ys
!
! Location of the neutral surface
Zo=slope*(-1.0D0*Epsilon_2) - Ys
!
! Checking if the neutral surface is within boundaries of cross-section
If (Zo >= (h+tf-Ys)) Zo=h+tf-Ys
If (Zo <= -Ys) Zo=-Ys
!
! The bottom of the flange
ZF=h-Ys
!
! The bottom of the web
Zb=-Ys
!
! The top of the flange
Zt=h+tf-Ys
!
Ze(1)=Zb; Ze(2)=Zp_t; Ze(3)=Zo; Ze(4)=ZF; Ze(5)=Zp_c; Ze(6)=Zt
!
! Arranging the Ze's from the smallest to the largest
Do I=1,5
  Small=Minval(Ze(I:6))
  MinLoc_array=MINLOC(Ze(I:6))
  Locmin=(I-1)+MinLoc_array(1)
  Ze(Locmin)=Ze(I)
  Ze(I)=Small
End do
!
! Initializing the load and moment
Fe=0.0D0
Me=0.0D0
!
! Calculating the load and moment
Do J=1,5
   dh=Ze(J+1)-Ze(J) ! Step size
   Z=Ze(J)
   !
   ! Checking for the width
   If(Z+(dh/2.0D0) > Zf) then
      Width= b
   Else
      Width=tw
   End if

   ! Calling the Strain function
   Epsilon=STRAIN(e,Kappa,Z)
   !
   ! Checking for the plastic stress
   If((Epsilon < Epsilon_y) .and. (Epsilon > - Epsilon_y)) then
      Sigma_e(J)=Epsilon * Em
   Else
      Sigma_e(J)=SIGN(Sigma_y,Epsilon)
   End If

   Z=Ze(J+1)
   ! Calling the Strain function
   Epsilon=STRAIN(e,Kappa,Z)
   !
   If((Epsilon < Epsilon_y) .and. (Epsilon > - Epsilon_y)) then
      Sigma_e(J+1)=Epsilon * Em
   Else
      Sigma_e(J+1)=SIGN(Sigma_y,Epsilon)
   End If

   If(Z <= Zo) then
      V= (Sigma_e(J+1)+0.5D0*(Sigma_e(J)-Sigma_e(J+1)))*dh*Width
      Fe=Fe+V
      ARM=(Z-dh+(0.5D0*Sigma_e(J+1)+(1.0D0/6.0D0)*(Sigma_e(J)-Sigma_e(J+1)))*dh&
         / (Sigma_e(J+1)+0.5D0*(Sigma_e(J)-Sigma_e(J+1))))
   Else
      V=(Sigma_e(J)+0.5D0*(Sigma_e(J+1)-Sigma_e(J)))*dh*Width
      Fe=Fe+V
      ARM=(Z-dh+(0.5D0*Sigma_e(J)+(1.0D0/3.0D0)*(Sigma_e(J+1)-Sigma_e(J)))*dh&
         / (Sigma_e(J)+0.5D0*(Sigma_e(J+1)-Sigma_e(J))))
      Me=Me+ARM*V
   End if
End do
P=Fe
M=Me
!write(1,*), P,M

!!**************************************************************************!!

! This section calculates the elastic section properties Area,"Ae"
! , the centriod "Yse", the second moment of area "Ise",
! the first moment of area,X_sec
! given the vertical dimensions of the elastic zone Zt & Zc

! Assigning the lower and upper dimensions of the plastic zone
! Z_sec(1)=Min(Zp_c,Zp_t) ! The lower plastic zone
! Z_sec(2)=Max(Zp_c,Zp_t) ! The upper plastic zone

he=h
tfe=tf
! Checking the flange plastic zone
If (Z_sec(2) <= h-Ys) then
   tfe=0.0D0 ! the flange is fully plastic => eliminate flange
   he=Z_sec(2)-Z_sec(1) ! The web contains two plastic zones
Else
   if (Z_sec(1) >= h-Ys) then
      tfe=Z_sec(2)-Z_sec(1) ! The flange contains two plastic zones
      he=0.0D0 ! The web is fully plastic => eliminate web
   else
      tfe=(Z_sec(2)+Ys)-h ! Subtract upper plastic zone from flange
      he=h-(Z_sec(1)+Ys) ! Subtract lower plastic zone from web
   End if
End if

! Elastic Area
Ae = b*tfe + he*tw
EA=Ae*Em
!
! Centroid of elastic area
Yse = (b*tfe*(he + 0.5D0*tfe) + he*tw*0.5D0*he)/Ae + (Z_sec(1)+Ys)
!
! First moment of elastic area
EX=Ax*(Yse-Ys)*Em
!
! Second moment of elastic area
EI=((b*tfe**3)/12.0D0 + b*tfe*(h + 0.5D0*tfe-Ys)**2 &
   + (tw*he**3)/12.0D0 + he*tw*(h - he/2.0D0-Ys)**2)*Em

200 End Subroutine TeeSec

DOUBLE PRECISION FUNCTION STRAIN(e,Kappa,Z)
DOUBLE PRECISION, INTENT(IN) :: e,Kappa,Z
Strain=e - Kappa*Z
END FUNCTION
Appendix B: Backward Euler Method to Obtain the Plastic Stresses

Problem Definition: Given the value of the stresses and strains at the increment $n$, find the values of the stresses and strains at increment $n + 1$, using the following relations:

$$\varepsilon_{n+1} = \varepsilon_n + \Delta \varepsilon$$  \hspace{1cm} (B.1.a)

$$\varepsilon^p_{n+1} = \varepsilon^p_n + \Delta \lambda_{n+1} \left( \frac{\partial f}{\partial \sigma_{n+1}} \right)$$  \hspace{1cm} (B.1.b)

$$\sigma_{n+1} = [C] \cdot (\varepsilon_{n+1} - \varepsilon^p_{n+1})$$  \hspace{1cm} (B.1.c)

$$f_{n+1} = f(\sigma_{n+1})$$  \hspace{1cm} (B.1.d)

where $\varepsilon = \begin{bmatrix} \varepsilon \\ \gamma \end{bmatrix}$ are the axial and shear strains, $\sigma_{n+1} = \begin{bmatrix} \sigma \\ \tau \end{bmatrix}$ are the axial and shear stresses, $[C] = \begin{bmatrix} E & 0 \\ 0 & G \end{bmatrix}$ is the matrix of elastic axial and shear modulus, and $f(\sigma) = \sqrt{\sigma^2 + 3\tau^2} - \sigma_y$ is the Von Misses yield function.

The plastic strain increment is given by
\[ \Delta \tilde{\varepsilon}_{n+1}^p = \tilde{\varepsilon}_{n+1}^p - \tilde{\varepsilon}_n^p = \Delta \lambda_{n+1} \left( \frac{\partial f}{\partial \tilde{\sigma}_{n+1}} \right) \]  

(B.2)

where \( \Delta \lambda_{n+1} \) is the plastic strain rate at increment \( n + 1 \).

Then the stress becomes

\[
\tilde{\sigma}_{n+1} = [C] \left( \tilde{\varepsilon}_{n+1}^p - \tilde{\varepsilon}_n^p - \Delta \tilde{\varepsilon}_{n+1}^p \right) \\
\tilde{\sigma}_{n+1} = [C] \left( \tilde{\varepsilon}_n + \Delta \tilde{\varepsilon} - \tilde{\varepsilon}_n^p - \Delta \tilde{\varepsilon}_{n+1}^p \right) = [C] \cdot (\tilde{\varepsilon}_n - \tilde{\varepsilon}_n^p + \Delta \tilde{\varepsilon} - \Delta \tilde{\varepsilon}_{n+1}^p) \\
\tilde{\sigma}_{n+1} = (\tilde{\sigma}_n + [C] \cdot \Delta \tilde{\varepsilon}) - [C] \cdot \Delta \tilde{\varepsilon}_{n+1}^p \\
\tilde{\sigma}_{n+1} = \tilde{\sigma}_{n+1}^{trial} - \Delta \lambda_{n+1} [C] \cdot \left( \frac{\partial f}{\partial \tilde{\sigma}_{n+1}} \right) \]  

(B.3)

i.e. \( \tilde{\sigma}_{n+1}^{trial} \) is the elastic predictor stress and \( \Delta \lambda_{n+1} [C] \cdot \left( \frac{\partial f}{\partial \tilde{\sigma}_{n+1}} \right) \) is the plastic corrector stress.

It is important to note that during the elastic predictor the plastic strain remains fixed and during the plastic corrector stage the total strain remains fixed.

This implies that during the plastic corrector stage

\[ \Delta \tilde{\sigma}_{n+1} = -\Delta \lambda_{n+1} [C] \cdot \left( \frac{\partial f}{\partial \tilde{\sigma}_{n+1}} \right) \]  

(B.4)
The solution of the set of non-linear algebraic equations is obtained by using Newton method and the systematic linearization of the non-linear equations. However, since the total strain is constant during the corrector stage, the linearization will be performed only with respect to the increment in the plasticity parameter (\( \Delta \lambda \)), and will take the following form for any function \( g(\Delta \lambda) = 0 \)

\[
g^{(k)} + \left( \frac{dg}{d\Delta \lambda} \right)^{(k)} \delta \lambda^{(k)} = 0
\]  

(B.5)

where

\[
\Delta \lambda^{(k+1)} = \Delta \lambda^{(k)} + \delta \lambda^{(k)}
\]  

(B.6)

in which \( k \) is the iteration number and \( \delta \lambda^{(k)} \) is the increment of \( \Delta \lambda \) at the \( k \)th iteration.

Now, rewriting Eq. (B.1.b) and (B.1.d) in the form

\[
\tilde{a}_{n+1} = -\tilde{e}^p_{n+1} + \tilde{e}^p + \Delta \dot{\lambda}_{n+1} \left( \frac{\partial f}{\partial \sigma_{n+1}} \right) = 0
\]

(B.7)

\[
f_{n+1} = f(\tilde{\sigma}_{n+1}) = 0
\]

Linearizing Eq. (B.7) and using the relation in Eq. (B.4) \( \Delta \varepsilon^{\mu(k)} = -C^{-1} \cdot \Delta \tilde{\sigma}^{(k)}_{n+1} \); we get
\[
\vec{u}^{(k)}_{n+1} + \left[ C \right]^{-1} \cdot \Delta \sigma^{(k)}_{n+1} + \Delta \lambda^{(k)}_{n+1} \left[ H^{(k)}_{n+1} \right] \cdot \Delta \sigma^{(k)}_{n+1} + \delta \lambda \left( \frac{\partial f}{\partial \sigma^{(k)}_{n+1}} \right) = 0
\] (B.8)

Where \( \left[ H^{(k)}_{n+1} \right] = \frac{\partial^2 f}{\partial \sigma^{(k)}_{n+1}^2} \)

and

\[
f^{(k)}_{n+1} + \left( \frac{\partial f}{\partial \sigma^{(k)}_{n+1}} \right) \cdot \Delta \sigma^{(k)}_{n+1} = 0
\] (B.9)

Let’s rewrite Eq. (B.8) in the following form

\[
\left[ \left[ C \right]^{-1} + \Delta \lambda^{(k)}_{n+1} \left[ H^{(k)}_{n+1} \right] \right] \cdot \Delta \sigma^{(k)}_{n+1} = -\vec{u}^{(k)}_{n+1} - \delta \lambda \left( \frac{\partial f}{\partial \sigma^{(k)}_{n+1}} \right)
\] (B.10)

Solving for the stress increment \( \Delta \sigma^{(k)}_{n+1} \)

\[
\Delta \sigma^{(k)}_{n+1} = \left[ \left[ C \right]^{-1} + \Delta \lambda^{(k)}_{n+1} \left[ H^{(k)}_{n+1} \right] \right]^{-1} \cdot \vec{u}^{(k)}_{n+1} - \delta \lambda \left[ C \right]^{-1} \cdot \Delta \lambda^{(k)}_{n+1} \left[ H^{(k)}_{n+1} \right]^{-1} \cdot \left( \frac{\partial f}{\partial \sigma^{(k)}_{n+1}} \right)
\] (B.11)
Substituting $\Delta \tilde{\sigma}_{n+1}^{(k)}$ in Eq. (B.9) and solving for $\delta \lambda$

\[
\delta \lambda^{(k)} = \frac{f^{(k)} - \left( \frac{\partial f}{\partial \tilde{\sigma}_{n+1}^{(k)}} \right)^T [C]^{-1} + \Delta \lambda_{n+1}^{(k)} [H_{n+1}^{(k)}]^{-1} \tilde{\sigma}_{n+1}^{(k)}}{\left( \frac{\partial f}{\partial \tilde{\sigma}_{n+1}^{(k)}} \right)^T [C]^{-1} + \Delta \lambda_{n+1}^{(k)} [H_{n+1}^{(k)}]^{-1} \left( \frac{\partial f}{\partial \tilde{\sigma}_{n+1}^{(k)}} \right)}
\]  

(B.12)

Hence, we can solve the problem as in the following flowchart.
At the end of the last incremental step (n): set

\[ k = 0 \]
\[ \tilde{\varepsilon}^{p(0)} = \tilde{\varepsilon}_n \]
\[ \Delta \lambda^{(0)} = 0 \]

\[ \tilde{\varepsilon}_{n+1} = \tilde{\varepsilon}_n + \Delta \tilde{\varepsilon} \]
\[ \tilde{\sigma}_{(n+1)}^{(0)} = \tilde{C} \cdot (\tilde{\varepsilon}_{n+1} - \tilde{\varepsilon}^{p(0)}) \]

Check convergence

\[ f(\tilde{\sigma}_{n+1}^{(k)}) \leq Tol_1 \]
and
\[ \left\| \tilde{\varepsilon}_{(n+1)}^{(p)} - \tilde{\varepsilon}_{(n)}^{(p)} + \Delta \lambda \left( \frac{\partial f}{\partial \tilde{\sigma}} \right)_{(n+1)} \right\| \leq Tol_2 \]

Calculate
\[ \Delta \lambda^{(k)} \]
\[ \text{and} \]
\[ \Delta \tilde{\sigma}^{(k)} \]

Update
\[ \tilde{\varepsilon}^{p(k+1)} = \tilde{\varepsilon}^{p(k)} - \tilde{C}^{-1} \cdot \Delta \tilde{\sigma}^{(k)} \]
\[ \Delta \lambda^{(k+1)} = \Delta \lambda^{(k)} + \Delta \lambda^{(k)} \]
\[ \tilde{\sigma}^{(k+1)} = \tilde{\sigma}^{(k)} + \Delta \tilde{\sigma}^{(k)} \]

Set
\[ k + 1 \rightarrow k \]
Appendix C: FORTRAN 90 Code Listing for the 6 DOF Nonlinear Elastic-Plastic T-Beam

Program FEM_Analysis

! This Program is used as the Analysis Code for Visual Doc Optimizer
! If used as a stand alone, the user should provide the input files or enter the values manually
! It Simulates the Behavior of a Nonlinear 14 DOF Elastic-Plastic Beam

Use Definitions

Implicit None
Double Precision:: G1, G2, G3, G4, G5, G6, G7, Weight, OBJ
Integer:: I

OPEN(UNIT = 1, FILE = 'dvar.vef', ACTION = "READ")
READ(UNIT = 1,FMT = *) h
READ(UNIT = 1,FMT = *) tw
READ(UNIT = 1,FMT = *) b
READ(UNIT = 1,FMT = *) tf
READ(UNIT = 1,FMT = *) P_load
READ(UNIT = 1,FMT = *) q_load
READ(UNIT = 1,FMT = *) Ecc
READ(UNIT = 1,FMT = *) FS
CLOSE(1)

! Beam dimensions
! Cross-section Dimensions
! Beam Length
L = 1.9D0
!
! Material Properties
Em = 70.0D9
Sigma_Y = 400.0D6
Nu = 0.375D0
Gm = 27.0D9

Open(1,FILE = 'resp.vef',FORM='FORMATTED',ACCESS = 'SEQUENTIAL',STATUS='UNKNOWN')
rewind(1)

! Calling the Analysis Code
Call FEM(G1, G2, G3, G4, G5, G6, G7, Weight)
!
OBJ = DSQRT(((P_load-225000.0D0)/(0.25D0*225000.0D0))**2 + & 
((Q_load-38000D0)/(0.35D0*38000D0))**2 +((Ecc-0.004D0)/(0.1D0*0.004D0))**2)

write(1,'(E16.4)') G1
write(1,'(E16.4)') G2
write(1,'(E16.4)') G3
write(1,'(E16.4)') G4
write(1,'(E16.4)') G5
write(1,'(E16.4)') G6
write(1,'(E16.4)') G7
write(1,'(E16.4)') Weight
write(1,'(E16.4)') OBJ

close (1)

END PROGRAM

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

MODULE Definitions
SAVE
DOUBLE PRECISION :: h, tw, b, tf
DOUBLE PRECISION :: L, LE
DOUBLE PRECISION :: Em, Sigma_Y, Nu, Gm
DOUBLE PRECISION :: FS, P_load, q_load, Ecc
Integer:: SectionConfig
Integer, Parameter::NE=8, m = 6
END MODULE

Subroutine AssembleP (flag1, qo, qn, Kg, K_sigma, K_element, Fg, Check)

!_________________________________________________________________________________________________________________
!                                                                                                           
!   This subroutine assembles the Plastic Global stiffness matrices [Kg], [K_sigma], [K_element], and the internal nodal forces vector [Fg].
!                                                                                                           
! INPUTS: nodal displacement vector {q}, the displacements increment vector
! OUTPUTS: the Global stiffness matrices [Kg] and the internal nodal forces vector [Fg]
! Subroutine Calls: KF_integ (U, K_int, K_sigma_e, K_element_e, F_int)
!__________________________________________________________________________________________________________________!

Use Definitions

Implicit None

Double Precision, Intent(IN):: qo(7+7*NE), qn(7+7*NE)
Double Precision, Intent(OUT):: Kg(7+7*NE,7+7*NE), Fg(7+7*NE), K_sigma(7+7*NE,7+7*NE), K_element(7+7*NE, 7+7*NE)
Integer, Intent(INOUT):: flag1
Integer, Intent(OUT):: Check
Integer:: I, CNT, Element
Double Precision:: Uo(14), Un(14)
Double Precision:: K_int(14,14), K_sigma_e(14,14), K_element_e(14,14), F_int(14)
Double Precision:: SigmaBeamOld(NE,2,3,m,m,2), SigmaBeamNew(NE,2,3,m,m,2)

! Initializing
Kg = 0.0D0                  ! Total Global Stiffness Matrix
Fg = 0.0D0                   ! Internal Nodal Force vector
K_sigma = 0.0D0        
K_element = 0.0D0
F_int=0.0D0

If (flag1 == 0) SigmaBeamOld = 0.0D0

! Assembling
CNT=1

Do I=0,NE-1
   Element = I + 1
! Assigning incremental displacements
   Uo = qo(CNT:CNT+13);
   Un = qn(CNT:CNT+13);

! Calling the integrated element tangent stiffness matrix (K_int) and the element internal forces vector (F_int)
! INPUTS : U (Element Displacements)
! OUTPUTS : K_int, K_sigma_e, K_element_e, F_int

Call KF_integP (SigmaBeamOld, SigmaBeamNew, Element, Uo, Un, K_int, K_sigma_e, K_element_e, F_int, Check)

! Assembling
Kg(I*7+1:I*7+14, I*7+1:I*7+14) = Kg(I*7+1:I*7+14, I*7+1:I*7+14) + K_int
K_sigma(I*7+1:I*7+14, I*7+1:I*7+14) = K_sigma(I*7+1:I*7+14, I*7+1:I*7+14) + K_sigma_e
K_element(I*7+1:I*7+14, I*7+1:I*7+14) = K_element(I*7+1:I*7+14, I*7+1:I*7+14) + K_element_e

Fg(I*7+1:I*7+14) = Fg(I*7+1:I*7+14) + F_int
SUBROUTINE BDPredictor(N,CDOF,PDOF,Kt,Fex,Fin,dqt,Lambda,delta,flag)
IMPLICIT NONE
INTEGER, INTENT(IN) :: N, CDOF(N),PDOF
DOUBLE PRECISION, INTENT(INOUT) :: Kt(N,N),Fin(N),Lambda
DOUBLE PRECISION, INTENT(OUT) :: dqt(N)
DOUBLE PRECISION, INTENT(IN) :: Fex(N),delta
Integer, Intent(OUT):: flag

DOUBLE PRECISION :: FinPDOF,FexPDOF, a1,a2,dLambda
flag = 0

! The inputs
! N,Kg(N,N), CDOF

CALL Cholesky(N,Kt,CDOF,flag)
If (flag == 1) Goto 100

! output: Factored Kt

Fin = Lambda*Fex-Fin  ! Fin= g, the unbalanced forces
FinPDOF = Fin(PDOF)
dqt = Fex
FexPDOF = Fex(PDOF)
WHERE(CDOF .EQ. 1)
dqt = 0.0D0; Fin = 0.0D0
END WHERE

! Assigning the displacement control DOF
Fin(PDOF) = delta   ! delta= INC

! INPUT: Factored Kt, Fin = g

CALL Solve(N,Kt,Fin,CDOF)
! Output: Fin = q unblanced

! Input: Factored Kt, dqt=Fex

CALL Solve(N,Kt,dqt,CDOF)
! Output: dqt= displacements due to Fex

! a1 = (Kt * q) - g ....> at PDOF
a1 = DOT_PRODUCT(Kt(PDOF,:),Fin)-FinPDOF                            

!! Kt(PDOF,:) is not factorized

! a2 = (Kt * qex) - Fex ....> at PDOF
a2 = DOT_PRODUCT(Kt(PDOF,:),dqt)-FexPDOF

dLambda = - a1/a2
write(*,*) a1, a2
! Updating the displacements
! q = q + dLambda * qFex
Fin = Fin + dLambda*dq

! Updating the Load
Lambda = Lambda + dLambda

100 END SUBROUTINE
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

SUBROUTINE BackwardEuler(deps,sigma,Cmt,Check)
USE Definitions
IMPLICIT NONE
DOUBLE PRECISION, INTENT(IN) :: deps(2)
DOUBLE PRECISION, INTENT(INOUT) :: sigma(2)
DOUBLE PRECISION, INTENT(OUT) :: Cmt(2,2)
Integer, Intent(OUT):: Check
DOUBLE PRECISION :. Cm(2,2), Cm_inv(2,2), dlambda, ddlambda, dsigma(2), f, A(2,2), H1(2,2), r(2), e(2), g(2)

Integer:: CNT
Cm = 0.0D0; Cm(1,1) = Em; Cm(2,2) = Gm
Cm_inv = 0.0D0; Cm_inv(1,1) = 1.0D0/Em; Cm_inv(2,2) = 1.0D0/Gm
sigma = sigma + MATMUL(Cm,deps)

! Checking for Elastisity
if(VonMisesStress(sigma) < Sigma_Y) then
   Cmt = Cm
   return
end if

Check = 0

dsimga = 0.0D0; dlambda = 0.0D0

CNT = 1

Do
   sigma = sigma + dsigma
   CALL VonMises(sigma,f,r,H1)
   A = Invert2x2(Cm_inv + dlambda*H1)
   e = MATMUL(A,MATMUL(Cm_inv,dsigma)+dlambda*r)
   g = MATMUL(A,r)
   ddlambda = (f-Sigma_Y-DOT_PRODUCT(r,e))/DOT_PRODUCT(r,g)
   dlambda = dlambda + ddlambda
   dsigma = -ddlambda*g-e
   IF((DABS(1.0D0-f/Sigma_Y) < 1.0D-4) .or. (CNT > 100)) Exit
   CNT = CNT + 1
End Do

Cmt = (Cm-MATMUL(reshape(g,(/2,1/)),reshape(g,(/1,2/)))/DOT_PRODUCT(g,r))

CONTAINS
DOUBLE PRECISION FUNCTION VonMisesStress(sigma)
DOUBLE PRECISION, INTENT(IN) :: sigma(2)
VonMisesStress = DSQRT(sigma(1)**2 + 3.0D0*sigma(2)**2)
END FUNCTION
SUBROUTINE VonMises(sigma,f,r,H1)
DOUBLE PRECISION, INTENT(IN) :: sigma(2)
DOUBLE PRECISION, INTENT(OUT) :: f, r(2), H1(2,2)
f = DSQRT(sigma(1)**2 + 3.0D0*sigma(2)**2)
r = (/sigma(1),3.0D0*sigma(2))/f
H1 = 3.0D0*reshape( (/sigma(2)**2,-sigma(1)*sigma(2),-sigma(1)*sigma(2),sigma(2)**2/),(/2,2/) ) / f**3

END SUBROUTINE

FUNCTION Invert2x2(B) RESULT(F)
DOUBLE PRECISION, INTENT(IN) :: B(2,2)
DOUBLE PRECISION :: F(2,2)
F = reshape((/B(2,2),-B(2,1),-B(1,2),B(1,1)/),(/2,2/))/(B(1,1)*B(2,2)-B(1,2)*B(2,1))
END FUNCTION

END SUBROUTINE

SUBROUTINE Cholesky
USE Definitions
IMPLICIT NONE
INTEGER, INTENT(IN) :: N
DOUBLE PRECISION, INTENT(INOUT) :: A(N,N)
INTEGER, INTENT(IN) :: CDOF(N)
INTEGER, Intent(OUT):: Flag
DOUBLE PRECISION :: x,y
INTEGER :: i,j,k
!
! Calling Convention: (Note I is INTEGER D is DOUBLE PRECISION)
! CALL Cholesky(N,A,CDOF)
!
! N    : D Size of Coefficients Matrix
! A    : D Coefficients Matrix
! CDOF : I Vector of length N (1) for constrained DOF and 0 for free DOF
!
!
flag = 0
Do i=1,N
IF(CDOF(i)) CYCLE  
DO j=i,N
IF(CDOF(j)) CYCLE
  x=A(j,i)
  Do k=1,i-1
    IF(CDOF(k)) CYCLE
    x=x-A(i,k)*A(j,k)
  END DO
  IF(i==j .and. x < 0.0) then
    ! write(*,*) 'Matrix is becoming singular !!'
    flag = 1
    goto 100
  End If
  IF(i==j) y=DSQRT(x)
  A(j,i)=x/y
END DO
END DO
100 END SUBROUTINE

SUBROUTINE Solve
IMPLICIT NONE
INTEGER, INTENT(IN) :: N
DOUBLE PRECISION, INTENT(IN) :: A(N,N)
INTEGER, INTENT(IN) :: CDOF(N)
DOUBLE PRECISION, INTENT(INOUT) :: b(N)
INTEGER :: i,j
DOUBLE PRECISION :: x
!

Forward Substitution

! Calling Convention:

CALL Solve(N,A,b,CDOF)

! N : D  Size of Coeffcients Matrix
! A : D  Coeffcients Matrix
! b : D  On input b contains the right hand side
! on output it contains the solution
! constrained DOF are set to zero
! CDOF : I  Vector of length N (1) for constrained DOF and 0 for free DOF

DO i=1,N
IF(CDOF(i)) CYCLE
DO j=1,N
IF(1-CDOF(j)) CYCLE
IF(i<j) Then
  x = A(i,j)
Else
  x = A(j,i)
End If
b(i)=b(i)-x*b(j)
END DO
END DO

DO i=1,N
IF(CDOF(i)) CYCLE
DO j=i+1,N
IF(CDOF(j)) CYCLE
b(i)=b(i)-A(i,j)*b(j)
END DO
b(i)=b(i)/A(i,i)
END DO

END SUBROUTINE

Subroutine Converged_step(INC,Lambda,q,qo,qn,flag2)
Use Definitions
Implicit none
Double Precision, Intent(INOUT):: INC, Lambda,q(7+7*NE), qo(7+7*NE),qn(7+7*NE)
Integer, Intent(INOUT):: flag2
Double Precision:: Lambda_old, q_old(7+7*NE), qo_old(7+7*NE),qn_old(7+7*NE), INC_old

If (flag2 == 0) then
  INC = INC *0.9D0
Elseif(flag2 ==1) then
  Lambda_old= Lambda
  q_old= q
  qo_old = qo
  qn_old = qn
  INC_old= INC
Elseif (flag2 ==2) then
  Lambda = Lambda_old
  q = q_old
  qo = qo_old
End Subroutine Converged_step
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

Subroutine dKF_LP (SigmaBeamOld, SigmaBeamNew,Element, Gauss,Uo, Un, z, dK_sigma, dK_element, dF_in,Check)

!________________________________________________________________________________________________________________________!
!
! This subroutine gets the section stiffness matrix and the internal forces vector per unit length (at gauss points)
!
! INPUT: old and incremented element displacements vector (Uo, Un), Gauss point location along the element (z)
!
! OUTPUT: the section stiffness matrix (dK_sigma, dK_element), the internal force vector (dF_in)
!
! IN-OUT: element stresses at each Gauss point (Sigmab_eg)
!
! Subroutine Calls: qz_Matrices (U, z, Gamma, A, Zc), Section (U, z, C)

!________________________________________________________________________________________________________________________!

Use definitions
Implicit none
Integer, Intent(IN):: Element, Gauss
Double Precision, Intent(IN):: Uo(1:14), Un(1:14), z
Double Precision, Intent(OUT):: dK_sigma(1:14,1:14), dK_element(1:14,1:14), dF_in(1:14)
Integer, Intent(OUT):: Check
Double Precision, Intent(IN):: SigmaBeamOld(NE,2,3,m,m,2)
Double Precision, Intent(OUT):: SigmaBeamNew(NE,2,3,m,m,2)
Double Precision:: U(1:14), Gamma(1:6), GammaO(1:6), GammaN(1:6), dGamma(1:6)
Double Precision:: A(1:8,1:6), Zc(1:14,1:8), Ctde(1:6,1:6)
Double Precision:: OmegaPde(1:6), Sq (1:8,1:8), OmegaPde1(6)
Double Precision:: K_element1 (1:14,1:6), K_element2 (1:14,1:6), B_mat(1:14,1:6)
Double Precision:: Xo, Yo, a1(3), b1(3), c1(3), d1(3)
integer :: i

Sq=0.0D0                                                                                                      ! Initializing [Sq (8,8)] (a component of the stiffness matrix)

! Calling qz_Matrices (the displacement dependant matrices)
!
! INPUTS  : U (element displacement vector), z (axial Gauss point integration point)
!
! OUTPUTS : Gamma (longitudenal element strains), A (strain-disp matrix), Zc (interpolation functions derivatives)
!......................................................................................................................................

U = Uo                                                                         ! Assigning the old element displacements
Call qz_Matrices (U, z, Gamma, A, Zc)                      ! Assigning the old element Strains
GammaO = Gamma
!
U = Un
Call qz_Matrices (U, z, Gamma, A, Zc)
GammaN = Gamma                                           ! Assigning the old element Strains
dGamma = GammaN - GammaO                                  ! Obtaining the element Strain increment
!
! Calling Cross-Section constants
!
! INPUTS : Old element Strains (GammaO), element Strain increment (dGamma)
!
! OUTPUTS : element stresses at each Gauss point (Sigmab_eg), modular element tangent matrix [6X6] (Ctde), Element stresses {6} (OmegaPde)
!......................................................................................................................................

integer :: i
Do i=1,6
IF (DFABS(OmegaPde(i)) < 1.0D-15) OmegaPde(i) = 0.0D0
End do
Do i=1,6
   IF (DABS(OmegaPde(i)) < 1.0D-15) OmegaPde(i) = 0.0D0
End do
!

The SQ matrix (a component of the stiffness matrix)
Call SectionProperties1(Xo, Yo, a1, b1, c1, d1)

Sq(2,2) = OmegaPde(1)
Sq(2,7) = Yo * OmegaPde(1)
Sq(3,6) = - OmegaPde(3)
Sq(4,4) = OmegaPde(1)
Sq(4,7) = -Xo * OmegaPde(1)
Sq(5,6) = OmegaPde(2)
Sq(6,3) = Sq(3,6)
Sq(6,5) = Sq(5,6)
Sq(7,2) = Sq(2,7)
Sq(7,4) = Sq(4,7)
Sq(7,7) = 2.0D0 * OmegaPde(5)
!

K_sigma1= MATMUL(Zc, Sq)        ! Intermediate variable
dK_sigma = MATMUL(K_sigma1, TRANSPOSE(Zc))     ! stiffness matrix point values
K_element1 = MATMUL(Zc, A)                                          ! Intermediate variable
K_element2 = MATMUL(K_element1, Ctde)        ! Intermediate variable
dK_element  = MATMUL(K_element2, TRANSPOSE(K_element1))                                        ! stiffness matrix point values
B_mat = K_element1                                   ! The B matrix
dF_in = MATMUL(B_mat,OmegaPde)                                            ! Internal force vector point values
End Subroutine dKF_LP

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

Subroutine equation (Ind, x, y, Sigma, deps, Ct, OmegaP,Check)

!______________________________________________________________________________________________________________________!
!
!   This subroutine calculates the matrices for each cross-section integration point (Lobbatto)
!   the tangent modular matrix                                           
!   and the element stresses vector                                      
!
!
! INPUT: member index (Ind), the value of Lobatto points (x and y), Element Strains Increment deps [2]
! OUTPUT: [Ct(6,6)], {OmegaP(6)}
!

Use Definitions
Implicit None

Integer, Intent(IN):: Ind
Double Precision, Intent(IN):: x, y, deps(2)
Double Precision, Intent(OUT):: Ct(6,6), OmegaP(6)
Double Precision, Intent(INOUT):: Sigma(2)
Integer, Intent(OUT):: Check
Double Precision:: G(6,2), Cmt(2,2)

! Calculating the G matrix for each value of Lobatto points (x and y) [G(x,y)] = [6X2]
Call Gmatrix (Ind, x, y, G)

! Calculating the stresses at each integration point (x,y,z) sigma [2], and the corresponding modular matrix [Cmt(2,2)]
Call BackwardEuler(deps,Sigma,Cmt,Check)
Ct = Matmul(Matmul(G,Cmt),Transpose(G))                            ! [Ct(6,6)] = [G(6,2)] [Cmt(2,2)] [G(2,6)]T
OmegaP = MatMul(G, Sigma)                                                     ! {OmegaP(6)} = [G(6,2)] {Sigma(2)}

End Subroutine equation
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Subroutine FEM(G1, G2, G3, G4, G5, G6, G7, Weight)

! Subroutine FEM
! Calculates the displacements and Buckling load of 14 a DOF non linear beam ()
! INPUTS : Cross-section Dimensions (h, b, tw, tf), Beam length (L), Material Properties (Em, Sigma, Nu), External loads (Fe)
! OUTPUTS : Load Fraction (lambda) vs. Displacement (q), Buckling load Fraction
! Subroutine Calls: Assemble, BD predictor

Use Definitions
Implicit None

Double Precision, Intent(OUT):: G1, G2, G3, G4, G5, G6, G7, Weight
Double Precision:: q(7+7*NE), qo(7+7*NE), qn(7+7*NE), dq(7+7*Ne)
Double Precision:: Kg(7+7*NE, 7+7*NE), K_sigma(1:7+7*NE, 7+7*NE), K_element(7+7*NE, 7+7*NE), Fg(7+7*NE)
DOUBLE PRECISION:: Fex(7+7*Ne)
DOUBLE PRECISION:: Tol, Lambda, dLambda, Lambda_1, Lambda_2, Lambda_3
DOUBLE PRECISION:: Max_axial, Max_q
Double Precision:: Lambda_f, Lat_f, Rot_f, Out_f, Lat_s, Rot_s, Out_s
Double Precision:: R_Sigma_WC, R_Sigma_WT, R_Sigma_FC
Double Precision:: Lambda_old(2), q_old(2,7+7*NE), qo_old(2,7+7*NE)
Double Precision:: INC_old(2), Max_stress_old(2,NE,2,3,m,m,2)
Double Precision:: Slope(2)
Integer:: NDOF, PDOF, CDOF(7+7*Ne)
Integer:: I, J, Check, flag, flag1, counter, flag2, counter1, Flag3, Flag4, Countf, flag5

Open (Unit=2,File="check.dat",Status="unknown")

! Number of degrees of freedome NDOF
NDOF=7+7*Ne
!

SectionConfig = 2 ! T-Section Beam
!

! Element Length
LE = L / NE
!

! Displacements incerement
INC =0.00005D0
!

! flag2 = 0
!

! Initializing matrices and constants
!
Fex(1:NDOF) = 0.0D0 ! Fex : external loads vector
Fg(1:NDOF) = 0.0D0 ! Fg : Internal loads vector
q(1:NDOF) = 0.0D0 ! q : Displacements vector
CDOF(1:NDOF) = 0 ! CDOF : Degrees of Freedom vector "0 = active; 1 = fixed"
dq = 0.0D0!
Lambda = 0.0D0 ! Lambda: External Loads fraction
qo = q
qn = q
q_f = 0.0D0
check = 0!
flag = 0
Flag1 = 0
Flag3 = 0
Flag4 = 0
Flag5 = 0
Counter = 1
Counter1 = 1
Countf = 0
Lambda_3= 0.0D0
Lambda_f= 0.0D0
Lambda_old=0.0D0
q_old=0.0D0
qo_old=0.0D0
qn_old=0.0D0
INC_old=0.0D0
Slope = 0.0D0
Lat_s =0.0D0; Lat_f=0.0D0 ; Rot_s=0.0D0; Rot_f=0.0D0; Out_s=0.0D0; Out_f=0.0D0;

! Specifying Constraints (Boundary Conditions)
CDOF(1) = 1        ! Simply Supported q1 (w0)
CDOF(2) = 1        ! Simply Supported q2 (u0)
CDOF(3) = 1        ! Simply Supported q3 (v0)
CDOF(4) = 1        ! Simply Supported q4 (Pphi0)
CDOF(NDOF-4) = 1  ! Simply Supported q10 (vL)
CDOF(NDOF-5) = 1  ! Simply Supported q9  (uL)

! Controlling displacement
PDOF = 7 * NE /2 + 4                         ! Mid-Span Axial Rotation

! External load magnitudes
! External loads
Do J = 3, NDOF, 7
    Fex(J) = - q_load * LE * FS           ! Uniformly distributed load
    Fex(J+1) = - q_load * LE * Ecc  * FS * FS                      ! Uniformly Distributed Torque
End do
Fex(5) = - q_load / 12.0D0 * LE**2 * FS                     ! Moment at the lift end of the first element
Fex(NDOF-2) = q_load / 12.0D0 * LE**2 * FS                                     ! Moment at the right end of the last element
Fex(NDOF-6) = -P_load * FS              ! Axial load at the end of the beam

! The predictor loop
Do
10 Call Converged_step(INC,Lambda,q,qo,qn,flag2)
   Tol = 1.0D0                 ! Tol : Tolerance
   I = 0
   ! Calling the Assemble subroutine :
   ! INPUTS : q (Nodal displacements)
   ! OUTPUTS: Kg (Global Stiffness matrix)
   ! K_sigma , K_element
   ! Fg (internal force vector)
   ! Call AssembleP (flag1,qo, qn, Kg, K_sigma, K_element, Fg,Check)
   !
   ! Fixing the controlling displacement
   CDOF(PDOF) = 1
   !
   ! Calling the solver to determine the displacements increment and the resulting loads step
   ! INPUTS: NDOF (Number of degrees of freedom), CDOF(fixed DOF),PDOF (displacement control DOF)
   ! Fex(external loads vector), INC (displacement initial increment)
   ! INOUT : Kg(global stiffness matrix/factored),Fg(internal loads vector/unbalanced displacements),Lambda (loads step)
   ! OUTPUT: dq(displacements increment)
   ! CALL BDPredictor(NDOF,CDOF,PDOF,Kg,Fex,dq,lambda,INC,flag)
   ! If(flag ==1) Goto 100
   ! Updating the Displacements Fg (unbalanced displacements)
   q = q + Fg
   qo = q
   !
   ! Iterating to balance the external and internal forces
Do while (Tol > 1.0D-4)
I = I + 1
! Calling the Global stiffness matrix and the internal force vector
Call AssembleP (flag1,qo, qn, Kg, K_sigma, K_element, Fg,Check)
If (Check ==1) GOTO 100
Lambda_2 = Lambda
! Calling the solver to determine the displacements increment and the resulting loads step
Call BDPredictor(NDOF,CDOF,PDOF,Kg,Fex,Fg,dq,Lambda,0.0D0,flag)
If (flag ==1) Goto 100
! Updating the displacement
dq = Fg
q = q + Fg
qn = q
! Checking Tolerance
Tol=DABS((Lambda-Lambda_2)/(Lambda+1.0D-16))
! Limiting the number of iterations
If (I > 5) then
! write(*,*) 'Have exceeded 5 Iterations, decreasing increment !!!'
If (flag2 == 1) then
flag2=2
Call Converged_step(INC,Lambda,q,qo,qn,flag2)
Goto 10
Else
Call Converged_step(INC,Lambda,q,qo,qn,flag2)
Goto 5
Endif
End if
End do
!________________________________________________________________________________________________________________________
If(DABS(Lambda) .NE. 0.0D0) then
flag2=1
Call Converged_step(INC,Lambda,q,qo,qn,flag2)
Endif
Lambda_f= Lambda; Lat_f= -q(PDOF-1); Rot_f= -q(PDOF); Out_f= -q(PDOF-2)
! To Calculate the maximum displacement at either the maximum load if Lambda < 1.0
! or the maximum displacement at Lambda = 1.0
If (Counter == 1) then
Lat_s = Lat_f ; Rot_s= Rot_f; Out_s=Out_f
If (Lambda_f >= 1.0) then
Lat_s = Lat_f ; Rot_s= Rot_f; Out_s=Out_f
Counter = 2
End If
Endif
If (Lambda < Lambda_3) GOTO 100                   ! If the solution reached its peek
Lambda_3 =Lambda
q_f= qn
qo = qn
Counter1 = Counter1 +1
!____________________________________________________________________________________________________________________
!  Curvature Step Control
Lambda_old(1)= Lambda_old(2)
q_old(1,:)= q_old(2,:)
qo_old(1,:)= qo_old(2,:)
qn_old(1,:)= qn_old(2,:)
INC_old(1)= INC_old(2)

End
Lambda_old(2) = Lambda
q_old(2,:) = q
qo_old(2,:) = qo
qn_old(2,:) = qn
INC_old(2) = INC

Slope(1) = Slope(2)
Slope(2) = (Lambda_old(2)/q_old(2,pdof))/(Lambda_old(1)/q_old(1,pdof))

If ((DABS((Lambda_old(2)/q_old(2,pdof))/(Lambda_old(1)/q_old(1,pdof))) < 1.0D0) .and. (flag3 ==0)) then
   INC=INC*0.5
   flag3=1
Endif

If ((DABS((Lambda_old(2)/q_old(2,pdof))/(Lambda_old(1)/q_old(1,pdof))) < 0.9999D0) .and. (flag3 ==1)) then
   INC=INC*0.5
   flag3=2
Endif

If ((DABS((Lambda_old(2)/q_old(2,pdof))/(Lambda_old(1)/q_old(1,pdof))) <= 0.999D0) .and. (flag3 ==2)) then
   INC=INC*0.5
   flag3=3
Endif

If ((DABS((Lambda_old(2)/q_old(2,pdof))/(Lambda_old(1)/q_old(1,pdof))) <= 0.998D0) .and. (flag3 ==3)) then
   INC=INC*0.5
   flag3=4
Endif

If ((DABS((Lambda_old(2)/q_old(2,pdof))/(Lambda_old(1)/q_old(1,pdof))) <= 0.998D0) .and. (flag3 ==4)) then
   INC=INC*0.5
   flag3=5
Endif

If ((DABS((Lambda_old(2)/q_old(2,pdof))/(Lambda_old(1)/q_old(1,pdof))) <= 0.998D0) .and. (flag3 ==5)) then
   INC=INC*0.6
   flag3=6
Endif

If ((Slope(2)/Slope(1) < 1.0D0) .and. (flag4 ==0)) then
   INC=INC*0.05
   Countf = Counter1
   flag4=1
Endif

If ((flag4 ==1) .and. (Counter1> Countf+60)) then
   INC=INC* 10.0D0
   flag4=2
   flag3=0
Endif

End Do

If((Counter1 < 250) .and. (flag5==0)) then
   Counter1=1
   flag5=1
   INC=-0.00001
   Goto 5
Endif

100  write(*,*) 'No. of Inc=',Counter1
Max_q = Lambda_f* Q_load *FS

! Local Buckling Calculations
Call Local_Buckling(Max_axial, Max_q, R_Sigma_WC, R_Sigma_WT, R_Sigma_FC)
G1 = \( \Lambda_f - 1.0D0 \)
G2 = \( (0.025D0/Lat_s)^2 - 1.0D0 \)
G3 = \( (0.05D0/Rot_s)^2 - 1.0D0 \)
G4 = \( (0.00125D0/Out_s)^2 - 1.0D0 \)
G5 = \( \Lambda_f - R_{\Sigma_WC} \)
G6 = \( \Lambda_f - R_{\Sigma_WT} \)
G7 = \( \Lambda_f - R_{\Sigma_FC} \)

Weight = \( 2800.0D0 \times L \times (h*tw+b*tf) \)

End Subroutine FEM

Subroutine Gmatrix

This subroutine calculates the G matrix for each value of Lobatto points (x and y) \([G(x,y)] = [6 \times 2]\)

Input:
- Member index \( \text{Ind} \)
- Value of Lobatto points \( x, y \)

Output:
- \([ G(6,2) ]\)

End Subroutine

Subroutine KF_integP

This subroutine calculates the plastic element stiffness matrix and the internal forces vector

Input: Old and incremented element displacements vector \( (Uo, Un) \)

Output: The element stiffness matrix \( K_{\text{int}} \), the internal force vector \( F_{\text{int}} \)

End Subroutine
Integer:: Gauss

! Gauss integration 2 points

! Calling the dKF_L subroutine to calculate the stiffness and internal force properties at the cross-section
! INPUTS : Uo and Un (the element displacements), z (axial location)
! OUTPUTS : dK_sigma (element stiffness point properties), dK_element, dF_in (element internal force point properties)

Gauss = 1
z = ((1.0D0 / DSQRT(3.0D0) + 1.0D0) / 2.0D0) * LE       ! Z axial distance along the element
Call dKF_LP(SigmaBeamOld, SigmaBeamNew, Element, Gauss, Uo, Un, z, dK_sigma, dK_element, dF_in,Check)

Gauss = 2
z = ((-1.0D0 / DSQRT(3.0D0) + 1.0D0) / 2.0D0) * LE
Call dKF_LP(SigmaBeamOld, SigmaBeamNew, Element, Gauss, Uo, Un, z, dK_sigma, dK_element, dF_in,Check)

k_sigma = 0.5D0 * (dK_sigmaL1 + dK_sigmaL2) * LE
k_element = 0.5D0 * (dK_elementL1 + dK_elementL2) * LE

K_int = K_sigma + K_element

End Subroutine KF_integP

Subroutine Lobatto_intP(SigmaBeamOld, SigmaBeamNew, Element, Gauss, Ind, a1, b1, c1, d1, GammaO, dGamma, Ct_int, OmegaP_int,Check)

! This Subroutine performs the Lobatto integration for each member of the cross-section and calculates the following matrices:
! the tangent modular matrix  [Ct_int(E,G,Sigma,Tau,x,y)] = [6X6],
! and the element stresses vector  [OmegaP_int(Tau,Sigma,x,y)] = [6]
! INPUT: member index (Ind), number of Lobatto integration points per member, limits of integration of each member (a,b1,c,d),
! element strains {GammaO(6)}, element strains increment {dGamma(6)}
! OUTPUT: Ct_int(6,6), G_int(6,2), OmegaP_int(6)

Use Definitions
Implicit None
Integer, Intent(IN):: Element, Gauss, Ind
Double Precision, Intent(IN) :: a1, b1, c1, d1, GammaO(1:6), dGamma(1:6), SigmaBeamOld(NE,2,3,m,m,2)
Double Precision, Intent(OUT):: Ct_int(6,6), OmegaP_int(6)
Integer, Intent(OUT):: Check
Double Precision, Intent(OUT):: SigmaBeamNew(NE,2,3,m,m,2)

Double Precision:: r(1:10,1:10), w(1:10,1:10)
Double Precision:: JCt(6,6), JXCt(6,6), JOmegaP(6), JXOmegaP(6)
Double Precision:: deps(2), Ct(6,6), OmegaP(6)
Double Precision:: h1, h2, k1, k2
Double Precision:: x, y, Cm(2,2)
Integer :: I, t

Cm = 0.0D0; Cm(1,1) = Em; Cm(2,2) = Gm       ! Elastic Modular Matrix

! Initializing
Ct = 0.0D0
OmegaP = 0.0D0
JCt = 0.0D0
JOMegaP = 0.0D0
JXCT = 0.0D0
JXOmegaP = 0.0D0
Sigma= 0.0D0

! Transforming the limits of integration
h1 = (b1 - a1) / 2.0D0
h2 = (b1 + a1) / 2.0D0
k1 = (d1 - c1) / 2.0D0
k2 = (d1 + c1) / 2.0D0

! Getting the integration roots and weights
Call Lobatto(r, w)

! For all the integration points m x m
Do l = 1, m
   JXCl = 0.0D0
   JXOmegaP = 0.0D0
   x = h1 * r(m, l) + h2
   Do t = 1, m
      y = k1 * r(m, t) + k2
      Call Gmatrix (Ind, x, y, G)
   Sigma(1) = SigmaBeamOld(Element, Gauss, Ind, l, t, 1) ! Old Stresses
   Sigma(2) = SigmaBeamOld(Element, Gauss, Ind, l, t, 2)
   If( (x == 0.0D0) .and. (y == 0.0D0) ) G = 0.0D0 ! Zero [G] where no material exists
   deps = MatMul(dGamma, G) ! Strains increment
   ! Calling the equations
   Call equation (Ind, x, y, Sigma, deps, Ct, OmegaP, check)
   SigmaBeamNew(Element, Gauss, Ind, l, t, 1) = Sigma(1) ! Updated Stresses
   SigmaBeamNew(Element, Gauss, Ind, l, t, 2) = Sigma(2)
   JXCl = JXCl + w(m, l) * Ct
   JXOmegaP = JXOmegaP + w(m, l) * OmegaP
   End Do
   JCt = JCt + w(m, l) * JXCl
   JOmegaP = JOmegaP + w(m, l) * k1 * JXOmegaP
   JXOmegaP = k1 * JXOmegaP
   End Do
   JCt = h1 * JCt
   JOmegaP = h1 * JOmegaP
   Ct_int = JCt
   OmegaP_int = JOmegaP

End Subroutine Lobatto_intP

Subroutine Lobatto(r, w)
!
! This subroutine stores Lobatto roots and weights
! OUTPUT: r(n,i), w(n,i)
!
Implicit none
Double Precision, Intent(OUT):: r(1:10,1:10), w(1:10,1:10)

! Zeroing out the unneeded elements
r(1:10,1:10) = 0.0D0
w(1:10,1:10) = 0.0D0

! Lobatto roots and weights
r(3,1) = -1.0D0
r(3,2) = 0.0D0
r(3,3) = 1.0D0
w(3,1) = 0.3333333333333D0
\[
\begin{array}{l}
w(3,2) = 1.3333333333D0 \\
w(3,3) = 0.3333333333D0 \\
n(4,1) = -1.0D0 \\
n(4,2) = -0.44721360D0 \\
n(4,3) =  0.44721360D0 \\
n(4,4) =  1.D0 \\
w(4,1) = 0.1666666667D0 \\
w(4,2) = 0.8333333333D0 \\
w(4,3) = 0.8333333333D0 \\
w(4,4) = 0.1666666667D0 \\
n(5,1) = -1.0D0 \\
n(5,2) = -0.65465367D0 \\
n(5,3) =  0.0D0 \\
n(5,4) =  0.65465367D0 \\
n(5,5) =  1.0D0 \\
w(5,1) = 0.1D0 \\
w(5,2) = 0.5444444444D0 \\
w(5,3) = 0.7111111111D0 \\
w(5,4) = 0.5444444444D0 \\
w(5,5) = 0.1D0 \\
n(6,1) = -1.0D0 \\
n(6,2) = -0.76505532D0 \\
n(6,3) = -0.28523152D0 \\
n(6,4) =  0.28523152D0 \\
n(6,5) =  0.76505532D0 \\
n(6,6) =  1.0D0 \\
w(6,1) = 0.06666667D0 \\
w(6,2) = 0.3784796D0 \\
w(6,3) = 0.55485838D0 \\
w(6,4) = 0.55485838D0 \\
w(6,5) = 0.3784796D0 \\
w(6,6) = 0.06666667D0 \\
n(7,1) = -1.0D0 \\
n(7,2) = -0.83023390D0 \\
n(7,3) = -0.46884879D0 \\
n(7,4) =  0.0D0 \\
n(7,5) =  0.46884879D0 \\
n(7,6) =  0.83023390D0 \\
n(7,7) =  1.0D0 \\
w(7,1) = 0.04761904D0 \\
w(7,2) = 0.27682604D0 \\
w(7,3) = 0.43174538D0 \\
w(7,4) = 0.48761904D0 \\
w(7,5) = 0.43174538D0 \\
w(7,6) = 0.27682604D0 \\
w(7,7) = 0.04761904D0 \\
n(8,1) = -1.0D0 \\
n(8,2) = -0.87174015D0 \\
n(8,3) = -0.59170018D0 \\
n(8,4) = -0.20929922D0 \\
n(8,5) =  0.20929922D0 \\
n(8,6) =  0.59170018D0 \\
n(8,7) =  0.87174015D0 \\
n(8,8) =  1.0D0 \\
w(8,1) = 0.03571428D0 \\
\end{array}
\]
w(8,2) = 0.21070422D0
w(8,3) = 0.34112270D0
w(8,4) = 0.41245880D0
w(8,5) = 0.41245880D0
w(8,6) = 0.34112270D0
w(8,7) = 0.21070422D0
w(8,8) = 0.03571428D0

r(9,1) = -1.0D0
r(9,2) = -0.8997579954D0
r(9,3) = -0.6771862795D0
r(9,4) = -0.3631174638D0
r(9,5) =  0.0D0
r(9,6) =  0.3631174638D0
r(9,7) =  0.6771862795D0
r(9,8) =  0.8997579954D0
r(9,9) =  1.0D0

w(9,1) = 0.0277777778D0
w(9,2) = 0.1654953616D0
w(9,3) = 0.274387126D0
w(9,4) = 0.3464285110D0
w(9,5) = 0.3715192744D0
w(9,6) = 0.3464285110D0
w(9,7) = 0.274387126D0
w(9,8) = 0.1654953616D0
w(9,9) = 0.0277777778D0

r(10,1) = -1.0D0
r(10,2) = -0.9195339082D0
r(10,3) = -0.7387738651D0
r(10,4) = -0.4779249498D0
r(10,5) = -0.1652789577D0
r(10,6) =  0.1652789577D0
r(10,7) =  0.4779249498D0
r(10,8) =  0.7387738651D0
r(10,9) =  0.9195339082D0
r(10,10) =  1.0D0

w(10,1) = 0.0222222222D0
w(10,2) = 0.1333059908D0
w(10,3) = 0.2248894320D0
w(10,4) = 0.2920426836D0
w(10,5) = 0.3275397612D0
w(10,6) = 0.3275397612D0
w(10,7) = 0.2920426836D0
w(10,8) = 0.2248894320D0
w(10,9) = 0.1333059908D0
w(10,10) = 0.0222222222D0

End Subroutine lobatto

Subroutine Local_Buckling(Max_axial, Max_q, R_Sigma_WC, R_Sigma_WT, R_Sigma_FC)
!______________________________________________________________________________________________________________________
!
!  To Calculate the Local Buckling Constraints
!______________________________________________________________________________________________________________________

Use Definitions Implicit None
Double Precision, Intent(IN):: Max_axial, Max_q
Double Precision, Intent(OUT):: R_Sigma_WC, R_Sigma_WT, R_Sigma_FC
Double Precision:: A, Ys, Is, Jw
Double Precision:: Dw, Sigma_wc, Sigma_wt, Df, Sigma_fc, PI
Double Precision:: Ip, SS
Double Precision:: Sigma_wc1, Sigma_wt1, Sigma_fc1, delta1wc, delta2wc, delta1wt, delta2wt, delta1fc, delta2fc

End Subroutine Local_Buckling
PI = 3.14159D0
A = tf * h * tw  ! Area
Ys = (b * (h + tf / 2.0) * tf + (h**2 * tw) / 2) / A ! Centeroid
Is = (b * tf**4 / 3.0) / tw**3 * (1.0 - nu**2) ! Moment of Inertia
Jw = h * tw**3 / 3.0D0 ! Web Torsional Rigidity

Dw = (Em * tw**3) / (12.0D0 * (1.0D0 - nu**2)) ! Web Plate Buckling Constant

Sigma_wc = 4.5D0 * ((PI / h)**2) * (Dw / tw) ! Critical Stress Web Compression Local Buckling

Ip = (h**3 * tw) / 3.0 + (h * tw**3) / 12.0
SS = h**3 * tw * (h / 4.0 - 1.0 / 3.0 * Ys)
Sigma_wt = -(Gm * Jw) / Ip + (Max_q) / 12.0 * (L**2 / Is) * (SS / Ip) * (1.0 - 3.0 / (4.0 * PI**2)) ! Critical Stress Web Tripping

Df = (Em * tf**3) / (12.0D0 * (1.0D0 - nu**2)) ! Flange Plate Buckling Constant

Sigma_fc = 6.25D0 * ((PI / b)**2) * (Df / tf) ! Critical Stress Flange Compression Local Buckling

! Knocking-Down Factors (to reduce the Buckling Stress under the Yield Strength

delta1wc = Sigma_wc / Sigma_Y
delta2wc = 1.0D0 / DABS(delta1wc) - 1.0D0 / (4.0D0 * delta1wc**2)
delta1wt = Sigma_wt / Sigma_Y
delta2wt = 1.0D0 / DABS(delta1wt) - 1.0D0 / (4.0D0 * delta1wt**2)
delta1fc = Sigma_fc / Sigma_Y
delta2fc = 1.0D0 / DABS(delta1fc) - 1.0D0 / (4.0D0 * delta1fc**2)

! Modified Buckling Strengths
Sigma_wc1 = DABS(Sigma_wc * delta2wc)
Sigma_wt1 = DABS(Sigma_wt * delta2wt)
Sigma_fc1 = DABS(Sigma_fc * delta2fc)
R_Sigma_WC = DABS(400.0D6 / Sigma_wc1)
R_Sigma_WT = DABS(400.0D6 / Sigma_wt1)
R_Sigma_FC = DABS(400.0D6 / Sigma_fc1)

End Subroutine

Subroutine qz_Matrices (U, z, Gamma, A, Zc)

! This subroutine generates matrices that depend only on the element displacements q and the axial location z
! INPUT: The element displacement vector (U), axial dimension (z)
! OUTPUT: The nodal strains vector (Gamma), a strain-disp matrix (A), an interpolation function matrix (Zc)

Use Definitions
Implicit none

Double Precision, Intent(IN):: U(14), z
Double Precision, Intent(OUT):: Gamma(6), A(8,6), Zc(14,8)

Double Precision:: wd, ud, udd, vd, vdd, phi, phid, phidd
Double Precision:: wo, wL
Double Precision:: uu, uL, udo, udL
Double Precision:: vo, vL, vdo, vddL
Double Precision:: phi, phiL, phiLd, phiLD
Double Precision:: N1, N1d, N2, N2d
Double Precision:: H1, H1d, H2, H2d, H2dd, H3, H3d, H3dd, H4, H4d, H4dd
Double Precision:: Xo, Yo, a1(3), b1(3), c(3), d(3)

! Initializing matrices
A = 0.0D0
Zc=0.0D0

! Assigning interpolation points to nodal displacement vector
wo = U(1)
wL = U(8)
uo = U(2)
uL = U(9)
vo = U(3)
vL = U(10)
phio = U(4)
phiL = U(11)
udo = U(5)
udL = U(12)
vdo = U(6)
vdl = U(13)
phido = U(7)
phidL = U(14)

! The interpolation functions and derivatives
N1 = (1.0D0 - z / LE)
N1d = -1.0D0 / LE
N2 = z / LE
N2d = 1.0D0 / LE

H1 = 1.0D0 - 3.0D0 * z**2 / LE**2 + 2.0D0 * z**3 / LE**3
H1d = -6.0D0 * z / LE**2 + 6.0D0 * z**2 / LE**3
H1dd = -6.0D0 / LE**2 + 12.0D0 * z / LE**3

H2 = z - 2.0D0 * z**2 / LE + z**3 / LE**2
H2d = (1.0D0 - (4.0D0 * z) / LE + (3.0D0 * z**2) / LE**2)
H2dd = (-4.0D0 / LE + (6.0D0 * z) / LE**2)

H3 = 3.0D0 * z**2 / LE**2 - 2.0D0 * z**3 / LE**3
H3d = ((6.0D0 * z) / LE**2 - (6.0D0 * z**2) / LE**3)
H3dd = (6.0D0 / LE**2 - (12.0D0 * z) / LE**3)

H4 = - z**2 / LE + z**3 / LE**2
H4d = ((-2.0D0 * z) / LE + (3.0D0 * z**2) / LE**2)
H4dd = (-2.0D0 / LE + (6.0D0 * z) / LE**2)

! Assigning the displacement derivatives
wd = N1d * wo + N2d * wL
ud = uo * H1d + udo * H2d + uL * H3d + udL * H4d
udd = uo * H1dd + udo * H2dd + uL * H3dd + udL * H4dd

vd = vo * H1d + vdo * H2d + vL * H3d + vdl * H4d
vdd = vo * H1dd + vdo * H2dd + vL * H3dd + vdl * H4dd

phi = phio * H1 + phido * H2 + phiL * H3 + phidL * H4
phid = phio * H1d + phido * H2d + phiL * H3d + phidL * H4d
phidd = phio * H1dd + phido * H2dd + phiL * H3dd + phidL * H4dd

!
! Writing the expressions for Gamma
Call SectionProperties1(Xo, Yo, a1, b1, c, d)

\[
\begin{align*}
\Gamma(1) &= w_d + 0.5D_0 \times (u_d^2 + v_d^2) - X_o \times v_d \times \phi d + Y_o \times (u_d \times \phi d) \\
\Gamma(2) &= u_d d + v_d d \times \phi \\
\Gamma(3) &= v_d d - u_d d \times \phi \\
\Gamma(4) &= \phi d d \\
\Gamma(5) &= \phi d \times \phi d \\
\Gamma(6) &= \phi d
\end{align*}
\]

! The A matrix

\[
\begin{align*}
A(1,1) &= 1.0D_0 \\
A(2,1) &= u_d + y_o \times \phi d \\
A(3,2) &= 1.0D_0 \\
A(3,3) &= - \phi \\
A(4,1) &= v_d - x_o \times \phi d \\
A(5,2) &= \phi \\
A(5,3) &= 1.0D_0 \\
A(6,2) &= v_d d \\
A(6,3) &= - u_d d \\
A(7,1) &= y_o \times u_d - x_o \times v_d \\
A(7,5) &= 2.0D_0 \times \phi d \\
A(7,6) &= 1.0D_0 \\
A(8,4) &= 1.0D_0
\end{align*}
\]

! The Zc matrix

\[
\begin{align*}
Z_c(1,1) &= N_1d \\
Z_c(2,2) &= H_1d \\
Z_c(5,2) &= H_2d \\
Z_c(9,2) &= H_3d \\
Z_c(12,2) &= H_4d \\
Z_c(2,3) &= H_1dd \\
Z_c(5,3) &= H_2dd \\
Z_c(9,3) &= H_3dd \\
Z_c(12,3) &= H_4dd \\
Z_c(3,4) &= H_1d \\
Z_c(6,4) &= H_2d \\
Z_c(10,4) &= H_3d \\
Z_c(13,4) &= H_4d \\
Z_c(3,5) &= H_1dd \\
Z_c(6,5) &= H_2dd \\
Z_c(10,5) &= H_3dd \\
Z_c(13,5) &= H_4dd \\
Z_c(4,6) &= H_1 \\
Z_c(7,6) &= H_2 \\
Z_c(11,6) &= H_3 \\
Z_c(14,6) &= H_4 \\
Z_c(4,7) &= H_1d \\
Z_c(7,7) &= H_2d \\
Z_c(11,7) &= H_3d \\
Z_c(14,7) &= H_4d \\
Z_c(4,8) &= H_1dd \\
Z_c(7,8) &= H_2dd \\
Z_c(11,8) &= H_3dd \\
Z_c(14,8) &= H_4dd
\end{align*}
\]
Subroutine SectionP (SigmaBeamOld, SigmaBeamNew, Element, Gauss, GammaO, dGamma, Ctde, OmegaPde, Check)
!
! To calculate the cross-section matrices, using m X m Lobatto numerical integration
! tangent Plastic modular matrix [Ctde(E,G,Sigma,Tau,x,y)] = [6X6]
! and the element stresses vector {OmegaPt(Tau,Sigma,x,y)} = [6]
!
! INPUT:  The old element Strains {GammaO(6)} , the element strains increment {dGamma(6)}
! OUTPUT: [Ctde(E,G,Sigma,Tau,x,y)] = [6X6], and the element stresses vector {OmegaPt(Tau,Sigma,x,y)} = [6]
! Function Calls: Lobatto_int
!
Use Definitions
Implicit None

Integer, Intent(IN):: Element, Gauss
Double Precision, Intent(IN):: GammaO(6), dGamma(6)
Integer, Intent(OUT):: Check
Double Precision, Intent(IN):: SigmaBeamOld(NE,2,3,m,m,2)
Double Precision, Intent(OUT):: SigmaBeamNew(NE,2,3,m,m,2)
Double Precision, Intent(INOUT):: Ctde(6,6), OmegaPde(6)

Double Precision:: Xo, Yo, a1(3), b1(3), c1(3), d1(3)
Double Precision:: Sigmab_egi(m,m,2)
Double Precision:: Ct_int(6,6), OmegaP_int(6)
Double Precision:: Ct_member_1(6,6), OmegaP_member_1(6)
Double Precision:: Ct_member_2(6,6), OmegaP_member_2(6)
Double Precision:: Ct_member_3(6,6), OmegaP_member_3(6), OmegaPde1(6)

Integer:: Ind, I,J

Call SectionProperties1(Xo, Yo, a1, b1, c1, d1) ! Cross-section constants
! For the Lower flange
Ind = 1
Call Lobatto_intP (SigmaBeamOld, SigmaBeamNew, Element, Gauss, Ind, a1(1), b1(1), c1(1), d1(1),GammaO, dGamma, Ct_int, OmegaP_int, Check)
Ct_member_1 = Ct_int
OmegaP_member_1 = OmegaP_int

! For the web
Ind = 2
Call Lobatto_intP (SigmaBeamOld, SigmaBeamNew, Element, Gauss, Ind, a1(2), b1(2), c1(2), d1(2),GammaO, dGamma, Ct_int, OmegaP_int, Check)
Ct_member_2 = Ct_int
OmegaP_member_2 = OmegaP_int

! For Upper the flange
Ind = 3
Call Lobatto_intP (SigmaBeamOld, SigmaBeamNew, Element, Gauss, Ind, a1(3), b1(3), c1(3), d1(3),GammaO, dGamma, Ct_int, OmegaP_int, Check)
Ct_member_3 = Ct_int
OmegaP_member_3 = OmegaP_int

! For the whole section
Ctde = Ct_member_1 + Ct_member_2 + Ct_member_3
OmegaPde = OmegaP_member_1 + OmegaP_member_2 + OmegaP_member_3

! Zeroing out the numerical integration errors
Do I=1,6
Do J=1,6
If(Ctde(I,J) < 1.0D-2) Ctde(I,J) = 0.0D0
End do
End Do
Do i=1,6
IF (DABS(OmegaPde1(i)) < 1.0D-12) OmegaPde1(i) = 0.0D0
End do

OmegaPde1 = OmegaPde

End subroutine SectionP

Subroutine SectionProperties1(Xo, Yo, a1, b1, c1, d1)

! This Subroutine provides the following cross-section properties: (Xo, Yo) the shear center location,
! a, b1 the integration limits in the Y direction
! c, d the integration limits in the X direction
!
! INPUT: Cross-section Dimensions (h, tw, b, tf): From the main Program
!
! OUTPUT: Xo, Yo, a1(3), b1(3), c1(3), d1(3)

Use Definitions
Implicit None
Double Precision, Intent(OUT):: Xo, Yo, a1(3), b1(3), c1(3), d1(3)
Double Precision:: Area, Ys

If (SectionConfig == 1) Then ! SectionConfig = 1 ==> I-section
! Geometric Properties
Area = h * tw + 2.0D0 * b * tf ! Section Area
Ys = tf + h / 2.0D0 ! Centroid
Xo = 0.0D0 ! Shear Center (measured from the Centroid)
Yo = 0.0D0 ! Shear Center (measured from the Centroid)

! Integration Limits (origin at the centroid)  aSb cSd dxdy
a1(1) = -b / 2.0D0
b1(1) = b / 2.0D0
!
! Lower flange (sec = 1)
c1(1) = - Ys
d1(1) = - Ys + tf
If (c1(1) == d1(1)) then
c1(1) = 0.0D0
d1(1) = 0.0D0
End if

a1(2) = -tw / 2.0D0
b1(2) = tw / 2.0D0
!
! Web (sec = 2)
c1(2) = -Ys + tf
d1(2) = Ys - tf
If (c1(2) == d1(2)) then
c1(2) = 0.0D0
d1(2) = 0.0D0
End if

a1(3) = -b / 2.0D0
b1(3) = b / 2.0D0
!
! Upper flange (sec = 3)
c1(3) = Ys - tf
d1(3) = Ys
If (c1(3) == d1(3)) then
c1(3) = 0.0D0
d1(3) = 0.0D0
End if

End subroutine SectionProperties1(Xo, Yo, a1, b1, c1, d1)
End if
Else

! Geometric Properties
! SectionConfig = 2 ==> T-section

! Area = tf * b + h * tw
Ys = (b * tf * (h + tf / 2.0D0) + h * tw * (h / 2.0D0)) / Area
Xo = 0.0D0
Yo = h + tf / 2.0D0 - Ys

! Integration Limits (origin at the centroid) aSb cSd dx dy

End If
End Subroutine

Subroutine SectionProperties2(x, y, n, s, omegaW)

! This Subroutine provides the warping properties of each member of the cross-section
!
! (n, s) the thickness and Contour coordinates
!
! omegaW the Warping function
!
! INPUT: The Lobatto integration Points (x,y)
! OUTPUT: n, s, omegaW
!

Use Definitions
Implicit None
Double Precision, Intent (IN):: x, y
Double Precision, Intent(OUT):: n(3), s(3), omegaW(3)
Double Precision:: Area, Ys

If (SectionConfig == 1) Then

! Geometric Properties
! SectionConfig = 1 ==> I-section

! Area = h * tw + 2.0D0 * b * tf
Ys = tf + h / 2.0D0

! Section Warping Parameters
n(1) = y + (h + tf) / 2.0D0
s(1) = x
omegaW(1) = - (h + tf) / 2.0D0 * x
Else

  ! Geometric Properties
  Area = t_f * b + h * t_w
  ! Section Area
  Ys = (b * t_f * (h + t_f / 2.0D0) + h * t_w * (h / 2.0D0)) / Area  ! Centroid

  ! Section Warping Parameters
  n(1) = 0.0D0
  s(1) = 0.0D0
  omegaW(1) = 0.0D0

  n(2) = x
  s(2) = y + (h / 2.0D0) - Ys
  omegaW(2) = 0.0D0

  n(3) = y - (h + (t_f / 2.0D0) - Ys )
  s(3) = - x
  omegaW(3) = 0.0D0

EndIf

End Subroutine
Vita

Mazen A. Ba-abbad was born in Cairo, Egypt on July 1970. He earned his B.S. degree in Mechanical Engineering from King Fahd University of Petroleum and Minerals in Dhahran Saudi Arabia in 1993. After graduation he worked as an airframe structural engineer in King Abdul Aziz Air Base until 1995. Then in 1996, he joined the Mechanical Engineering department at the Lamar University in Beaumont Texas, where he earned his M.S. in Mechanical Engineering in 1998.