NOTE TO USERS

This reproduction is the best copy available.

UMI®
RICE UNIVERSITY

An Optimization Algorithm for Minimum Weight Design of Steel Frames with Nonsmooth Stress Constraints

by

Steven M. Wilkerson

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE

Doctor of Philosophy

APPROVED, THESIS COMMITTEE:

Dr. Satish Nagarajaiah, Chair
Associate Professor
Civil and Environmental Engineering
Mechanical Engineering and Material Science

Dr. Matthias Heinkenschloss
Associate Professor
Computational and Applied Mathematics

Dr. Ahmad Durrani
Professor
Civil and Environmental Engineering

Dr. Jasbir Arora
F. Wendell Miller Distinguished Professor
Civil and Environmental Engineering
The University of Iowa

HOUSTON, TEXAS

APRIL 2005
INFORMATION TO USERS

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleed-through, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.
Abstract

An Optimization Algorithm for Minimum Weight Design of Steel Frames with Nonsmooth Stress Constraints

by

Steven M. Wilkerson

A new algorithm is presented for the solution of structural optimization problems in which the stress constraints are nonsmooth. The allowable stresses of structural members may be governed by one of three types of behavior: yielding, inelastic buckling, or elastic buckling. Consequently, the strength of members is defined by a piecewise function that depends on the cross-section and other design parameters. Some of these allowable stress functions are discontinuous, while some are continuous but nonsmooth. The allowable stress functions are sometimes defined by nonsmooth envelope functions, wherein the strength is determined by the controlling failure mechanism. Absolute values of stresses are compared to positive allowable stresses for simplicity, and the absolute value function is nonsmooth at zero.

Optimization of structural members with such nonsmooth constraint functions is limited, because derivative-based algorithms assume that the objective and constraint functions are smooth. Typical approaches in current practice are to oversimplify the constraints, use slower, derivative free methods, apply ad-hoc solutions, or ignore the problem altogether. In the approach taken here, the causes of the nonsmooth constraints in a typical design code are systematically identified and replaced with nearly
equivalent alternatives so that the problem can be solved using readily available and powerful derivative-based optimization methods. Theoretical models, finite element models, and experimental data are used as benchmarks to predict the behavior. These are used to fit an appropriate set of curves for use in design.

The new optimization algorithm presented uses a combination of a continuation method, then a judicious choice of added "secondary constraints" to transform the original nonsmooth problem to an equivalent smooth one. First, a solution is obtained for a smooth approximation of the original problem. It is used as a starting value to successively solve more and more nonsmooth, but closer approximations until a reasonably close solution to the original problem is determined. This solution is used to constrain the variables governing each of the piecewise defined functions. The original problem is thus transformed to a smooth problem with the added secondary constraints, and is solved using a standard derivative-based optimization method.
Acknowledgements

First I must acknowledge the sacrifice of my family to spare me during these years of study. My wife Janet Greenberg, and children Walt, Joe, and Elaine have contributed with their support, and encouragement. Janet has listened to hours of discussion about this idea, and that, and has provided helpful advice throughout. Janet also proofread this thesis, which is more help than anyone can ask.

The members of my committee deserve individual recognition. I would like to thank my advisor, Dr. Satish Nagarajaiah who has provided constant and valuable support and encouragement throughout my time at Rice. I would also like to thank him for his timely advice in completion of this thesis, and particularly for introducing me to Dr. Jasbir Arora. Dr. Michael Terk, who was my first adviser, got me on track and was always there with advice. Dr. Matthias Heinkenschloss deserves my thanks for showing an interest in what I was doing since our first meeting in 1998. Without the interest of these three, I would not have developed my thesis topic. Dr. Ahmad Durrani was my first point of contact upon applying for study at Rice in 2000. My thanks to him for his encouragement and support. Dr. Jasbir Arora should also be thanked for stepping in to provide his unique input as a member of the structural optimization community.

I would also like to mention Dr. Shankar Bhat, my point of contact at Shell International E&P. A portion of this research was funded by a grant from Shell International, which is gratefully acknowledged.

I would next like to thank a few outstanding professors whose excellent courses have given me the background I needed to pursue this study. Dr. Angelo Miele, who has always shown an interest in my work, taught my first course in constrained optimization. Dr. Richard Tapia was my instructor for two invaluable courses, and he has been helpful in giving direction and recommending papers for my use. I
credit these two professors with helping to start the most important part of my time here. Dr. John Dennis was my first professor in applied numerical methods, and he helped me explore the world of pattern search methods. I credit his first course with generating the interest that would eventually lead to this thesis. Dr. William Symes taught my first course in analysis. His excellent instruction technique helped me develop my appreciation for the subtleties of the real numbers.

Lt. Col. Mark Abramson deserves special mention for offering his advice, software, and technical support. We spent many hours discussing difficult topics that I may not have otherwise appreciated.

Finally, the trio of Walt Fish, Jeff Dyck, and Nikolaos Politis have earned my special thanks for providing moral support and for being there as friends. The time we spent inside and outside of the office was essential to the overall experience.
Contents

Abstract ii

Acknowledgements iv

List of Figures x

List of Tables xiii

1 Introduction 1
  1.1 Structural Optimization .............................. 1
  1.2 Purpose of the Research ............................ 4
  1.3 Thesis Overview .................................. 5

2 Literature Survey 9
  2.1 Problem Formulation ................................ 10
  2.2 Problem Solution .................................. 15
    2.2.1 Optimality Criteria Methods ................... 15
    2.2.2 Direct Search Methods ........................ 20
    2.2.3 Ad-Hoc Methods ................................ 26
    2.2.4 Use of Derivatives ............................ 26
  2.3 Nonsmooth Optimization ............................ 28
  2.4 Surrogates in Optimization ........................ 31
  2.5 Design Grouping .................................. 33
3 Optimization, Analysis and Design 35

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Problem Description and Formulation</td>
<td>35</td>
</tr>
<tr>
<td>3.2</td>
<td>Finite Element Solution</td>
<td>44</td>
</tr>
<tr>
<td>3.3</td>
<td>Derivatives and Sensitivities</td>
<td>49</td>
</tr>
<tr>
<td>3.4</td>
<td>Optimization Test Models</td>
<td>54</td>
</tr>
<tr>
<td>3.5</td>
<td>Optimization Software</td>
<td>59</td>
</tr>
<tr>
<td>3.6</td>
<td>Effects of Nonsmooth Constraints</td>
<td>63</td>
</tr>
<tr>
<td>3.7</td>
<td>Summary</td>
<td>67</td>
</tr>
</tbody>
</table>

4 Study of Failure Mechanisms 68

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1</td>
<td>Stress-Strain Model</td>
<td>69</td>
</tr>
<tr>
<td>4.2</td>
<td>Lateral-Torsional Buckling</td>
<td>73</td>
</tr>
<tr>
<td>4.2.1</td>
<td>Uniform Moment</td>
<td>74</td>
</tr>
<tr>
<td>4.2.2</td>
<td>Non-Uniform Moment</td>
<td>76</td>
</tr>
<tr>
<td>4.2.3</td>
<td>General Case</td>
<td>78</td>
</tr>
<tr>
<td>4.2.4</td>
<td>Finite Element Solution of Lateral-Torsional Buckling</td>
<td>79</td>
</tr>
<tr>
<td>4.2.5</td>
<td>Bounds on Inelastic Behavior</td>
<td>82</td>
</tr>
<tr>
<td>4.2.6</td>
<td>Iterative Solution</td>
<td>84</td>
</tr>
<tr>
<td>4.3</td>
<td>Flange Local Buckling</td>
<td>86</td>
</tr>
<tr>
<td>4.3.1</td>
<td>Bounds on Inelastic Behavior</td>
<td>88</td>
</tr>
<tr>
<td>4.4</td>
<td>Web Local Buckling</td>
<td>89</td>
</tr>
<tr>
<td>4.4.1</td>
<td>Elastic Stress Range</td>
<td>90</td>
</tr>
<tr>
<td>4.4.2</td>
<td>Inelastic Stress Range</td>
<td>91</td>
</tr>
<tr>
<td>4.4.3</td>
<td>Experimental Results of Plate Shear Failure</td>
<td>95</td>
</tr>
<tr>
<td>4.4.4</td>
<td>Finite Element Solution of Plate Shear Buckling</td>
<td>96</td>
</tr>
<tr>
<td>4.4.5</td>
<td>Bounds on Inelastic Behavior</td>
<td>102</td>
</tr>
<tr>
<td>4.4.6</td>
<td>Iterative Solution</td>
<td>103</td>
</tr>
</tbody>
</table>
5 Development of a Smooth Formulation

5.1 Allowable Stresses

5.1.1 Tensile Stress

5.1.2 Compressive Stress

5.1.3 Bending Stress

5.1.4 Shear Stress

5.2 Sample Calculations with Proposed Allowable Stresses

5.2.1 Lateral-Torsional Buckling

5.2.2 Flange Local Buckling

5.2.3 Web Local Buckling

5.3 Interaction Ratios

5.4 Comparison Between Proposal and Code Constraints

5.5 Summary

6 Two-Stage Algorithm

6.1 Implementation of Stage I

6.2 Implementation of Stage II

6.3 Prototype Example

6.4 Comparison of Methods on Test Set

6.5 Summary

7 Application to Tug Load Minimization

7.1 Problem Formulation

7.1.1 Environmental Loads

7.1.2 Equilibrium and Feasibility

7.2 Smooth Formulation

7.3 Comparison of Smooth and Nonsmooth Formulations

7.4 Summary
8 Conclusions and Recommendations

8.1 Thesis Summary ........................................ 191
8.2 Future Areas of Research ........................... 193

Bibliography ................................................. 196

A AISC ASD Allowable Stresses .......................... 205

B Polynomial Smoothing Functions .................. 209
List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Polling points on a mesh in the Pattern Search method.</td>
<td>22</td>
</tr>
<tr>
<td>2.2</td>
<td>Nonsmooth envelope function.</td>
<td>28</td>
</tr>
<tr>
<td>2.3</td>
<td>Nonsmooth absolute value function.</td>
<td>29</td>
</tr>
<tr>
<td>2.4</td>
<td>Nonsmooth piecewise defined function.</td>
<td>30</td>
</tr>
<tr>
<td>3.1</td>
<td>State variable details.</td>
<td>38</td>
</tr>
<tr>
<td>3.2</td>
<td>Wide flange geometric section property details.</td>
<td>39</td>
</tr>
<tr>
<td>3.3</td>
<td>Steel design parameter details.</td>
<td>40</td>
</tr>
<tr>
<td>3.4</td>
<td>Geometric constraint details.</td>
<td>41</td>
</tr>
<tr>
<td>3.5</td>
<td>Stress constraint details.</td>
<td>42</td>
</tr>
<tr>
<td>3.6</td>
<td>Objective function details.</td>
<td>43</td>
</tr>
<tr>
<td>3.7</td>
<td>Force and displacement sign convention for 2D frame element.</td>
<td>44</td>
</tr>
<tr>
<td>3.8</td>
<td>Two-dimensional structure (test model 19).</td>
<td>55</td>
</tr>
<tr>
<td>3.9</td>
<td>Small optimization test models.</td>
<td>56</td>
</tr>
<tr>
<td>3.10</td>
<td>Medium size test model bigframe ($n_u = 72$).</td>
<td>57</td>
</tr>
<tr>
<td>3.11</td>
<td>3D models</td>
<td>58</td>
</tr>
<tr>
<td>3.12</td>
<td>Nonsmooth design space of test model test19x.</td>
<td>65</td>
</tr>
<tr>
<td>4.1</td>
<td>Stress-Strain model.</td>
<td>71</td>
</tr>
<tr>
<td>4.2</td>
<td>Poisson’s ratio ($\nu$) model.</td>
<td>72</td>
</tr>
<tr>
<td>4.3</td>
<td>Lateral-torsional buckling behavior.</td>
<td>73</td>
</tr>
<tr>
<td>4.4</td>
<td>Beam properties.</td>
<td>75</td>
</tr>
<tr>
<td>Section</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>---------</td>
<td>----------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>4.5</td>
<td>Finite element and local loading.</td>
<td>80</td>
</tr>
<tr>
<td>4.6</td>
<td>Flange local buckling behavior.</td>
<td>86</td>
</tr>
<tr>
<td>4.7</td>
<td>Web local buckling behavior.</td>
<td>89</td>
</tr>
<tr>
<td>4.8</td>
<td>Plate properties.</td>
<td>91</td>
</tr>
<tr>
<td>4.9</td>
<td>Shear strength— theoretical curves.</td>
<td>94</td>
</tr>
<tr>
<td>4.10</td>
<td>Shear strength— finite element results.</td>
<td>103</td>
</tr>
<tr>
<td>5.1</td>
<td>Proposed LTB curve (W6X20, $F_y = 50ksi$).</td>
<td>114</td>
</tr>
<tr>
<td>5.2</td>
<td>Proposed flange restraint parameter ($k_r$) curve.</td>
<td>119</td>
</tr>
<tr>
<td>5.3</td>
<td>Proposed FLB curve.</td>
<td>120</td>
</tr>
<tr>
<td>5.4</td>
<td>Proposed WLB curve.</td>
<td>124</td>
</tr>
<tr>
<td>5.5</td>
<td>Proposed, smooth factor of safety curves.</td>
<td>125</td>
</tr>
<tr>
<td>6.1</td>
<td>Stage I Algorithm for the approximate solution of Problem 6.1.</td>
<td>140</td>
</tr>
<tr>
<td>6.2</td>
<td>Stage II Algorithm for the formulation of Problem 6.3.</td>
<td>142</td>
</tr>
<tr>
<td>6.3</td>
<td>Two-Stage Algorithm for the solution of Problem 6.1.</td>
<td>142</td>
</tr>
<tr>
<td>6.4</td>
<td>Proposed, smooth tensile interaction ratio ($IR_t$) curve.</td>
<td>145</td>
</tr>
<tr>
<td>6.5</td>
<td>Proposed, smooth compressive interaction ratio ($IR_c$) curve.</td>
<td>145</td>
</tr>
<tr>
<td>6.6</td>
<td>Proposed, smooth bending interaction ratio ($IR_b$) curve.</td>
<td>146</td>
</tr>
<tr>
<td>6.7</td>
<td>Proposed, smooth second-order moment magnifier ($B_1$) curve.</td>
<td>147</td>
</tr>
<tr>
<td>6.8</td>
<td>Interaction ratios of intermediate smoothness.</td>
<td>149</td>
</tr>
<tr>
<td>6.9</td>
<td>Proposed, smooth web buckling parameter ($k_v$) curve.</td>
<td>151</td>
</tr>
<tr>
<td>6.10</td>
<td>Dependence on design parameters in stress constraints.</td>
<td>153</td>
</tr>
<tr>
<td>6.11</td>
<td>Nonsmooth design space of Problem 6.4</td>
<td>158</td>
</tr>
<tr>
<td>6.12</td>
<td>Design space of Problem 6.4 during stage I solution (part 1).</td>
<td>160</td>
</tr>
<tr>
<td>6.13</td>
<td>Design space of Problem 6.4 during stage I solution (part 2).</td>
<td>161</td>
</tr>
<tr>
<td>6.14</td>
<td>NOMAD (GPS filter method) run statistics for model test20a.</td>
<td>165</td>
</tr>
<tr>
<td>6.15</td>
<td>Convergence history using Pattern Search and Two-Stage methods.</td>
<td>175</td>
</tr>
<tr>
<td>6.16</td>
<td>Convergence history of model bigframe.</td>
<td>176</td>
</tr>
</tbody>
</table>
7.1 Schematic diagram of offloading procedure. .......................... 179
7.2 Free body equilibrium of Tanker and FPSO. ......................... 184
## List of Tables

2.1 Section properties for column and beam section. 12
2.2 Some properties of various optimization algorithms. 16

3.1 Function evaluation timing data. 53
3.2 Test model problem data. 56
3.3 Test model 19 parameter data. 64
3.4 Test model 19 solution results using three algorithms. 66

4.1 Values of correction factor, \( \zeta \), for LTB slenderness at yield. 84
4.2 Experimental shear strength data. 95

5.1 Experimental and predicted critical FLB stresses. 118
5.2 Test model 19 solution using smooth and nonsmooth formulation. 136

6.1 Interval data for miscellaneous transition functions. 151
6.2 Design parameter bounds for allowable stresses. 154
6.3 Design parameter bounds for plate buckling coefficients. 154
6.4 Local solutions of Problem 6.4. 158
6.5 Stage I data for Problem 6.4. 159
6.6 Nearest rolled shapes corresponding to initial sizes for test models. 163
6.7 Initial objective and aggregate constraint violation for test models. 164
6.8 Pattern Search convergence results for test models. 166
6.9 Stage I data for model test19d1a. 168
6.10 Stage I data for model test20a. .......................... 168
6.11 Stage I data for model test21a. .......................... 168
6.12 Stage I data for model test23. .......................... 169
6.13 Stage I data for model bigframe. ........................ 169
6.14 Governing slenderness and axial stress parameters for model test20a. 171
6.15 Selected slenderness and axial stress parameters for model bigframe. 172
6.16 Final step data for the Two-Stage algorithm. ............. 173
6.17 Final comparison of results of two methods for test models. .... 174
6.18 Convergence rates for Two-Stage algorithm ................. 177

7.1 Tug load optimization test data. .......................... 187
7.2 Smooth and nonsmooth optimization of Problem 7.1. .......... 188

8.1 Optimal section properties for model bigframe. ............. 195
Chapter 1

Introduction

1.1 Structural Optimization

As structural engineers, we seek to improve design. Thus, we use optimization methods to determine feasible and optimal solutions of a posed continuous variable design problem. From this solution we obtain:

- The feasible and optimal sizes for all the members in the structure.

- An understanding of the behavior of each member in the optimal design be it governed by yield, buckling, local buckling, etc.

- The optimal classification of each member be it compact, non-compact or slender.

The goal of this research is to identify and study an optimization algorithm appropriate for the design of steel structures subject to code prescribed constraints. These constraint functions in general are not smooth or even continuous. Since the more efficient, derivative-based optimization techniques assume the objective and constraint functions are continuously differentiable (smooth), this creates a problem since the techniques may be less effective or ineffective when this assumption is violated. A
good deal of work has been done to date in structural optimization, though the issue of smoothness has not been adequately addressed.

In some studies [83], the constraints are simplified so that they are smooth, and standard optimization algorithms such as Newton's method or the gradient method may be used. In others [39], [66], [36], a direct search optimization method is used, which does not require continuous derivatives, or make any assumptions about the functions other than that they are well defined. A third class of studies includes those that use ad-hoc schemes to select feasible and optimal designs (e.g., [70], [33]). Lastly there are those studies that apply derivative-based algorithms to nonsmooth problems (e.g., [30], [35], pp. 255-261, [57]).

Clearly, one desires to refrain from over-simplification of the design constraints. The use of direct search (DS) methods is an excellent alternative for very complex engineering constraints [15], but the main drawback is their inefficiency compared with methods that use derivative information. Furthermore, as the number of variables increases, this becomes even more pronounced, since the sampling space grows geometrically. Ad-hoc approaches are not general enough to work for all classes of problems (see [16]).

The last category offers the most potential. There are two approaches: to ignore the nonsmoothness, or to account for it in some way so that a smooth optimization technique works. Ignoring nonsmoothness of the functions can give mixed results using a derivative-based method (see [77]). However, by using a mixed integer programming formulation, the nonsmoothness can be accounted for in such a way that the direct solution of the problem can be accomplished. Integer variables can be used to refer to specific code equations so that for a given set of these variables, only a subset of the constraint equations apply. Once the problem is so limited, the nonsmoothness due to code constraints disappears. Optimization via this method can be accomplished using a combinatorial approach to minimize the objective with respect to the discrete variables, while using a derivative-based method to minimize
the smooth, continuous subproblems. This idea seems attractive until the number of probable subproblems is examined. Since there are at least 4 integer variables per member group (one each for allowable compression, lateral-torsional buckling, flange local buckling, and web local buckling) with at least 2 possible values (inelastic or elastic buckling) in a structure with $N$ member groups (that is, $N$ distinct member sizes in the structure), the maximum number of possible subproblems is $2^{4N}$. For even 8 member groups, this is an astonishing 4 billion subproblems! Suppose the branch-and-bound algorithm is used and the number of combinations is reduced to only 1% of that, the number is still in the millions. If the number of subproblems can be somehow made independent of the problem size, the mixed integer concept becomes useful.

This study will propose an algorithm that uses derivative-based optimization methods, but simplifies the constraints so that they are nearly smooth. Although Goble and Moses [30] made some effort to make stress constraint functions continuous, no earlier work has addressed the smoothness requirements, and most do not even acknowledge the deficiency. The degree of smoothness in the constraints will be gradually decreased so that the subproblems are initially easy to solve. Once a reasonable approximation to the solution is known, we can use this information to impose secondary constraints limiting the range of parameters in the problem so that constraint functions are limited to a single, smooth interval of the piecewise defined stress constraints. These secondary constraints, analogous to the use of integer variables to denote certain failure modes, allow the formulation of a new, smooth optimization problem using a portion of the full design specifications. The number of subproblems required using this idea is a function of the number of steps taken to vary the smoothness of the problem. This number can be algorithmically specified and is completely independent of the problem size.
1.2 Purpose of the Research

Currently all structural optimization methods that apply code constraints fall into one of four categories, using:

- simplification of the constraints to produce a smooth formulation,
- optimization methods such as direct search, which do not require smoothness,
- ad-hoc schemes to circumvent the difficulties that may arise, or
- intentional application of an unsuitable algorithm to a nonsmooth problem.

As discussed in Section 1.1, all have drawbacks and are not well suited to solving general structural optimization problems. The intent of this study is to determine a way of simplifying the constraints so that they are smooth, then apply an efficient derivative-based method to the solution of this smooth problem.

The problem with optimizing using over-simplified constraints is that the resulting design does not meet real world specifications. The goal of this study is to establish the necessary simplifications to give a smooth formulation without losing the elements of the original design problem. This is accomplished by specifying an approximate formulation of the AISC ASD design code [3], which is different in only those areas where nonsmoothness has been identified. Each specific nonsmooth equation is reviewed to determine how best to approximate it with a smooth equation. In each area of review, the question of accuracy versus smoothness is weighed. The resulting alternative specification will have nearly smooth constraints and be accurate enough to use in lieu of the original.

Finally the specification proposed will be used in a Two-Stage algorithm, where the approximate smooth formulation is first solved. Once a solution to the simplified problem is known, a reduced exact formulation is posed, which contains only a subset of the constraints in the general problem. This reduction makes the final problem smooth by limiting the scope of the design for each member. For example, rather than
designing each member for any one of a dozen failure mechanisms, we determine the controlling mechanism, then design only for that. The effectiveness of this approach will be studied through numerical testing and comparison with a direct search method, which solves the original nonsmooth problem.

Completion of these tasks will yield:

- A solution algorithm for nonsmooth optimization. This algorithm will in general be applicable to nonsmooth problems if the source of nonsmoothness is due to discontinuities in the derivatives at the boundaries between piecewise defined functions.

- A smooth formulation of the minimum weight design problem using the AISC ASD [3] specification. The methods proposed may be used with equal success on other design specifications.

In addition to the solution algorithm and problem formulation, the software for their implementation will be developed and tested. The optimization problem can be solved using efficient, off-the-shelf software for nonlinear constrained optimization. Furthermore, there will be no limitation on the class of structure that can be designed using the software, although available computer hardware may limit the size of problems.

1.3 Thesis Overview

The thesis is organized into chapters resulting in the specification of an approximate, smooth formulation of the AISC ASD design code [3], and design and implementation of a Two-Stage algorithm for the solution of structural optimization problems with nonsmooth constraints. Chapter 2 discusses previous research in related fields to help gain perspective and motivate the study. First, methods of problem formulation and solution are reviewed. Second, a brief summary of nonsmooth optimization is given. Next, a discussion of the use and benefits of surrogates in optimization is
given. Finally, a brief discussion is given on why and how members are organized into groups for design efficiency.

Chapter 3 first details the specifics of the current formulation of the minimum weight design problem. Two formulations are given so that the advantages of explicit inclusion of the state equations (equations of equilibrium) as equality constraints can be understood. Next, the finite element method of analysis is reviewed and some characteristics of the state equations are pointed out. Analysis is an integral part of the design of structures because feasibility depends on knowledge of the state of stress in each part of the structure. Efficient analysis is necessary to improve the overall performance of the optimization. Next, calculation of derivatives and sensitivities is outlined. In particular, an example is given in the use of the chain rule to evaluate a complex derivative. Differentiation of the state equations to solve for the sensitivities (Jacobian matrix) of the state variables is explained. The use of sensitivities in evaluating derivatives of the state-dependent constraints is detailed. The next section presents the test set of structural models that are used for the validation and comparison of different methods. After that, the software modules used are explained. Last, a study of the effects of nonsmooth constraints in structural optimization is given, and the results are analyzed.

Chapter 4 provides background material for formulating an alternative design specification. The failure mechanisms of lateral-torsional buckling (LTB), flange local buckling (FLB), and web local buckling (WLB) are studied from a theoretical standpoint. Since determination of the strength of members in closed form is limited to elastic behavior, special attention is given to the inelastic range. Finite element models and experimental data are used to provide benchmarks for the proposed, alternative allowable stress equations. A stress-strain model for steel is proposed for use in the study.

In Chapter 5, allowable stresses are given for tension, compression, bending, and shear. These recommendations are made based on the background provided in Chap-
ter 4 and the need to provide smooth allowable stress functions. Tension and compression stresses are the same as in the original specification, but the equations are presented in a uniform format for completeness. Sample calculations for bending and shear stresses are given to clarify the use of the proposed equations, and to demonstrate their agreement with the original code. Finally, the interaction ratios\footnote{The ratio of actual to allowable stress.} themselves are reviewed from the point of view of nonsmoothness. Starting with the original code interaction ratios, an expanded set of stress checks is developed to eliminate the nonsmoothness associated with the use of the absolute value function and “min” or “max” functions. The primary dependence of the interaction ratio equations on the axial stress is discussed and an approach is given to generalize the cases where axial stress is (1) tensile, (2) compressive, but small, and (3) compressive, but large.

Chapter 6 presents the Two-Stage algorithm. In the opening section, the motivation for the algorithm is given, followed by the algorithm specifics. The next section details the implementation of the structural optimization problem formulated so that the stage I algorithm may be used. Specifics about the use of the smoothness parameter to transition from smooth to nonsmooth constraints are given. Stage II implementation is discussed in the section after that. Specifics consist of discussion of the secondary constraints to be used to limit the structural optimization problem to a smooth subset of the original problem. Finally, the results of testing of the Two-Stage algorithm on a prototype problem, and then the model “test set,” are reported. Comparisons are made with the augmented Lagrange and filter pattern search methods. From this, conclusions are drawn about the robustness and efficiency of the Two-Stage algorithm.

Chapter 7 shows the smooth formulation and solution of the “tug load” minimization problem. The concepts presented in Chapter 6 are potentially applicable to general nonsmooth optimization problems, provided an alternative smooth formulation can be determined. In this chapter, the nonsmooth characteristics of the
problem are determined and eliminated in a manner similar to the structural optimization problem in Chapter 5. Once a smooth alternative is found, the solution quality and efficiency are compared with the nonsmooth problem solution.

Chapter 8 gives conclusions, and summarizes the contributions of this work. Finally, recommendations are made for future research on related topics.
Chapter 2

Literature Survey

Structural optimization\(^1\) has been studied extensively since the 1960s. Various subjects range from size and shape optimization to topology optimization. This review focuses on size optimization assuming that the geometry of the structure and the loads and specifications for design are already determined. Based on that, the minimum cost sizes are to be selected. For simplicity, it is usually assumed that cost is proportional to the weight of structural steel. Questions on how to pose and solve the minimum weight design problem include:

1. Will the constraints be completely general or will some assumptions regarding the behavior of the structure be made?

2. Will the equilibrium constraints be treated implicitly or explicitly?

3. Should the variables fully define the cross-section, or should fewer, linked variables be used?

4. Should the problem be solved with discrete or continuous variables?

5. Are the objective and constraint functions smooth?

\(^1\)Originally this was called "design synthesis," to emphasize the purpose of the method.
6. Will mathematical programming, direct search, or ad-hoc methods be used to solve the minimum weight problem?

7. Will derivatives be used? If so, will they be analytical or finite difference approximations?

8. Will surrogate functions be used to reduce the amount of computation required during solution?

9. Can optimization variables for some members be grouped together?

The first four questions affect the formulation of the problem. Questions four through seven affect the choice of optimization algorithm and, ultimately, the software. Various ways of treating nonsmoothness are briefly described. The use of surrogates is an important topic, and it is discussed separately. Design grouping is a useful technique to reduce problem size and is discussed briefly.

2.1 Problem Formulation

There are multiple aspects to consider in the behavior of framed structures. Types of stresses include axial, shear, and bending. Axial and bending stresses can be either tensile or compressive. If stresses are compressive, they may cause failure by yield, inelastic buckling, or elastic buckling. Additionally, members in a general structure may have limit states defined by any of these. Due to the complexity of the systems, design codes have a variety of stress constraints, slenderness (geometric) constraints, and displacement constraints. In some cases (e.g., [69], pp. 100-107, [83]), a formulation may assume only a subset of these cases is possible, limiting the design, but simplifying the solution. The most obvious simplification (and the one taken in both of the studies cited above) is to assume that all members fail by yielding, which allows a single value for the maximum, or allowable stress. Most studies aimed at design use the full set, or a reasonably large subset of the code design constraints.
(e.g., [37], [30], and [5]). In order to be useful, the advantages gained by simplification must justify the loss of generality.

The most straightforward choice of optimization variables is to use only design parameters, such as the cross-sectional areas of members. This is known as the “black box” or NAND (nested analysis and design) method. An alternative approach (see [25]) called “all-at-once” or SAND (simultaneous analysis and design) defines the optimization variables by including design variables and state variables. The state variables solve the state equations governing equilibrium of the system (in structures, these are the nodal displacements). In order to formulate the optimization problem, equality constraints are explicitly included to impose equilibrium for feasible designs. Note that equilibrium is imposed implicitly in the NAND approach by solving for state variables in terms of the design variables.

Derivatives in the SAND approach are sparse and can be stored using much less memory (it may be advantageous to use only Hessian-vector products, and thus never form the Hessian). Another benefit from the SAND approach is that by separating the state and design variables, functions that involve both become less nonlinear. Since the constraints are linearized during the optimization process, this can reduce the number of iterations needed to reach a solution. The disadvantage of the SAND approach is that it adds considerable size to the optimization problem by including additional variables and constraints.

In specifying design variables, one can fully define a prismatic member’s cross-section by allowing the width and thickness of plate elements to vary continuously. For example, for a doubly symmetric wide flange section the design variables are: $u = [d \ t_w \ b_f \ t_f]^T$, where $d$ and $t_w$ are the section depth and web thickness and $b_f$ and $t_f$ are the flange width and thickness (see Figure 3.8, Section 3.4). Alternatively, fewer variables may be adequate, and this can reduce the problem size. In the case of trusses, the area of the cross-section is adequate to define its stiffness and strength when all members fail by yielding. If they are allowed to buckle, then the minimum
radius of gyration is also needed. For 2D frames, the area \((A)\), shear area \((A_v)\), strong-axis moment of inertia \((I_x)\), strong-axis section modulus \((S_x)\), and weak-axis radius of gyration \((r_y)\) are needed to define strength and stiffness. These quantities can be computed directly from the four design variables: depth, web thickness, and flange width and thickness.

It is reasonable to assume, and has been shown [21], that there is good correlation between the area and the remaining parameters \((A_v, I_z, S_z, \text{ and } r_y)\), so that \(A\) is commonly chosen as the single continuous design variable. Unfortunately, this choice does not provide any information about the slenderness of individual plate elements, so the code constraints cannot be fully determined. Furthermore, the correlation is poor in some cases. For example, two rolled shapes in Table 2.1 have nearly the same area, but they have significantly different values for their remaining properties. The W14 section is commonly used as a column, while the W36 is a beam. In order to avoid this difficulty, sections must be specifically categorized with respect to usage with different models to relate area and the other parameters.

<table>
<thead>
<tr>
<th>Shape</th>
<th>(d) (in)</th>
<th>(t_w) (in)</th>
<th>(b_f) (in)</th>
<th>(t_f) (in)</th>
<th>(A) (in(^2))</th>
<th>(I_x) (in(^4))</th>
<th>(S_x) (in(^3))</th>
<th>(I_y) (in(^4))</th>
<th>(r_y) (in)</th>
</tr>
</thead>
<tbody>
<tr>
<td>W14X257</td>
<td>16.4</td>
<td>1.18</td>
<td>16</td>
<td>1.89</td>
<td>75.6</td>
<td>3400</td>
<td>415</td>
<td>1290</td>
<td>4.13</td>
</tr>
<tr>
<td>W36X256</td>
<td>37.4</td>
<td>0.96</td>
<td>12.2</td>
<td>1.73</td>
<td>75.4</td>
<td>16800</td>
<td>895</td>
<td>528</td>
<td>2.65</td>
</tr>
</tbody>
</table>

Table 2.1: Section properties for column and beam section.

Variable linking can be useful to associate all of the relevant section properties through a single integer variable [6]. If the design space is limited to a subset of the available rolled shapes, and an integer value is assigned to each (say in order of ascending weight), then all of the needed quantities are fully and exactly defined through this link.

Since the design eventually results in a choice of sizes from a limited (discrete) number of rolled shapes, or by specifying built-up shapes with stock plate thicknesses, it is often thought of as a discrete problem. Optimization techniques for discrete
variable problems are very different from those for continuous variables. The discrete variable approach can generally be described as enumerating or directly searching the elements of the discrete variable domain. Practical methods use information about the designs to prevent the need to exhaust all combinations of the discrete variables. Methods such as branch-and-bound, simulated annealing, and genetic algorithms are commonly used in applications for structural optimization. Alternatively, the discrete problem may be solved as a continuous variable problem, in which the solution gives the information needed to choose the nearest sizes from those available. Accordingly, there is a need for robust methods to solve the continuous variable problem.

The obvious but simplistic approach of rounding continuous variables to their nearest discrete values may not be satisfactory from a feasibility or optimality point of view. There may not be a size that matches all of the relevant properties of the section \((A, A_v, I_x, S_x, r_y)\), so that the choice is not obvious. Based on the governing design constraints of a member, the best available section may be chosen by some heuristic procedure [5]. For example, for a member governed by flexure, the nearest area and section modulus may be sought. More systematic algorithms are designed to use the continuous variable problem as a subproblem for solving the discrete problem. For example, a method known as “dynamic rounding-up” [39] fixes a single variable to its nearest upper bound discrete value, then repeats the optimization to determine the corresponding values of the remaining design variables. This is repeated until all of the design variables have been assigned a value from the discrete set. This can be computationally expensive, so a variation on the method may instead choose several variables that do not have significant interaction at each step.

Another approach, the Lagrangian relaxation technique, is implemented by solving a minimax problem (see [32]). The solution of the minimax problem corresponds to a saddle point of the Lagrangian function. In the first stage, the Lagrangian is minimized for one optimization variable at a time over the discrete set. In the second stage, the resulting dual function, which is only defined on the discrete set is
maximized with respect to the Lagrange multipliers. The value of the optimization variables at the saddle point is a solution to the original discrete problem.
2.2 Problem Solution

The nonlinear inequality constrained minimization problem is posed as follows:

\[
\min_{x \in \mathbb{R}^n} f(x) \\
\text{s.t. } g(x) \leq 0
\]  

(2.1)

where: \( g : \mathbb{R}^n \rightarrow \mathbb{R}^m \)

The choice of solution algorithm for Problem 2.1 must take into account the nature of the formulation, problem size, applicability, and software availability. Methods that are strictly for discrete problems will not be discussed, since a continuous variable formulation will be posed in this research. Table 2.2 summarizes the applicability of common methods for a large number of design variables and indicates which will also work for nonsmooth problems. Note that there are not any methods in the list that are applicable for both large, and nonsmooth problems. Various commonly used methods will be discussed individually below. Software is available for most, but not all of the methods. Writing software for these algorithms can be quite complex, though some of the methods are easier to implement.

2.2.1 Optimality Criteria Methods

Methods in this category obtain a solution by seeking a point in the design space that satisfies the first-order optimality criteria (OC), or KKT conditions. The KKT conditions depend on a preliminary condition known as constraint qualification, which places a requirement on the constraint gradients so that Lagrange multipliers are guaranteed to exist. One such constraint qualification is the linear independence constraint qualification:

Definition 2.2.1. For a given point \( x^* \), if the set of active constraint gradients for Problem 2.1 \( \{ \nabla g_i(x^*) : g_i(x^*) = 0 \} \) is linearly independent, then the linear independence constraint qualification or LICQ holds at \( x^* \).
<table>
<thead>
<tr>
<th>Category</th>
<th>Algorithm</th>
<th>For Large Problems</th>
<th>For Nonsmooth Problems</th>
</tr>
</thead>
<tbody>
<tr>
<td>OC</td>
<td>FP</td>
<td>yes</td>
<td>no</td>
</tr>
<tr>
<td></td>
<td>IP</td>
<td>yes</td>
<td>no</td>
</tr>
<tr>
<td></td>
<td>SQP</td>
<td>yes</td>
<td>no</td>
</tr>
<tr>
<td></td>
<td>AL</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>DS</td>
<td>GA</td>
<td>no</td>
<td>yes</td>
</tr>
<tr>
<td></td>
<td>SA</td>
<td>no</td>
<td>yes</td>
</tr>
<tr>
<td></td>
<td>PS</td>
<td>no</td>
<td>yes</td>
</tr>
<tr>
<td>ad-hoc</td>
<td>FSD</td>
<td>yes</td>
<td>no</td>
</tr>
</tbody>
</table>

OC - Optimality Criteria  
DS - Direct Search  
FP - Fixed Point iteration  
IP - Interior Point  
SQP - Sequential Quadratic Programming  
AL - Augmented Lagrange  
GA - Genetic Algorithm  
SA - Simulated Annealing  
PS - Pattern Search  
FSD - Fully Stressed Design

Table 2.2: Some properties of various optimization algorithms.

*Depends on method of solving subproblem.

This condition allows the following definition:

**Definition 2.2.2.** If \( x^* \) is a local minimizer of Problem 2.1, and the LICQ (see Definition 2.2.1) holds at \( x^* \), then there exists a vector of Lagrange multipliers \( \lambda^* \), so
that the KKT conditions:

\[ \nabla_x L(x^*, \lambda^*) = 0 \]  \hspace{1cm} (2.2)

\[ g(x^*) \leq 0 \]  \hspace{1cm} (2.3)

\[ \lambda^*_i \geq 0, \ i = 1, \ldots, n_g \]  \hspace{1cm} (2.4)

\[ \lambda^*_i g_i(x^*) = 0, \ i = 1, \ldots, n_g \]  \hspace{1cm} (2.5)

hold, where \( L(x, \lambda) = f(x) + \lambda^T g(x) \) is the Lagrangian function. The gradient of the Lagrangian is \( \nabla_x L(x, \lambda) = \nabla_x f(x) + g(x)^T \lambda \), and the Jacobian of the constraint functions is given by \( g_x \).

Equations 2.2, 2.3, 2.4, and 2.5 are the KKT conditions. These provide a system of nonlinear equations (2.2 and 2.5) to solve for the optimization variables and Lagrange multipliers. They also provide a measure of feasibility (equation 2.3) and optimality (equation 2.2) of a solution, and are used in iterative numerical algorithms as stopping criteria. In general, OC methods are fast, find local solutions, and require locally continuously differentiable objective and constraint functions.

Fixed point (FP) iterations can be used to solve the KKT system. A contraction mapping, \( F \), can be devised that maps the current value of the design variables to a new value, so that \( F(x) - x = 0 \) is solved for \( x \) (see [21]). This update method is very efficient even for a large number of variables. Under certain conditions, namely if the objective and constraints are smooth and if the initial values of the variables are sufficiently close to the solution, the method will converge to a point satisfying the optimality criteria. Fixed point iteration methods are more generally used to solve systems of nonlinear equations, and are not specifically optimization methods. Since the algorithm is not designed to decrease the objective, it is possible to converge to a maximizer, if one exists. As a result, these methods require ad-hoc criteria to determine if a solution is locally a minimizer. If not, it is necessary to restart the algorithm from another initial point.

The interior point (IP) method is an iterative method that minimizes an equivalent
nonlinear equality constrained problem using a log barrier function to penalize points that are infeasible with respect to the inequality constraints. The log barrier function for solution of Problem 2.1 is:

\[ B(x, s; \kappa) = f(x) - \kappa \sum_{i=1}^{m} \ln s_i \]

where: \( \kappa > 0 \) is the barrier parameter

\( s \in \mathbb{R}^{m} \) is a vector of slack variables

The equality constrained subproblem is:

\[
\min_{x, s} B(x, s; \kappa) \tag{2.6}
\]

\[ \text{s.t. } g(x) + s = 0 \]

A perturbed KKT system is solved using Newton’s method so that the Hessian of the Lagrangian is required. Because Newton’s method is used, IP is very efficient even for a large number of variables, but it requires smooth objective and constraint functions to guarantee convergence to a solution.

Sequential quadratic programming (SQP) is an iterative method that solves a sequence of quadratic programming (QP) subproblems centered at the current iterate pair \((x_k, \lambda_k)\) (see [59], Chapter 18):

\[
\min_{p} \frac{1}{2} p^T \nabla_x^2 \mathcal{L}(x_k, \lambda_k)p + \nabla f(x_k)^T p \tag{2.7}
\]

\[ \text{s.t. } g_x(x_k)p + g(x_k) \leq 0 \]

so that: \( x_{k+1} = x_k + \alpha_k p_k \) is the next iterate

\( p_k \) is the solution to Problem 2.7

\( \alpha_k \) is an appropriate step size

\( \lambda_{k+1} \) is appropriately determined from the KKT conditions
The QP has a quadratic objective and linear constraints that are computed using Taylor series expansions about the current iterate, $x_k$. The expansion of the Lagrangian is second-order to capture its curvature, while that of the constraints is only first-order. The QP can be solved by various means including active set (see [59], pp. 457-475) and interior point methods. The SQP algorithm is very difficult to program because of many considerations for robustness, but software is readily available. The efficiency depends on the method used to solve the subproblems, but it is usually effective for a large number of variables. The SQP method requires smooth objective and constraint functions to guarantee convergence to a solution.

The augmented Lagrange (AL) method (see [58] and [23]) is an iterative method that solves a sequence of bound constrained subproblems. This is akin to a penalty function method because during the iterations a parameter is updated, which imposes a penalty on infeasibility. However, since the penalty function includes a Lagrange multiplier term, the penalty need not go to infinity in order to drive the constraints to zero. The augmented Lagrange function for solution of Problem 2.1 is:

$$P(x, s; \lambda; \mu) = f(x) + \lambda^T (g(x) + s) + \frac{1}{2\mu} \|g(x) + s\|_2^2$$

where: $\lambda \in \mathbb{R}^{n_s}$ is a vector of Lagrange multipliers

$\mu > 0$ is the penalty parameter

$s \in \mathbb{R}^{n_s}$ is a vector of slack variables

$\| \cdot \|_2$ refers to the Euclidean norm or SRSS.

The bound constrained subproblem is:

$$\min_{x, s} P(x, s; \lambda; \mu)$$

subject to $s \geq 0$ (2.8)

By setting the partial derivative of $P$ with respect to $s$ to zero, the value of $s$ at the optimum can be found in closed form. If the slack variables are eliminated in
this manner, the number of constraints does not affect the problem size. The applicability of the augmented Lagrange algorithm depends on the method used to solve the subproblems, which may be any method used for solving bound constrained problems (e.g. pattern search, interior point, or the gradient-projection method, see [59], pp. 476-481). In general, the augmented Lagrange method is less efficient than other OC methods because successive iterations require the solution of the nonlinear subproblem (Problem 2.8) with greater accuracy.

2.2.2 Direct Search Methods

Methods in this category obtain a solution by systematic evaluation of trial points until the minimizer, or an improved point, is found. These do not require derivatives, nor do they impose any requirements on the objective and constraint functions. They are more likely to find global minima than the previous class of methods.

The genetic algorithm (GA) is a stochastic method based on the evolutionary process in nature. A set of points in the design space is designated at each iteration as the “population.” An update consists of mating the population members to produce “offspring,” which become the population in the next iterate. The chance of any two members mating to produce offspring is nonzero; however, the more “fit” members are more likely to mate with each other producing even more fit offspring. Fitness is defined by some measure of feasibility and optimality, wherein higher fitness is associated with more feasible and optimal points. Since less fit offspring will occasionally be produced, the chance of finding only local minimizers is reduced. Additionally, an operation called a “mutation” is usually included to keep the population from stagnating. This method can be used to solve continuous or discrete problems. It is less effective for a large number of variables because the population must be large to provide sufficient diversity. It is equally effective on smooth and nonsmooth problems.

Simulated annealing (SA) is a stochastic method based on the process of the cooling of metals to form crystalline structures with minimum internal energy. An
initial temperature, $T_0$, is chosen. Successive iterates are generated randomly, but “close” to the current iterate. If a trial iterate is feasible and the objective is less than the current iterate, the trial is accepted and becomes the current iterate. If it is feasible, but the objective is more than the current iterate, and

$$r < e^{\frac{f(x_k) - f(x)}{T_k}}$$

where: $r$ is a randomly generated number between 0 and 1

$f(x)$ is the objective at the trial point

$f(x_k)$ is the objective at the current iterate

$T_k$ is the current temperature

the trial is accepted and becomes the current iterate, otherwise the current iterate is not changed. After each iteration the temperature is updated so that $T_{k+1} < T_k$. Depending on the temperature reduction rate, the method is more or less likely to accept nonoptimal iterates. This reduces the chance of finding only local minimizers. The method can be used to solve continuous or discrete problems. It is less effective for a large number of variables because as the order increases, the design space of potential iterates increases geometrically. It is equally effective on smooth and nonsmooth problems.

The pattern search (PS) method (see Figure 2.1) polls points on a mesh near the current iterate $x_k$ in a systematic fashion (for an overview, see [45]). Polling directions are defined by a predetermined pattern, $D_k$ chosen to contain a positive basis for the design variable space, and thus a descent direction. A positive basis is a set of vectors whose nonnegative linear combinations span the space. When a point in the pattern is found that improves the current iterate, it is accepted as the next iterate, otherwise the density of the mesh is increased and the process is repeated. In Figure 2.1 the objective function contours are shown and the minimum is near the center of the figure. Four polling points are indicated nearby the current iterate, $x_k$ and of these, two increase the objective, while two decrease it. In this case between
1 and 3 points must be polled before a new iterate is found.

\[ x_k + \Delta_k d \quad d \in D_k \]

Figure 2.1: Polling points on a mesh in the Pattern Search method.

Polling is guaranteed to improve the objective unless the current iterate is a local minimizer on the current mesh. Once the mesh is sufficiently dense, the method terminates with an approximate minimizer. The generalized pattern search (GPS) ([15]) includes an optional search step to precede polling around the current iterate to improve efficiency. The search can employ any method (e.g. a fast OC method) to return an improved point on the current mesh. At the most, polling can require \(2n\) trials (where \(n\) is the dimension of the design variable space), but the search method can be much more efficient if it is well suited to the problem. In the event that the search does not succeed, polling continues as usual to guarantee convergence. When the search uses a global surrogate (see Section 2.4) to scan the design space for regions of promising local minima, the chance of finding the global minimizer, or a deeper local minimizer is increased.

A variant of the pattern search method [2] uses gradient information of varying
fidelity to greatly reduce the number of polling steps. No assumptions on the characteristics of the objective function are made; however if it is discontinuous, and some partial derivatives are not available, the improvement made by using derivative information will be less pronounced. Essentially there is a one-to-one tradeoff between the maximum number of polling steps required and the number of partial derivatives available, so that if the entire gradient exists and is known, a total of \( n \) polling steps may be avoided. Additionally, the precision of the gradient may be very low and still useful because it is primarily used to determine directions of descent: the inner product of the gradient with a potential search direction can be used to determine whether moving along that direction will locally increase or decrease the objective function. If \( x_k \) is the current iterate, a direction \( d \) can be shown to locally decrease the objective by the Taylor series expansion:

\[
 f(x_k + \alpha d) = f(x_k) + \alpha d^T \nabla f(x_k) + O(\alpha^2)
\]

If \( \alpha > 0 \) is sufficiently small, and \( d^T \nabla f(x_k) < 0 \), then \( f(x_k + \alpha d) < f(x_k) \). If the calculation of the gradient is possible, and a low fidelity estimate can be determined without significant increase in the overall cost of a function evaluation, this approach, known as gradient “pruning,” can be much faster than the ordinary pattern search method.

The early pattern search method [38] was intended for unconstrained optimization, but there are several flavors of the PS for constrained optimization. Bound, or linearly constrained pattern search (see [49] and [50]) can be used to solve a problem with nonlinear constraints when nested inside an augmented Lagrange framework (see[51]). A filter method (see [7]), which solves the biobjective problem:

\[
\min_x (f(x), h(x)) \tag{2.9}
\]
where: \( h(x) = \|g_+(x)\|_2 \)

\[
g_+(x) = \begin{cases} 
g(x), & \text{if } g(x) > 0 \\
0, & \text{otherwise}
\end{cases}
\]

can also be used. Dominant points are used to build a filter of infeasible but relatively optimal, and nonoptimal but feasible points, similar to a Pareto set. A barrier function approach (see [2]) can be used in which the extended value objective function is minimized:

\[
\min_x f_B(x) \tag{2.10}
\]

where: \( f_B : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\} \)

\[
f_B(x) = \begin{cases} 
f(x), & \text{if } h(x) = 0 \\
\infty, & \text{otherwise}
\end{cases}
\]

\( h(x) \) is defined in Problem 2.9

The strengths and weaknesses of each method are discussed below.

The augmented Lagrange method performs as it was described in Section 2.2.1 using a pattern search method to solve the bound constrained subproblem (Problem 2.8). As a result, the convergence properties are excellent. However, it requires the solution of several increasingly difficult subproblems and can be slow when compared to methods that solve Problem 2.1 directly. Choice of the initial penalty parameter, \( \mu_0 \), may positively influence the solution efficiency; however, if it is chosen incorrectly the first few iterations will produce very difficult subproblems.

The filter method has the advantage that the solution of Problem 2.9 is made directly, and it tends to identify a solution much faster than the augmented Lagrange algorithm\(^2\). Because the filter contains a family of current iterates, polling must be

\(^2\)The convergence of the filter method is somewhat slow once it nears a solution. However, the initial location of a minimizer can be very fast.
done around multiple poll centers. Typically these are chosen as the best feasible iterate, and the least infeasible iterate, but all iterates in the filter may be used as poll centers. This may double the amount of polling compared to the barrier function approach. A disadvantage of the filter method is that the poll center strategy and choice of upper bound limit on infeasibility may affect both the efficiency of the algorithm, and the minimizer to which it converges.

The barrier method is essentially an unconstrained optimization method that cannot converge to an infeasible point. From this point of view it is potentially the fastest pattern search method for the solution of Problem 2.1, because it essentially solves it directly. However, since no information is retained about the constraint boundary as in the filter method, it is less likely to find points on the boundary, which are usually the most optimal. Another drawback is that convergence analysis of the pattern search method does not apply to extended value objective functions (see [74]). In practice the algorithm tends to stall at a point near the constraint boundary, unable to move in any direction that improves the objective because the barrier does not permit it. This is the reason for use of the filter method, which selectively allows infeasible points to be considered in order to avoid stalling. Mesh adaptive direct search (MADS) methods are now being studied to address this problem [8]. This stochastic approach is designed to prevent stalling by randomly providing alternative search directions, one of which may allow progression of the search along a constraint boundary. Random selection from a larger, but finite set of candidate directions is required because the arbitrary inclusion of search directions violates the key assumption that the pattern be determined \textit{a priori}.

In general, pattern search methods are less effective for a large number of variables because as the order increases, the design space of potential iterates increases geometrically. Depending on the implementation, they are more likely to find global minima. They are equally effective on smooth and nonsmooth problems.
2.2.3 Ad-Hoc Methods

Methods in this category obtain an improved design (perhaps a minimizer) by using methods that have proven effective in manual design. The properties of efficiency and convergence are completely dependent on the method.

Fully stressed design (FSD) (see [43], pp. 88-98, and [75], pp. 340-342) is an iterative method that resizes the members at each iteration so that they are fully stressed under the current loads (i.e., one of the stress constraints is active for every member). Once the structure is reanalyzed with the new sizes, forces redistribute and the members are not all fully stressed. However, as the method converges, successive iterations produce less redistribution of forces and the members are close to fully stressed at the solution. The method does not necessarily produce an optimal design, though optimality is possible. If it is convergent, it will produce an improved, feasible design. The method is very efficient even for a large number of variables. It requires smooth constraint functions, but there is no guarantee of convergence (see [16]).

2.2.4 Use of Derivatives

Derivative-based optimization methods use derivatives of the objective function \( f(x) \) to obtain search directions for finding the minimizer. The \( k \)-th step in an iterative method for unconstrained optimization computes the search direction \( d_k \) that solves \( B_k d_k = -\nabla f(x_k) \). If \( B_k \) is invertible, \( d_k \) exists and is unique. In Newton’s method, \( B_k = \nabla^2 f(x_k) \) is the Hessian of \( f \). If the Hessian is positive definite, it is invertible. In the gradient method (or method of steepest descent), \( B_k = I \) (the identity matrix), and \( d_k = -\nabla f(x_k) \) is the negative gradient of \( f \), which is locally a direction of descent of the objective function.

Analytical derivatives are not available if the associated functions come from a “black box.” For example, if a finite element analysis program computes the structure response to loading, the state equations are solved, but their derivatives are not needed for the analysis, and so they are not provided. When calculation of the functions
is done directly, analytical gradients and Hessians can be computed. Automatic differentiation methods [13] are an aid in this. These parse the program source code to interpret a function and generate further source code for the calculation of its partial derivatives. Rules of differentiation such as Liebnitz’ rule and the chain rule are applied as they are when deriving derivatives manually.

The question of using finite difference vs. analytical derivatives should be decided in terms of efficiency. If it is possible to compute the analytical gradients and Hessians for the objective and constraints, it must be determined whether the additional accuracy is worth the time required in computation (most applications use analytical gradients but Hessian approximations). A finite difference gradient of a function in $n$ variables may be computed with $n + 1$ to $2n$ function evaluations using either forward or centered finite difference approximations. Given a differentiable function $f : \mathbb{R}^n \to \mathbb{R}$, its partial derivatives can be approximated by:

$$\frac{\partial f}{\partial x_i} \approx \frac{f(x + \delta) - f(x)}{\delta}, i = 1, \ldots, n$$

or

$$\frac{\partial f}{\partial x_i} \approx \frac{f(x + \delta) - f(x - \delta)}{2\delta}, i = 1, \ldots, n$$

using a suitably small $\delta$. Finite difference Hessians are not computed using additional function evaluations. Instead a secant method such as the BFGS update is used to update $B_k$, an estimate of the Hessian (see [59], Section 8.1):

$$B_{k+1} = B_k - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \frac{y_k y_k^T}{y_k^T s_k}$$

where: $s_k = x_{k+1} - x_k$, $y_k = \nabla f(x_{k+1}) - \nabla f(x_k)$

The BFGS Quasi-Newton method typically requires more iterations than would Newton’s method with exact derivative information. However, the efficiency of the BFGS update may result in overall better algorithm performance. The Newton-CG (conjugate gradient) method uses an alternative to forming the Hessian. Instead of the actual Hessian it requires only the product of the Hessian and a vector, which may require significantly less storage and can be more efficient.
2.3 Nonsmooth Optimization

It is not possible to solve a general nonsmooth optimization problem with optimality criteria methods. However, some strategies exist for solving special cases. The first approach is by reformulation of the nonsmooth problem to an equivalent smooth one. For example, if a constraint exists on the envelope (or minimum) of several other functions as in Figure 2.2, it is nonsmooth at the intersection of the constituent curves (shown circled in the figure), here where $g_1$ and $g_2$ intersect. The single constraint may be expanded to a set of constraints on the individual functions involved, that is:

$$\min (g_1(x), g_2(x)) \leq 0 \Rightarrow \begin{cases} 
g_1(x) \leq 0, 
g_2(x) \leq 0 
\end{cases}$$

![Figure 2.2: Nonsmooth envelope function.](image)

Similarly, if a constraint function involves an absolute value as in Figure 2.3 it is nonsmooth at the point where the argument of the absolute value, here $f(x)$ is zero (shown circled in the figure). The single constraint may be expanded to 2 constraints:

$$|f(x)| \leq 1 \Rightarrow -1 \leq f(x) \leq 1$$
Another method is to limit the scope of the problem so that the nonsmooth functions are locally smooth. For example, if a constraint function is defined piecewise by several smooth functions with discontinuous derivatives at the boundaries between pieces as in Figure 2.4 (the discontinuous derivatives are shown circled in the figure), it may be possible to find the solution by minimizing while considering each smooth piece individually. Consider the problem:

$$\min_{x \geq 0} f(x)$$

s.t. $g(x) \leq 0$

where: $g(x) = \begin{cases} 
  g_1(x), & 0 \leq x < 0.56 \\
  g_2(x), & 0.56 \leq x < 1.15 \\
  g_3(x), & x \geq 1.15 
\end{cases}$

If three subproblems are solved:

$$\min_{0 \leq x < 0.56} f(x) \quad \min_{0.56 \leq x < 1.15} f(x) \quad \min_{x \geq 1.15} f(x)$$

s.t. $g_1(x) \leq 0$ s.t. $g_2(x) \leq 0$ s.t. $g_3(x) \leq 0$
then $x^*$, the solution of the original nonsmooth problem, corresponds to the solution of the smooth subproblem with minimum objective. Depending on the number of such constraints, this may or may not be practical since all combinations of the constraint segments must be somehow considered.

If a function is continuous, but nondifferentiable at a few points, methods can be employed using generalized gradients to identify local minimizers. A class of methods called “bundle” methods [47] are aimed at unconstrained problems, but can be extended to constrained problems using a penalty function approach. The cutting-plane method is an example, which may be unstable (and numerically ineffective) and requires a convex objective function. Methods for solution of nonsmooth nonlinear equations (e.g. see [31]) may be adapted to optimization, but they are not robust unless a successful globalization scheme is used.

The NEOS Optimization Software Guide [19] is an online guide to software for optimization and nonlinear equations. This source lists hundreds of codes for smooth optimization (such as Lancelot), but only one for “Nondifferentiable Optimization.” This code, called BT, is a bundle method, but it is limited to linearly constrained problems [61].
2.4 Surrogates in Optimization

Surrogates are used to increase efficiency in optimization when the evaluation of objective or constraint functions is computationally expensive. A surrogate of a function is any mapping from the domain to the range that approximates the function values at some or all points in the domain. If a surrogate is intended for use over the entire domain, it is generally less accurate locally. Conversely, if it is intended for use in a specific region of the domain, it is more accurate locally, but of little or no use outside of the intended region. Surrogates are commonly used as part of an iterative optimization algorithm (e.g. SQP) that updates the surrogate so that it becomes more accurate near the solution. Eventually it is sufficiently accurate so that optimizing the surrogate function is numerically equivalent to optimizing the original function.

Low order interpolating polynomials, or response surfaces, are commonly used global surrogates. A polynomial model can be obtained using a least squares fit, but these models are increasingly difficult to build in higher order dimensions. If the data points in the model sufficiently represent the behavior of the function over the entire domain, it can be a useful coarse model for detecting regions where local minima exist. Global models must use “space filling” data. That is, the points used to build the model must uniformly fill the design space. In higher order dimensions, more points are needed so that all global models are limited in use to relatively small problems.

Series expansions are commonly used local surrogates. The Taylor series is the most often used (e.g., [54]) because gradients (and sometimes Hessians) are used for optimization and are readily available. If the function and its derivatives are known at a point $x_0$, then the function at any $x$ is represented by the infinite Taylor series expansion:

$$f(x) = f(x_0) + f'(x_0)(x - x_0) + \frac{1}{2}f''(x_0)(x - x_0)^2 + \frac{1}{6}f'''(x_0)(x - x_0)^3 + \cdots$$
When $x$ is a vector valued argument, only the first three terms are practical to evaluate. This results in a quadratic model that captures the curvature of the function and approximates the values locally.

A binomial series expansion of a matrix using a few terms (see [48]) can be developed to approximately solve the linear system $Ax = b$, where $A = (A_0 + \Delta A)$, $A_0 \in \mathbb{R}^{n \times n}$ is a known factored matrix, and $\Delta A$ is a small perturbation in $A_0$. If $A$ and $A_0$ are symmetric, the infinite binomial series expansion of $A^{-1}$ is:

$$A^{-1} = (A_0 + \Delta A)^{-1}$$

$$= A_0^{-1} - A_0^{-2} \Delta A + A_0^{-3} \Delta A^2 - A_0^{-4} \Delta A^3 + \cdots$$

$$= (I - A_0^{-1} \Delta A + A_0^{-2} \Delta A^2 - A_0^{-3} \Delta A^3 + \cdots) A_0^{-1}$$

Let $B = A_0^{-1} \Delta A$,

then $(A_0 + \Delta A)^{-1} = (I - B + B^2 - B^3 + \cdots) A_0^{-1}$

A reduced basis is introduced with dimension $q \ll n$. The basis vectors are determined from the first $q$ terms of the binomial expansion. Since $x = (A_0 + \Delta A)^{-1} b$, the first vector is $r_1 = A_0^{-1} b$. The following recurrence relation can be used to determine the remaining $q - 1$ basis vectors:

$$r_2 = -BA_0^{-1} b = -Br_1$$

$$r_3 = -B^2 A_0^{-1} b = -Br_2$$

$$\vdots$$

$$r_q = -B^{q-1} A_0^{-1} b = -Br_{q-1}$$

Let $x = Ry$, $R \in \mathbb{R}^{n \times q}$ is $[r_1 r_2 \cdots r_q]$, then if the columns of $R$ are linearly independent, there is a solution to $R^T ARy = R^T b$, and an approximation to $x$ is computed from $y$ and $R$. The original factorization of $A_0$ is done only once. The remaining operations involve solving order $n$ triangular systems from this factorization and solving order $q$ systems to determine the basis coefficients, $y$. This surrogate can be used, for
example, to solve for the state variables or displacements in a structural analysis by taking $A_0$ to be the initial stiffness matrix, $\Delta A$ any perturbations in stiffness from changes in the design, $x$ the displacements, and $b$ the external loads.

If the surrogate is only accurate locally, it is standard practice to impose "move limits" or bounds on the design variables as in trust region methods. We know from the first-order Taylor series that accuracy improves near the center of the expansion but decreases at the rate of distance squared away from the center.

There are also many miscellaneous models, such as the "two-point" methods that match the function at two points. A particularly interesting implementation [82] includes two terms, which are essentially a first-order Taylor series in both the original and reciprocal variables. This method is only useful as a local surrogate, but has the potential to accurately represent highly nonlinear functions.

Hybrid models attempt to provide accuracy globally and locally. One example is "combined approximations" (CA) [44], which combines a local (binomial expansion) and global (reduced basis) model. The surrogate is a linear combination of basis vectors determined by the local model of the original function. The weighting parameters can be determined to fit the function globally in a least squares sense. Another example is kriging models (see [68]), which match the function exactly at known data points and provide a smooth variation between. Kriging model parameters are determined by maximum likelihood estimation (MLE), where the parameters are the best guesses to fit the given data. The number of data points used in the model is limited because as points are added, the linear system used to compute model parameters becomes increasingly ill-conditioned.

### 2.5 Design Grouping

A design group consists of one or more members with the same cross-sectional properties. The groups are chosen according to structural function and proximity. It is
expected that if two members in a design group were allowed to vary independently, the resulting optimal design variables would differ by a small percentage. Besides reducing problem size, this concept introduces a degree of practicality into the design from the point of view of simplifying the fabrication and erection of the structure by introducing fewer differing elements.

For example, beams and columns would be in different groups because of their different structural function. In a multi-story structure, it is reasonable that the stories would also divide into groups since forces tend to be cumulative from the top down. It is not practical to change column sizes at every story, so a group may be made up of similar members in several adjacent stories. If a symmetric design is desired, groups may be chosen with this in mind. Since all members in a group do not behave exactly the same, a few members in each group control the design while others are under-utilized. Though nonoptimal, this result has many practical benefits.

### 2.6 Summary

There is not sufficient research available on structural optimization methods for problems with nonsmooth stress constraints using optimality criteria methods. The majority of research done on nonsmooth problems in recent years has concentrated on the use of genetic algorithms (e.g., [17], and [36]). Other research in structural optimization has not addressed the issue of nonsmoothness (e.g., [57]).

The general inequality constrained optimization problem (Problem 2.1) is well understood from the point of view of algorithmic limitations. Based on these, it is clear that either a reformulation strategy or the use of a direct search method should be used to accommodate nonsmooth problems. In light of the large body of research already in existence on genetic algorithms, this study seeks to consider the former alternative. It is also anticipated that the use of optimality criteria methods will result in more efficient solution of the nonsmooth problem, if managed properly.
Chapter 3

Optimization, Analysis and Design

Details of the structural optimization problem are presented below in Section 3.1. The FEM analysis and evaluation of the objective and constraint functions and derivatives are discussed in Sections 3.2 and 3.3. A test set of structural optimization problems has been designed and is presented in Section 3.4. Use of the software for the augmented Lagrange, filter pattern search, interior point, and sequential quadratic programming software packages are discussed in Section 3.5. A preliminary study has been done on the effects of nonsmooth constraints in zero-, first-, and second-order optimization methods. The results are discussed in Section 3.6.

Note: In this chapter the symbols “x” and “y” refer to optimization variables. In a few instances the same symbols are used in a structural engineering context to refer to other quantities. The intended usage is pointed out in the relevant section to avoid confusion.

3.1 Problem Description and Formulation

The nonlinear constrained optimization problem for the design of steel structures is posed below using continuous design and state variables. Continuous rather than
discrete variables are used so that derivative-based methods of problem solution may be employed. While it is recognized that the design problem ultimately has a discrete solution, a continuous solution can provide information for the solution to the discrete problem (see Section 2.1). Furthermore, any advances in the solution of the continuous problem will clearly advance the broader state-of-the-art. The cross-sections are assumed to be doubly symmetric wide flanges (see Figure 3.8, Section 3.4). Design grouping is assumed, although not necessary as a consequence of the formulation. Both the NAND and SAND formulations are presented since neither has demonstrated a clear advantage at this point.

The objective is to minimize the weight of structural steel. The constraints limit stresses, displacements, and certain aspects of the cross-section geometry. The axial, bending, and shear stresses have limits dictated by code strength requirements. The nonlinear geometric constraints impose limits on (a) the ratio of weak-to-strong-axis stiffness to guarantee that the member buckles about the weak-axis, (b) the code prescribed slenderness of the cross-section, and (c) the ratio of width-to-thickness of the flanges and web (this may be prescribed by the code, the user, or some practical default value). The linear geometric constraints impose limits so that (a) the thickness of the flanges does not exceed the depth, and (b) the thickness of the web does not exceed the flange width. All variables may have upper and lower bounds. The design variables are practically limited by values of available sections or plates. The state variables may have limits dictated by code serviceability (i.e. deflection) requirements.

Implicit state formulation (NAND):

\[
\min_u f(u) \quad (3.1)
\]

s.t. \( g(u, y(u)) \leq 0, \)

\[
\begin{bmatrix}
    u^L \\
    y^L
\end{bmatrix} \leq \begin{bmatrix}
    u \\
    y(u)
\end{bmatrix} \leq \begin{bmatrix}
    u^U \\
    y^U
\end{bmatrix}
\]
where: \( u \in \mathbb{R}^{n_u} \) is a vector of design variables, e.g. for a single design group: \( u_i = [d_i \ t_{wi} \ b_i \ t_{fi}]^T, i = 1, \ldots, N \)

\( y \in \mathbb{R}^{n_y} \) is a vector of state variables: \( y \) solves \( K(u)y = p \)

\( f : \mathbb{R}^{n_u} \rightarrow \mathbb{R} \) is the objective function: \( f(u) = \frac{1}{2} u^T H u \)

\( g : \mathbb{R}^{n_u \times n_y} \rightarrow \mathbb{R}^{n_g} \) is a vector function of inequality constraints:

\( g = [g_g(u), g_s(u, y(u))]^T \)

\( u^L > 0 \) and \( u^U \) are upper and lower bounds on \( u \) (box constraints)

\( y^L \) and \( y^U \) are upper and lower bounds on \( y \) (box constraints)

Figure 3.6 gives details of the objective function, \( f \). Details of the solution of the state equations, state variables \( y \), and calculation of member forces and stresses are given in Figure 3.1 and later in Section 3.2. Figures 3.4 and 3.5 give details of the inequality constraints, \( g_g \) and \( g_s \).

**Explicit state formulation (SAND):** (see also Problem 3.1)

\[
\begin{align*}
\min_x f(u) \\
\text{s.t. } c(u, y) &= 0, \\
&\quad g(u, y) \leq 0, \\
&\quad x^L \leq x \leq x^U
\end{align*}
\]

where: \( x \in \mathbb{R}^{n_x} \) is a vector of design and state variables:

\( x = \begin{bmatrix} u \\ y \end{bmatrix}, \quad n_x = n_u + n_y \)

\( c : \mathbb{R}^{n_u \times n_y} \rightarrow \mathbb{R}^{n_c} \) is a vector function of state equations:

\( c(u, y) = K(u)y - p \)

\( g : \mathbb{R}^{n_u \times n_y} \rightarrow \mathbb{R}^{n_g} \) is a vector function of stress constraints:

\( g = [g_g(u), g_s(u, y)]^T \)
\[ x^L = \begin{bmatrix} u^L \\ y^L \end{bmatrix}, \text{ and } x^U = \begin{bmatrix} u^U \\ y^U \end{bmatrix}, \text{ define the box constraints on } x \]

Figure 3.6 gives details of the objective function, \( f \). Details of the solution of the state equations, \( c \), state variables \( y \), and calculation of member forces and stresses are given in Figure 3.1 and later in Section 3.2. Figures 3.4 and 3.5 give details of the inequality constraints, \( g_s \) and \( g_a \).

Figure 3.1 defines the state equations and the state variables used in Problems 3.1 and 3.2. The state equations are linear in \( y \), but nonlinear in \( u \), since the stiffness, \( K \), depends on the design variables. The global load vector, \( p \), is assumed to be constant.

\[
\begin{align*}
y &\in \mathbb{R}^{n_y} \text{ is a vector of state variables: } y \text{ solves } K(u)y = p \\
K &\in \mathbb{R}^{n_y \times n_y} \text{ is a global stiffness matrix} \\
p &\in \mathbb{R}^{n_u} \text{ is a global load vector}
\end{align*}
\]

Figure 3.1: State variable details.

Figure 3.2 summarizes the design variables used in Problems 3.1 and 3.2 and the geometric cross-sectional properties needed for wide flange members. The dependencies of geometric properties on the design variables, \( u \), are stated in the figure. Constant parameters associated with steel design are summarized in Figure 3.3. These are fixed parameters and do not depend on any of the optimization variables. Note: In both of these figures, several quantities have the subscripts “\( x \)” and “\( y \)”. These refer to the axis, or direction in which the value is defined, and should not be confused with the state variables, \( y \), or the more general optimization variables \( x \) in Problems 3.1 and 3.2.

The inequality constraints in Problems 3.1 and 3.2 are partitioned first into stress constraints and geometric constraints. The division is made to separate state-
for $u = \begin{bmatrix} d & t_w & b_f & t_f \end{bmatrix}^T$

$d$ is the overall section depth
$t_w$ is the web thickness
$b_f$ and $t_f$ are the flange width and thickness

Additionally:

$h(u) = d - 2t_f$, $h_c(u) = d - t_f$
$A_f(u) = b_f t_f$, $A(u) = 2A_f(u) + h(u)t_w$
$A_v(u) = \text{the effective net area (see [3], Chapter B)}$

$A_v(u) = dt_w$, $A_T(u) = A_f(u) + \frac{1}{6}h(u)t_w$

$I_x(u) = \frac{1}{12} \left( 2b_f t_f^3 + t_w h(u)^3 \right) + \frac{1}{2}A_f(u)h_c(u)^2$, $r_x(u) = \sqrt{\frac{I_x(u)}{A(u)}}$

$I_y(u) = \frac{1}{12} \left( 2t_f b_f^3 + h(u) t_w^3 \right)$, $r_y(u) = \sqrt{\frac{I_y(u)}{A(u)}}$

$I_T(u) = \frac{1}{12} \left( t_f b_f^3 + \frac{1}{6}h(u) t_w^3 \right)$, $r_T(u) = \sqrt{\frac{I_T(u)}{A_T(u)}}$

$S_z(u) = \frac{2I_x(u)}{d}$

$J(u) = \frac{1}{3} \left( 2b_f t_f^3 + h(u) t_w^3 \right)$, $C_w(u) = \frac{1}{4}I_y(u)h_c(u)^2$

Figure 3.2: Wide flange geometric section property details.

dependent and state-independent constraints, primarily due to the cost of evaluation of the state-dependent constraints and their derivatives. The geometric constraints (see Figure 3.4) are further partitioned into a section proportioning constraint, limit constraints, and wide flange shape constraints. The first of these guarantees that the
$E = 29,000$ ksi ($200,000$ MPa) is the modulus of elasticity of steel

$G = 11,200$ ksi ($77,200$ MPa) is the shearing modulus of steel

$E_{st} = 700$ to $900$ ksi ($4800$ to $6200$ MPa) is the strain hardening modulus of steel

$\rho_s = 490$ lb/ft$^3$ ($7.85$ g/cm$^3$) is the density of steel

$F_y$ is the yield stress of the member

$F_u$ is the ultimate stress of the member

$K_x$ is the strong-axis effective length factor of the member

$K_y$ is the weak-axis effective length factor of the member

$L_x$ is the strong-axis unbraced length of the member

$L_y$ is the weak-axis unbraced length of the member

$L_b$ is the compression flange brace spacing of the member

$C_b \in [1, 2.3]$ is the moment gradient coefficient of the member

$C_m \in [0.2, 1]$ is the applied moment coefficient of the member

$a$ is the spacing of transverse web stiffeners

Figure 3.3: Steel design parameter details.

allowable bending stress is less than the Euler strong-axis buckling stress ($F_a < F'_e$, see Appendix A for their definitions). The limit constraints\(^1\) ($g_{lim}$) allow design for particular slenderness ratios (for control of member global and local buckling). The wide flange shape constraints ($g_{WF}$) guarantee that the shape of a wide flange is

\(^1\)No actual limit is specified for $L_b$, though $L_b/d \leq 30$ is recommended. See [3] Table B5.1 or Appendix B5.2.a for guidelines on flange slenderness. See [3] Section F4 or G1 for guidelines on web slenderness.
$g_g$ is a vector function of geometric constraints, e.g. for a specific design group: $g_{gi}(u) = \left[ B_{xy} \frac{I_x(u)}{I_y(u)} - 1, g_{lim}(u), g_{WF}(u) \right]^T, i = 1, \ldots, N$

$g_{lim}$ is a vector function of limit constraints:

$$
g_{lim}(u) = \left[ \frac{K_y L_y}{r_y(u)} - \left( \frac{K L}{r} \right)^U, \frac{L_b}{r_y(u)} - \left( \frac{L_b}{r_y} \right)^U, \ldots, \right.$$

$$\left. \frac{b_f}{2t_f} - \left( \frac{b_f}{2t_f} \right)^U, \frac{h(u)}{t_w} - \left( \frac{h}{t_w} \right)^U \right]^T$

$g_{WF}$ is a vector function of wide flange shape constraints:

$$g_{WF}(u) = [2t_f - d, t_w - b_f]^T$$

$B_{xy} > \max (1, (K_x L_x / K_y L_y)^2)$ is a section proportioning constant

$I_x(u), I_y(u)$ are the strong and weak-axis moments of inertia

$r_y(u)$ is the weak-axis (minimum) radius of gyration

$(K L/r)^U$ is 200 for compression members, 300 for others

$(L_b/r_y)^U, (b_f/2t_f)^U$, and $(h/t_w)^U$, are user-defined limits.

Figure 3.4: Geometric constraint details.

In $g_s$ (see Figure 3.5 and Appendix A for definitions of AISC ASD [3] allowable stresses), we have two basic stress constraint equations: unity checks $(UC)$, plus a check on the axial stress to prevent division by zero while computing $B_1$ in the unity check equations, in the event the axial stress is compressive and approaches the Euler buckling stress $(f_a \rightarrow -F'_a)$. Specifically, given the allowable axial stress is less than
the Euler buckling stress \((F_a < F'_e)\) and:

\[-f_a \leq F_a, \text{ when } (f_a < 0)\]

then:

\[-f_a \leq F_a < F'_e\]

\[-f_a < F'_e\]

\[-\frac{f_a}{F'_e} < 1\]

\[1 + \frac{f_a}{F'_e} > 0\]

The stress checks apply at each cross-section of a member but in practice are applied at as many points as required to find critical stress combinations.

\[g_a \text{ is a vector function of stress constraints, e.g. at a specific cross-section: } g_{a_j}(u, y) = \left(UC(u, y) - 1, -\frac{f_a(u, y)}{F_a(u)} - 1\right)^T, j = 1, \ldots, M\]

\(UC(u, y)\) is the maximum unity check (see Appendix A)

\(f_a(u, y), f_v(u, y), \text{ and } f_b(u, y)\) are the axial, shear, and bending stresses

\(F_t(u), F_a(u), F_v(u), \text{ and } F_b(u)\) are code prescribed allowable tensile, compressive, shear, and bending stresses

Figure 3.5: Stress constraint details.

Details of the objective function in Problems 3.1 and 3.2 are specified in Figure 3.6. Note that the Hessian of the objective function, \(H\) is symmetric and indefinite. However, when

\[g_{WF} \leq 0\]

\[u \geq u^L\]
$f(u)$ is strictly positive, and $f$ is strictly convex in the feasible domain. The inclusion of these intuitively obvious constraints are sufficient to ensure positive member weights. In the SAND formulation (Problem 3.2), the presence of nonlinear equality constraints makes the problem nonconvex. The stress constraints are in general nonconvex, so Problem 3.1 is almost always nonconvex.

$$f(u) = \frac{1}{2} u^T H u$$ is the objective function

$$H = \rho_s \begin{bmatrix} H_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & H_n \end{bmatrix}, \quad H_i = L_i \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & -2 \\ 0 & 0 & 0 & 2 \\ 0 & -2 & 2 & 0 \end{bmatrix}$$

$L_i$ is the sum of lengths of members in a single design group

Figure 3.6: Objective function details.

Details on the calculation of code allowable stresses are given in Appendix A. The modified allowable stresses for the smooth formulation are given in Section 5.1. Some timing results comparing function and derivative evaluations for the NAND and SAND formulations are given in Table 3.1, Section 3.3.
3.2 Finite Element Solution

The finite element method (FEM) is used to discretize and solve the partial differential equations (PDEs) governing equilibrium of the structure. Two-dimensional, prismatic beam elements with third order polynomial shape functions and three degrees of freedom (DOF) per node are used. Figure 3.7 shows the FEM sign convention for element degrees of freedom and end forces. **Note:** In that figure “$x$” and “$y$” refer to the axes in the plane of the cross-section, and should not be confused with the state variables, $y$, or the more general optimization variables, $x$, in Problems 3.1 and 3.2.

![Figure 3.7: Force and displacement sign convention for 2D frame element.](image)

The element stiffness matrix $K_{el}$ is 6 by 6 and contains contributions from axial resistance and coupled shear and moment resistance. The element stiffness (and thus the global stiffness) depends on the optimization variables, $u$, as indicated. The element transformation matrix $\Gamma_e$ is 6 by 6 and is used to transform coordinates.
between local (element) and global (structure) coordinate systems.

\[
K_{eL}(u) = \begin{bmatrix}
\frac{EA(u)}{L} & 0 & -\frac{EA(u)}{L} & 0 & 0 \\
0 & \frac{12EI_s(u)}{L^3} & \frac{6EI_s(u)}{L^2} & 0 & -\frac{12EI_s(u)}{L^3} \\
0 & \frac{6EI_s(u)}{L^2} & \frac{4EI_s(u)}{L} & 0 & \frac{6EI_s(u)}{L^2} \\
-\frac{EA(u)}{L} & 0 & \frac{EA(u)}{L} & 0 & 0 \\
0 & -\frac{12EI_s(u)}{L^3} & -\frac{6EI_s(u)}{L^2} & 0 & -\frac{12EI_s(u)}{L^3} \\
0 & \frac{6EI_s(u)}{L^2} & \frac{2EI_s(u)}{L} & 0 & \frac{6EI_s(u)}{L^2}
\end{bmatrix}
\]

\[
\Gamma_e = \begin{bmatrix}
\cos \theta & \sin \theta & 0 & 0 & 0 \\
-\sin \theta & \cos \theta & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & \cos \theta & \sin \theta \\
0 & 0 & 0 & -\sin \theta & \cos \theta \\
0 & 0 & 0 & 0 & 1
\end{bmatrix}
\]

The stiffness method (see [56], pp. 41-59) is used to assemble element stiffness contributions, \( \Gamma_e^T K_{eL} \Gamma_e \) into a global stiffness matrix \( K_G \), which is partitioned into free and support DOFs. The displacements \( y_G \) and applied loads \( p_G \) are partitioned in the same way. The discretized equations of equilibrium are \( K_G y_G = p_G \). The displacements at support DOFs and loads at free DOFs are known. The free displacements and support loads are unknown.

\[
K_G = \begin{bmatrix}
K_{ff} & K_{fs} \\
K_{sf} & K_{ss}
\end{bmatrix}, \quad y_G = \begin{bmatrix}
y_f \\
y_s
\end{bmatrix}, \quad p_G = \begin{bmatrix}
p_f \\
p_s
\end{bmatrix}
\]

Since \( y_s \) is known, the first set of equations in the partitioned system are solved for the unknown displacements, \( y_f \):

\[
K_{ff} y_f + K_{fs} y_s = p_f
\]

so that:

\[
y_f = K_{ff}^{-1} (p_f - K_{fs} y_s)
\]

(3.3)

then the unknown support loads, \( p_s \), are computed directly:

\[
p_s = K_{sf} y_f + K_{ss} y_s
\]

For brevity, \( p_f, y_f, \) and \( K_{ff} \) are usually referred to simply as \( p, y, \) and \( K \). The global stiffness depends on the optimization variables, \( u \), in the same way as the element stiffness. This dependence is omitted for clarity in the global equations.
The existence and uniqueness of \( y \) can be shown using an energy interpretation. The total energy for a static system in equilibrium is the internal (strain) energy minus the external (potential) energy.

\[
E(y) = \frac{1}{2} y^T K y - p^T y
\]

For simplicity, the support displacements \( y_s \) are assumed to be zero, so there is no energy associated with them. At equilibrium, the state \( y \) minimizes \( E(y) \).

\[
\min_y \frac{1}{2} y^T K y - p^T y
\]

which has a solution when:

\[
\nabla E(y) = 0
\]

\[
\Rightarrow K y - p = 0
\]

\[
K y = p
\]

\[
y = K^{-1} p
\]

If \( K \) is invertible, then there is a unique solution, \( y \). The elimination of support degrees of freedom by partitioning the positive semidefinite global stiffness matrix \( K_G \) results in a positive definite and invertible matrix, \( K \) (a.k.a. \( K_{ff} \)) (see Bathe [11], Chapter 8). We retain the caveat that the system must be stable (that is, all degrees of freedom must have some means of resistance). We also suppose that the feasibility restrictions (see Problems 3.1 and 3.2) on the design variables result in a well posed system.

Additionally, \( K \) is symmetric as a consequence of Maxwell’s reciprocal theorem. The most efficient factorization (in terms of computation time and storage) to solve for \( y \) with symmetric positive definite \( K \), is a Cholesky decomposition \( R^T R \), where \( R \) is a right triangular matrix:

let \( R^T R = K \), then \( R^T R y = p \)

let \( z = R y \), then \( z \) solves \( R^T z = p \), and \( y \) solves \( R y = z \)
Since \( R \) is triangular, the solution of \( z \) and \( y \) are accomplished directly by forward and back substitution.

With the displacements known, member forces are calculated from:

\[
p_e(u, y) = K_{el}(u)y_{el}(y) + FEM_e
\]

where: 
\[
y_{el}(y) = \Gamma_e T_e y \equiv \begin{bmatrix} x_A \\ y_A \\ \phi_A \\ x_B \\ y_B \\ \phi_B \end{bmatrix}, \quad FEM_e = \begin{bmatrix} 0 \\ FEV_{AB} \\ FEM_{AB} \\ 0 \\ FEV_{BA} \\ FEM_{BA} \end{bmatrix}
\]

\[
\Delta_x = x_B - x_A, \quad \Delta_y = y_B - y_A,
\]

and: \( T_e \in \mathbb{R}^{6 \times 6} \) maps \( y \) to the member nodal displacements \( FEV_{XX} \) and \( FEM_{XX} \) are constant fixed-end shears/

moments from transverse loads between the member ends.

The usual “strength of materials” sign convention for forces (as in Figure 3.7) is achieved with the following transformation:

\[
\begin{bmatrix} P_A \\ V_A \\ M_A \\ P_B \\ V_B \\ M_B \end{bmatrix} = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_e \end{bmatrix}
\]

(3.6)

Forces within the span can be determined through static analysis treating the member as a free body with applied intermediate and end loads. Element stresses at any point are then calculated as follows:

\[
\begin{align*}
f_a(u, y) &= \frac{P(u, y)}{A(u)} \\
f_v(u, y) &= \frac{V(u, y)}{A_v(u)} \\
f_b(u, y) &= \frac{M(u, y)}{S_z(u)} \\
\end{align*}
\]

(3.7)
The section forces and stresses are thus defined at a general cross-section. **Note:** These vary along the $x$-axis of the member, though the dependency on this variable is omitted for simplicity. The dependency of the stresses on the optimization and state variables ($u$ and $y$) is emphasized here because the stress constraints are of primary importance in the solution of Problems 3.1 and 3.2. Refer to Figure 3.2, Section 3.1 for the definition of the cross-section geometric properties.

Efficient storage of the stiffness matrix can be accomplished using either a banded or sparse storage scheme. Stiffness, $K$, for larger problems is usually very sparse, so this is the better choice. In order to speed up computation using the Cholesky decomposition, a reordering scheme (see [27]) is used on the rows and columns of $K$ to minimize fill in of $R$. If banded storage is used, the preferred reordering is reverse Cuthill-McKee. Using sparse storage, the better scheme is minimum degree ordering.
3.3 Derivatives and Sensitivities

The use of derivatives in optimization is essential for efficiency of solution. Finite
difference derivatives are inaccurate and time consuming to evaluate, so analytical
derivatives are preferred. Gradients are much easier to evaluate than Hessians, but
both can be computed for Problems 3.1 and 3.2. Since the objective function $f(u)$ is
quadratic, its derivatives are simple to evaluate:

$$f(u) = \frac{1}{2} u^T H u$$
$$\nabla f(u) = H u$$
$$\nabla^2 f(u) = H$$

Similarly, derivatives of the box constraint functions are trivial:

$$\nabla (u^L - u) = -1$$
$$\nabla (u - u^U) = 1$$

where: $1 = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}$

Derivatives of the constraint functions $g$ and $c$ are more involved. But, since the
component $g_g$ is only a function of $u$, its derivatives are relatively straightforward. The
derivatives are evaluated using the chain rule. Derivatives of the geometric properties
are built up from their relationships with one another and the basic design variables
$u = [d \ t_w \ b_f \ t_f]^T$. For example, the second equation in $g_g$ has derivatives as shown
below (dependencies on $u$ are omitted for clarity in the derivative expressions):

$$g_{g2}(u) = \frac{K_y L_y}{r_y(u)} - \left( \frac{K L}{r} \right)^U$$

where: $r_y(u) = \sqrt{\frac{I_y(u)}{A(u)}}$

$$A(u) = 2b_f t_f + h(u) t_w$$
\[ I_y(u) = \frac{1}{12} (2t_f b_f^3 + h(u)t_w^3) \]
\[ h(u) = d - 2t_f \]

differentiating once:
\[ \nabla g_{g2}(u) = -\frac{K_y L_y}{r_y^2} \nabla r_y = -\frac{g_{g2}}{r_y} \nabla r_y \]
where: \( \nabla r_y(u) = \frac{1}{2} \left( \frac{I_y}{A} \right)^{-1/2} \frac{1}{A^2} \left( A \nabla I_y - I_y \nabla A \right) \)
\[ = \frac{1}{2A r_y} \left( \nabla I_y - \frac{I_y}{A} \nabla A \right) \]
\[ = \frac{1}{2A} \left( \frac{1}{r_y} \nabla I_y - r_y \nabla A \right) \]
\[ \nabla A(u) = \begin{bmatrix} t_w \\ h \\ 2t_f \end{bmatrix}, \quad \nabla I_y(u) = \begin{bmatrix} \frac{1}{12} t_w^3 \\ \frac{1}{2} h(t_w^2) \\ \frac{1}{2} t_f b_f^2 \end{bmatrix} \]

differentiating twice:
\[ \nabla^2 g_{g2}(u) = -\frac{1}{r_y^2} \nabla r_y \left( r_y \nabla g_{g2}^T - g_{g2} \nabla r_y^T \right) - \frac{g_{g2}}{r_y} \nabla^2 r_y \]
\[ = -\frac{1}{r_y} \nabla r_y \left( \frac{g_{g2}}{r_y} - \frac{g_{g2}}{r_y} \right) \nabla r_y^T - \frac{g_{g2}}{r_y} \nabla^2 r_y \]
\[ = \frac{g_{g2}}{r_y^2} \left( \frac{2}{r_y} \nabla r_y \nabla r_y^T - \nabla^2 r_y \right) \]
where: \( \nabla^2 r_y(u) = \frac{1}{2A} \left[ \frac{1}{r_y^2} \left( r_y \nabla^2 I_y - \nabla I_y \nabla r_y^T \right) - r_y \nabla^2 A - \nabla A \nabla r_y^T \right] \]
\[ - \frac{1}{2A^2} \left( \frac{1}{r_y} \nabla I_y - r_y \nabla A \right) \nabla A^T \]
\[ = \frac{1}{2A} \left[ \frac{1}{r_y} \nabla^2 I_y - \frac{1}{2A} \left( \frac{1}{r_y^2} \nabla I_y + \nabla A \right) \left( \frac{1}{r_y} \nabla I_y - r_y \nabla A \right)^T \right. \]
\[ - \frac{1}{Ar_y} \nabla I_y \nabla A^T + \frac{r_y}{A} \nabla A \nabla A^T - r_y \nabla^2 A \]
\[ = \frac{1}{2A} \left( \frac{1}{r_y} \nabla^2 I_y - \frac{1}{2Ar_y} \nabla I_y \nabla I_y^T + \frac{1}{2Ar_y} \nabla I_y \nabla A^T - \frac{1}{2Ar_y} \nabla A \nabla I_y^T \right. \]
\[ + \frac{r_y}{2A} \nabla A \nabla A^T - \frac{1}{Ar_y} \nabla I_y \nabla A^T + \frac{r_y}{A} \nabla A \nabla A^T - r_y \nabla^2 A \right) \]
\[
\begin{align*}
\n\n= & \frac{1}{2A} \left[ \frac{1}{r_y} \nabla^2 I_y - \frac{1}{2Ar_y^3} \nabla I_y \nabla I_y^T + \frac{1}{2A} \nabla A \nabla A^T - r_y \nabla^2 A \right] \\
= & \frac{1}{4A^2} \left[ 2A \nabla^2 I_y - \frac{1}{r_y^3} \nabla I_y \nabla I_y^T \right. \\
& \left. - \frac{1}{r_y} \left( \nabla I_y \nabla A^T + \nabla A \nabla I_y^T \right) + 3r_y \nabla A \nabla A^T - 2Ar_y \nabla^2 A \right]
\end{align*}
\]

\[\nabla^2 A(u) = \begin{bmatrix}
0 & 1 & 0 & 0 \\
1 & 0 & 0 & -2 \\
0 & 0 & 0 & 2 \\
0 & -2 & 2 & 0
\end{bmatrix}, \quad \nabla^2 I_y(u) = \begin{bmatrix}
0 & \frac{1}{4}t_w^2 & 0 & 0 \\
\frac{1}{4}t_w^2 & \frac{1}{2}ht_w & 0 & -\frac{1}{2}t_w^2 \\
0 & 0 & b_t t_f & \frac{1}{2}b_f^2 \\
0 & -\frac{1}{2}t_w^2 & \frac{1}{2}b_f^2 & 0
\end{bmatrix}\]

Note that the Hessians are all symmetric. If \(y\) depends on \(u\) (as in Problem 3.1), derivatives of \(g(u, y(u))\) require further use of the chain rule and the determination of sensitivities of \(y\) with respect to \(u\). When the number of constraints is larger than the number of variables, using sensitivities is more efficient than the adjoint method for gradients, requiring solution of \(n_u\) linear systems as compared to \(n_g\) for stress constraints (typically \(n_g \gg n_u\)). If Hessians are computed, sensitivities are much more efficient, requiring the solution of an additional \(n_u^2\) linear systems, while adjoints require \(2n_g n_u\) more.

The dependence of \(y\) on \(u\) is only explicitly stated in cases where it need be emphasized, as below. Jacobian matrices such as the matrix of partial derivatives of \(g_s\) with respect to \(u\) are denoted \((g_s)_u\). In general for:

\[\hat{g}_s(u) = g_s(u, y(u))\]
\[(\hat{g}_s)_u = (g_s)_u + y_u^T (g_s)_y\]
\[(\hat{g}_s)_{uu} = (g_s)_{uu} + y_u^T (g_s)_{yy} + \left[ y_u^T (g_s)_{yy} + (g_s)_{uy} \right] y_u + y_{uu}^T (g_s)_y\]

The Jacobians \(y_u \in \mathbb{R}^{n_y \times n_u}\) and \(y_{uu} \in \mathbb{R}^{n_y \times n_u \times n_u}\) are the first- and second-order
sensitivities. These are determined by differentiating the state equations:

\[
\text{if } y(u) \text{ solves } c(u, y(u)) = 0
\]

then \( c_u + c_y y_u = 0 \)

\[\Rightarrow y_u = -c_y^{-1} c_u \]

if \((c_u + c_y y_u) v = 0, \text{ for } v = e_i, i = 1, \ldots, n_u\)

\[
(c_{uu} + c_{uy} y_u) v + c_y y_{uu} v + v^T y_{uu}^T (c_{yu} + c_{yy} y_u) = 0
\]

\[\Rightarrow y_{uu} v = -c_y^{-1} [(c_{uu} + c_{uy} y_u) v + v^T y_{uu}^T (c_{yu} + c_{yy} y_u)] \]

Because \( y_{uu} \) is a tensor, its elements are determined \( n_y \) by \( n_u \) matrix-wise using coordinate vectors \( e_i \). Since partial derivatives of the state equations give:

\[
c_u = y^T K_u, \quad c_y = K
\]

\[c_{uu} = y^T K_{uu}, \quad c_{uy} = K_u^T, \quad c_{yu} = K_u, \quad c_{yy} = 0\]

the sensitivities can be computed in terms of the stiffness \( K \), and displacements, \( y \).

Note that the Jacobian of \( c \) with respect to \( y \) \( (c_y) \) is the stiffness matrix \( K \), so that calculation of \( y_u \) and \( y_{uu} \) can use the Cholesky factorization from the solution of \( y \) (see equation 3.3, Section 3.2).

Derivatives of \( g_s \) and \( c \) do not require sensitivities if \( u \) and \( y \) are considered independent variables (as in Problem 3.2). For \( x = \binom{u}{y} \), they are simply the combined partial derivatives with respect to \( u \) and \( y \):

\[
(g_s)_x = \begin{bmatrix} (g_s)_u \\ (g_s)_y \end{bmatrix}, \quad (g_s)_{xx} = \begin{bmatrix} (g_s)_{uu} & (g_s)_{uy} \\ (g_s)_{yu} & (g_s)_{yy} \end{bmatrix}
\]

\[c_x = \begin{bmatrix} c_u \\ c_y \end{bmatrix}, \quad c_{xx} = \begin{bmatrix} c_{uu} & c_{uy} \\ c_{yu} & c_{yy} \end{bmatrix}\]

The partial derivatives of \( g_s \) and \( K \) are computed using the chain rule, as shown for \( g_g \) above.

Typically analytical derivatives reduce the number of function evaluations required to reach a solution. Table 3.1 gives function evaluation timing data for two models
from the test set. In both formulations, function evaluations plus gradients take an order of magnitude longer than function evaluations alone. Since finite difference gradients need one or two function evaluations for each partial derivative, there is little loss of efficiency in using finite differences for a small number of variables. Although for a larger number of variables, analytical gradients are more efficient. In the NAND formulation, function evaluations plus gradients and Hessians take 10 to 60 times longer than function evaluations and gradients alone. In the SAND formulation the ratio is only about 3 to 1.

<table>
<thead>
<tr>
<th>Test Model</th>
<th>Order</th>
<th>Average Time (sec)(^a)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>bigframe</td>
</tr>
<tr>
<td>NAND (Problem 3.1)</td>
<td>2</td>
<td>1320</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>21.9</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>1.11</td>
</tr>
<tr>
<td>SAND (Problem 3.2)</td>
<td>2</td>
<td>67.4</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>24.8</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>1.09</td>
</tr>
</tbody>
</table>

Table 3.1: Function evaluation timing data.

\(^a\)for evaluation of nonlinear constraints and analytical derivatives
\(^b\)2\(^{nd}\) order includes Hessians and gradients, 1\(^{st}\) order includes gradients, and 0 order is a simple function evaluation without derivatives.

In forming analytical Hessians for the NAND formulation much of the time goes into computing the second-order sensitivities \(y_{uu}\), which are not computed for the SAND formulation. Based on the timing results in Table 3.1, there is a significant need for an alternative to computing the analytical Hessian in the NAND, but not in the SAND formulation (see for example, the BFGS method, described in Section 2.2.4). Use of analytical Hessians will thus depend on which formulation is used and which method is used to solve the optimization problem. Calculating analytical gradients is clearly beneficial in both formulations.
3.4 Optimization Test Models

Test models have been developed to aid in software and algorithm development. These include:

1. Model “test19x” (see Figure 3.8), a two span continuous beam with two design groups and one load case. This is a small problem that is solved in under a minute. Each specific test19x model depends on various geometry and load parameters $L_1$, $L_2$, $P_1$, $P_2$, $a_1$, $a_2$, and design parameters, $L_{B1}$ and $L_{B2}$ (see Table 3.3 in Section 3.6, where the ID in the table replaces the “x” to form the model name as in “test19d1a”). Various different solutions governed by different failure modes can be devised by varying these parameters (see [77]). Introducing equality constraints for six of the eight design variables results in a two-dimensional problem that is easily studied graphically.

2. Model “test20a” (see Figure 3.9), a one story, one bay portal frame with two design groups and one load case. This is a small problem that is solved in under a minute. It includes flexural and compression members so that all of the stress checks are applicable.

3. Model “test21a” (see Figure 3.9), a six span continuous beam with six design groups and one load case. This problem takes slightly longer to solve (about two minutes). The beams in groups G-3, G-4, and G-5 have nearly the same forces in the solution, so that some active constraints may have nearly collinear gradients. This model may exhibit difficulties due to dependent constraints.

4. Model “test23” (see Figure 3.9), a three story, two bay, braced frame with sloping columns. It has five design groups and two load cases (the second is shown). It takes more than ten minutes to solve. Due to symmetry the moment in columns in group G-2 is zero. However, a nonzero value is usually obtained due to roundoff in solving the state equations. This model may exhibit
some of the behavior described for nonsmooth problems because the absolute value function is used in the bending stress constraint and \(|f_b| > 0\), but small perturbations in \(u\) can modify the sign of \(f_b\).

5. Model "bigframe" (see Figure 3.10), a twelve story, five bay plane frame. It has eighteen design groups and two load cases. This is a medium size problem and can take over an hour to solve. This model is designed for studying the efficiency of different aspects of the optimization.

![Figure 3.8: Two-dimensional structure (test model 19).](image)

Except for certain cases of the model test19x, the test problems are nonsmooth, and nonconvex (have more than one local minimum). See Table 3.2 for the full list of problem size parameters. For additional reference, two three-dimensional space frame models\(^2\) are included in the table and shown in Figure 3.11.

The parameters in Table 3.2 indicate that the problem size is in the medium range \((n_u < 500)\) for the NAND formulation. However, if state variables are included, problem sizes get very large \((n_x > 1000)\). The constraints are counted assuming one check of the combined axial and bending stress plus one shear stress check per section.

\(^2\)The 3D models are not designed using the current software, although the optimization algorithm outlined in Chapter 6 is equally applicable to these structures.
Figure 3.9: Small optimization test models.

Table 3.2: Test model problem data.

and allows for four stress constraints per column (2 sections) and six per beam (3 sections) for each load case. Clearly the stress constraints are under-represented; however, the trend shows that the number of constraints is easily in the thousands.
Figure 3.10: Medium size test model bigframe \((n_u = 72)\).

If additional redundant stress constraints are added, this may number in the tens of thousands for a medium size structure (e.g. the 400 foot offshore platform). The number of nonzero entries in the stiffness matrix is shown to give an idea of the memory required to store derivatives of the state equations, which require \(K_{uu}\) (only the nonzero entries need be computed and stored).
Figure 3.11: 3D models—50 story office building (left) & 400 ft offshore platform (right).
3.5 Optimization Software

Four different optimization algorithms are used in this study: two pattern search methods (augmented Lagrange, and filter), the interior point method, and the sequential quadratic programming method (see Section 2.2). The software system is written almost entirely in the Matlab\textsuperscript{3} programming language, except for some interface routines written in C. Third-party packages have been used when software was readily available, otherwise new software was written.

The augmented Lagrange algorithm (Section 2.2.1) is based on Lancelot (Conn, Gould, & Toint [24], Section 3.4), with extensions for pattern search by Lewis and Torczon [51]. Lancelot has good documentation and has been widely used and tested. The penalty parameter $\mu$ is decreased until a suitable level of feasibility is obtained. Initially, the Lagrange multipliers are set to zero. This results in the algorithm acting as a penalty function method for a few iterations until the first multiplier estimates are computed. Once feasibility is acceptable, the Lagrange multipliers are updated. At the $k$-th iteration:

$$\lambda_{k+1} = \lambda_k + \max \left( 0, \frac{g(x_k)}{\mu_k} \right)$$  \hspace{1cm} (3.8)

The feasibility and optimality stopping criteria are:

$$h(x_k) \leq \eta$$
$$\delta_k \leq \omega$$

where: $x_k$ is the approximate solution at the $k$-th iteration
$\eta$ is the user-defined feasibility tolerance
$\omega$ is the user-defined optimality tolerance
$$\delta_k = \frac{\omega_k}{1 + \|\lambda_k\|_2 + \frac{1}{\mu_k}}$$
$$\omega_k$$ is the required optimality tolerance at the $k$-th iteration

\textsuperscript{3}©1984-2005, The Mathworks, Inc.
Note that the complementarity condition, KKT equation 2.5 is automatically satisfied at the solution because equation 3.8 sets the Lagrange multipliers to zero for inactive constraints.

A default value of 0.1 is used for $\mu_0$. The starting value of the penalty parameter may affect the number of iterations required to reach a solution, but there is no general method to determine a suitable value. Other algorithm parameters use the recommended starting values in the Lancelot manual. The augmented Lagrange algorithm is a Matlab macro, but the gradient-projection method (for subproblems) is written in Fortran, and accessed through a C interface. The Fortran code is part of the public domain package Lancelot.

The pattern search filter method is based on specifications for the algorithm by Audet and Dennis [7], using multiple poll centers. This software is part of the research package NOMADm [1] developed for Matlab by Mark A. Abramson. The filter is used to keep infeasible points, presumably near the optimum for polling in addition to polling around the current iterate. In this way, it is possible to make good progress along the border of the feasible region.

Trial iterates $x_k$ are considered feasible when $h(x_k) \leq \eta$, where $\eta$ is the user-defined feasibility tolerance. Infeasible points fall into two categories and are either added to the filter when $\eta < h(x_k) \leq 1$, or disregarded. The optimality stopping criteria is:

$$\Delta_k \leq \omega$$

where: $\Delta_k$ is the mesh parameter at the $k$-th iteration

$\omega$ is the user-defined optimality tolerance

A successful iteration is declared when polling either improves the current "best feasible" iterate or adds a new point to the filter. The mesh parameter is controlled
by:

$$\Delta_{k+1} = \begin{cases} 
\sigma \Delta_k, & \text{if the iteration is successful} \\
\tau \Delta_k, & \text{otherwise}
\end{cases}$$ \quad (3.9)

where: $\sigma \geq 1$ is the mesh coarsening factor

$0 < \tau < 1$ is the mesh refining factor

The mesh update parameters are taken as $\sigma = 2$ and $\tau = 0.5$. Search directions are from the maximal positive basis $[I - I]$ as in the method of Hooke and Jeeves [38], allowing movement parallel to active box constraints. The search directions are scaled so that unit moves are relatively equivalent in all directions, regardless of the size of the optimization variable.

The GPS for subproblems in the AL (which also functions as a core for the filter method), is based on an algorithm for solving a linearly constrained problem (Lewis and Torczon [49] and [50]). At the beginning of each problem, the initial solution is checked for feasibility with respect to the linear constraints. If it is infeasible, a feasible starting point is found using a phase-1 linear program (see [28], pp. 311-316). For bound constrained problems, the maximal positive basis $[I - I]$ is used as in the method of Hooke and Jeeves [38], allowing movement parallel to active constraints. When general linear constraints are active, this set of search directions is insufficient. It is modified to include a linearly independent set of generators for the tangent cone of the active constraints (see [50] for the generation algorithm and details). Using this approach all generated iterates are feasible with respect to the bound and linear constraints. Since this method is used to solve subproblems for the augmented Lagrange algorithm, the feasibility of the nonlinear constraints is handled externally. The optimality stopping criteria is:

$$\Delta_k \leq \omega$$
where: $\Delta_k$ is the mesh parameter at the $k$-th iteration

$\omega$ is the user-defined optimality tolerance

A successful iteration is declared when polling improves the current iterate. The mesh parameter is controlled in the same way as the filter method (see above).

The interior point software is the research and commercial package, Knitro\textsuperscript{4} (Waltz & Nocedal [76]). The interface to Matlab is written in C. The main routine takes the Matlab objects, converts the data to arrays, and calls the reverse communication Knitro function repeatedly until a stopping point is reached. The feasibility and optimality stopping criteria are (also see Section 2.2.1 and Definition 2.2.2):

\[
\|g_+(x_k)\|_\infty \leq \eta \|g_+(x_0)\|_\infty \\
\|\nabla_x L(x_k, \lambda_k)\|_\infty \leq \omega \|\nabla f(x_k)\|_\infty \\
\|\lambda_k \circ g(x_k)\|_\infty \leq \omega
\]

where: $\| \cdot \|_\infty$ is the infinity norm or element of maximum magnitude

$\lambda_k \circ g(x_k)$ is the component-wise product of $\lambda_k$ and $g(x_k)$

$(x_k, \lambda_k)$ is the approximate solution at the $k$-th iteration

$x_0$ is the initial solution

$\eta$ is the user-defined feasibility tolerance

$\omega$ is the user-defined optimality tolerance

Auxiliary routines take the results of function calls (objective and constraints) as Matlab objects and convert these to function, gradient, and Hessian information for use in Knitro.

The SQP software is native to Matlab. It uses an active set strategy to solve the QP subproblems, and a line search method to provide sufficient decrease in the merit

\textsuperscript{4}\textsuperscript{©}2001-2005 by Northwestern University.
function:

$$\Phi(x_k, \lambda_k) = f(x_k) + r_k^T g_+(x_k)$$  \hspace{1cm} (3.10)

where: $r_k = \max(\lambda_k, \frac{1}{2}(r_{k-1} + \lambda_k))$

initially $(r_0)_i = \frac{\|\nabla f(x_0)\|}{\|\nabla g_i(x_0)\|}$, $i = 1, \ldots, n_g$

The software requires first derivatives of the objective and constraints and uses a BFGS update to estimate the Hessian of the Lagrangian. The feasibility and optimality stopping criteria are (also see above):

$$\|g_+(x_k)\|_{\infty} \leq \eta$$

$$\|\nabla \mathcal{L}(x_k, \lambda_k)\|_{\infty} \leq \omega$$

where: $(x_k, \lambda_k)$ is the approximate solution at the $k$-th iteration

$\eta$ is the user-defined feasibility tolerance

$\omega$ is the user-defined optimality tolerance

Note that the complementarity condition, KKT equation 2.5, is automatically satisfied for feasible solutions because the active set method sets the Lagrange multipliers of inactive constraints to zero. See the user’s guide [55], Chapters 2, 3, and 4 for further details.

### 3.6 Effects of Nonsmooth Constraints

Convergence theory for optimization algorithms draws conclusions based on the worst-case behavior. In the event that a problem is attempted that has the worst-case qualities, the algorithm should still behave as intended. Conversely, even though an algorithm is not guaranteed to converge on a certain class of problem, it may in fact succeed. A study [77] was made to determine the extent to which nonsmooth constraints would affect the convergence of (a) the augmented Lagrange algorithm
with subproblems solved using a gradient-projection method (Lancelot) and (b) the interior point method (Knitro). Method (c), the augmented Lagrange with pattern search, was used as a benchmark. Since methods (a) and (b) make assumptions regarding the continuity of the partial derivatives of the constraint functions, the intent was to show that these would fail to solve some or all of the nonsmooth problems, while method (c) would succeed on all of them. As discussed in Section 2.2.2, direct search methods do not use derivatives nor do they rely on continuity.

The test set of problems was based on the model test19x (see Figure 3.8, Section 3.4). Parameters (see Table 3.3) defining span length \( L \), load placement \( a \) and magnitude \( P \), unbraced length \( L_b \), and moment magnification coefficient \( C_b \) were varied to create scenarios that were smooth (cases c, d1, d2, d3, and d4), nonsmooth (cases d2a and d4a), or discontinuous (cases d1a and d2b) at the solution.

<table>
<thead>
<tr>
<th>ID</th>
<th>( L_1 ) (ft)</th>
<th>( a_1 ) (ft)</th>
<th>( L_b1 ) (ft)</th>
<th>( C_b1 ) (k)</th>
<th>( P_1 ) (ft)</th>
<th>( a_2 ) (ft)</th>
<th>( L_b2 ) (ft)</th>
<th>( C_b2 ) (k)</th>
<th>( P_2 ) (k)</th>
<th>( F_y ) (ksi)</th>
</tr>
</thead>
<tbody>
<tr>
<td>c</td>
<td>30</td>
<td>15</td>
<td>0</td>
<td>1</td>
<td>60</td>
<td>20</td>
<td>10</td>
<td>0</td>
<td>1</td>
<td>40</td>
</tr>
<tr>
<td>d1</td>
<td>30</td>
<td>10</td>
<td>5</td>
<td>1</td>
<td>60</td>
<td>5</td>
<td>n/a</td>
<td>5</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>d1a</td>
<td>30</td>
<td>10</td>
<td>5</td>
<td>1</td>
<td>35</td>
<td>5</td>
<td>n/a</td>
<td>5</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>d2</td>
<td>30</td>
<td>10</td>
<td>10</td>
<td>1</td>
<td>60</td>
<td>5</td>
<td>n/a</td>
<td>5</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>d2a</td>
<td>30</td>
<td>10</td>
<td>10</td>
<td>1</td>
<td>38.854</td>
<td>5</td>
<td>n/a</td>
<td>5</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>d2b</td>
<td>30</td>
<td>10</td>
<td>10</td>
<td>1</td>
<td>160</td>
<td>5</td>
<td>n/a</td>
<td>5</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>d3</td>
<td>30</td>
<td>10</td>
<td>15</td>
<td>1</td>
<td>60</td>
<td>5</td>
<td>n/a</td>
<td>5</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>d4</td>
<td>30</td>
<td>10</td>
<td>30</td>
<td>1</td>
<td>60</td>
<td>5</td>
<td>n/a</td>
<td>5</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>d4a</td>
<td>30</td>
<td>10</td>
<td>30</td>
<td>1</td>
<td>1.6681</td>
<td>5</td>
<td>n/a</td>
<td>5</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 3.3: Test model 19 parameter data.

In order to better study these problems, the number of optimization variables was
reduced to two ($d_1$ and $d_2$) by using equality constraints for both design groups:

\[ t_w = \frac{d \sqrt{F_y}}{640} \]
\[ b_f = \frac{d}{3} \]
\[ t_f = \frac{b_f \sqrt{F_y}}{2} \frac{65}{65} \]

The design space was plotted in two-dimensions with the stress constraint functions overlaid on contours of the objective (see Figures 3.12(a) and 3.12(b)). In Figure 3.12(a) two local solutions are marked. Solution 1 is the global minimum and occurs at a point where the constraint on moment $M_{CE}$ is discontinuous. A range of feasible points are found when $d_2$ is about 21 and $d_1$ is between about 25 and 27. The minimum thus occurs roughly at $u^* = [25, \ 21]^T$. In Figure 3.12(b) two local solutions are marked. Solution 1 is the global minimum at $d_2 = 10$ and $d_1$ about 18. Solution 2 is a local minimizer occurring roughly at $u^* = [18, \ 12]^T$. Solution 2 occurs at a point where the constraint on moment $M_{B}$ is nonsmooth. Regions of nonsmoothness are generally less visually detectable than discontinuities.

![Diagram](image)

(a) Case d1a: $P = 35, L_b = 60, C_b = 1$. 
(b) Case d4a: $P = 1.6681, L_b = 360, C_b = 1$.

Figure 3.12: Nonsmooth design space of test model test19x.

Table 3.4 summarizes the results of attempting to solve the design examples using
the three algorithms. For each problem the number of function evaluations and the optimal value of the objective are tabulated. Cases where the method failed to converge are noted in the table. Some cases using Lancelot returned a solution that was not optimal to the required tolerance. However, upon inspection, the results were near the solution (also noted in the table).

<table>
<thead>
<tr>
<th>ID</th>
<th>AL (Lancelot)</th>
<th>AL (Pattern Search)</th>
<th>IP (Knitro)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Function Eval'ns $f(u^*)$</td>
<td>Function Eval'ns $f(u^*)$</td>
<td>Function Eval'ns $f(u^*)$</td>
</tr>
<tr>
<td>c</td>
<td>153 2.675</td>
<td>1772 2.675</td>
<td>11 2.675</td>
</tr>
<tr>
<td>d1</td>
<td>411 1.954</td>
<td>1627 1.954</td>
<td>10 1.954</td>
</tr>
<tr>
<td>d1a</td>
<td>* -</td>
<td>1850 1.365</td>
<td>* -</td>
</tr>
<tr>
<td>d2</td>
<td>202 2.075</td>
<td>1726 2.075</td>
<td>12 2.075</td>
</tr>
<tr>
<td>d2a</td>
<td>4358† 1.553</td>
<td>1966 1.553</td>
<td>13 1.553</td>
</tr>
<tr>
<td>d2b</td>
<td>* -</td>
<td>1586 3.894</td>
<td>* -</td>
</tr>
<tr>
<td>d3</td>
<td>510 2.169</td>
<td>1789 2.169</td>
<td>11 2.169</td>
</tr>
<tr>
<td>d4</td>
<td>140 2.909</td>
<td>21247 2.909</td>
<td>16 2.920</td>
</tr>
<tr>
<td>d4a</td>
<td>15761† 0.695</td>
<td>24884 0.695</td>
<td>13 0.695</td>
</tr>
</tbody>
</table>

Table 3.4: Test model 19 solution results using three algorithms.

*Failed to converge.  
†Did not meet required stopping criteria.

Among the nonsmooth problems: “d1a” and “d2b” (with discontinuous constraints) failed to converge with Knitro and Lancelot, but were solved by the pattern search method, while problems “d2a” and “d4a” (with continuous, but nonsmooth constraints) converged relatively well with Knitro, but very slowly with Lancelot (failing to meet the optimality criteria). Problems “d4” and “d4a” were the only two for which the number of function evaluations using the pattern search method varied significantly, although it converged satisfactorily on all problems. Knitro found a local minimizer of problem “d4”, but in all other cases the algorithms (if successful) found the global solution.

As expected, the only robust algorithm is the pattern search method. The ability of the interior point method to work well on continuous but nonsmooth problems
does not indicate that this is the case in general. However, it suggests that a certain amount of nonsmoothness is tolerable. This is the basis for the Two-Stage algorithm, outlined in Chapter 6, which solves a problem with nonsmooth constraints by first solving a nearby smooth problem, then successively solving more and more nonsmooth variants of the original problem.

3.7 Summary

The implicit state (NAND) and explicit state (SAND) formulations have been detailed for the minimum weight design problem. The integration of analysis and design has been discussed in the solution of these formulations and for derivative calculations. Details of the finite element analysis have been given for reference, and to clarify the solution of the equations of equilibrium, or state equations.

A set of test models has been described for algorithm verification. The software used for optimization problem solution has been discussed to provide some understanding of the implementation of the tests.

Finally, the detrimental effects of nonsmooth constraints using optimality criteria methods have been clearly demonstrated with the study of a small test structure. It has been shown that only direct search is a robust method for the class of problems that arise in structural engineering in which the allowable stresses are defined piecewise over the domain of their design variables.
Chapter 4

Study of Failure Mechanisms

The development of a smooth formulation requires the use of alternative allowable stress equations. In order to remain within the bounds of the design code, these alternatives must be justified by analysis. In particular, failure of members by:

- Lateral-torsional buckling (Section 4.2),
- Flange local buckling (Section 4.3), and
- Web local buckling (Section 4.4)

is studied, with particular interest in inelastic buckling failure. Section 4.1 presents a stress-strain model, which is used in all of the following analysis.

Note: In this chapter the symbols “x”, “y” and “θ” refer to directions or displacements in the structural engineering sense (rather than optimization or state variables). The symbol “λ” refers to a slenderness parameter (rather than to a Lagrange multiplier). The symbol “η” refers to a reduced modulus factor (rather than to a feasibility tolerance). Although the failure stresses derived in this chapter are a function of the optimization variables u, the dependency is omitted in this chapter to avoid confusion.
4.1 Stress-Strain Model

A nonlinear and ductile stress-strain model is assumed, so that stiffness associated with plastic strains may be properly represented. The tangent modulus, $E_t = \frac{d\sigma}{d\varepsilon}$ (the slope of the stress strain curve), is assumed to vary according to:

$$
E_t(\varepsilon) = \begin{cases} 
E, & 0 \leq \varepsilon \leq \varepsilon_p \\
E \frac{\varepsilon - \varepsilon_p}{\varepsilon_y - \varepsilon_p}, & \varepsilon_p < \varepsilon \leq \varepsilon_y \\
0, & \varepsilon_y < \varepsilon \leq \varepsilon_{st} \\
E_{st} \frac{\varepsilon - \varepsilon_{st}}{\varepsilon_f - \varepsilon_{st}}, & \varepsilon_{st} < \varepsilon \leq \varepsilon_f \\
E_{st}, & \varepsilon_f < \varepsilon \leq \varepsilon_u \\
0, & \varepsilon > \varepsilon_u 
\end{cases}
$$

(4.1)

where: $\varepsilon$ is the normal strain

$F_u$ is the ultimate stress

$F_y$ is the yield stress

$F_p$ is the proportional limit

$E = 29,000$ ksi $(200,000$ MPA$)$ is the elastic modulus

$E_{st} = 700-900$ ksi $(4800-6200$ MPA$)$ is the strain hardening modulus

$\varepsilon_p = \frac{F_p}{E}$ is the strain at the proportional limit

$\varepsilon_y = \frac{1}{E} (2F_y - F_p)$ is the strain at yield

$\varepsilon_{st}$, assumed $= 10\varepsilon_y$ is the strain at onset of strain hardening

$\varepsilon_f$, assumed $= 1.05\varepsilon_{st}$ is the strain at full strain hardening

$\varepsilon_u = \frac{1}{E_{st}} (F_u - F_y) + \frac{1}{2} (\varepsilon_f + \varepsilon_{st})$ is the ultimate strain

This model is chosen so that the tangent modulus is continuous, and the stress (as a function of strain) is continuously differentiable or smooth. Further, by choosing a piecewise linear curve, the proportional limit distinctly marks the transition to
nonlinear (inelastic) behavior. The magnitude of the proportional limit is chosen to indirectly account for the presence of residual stresses\(^1\). Unloading of the material is not modeled, as inelastic buckling theory assumes no strain reversal takes place [67]. The stress-strain relationship is defined by:

\[
\sigma(\epsilon_0) = \int_0^{\epsilon_0} E_\epsilon(\epsilon) d\epsilon
\]  

(4.2)

so that for a particular value of normal strain, \(\epsilon_0\), the corresponding normal stress can be determined. Figure 4.1(a) shows the stress-strain curve for A441 steel (\(F_y = 50\) ksi (340 MPa), \(F_u = 70\) ksi (480 MPa)), with coupon test data overlaid. Figure 4.1(b) shows a local portion of the curve to strain hardening. The yield strain, \(\epsilon_y\), is 0.25\%. The onset of strain-hardening, \(\epsilon_{st}\), is at ten times the yield strain, or 2.5\%. The ultimate strain of the model, \(\epsilon_u\), is about 5.5\%. The model is undefined beyond this strain, but strains in excess of 2\(\epsilon_{st}\) are rarely required.

The ratio of transverse contraction to longitudinal extension, or Poisson’s ratio, is associated with elastic strains, up to the proportional limit. For plastic strains, the ratio is \(\frac{1}{2}\). For elasto-plastic strains, the ratio is the weighted average of elastic and plastic strains (see [63] pp. 5-6). In general:

\[
\nu_{\epsilon}(\epsilon) = \begin{cases} 
\nu, & 0 \leq \epsilon \leq \epsilon_p \\
\nu' + \frac{1}{2} \frac{\epsilon'}{\epsilon' + \epsilon''}, & \epsilon > \epsilon_p
\end{cases}
\]  

(4.3)

\(^1\)A proportional limit of 0.5\(F_y\) is usually taken for uniformly compressed flanges. In accordance with the current code [3] the slightly higher value of 0.55\(F_y\) is used for lateral-torsional buckling, which considers the flexural stress variation over the cross-section. A proportional limit of 0.8\(F_y\) is usually taken for webs in shear. This reflects the lower residual stresses seen in webs, which have less restraint than flanges.
where: $\nu$ is Poisson's ratio (0.3 for steel)

$$\epsilon'(\epsilon) = \frac{\sigma(\epsilon)}{E}$$ is the elastic strain

$$\epsilon''(\epsilon) = \epsilon - \epsilon'$$ is the plastic strain

$\sigma(\epsilon)$ is the normal stress defined by equation 4.2

Figure 4.1 shows the Poisson’s ratio model up to about 5 times the yield strain. Strain at the proportional limit of about 0.09% is marked by a dashed line and above this, $\nu$ is greater than its elastic value of 0.3. As the strain increases, $\nu$ approaches its
theoretical plastic limit of 0.5 (shown by a dashed line).

Figure 4.2: Poisson’s ratio (\(\nu\)) model.
4.2 Lateral-Torsional Buckling

The lateral-torsional buckling of flexural members (LTB) is primarily a function of the beam slenderness, or the unbraced length divided by the weak-axis radius of gyration. Over the range of slenderness, failure is controlled by yield for the lowest values, and by elastic buckling for the highest. In the intermediate range, when the stress exceeds the proportional limit, inelastic buckling theory predicts the failure load. Figure 4.3 shows the buckled configuration associated with lateral-torsional buckling. Buckling is due to simultaneous twisting ($\beta$) and lateral displacement ($u$) of the beam.

Figure 4.3: Lateral-torsional buckling behavior (original figure by A. Chajes [20]).

A general solution for elastic buckling exists with some limitations; however, the solution for inelastic buckling must be obtained numerically. The current study uses finite element analysis to solve the differential equations of equilibrium for lateral-torsional buckling. In Section 5.1.3.1 this solution is used to validate a simple smooth equation, which is proposed for design.

The differential equations of equilibrium (see [14], articles 47 and 49) for a trans-
versely loaded wide flange beam in lateral-torsional buckling are:

\[ EI_y \frac{d^4 u}{dz^4} + \frac{d^2}{dz^2} (M \beta) = 0 \]  \hspace{1cm} (4.4)

\[ EC_w \frac{d^4 \beta}{dz^4} - GJ \frac{d^2 \beta}{dz^2} - w a_0 \beta + M \frac{d^2 u}{dz^2} = 0 \]  \hspace{1cm} (4.5)

where (also see Figure 4.4):

- \( z \) is the distance along the axis of the beam
- \( u(z) \) is the lateral (weak-axis) displacement
- \( \beta(z) \) is the rotation about the beam axis
- \( w(z) \) is the transverse loading
- \( M(z) \) is the transverse (strong-axis) moment
- \( G = \frac{E}{2(1 + \nu)} \) is the shear modulus
- \( a_0 \) is the \( y \) coordinate (measured from the shear center) to the point of load application
- \( A \) is the area of the cross-section
- \( S_x \) is the strong-axis elastic section modulus
- \( I_y \) is the weak-axis moment of inertia
- \( J \) is the torsional rigidity
- \( C_w = \frac{1}{4} I_y h_c^2 \) is the warping constant
- \( h_c \) is the center-to-center distance between flanges
- \( d \) is the depth of the cross-section

### 4.2.1 Uniform Moment

The simplest case of lateral-torsional buckling is limited to elastic stresses (below the proportional limit) and uniform (constant) moment. The following assumptions are
required:

1. The cross-section is constant.

2. Deflections are small. The deformations of the beam are such that the cross-section does not change shape.

3. The direction of the external load remains constant.

4. Flexural stresses due to buckling are negligible compared with those due to the direct load.

5. Prior to loading, the beam is straight and stress free (i.e., imperfections and residual stresses are not directly considered).

6. Flexural stress due to the direct load does not exceed the proportional limit.

7. The boundary conditions for torsion are “pinned ends.”

8. The boundary conditions for lateral shear and bending are “pinned ends.”

9. The moment $M(z)$ is uniform along the length of the beam.
The pinned end boundary conditions are:

\[ \text{at } z = 0 \text{ and } L, u = \beta = 0 \]
\[ \text{at } z = 0 \text{ and } L, \frac{d^2 u}{dz^2} = \frac{d^2 \beta}{dz^2} = 0 \]

where: \( L \) is the span length (i.e. \( 0 \leq z \leq L \))

These are typically assumed for design conditions, because the buckling load is conservatively estimated when the end restraint is ignored. The boundary conditions may be used to eliminate the unknown \( u(z) \). Twice integrating equation 4.4:

\[ EI_u \frac{d^2 u}{dz^2} + M \beta + c_1 z + c_2 = 0 \]

and applying boundary conditions at \( z = 0 \) and \( z = L \) determines that \( c_1 = c_2 = 0 \). Solving for the second derivative of \( u \) gives:

\[ \frac{d^2 u}{dz^2} = -\frac{M}{EI_y} \beta \]

When this is substituted into equation 4.5, a single fourth order equation in \( \beta \) remains:

\[ EC_w \frac{d^4 \beta}{dz^4} - GJ \frac{d^2 \beta}{dz^2} - \frac{M^2}{EI_y} \beta = 0 \quad (4.6) \]

Since equation 4.6 has constant coefficients, the analytical solution may be obtained (see [73], Section 6.2). The characteristic value at the solution is the critical or buckling moment:

\[ M_{cr} = \frac{\pi}{L} \sqrt{EI_y GJ + I_y C_w \left( \frac{\pi E}{L} \right)^2} \quad (4.7) \]

The critical bending stress is:

\[ F_{crLTB} = \frac{M_{cr}}{S_z} = \frac{\pi}{S_z L} \sqrt{EI_y GJ + I_y C_w \left( \frac{\pi E}{L} \right)^2} \quad (4.8) \]

### 4.2.2 Non-Uniform Moment

If Assumption 9 in Section 4.2.1 is relaxed, various configurations of transverse loading can be considered. Although closed form solutions are not available in general, solutions for common cases are found in the literature.
It has been shown (e.g. [73], Section 6.4) that a moment gradient along the length of the beam modifies the critical moment by a factor $C_b$, depending on the boundary conditions and loading (see [22] for a summary of cases). Salvadori [65] proposed a lower bound estimate of this factor from the ratio of moments on either end of an unbraced length of the span ($L_b$). Kirby and Nethercot ([42], Chapter 3) have given a simple, general expression for estimating the factor using moments at quarter points of the unbraced length.

Loads are commonly applied to the top flange, further decreasing beam stability. Clark and Hill [22] give the solution for critical stress, incorporating factors $C_b$ for moment gradient and $C_a$ ($C_1$ and $C_2$ in the original paper) for point of load application, and tabulating many common cases of the factor $C_a$ for various boundary conditions and loading. More general expressions for $C_b$ and $C_a$ have been proposed in [78]. The critical stress is:

$$F_{crLTB} = C_b \frac{\pi}{S_2L_b} \left[ \sqrt{EI_yGJ + \left( \frac{C_w}{I_y} + C_a^2a_0^2 \right) I_y^2 \left( \frac{\pi E}{L_b} \right)^2} - C_a a_0 I_y \frac{\pi E}{L_b} \right]$$

$$= C_b \left( \frac{X_1}{\lambda_{LTB}} \right) \left[ \sqrt{2 + X_2 \left( 1 + C_a^2a_0^2 \frac{I_y}{C_w} \right)} \left( \frac{X_1}{\lambda_{LTB}} \right)^2 - C_a a_0 \frac{X_3}{\lambda_{LTB}} \right] \quad (4.9)$$

where: $X_1 = \frac{\pi}{S_2} \sqrt{\frac{EGJA}{2}}, \quad X_2 = \frac{4C_w}{I_y} \left( \frac{S_x}{GJ} \right)^2, \quad X_3 = \frac{\pi}{r_y} \frac{\sqrt{2EI_y}}{GJ}$

$$\lambda_{LTB} = \frac{L_b}{r_y} \quad (4.10)$$

The current AISC specifications allow use of the $C_b$ coefficient and provide an equation to determine its value. The ASD code [3] uses an equation based on Salvadori’s [65] lower bound factors, with limitations on its use for cantilevers and beams subjected to transverse loading. The LRFD code [4] uses a slightly modified version of Kirby and Nethercot’s equation [42] and suggests using $X_2 = X_3 = 0$ for top flange loading.
4.2.3 General Case

Retaining Assumptions 1, 2, 3, 4, and 5 and relaxing Assumptions 6, 7, 8, and 9 from Section 4.2.1, the following additional assumptions are required for the general case:

10. The shear along the length of the member is linear, or nearly so.

11. The material is ductile and strain hardens after yielding.

12. The shear strains are elastic so that the shearing modulus \( G \) remains constant.

13. The tangent modulus is nearly constant within a discrete portion of the beam.

Relaxation of Assumptions 6, 7, 8, and 9 allows treatment of non-uniform moments, various boundary conditions, and inelastic behavior. However, a closed-form solution cannot be determined, and equations 4.4 and 4.5 must be solved numerically. For the finite element formulation, the total system energy \( U \) is given below (see [14] article 47) as the sum of strain energy and external work:

\[
U = \frac{1}{2} \int_0^L \left[ \bar{E}_t I_y \left( \frac{d^2 u}{dz^2} \right)^2 + \bar{E}_t C_w \left( \frac{d^\beta}{dz} \right)^2 + GJ \left( \frac{d\beta}{dz} \right)^2 \right] dz
- \frac{1}{2} \int_0^L \left[ 2 \frac{du}{dz} \left( M \frac{d\beta}{dz} + V\beta \right) + wa_0\beta^2 \right] dz \quad (4.11)
\]

where: \( V(z) = \frac{dM}{dz} \) is the transverse (strong-axis) shear

\( \bar{E}_t(z) \) is the average tangent modulus of the cross-section

The average tangent modulus is defined as:

\[
\bar{E}_t = \frac{\int_A E_t dA}{\int_A dA} = \frac{\int_A E_t dA}{A}
\]

Note that the shear modulus \( G \) is taken as a constant in equation 4.11. The assumption of elastic shear strains is based on the following:

- Relatively long beam spans typically result in shears much less than the plastic shear (web) capacity.
• St. Venant torsional stiffness is known to be of less importance in the inelastic range of bending stress ([46]).

Assumptions 10 and 13 on variation of shear and tangent modulus are realized by using a sufficient number of finite elements in the solution.

4.2.4 Finite Element Solution of Lateral-Torsional Buckling

The differential equations of equilibrium (4.4 and 4.5) are solved using the finite element method. This section outlines the formulation and solution of a discrete eigenvalue system for the determination of the critical buckling moment and bending stress.

In order to form the eigenvalue system, element stiffness matrices for the lateraltorsional buckling problem are required. First, the element end deformations are defined as:

\[ d_e = [u_A \, \theta y_A \, u_B \, \theta y_B \, \beta_A \, \theta_A \, \beta_B \, \theta_B]^T \]

where (refer to Figure 4.5):

\[ u_A = u(z_A), \, \theta y_A = \left. \frac{du}{dz} \right|_{z_A}, \, u_B = u(z_B), \, \theta y_B = \left. \frac{du}{dz} \right|_{z_B} \]

\[ \beta_A = \beta(z_A), \, \theta_A = \left. \frac{d\beta}{dz} \right|_{z_A}, \, \beta_B = \beta(z_B), \, \theta_B = \left. \frac{d\beta}{dz} \right|_{z_B} \]

Using cubic shape functions for \( u \) and \( \beta \) gives:

\[
\begin{bmatrix}
  u \\
  \beta
\end{bmatrix} = N d_e,
\begin{bmatrix}
  \frac{du}{dz} \\
  \frac{d\beta}{dz}
\end{bmatrix} = P d_e,
\begin{bmatrix}
  \frac{d^2 u}{dz^2} \\
  \frac{d^2 \beta}{dz^2}
\end{bmatrix} = Q d_e
\]

where: \( N = \begin{bmatrix} H & 0 \\ 0 & H \end{bmatrix} \), \( P = \frac{d\xi}{dz} \begin{bmatrix} \frac{dH}{d\xi} & 0 \\ 0 & \frac{dH}{d\xi} \end{bmatrix} \), \( Q = \left( \frac{d\xi}{dz} \right)^2 \begin{bmatrix} \frac{d^2 H}{d\xi^2} & 0 \\ 0 & \frac{d^2 H}{d\xi^2} \end{bmatrix} \)

\[ H(\xi)^T = \begin{bmatrix}
1 - 3\xi^2 + 2\xi^3 \\
\xi h(\xi - 1)^2 \\
3\xi^2 - 2\xi^3 \\
\xi h(\xi^2 - \xi)
\end{bmatrix} \]

\[ \xi = \frac{z - z_A}{h}, \, \frac{d\xi}{dz} = \frac{1}{h}, \, h = z_B - z_A \]
Substituting the discrete forms into equation 4.11 (for a single element) gives:

\[
U_e = \frac{1}{2} \int_0^h d_e^T (Q^T AQ + P^T BP) d_e dz \\
- \frac{1}{2} \int_0^h d_e^T (P^T D_1 P + P^T D_2 N + N^T D_2^T P + N^T D_3 N) d_e dz
\]  

(4.12)

where:

\[
A = \begin{bmatrix}
\bar{E} I_y & 0 \\
0 & \bar{E} I_w
\end{bmatrix}, \quad B = \begin{bmatrix}
0 & 0 \\
0 & GJ
\end{bmatrix}
\]

\[
D_1 = \begin{bmatrix}
0 & M \\
M & 0
\end{bmatrix}, \quad D_2 = \begin{bmatrix}
0 & V \\
0 & 0
\end{bmatrix}, \quad D_3 = \begin{bmatrix}
0 & 0 \\
0 & wa_0
\end{bmatrix}
\]

Figure 4.5: Finite element and local loading.

At equilibrium, the nodal displacements minimize the energy in the system. Differentiating equation 4.12 with respect to the nodal displacements \(d_e\) and setting the result equal to zero gives the discrete solution to the equations of equilibrium. After moving the loading terms to the right hand side:

\[
\int_0^h (Q^T AQ + P^T BP) d_e dz \\
= \int_0^h (P^T D_1 P + P^T D_2 N + N^T D_2^T P + N^T D_3 N) d_e dz
\]  

(4.13)

Assuming a linear variation in shear, for a single element (refer Figure 4.5) the local forces are:

\[
\tilde{w}(\xi) = \frac{1}{h}(V_A - V_B)
\]

\[
\tilde{V}(\xi) = V_A + \xi(V_B - V_A)
\]

\[
\tilde{M}(\xi) = M_A + hV_A \xi + \frac{1}{2}h(V_B - V_A)\xi^2
\]
Let the maximum moment in the member be defined as:

\[
M_{\text{max}} = \max_{0 \leq z \leq L} |M(z)|
\]

Since the beam buckles at the section with the largest moment, the buckling moment is equal in magnitude to \( M_{\text{max}} \). Since the geometric stiffness terms in equation 4.13 must be evaluated prior to knowing the buckling moment, the equation is modified by taking the critical moment (a constant) out of the integral, while the terms inside the integral are divided through by \( M_{\text{max}} \):

\[
K_ed_e = M_{\text{cr}}G_ed_e \tag{4.14}
\]

where:

\[
K_e = h \int_0^1 (Q^TAQ + P^TB) d\xi
\]

\[
G_e = h \int_0^1 (P^T\bar{D}_1 P + P^T\bar{D}_2 N + N^T\bar{D}_2^T P + N^T\bar{D}_3 N) d\xi
\]

\[
\bar{D}_1 = \frac{1}{M_{\text{max}}} \begin{bmatrix}
0 & \bar{M} \\
\bar{M} & 0
\end{bmatrix}, \quad \bar{D}_2 = \frac{1}{M_{\text{max}}} \begin{bmatrix}
0 & \bar{V} \\
0 & 0
\end{bmatrix}, \quad \bar{D}_3 = \frac{1}{M_{\text{max}}} \begin{bmatrix}
0 & 0 \\
0 & \bar{w}_a_0
\end{bmatrix}
\]

After summing the contribution of the element stiffnesses and assembling the global matrices:

\[
K(M)d = M_{\text{cr}}Gd \tag{4.15}
\]

where:

\( K \) is the global stiffness matrix

\( G \) is the global geometric stiffness matrix

\( d \) is the global displacement vector

\( M_{\text{cr}} \) the critical buckling moment, is the minimum positive eigenvalue of equation 4.15

Note that the stiffness is a function of the moment along the length of the beam so that equation 4.15 is a nonlinear eigenvalue system, and it must be solved iteratively (see Section 4.2.6).
4.2.5 Bounds on Inelastic Behavior

The lower bound \( \lambda_u \) is found by setting the critical stress (equation 4.9) equal to the proportional limit and solving for \( \lambda_{LTB} \). The resulting slenderness is:

\[
\lambda_u = \frac{C_b X_1}{0.55 F_y} \cdot \frac{1}{\sqrt{1 + X_2 \left(1 + \frac{C_a a_0^2 I_y}{C_w} \right)^2 \left(\frac{0.55 F_y}{C_b} \right)^2 - 2 \frac{C_a a_0 X_3}{C_b X_1/0.55 F_y} - \frac{C_a a_0 X_3}{C_b X_1/0.55 F_y}}} 
\]

(4.16)

This differs from the LRFD code [4] in which the parameter \( C_b \) is excluded. In that specification, the inelastic strength (moment) is given as:

\[
M_n = C_b \left[ M_p - (M_p - M_r) \left(\frac{\lambda_{LTB} - \lambda_F}{\lambda_r - \lambda_p} \right) \right] \leq M_p
\]

where:

- \( M_p \) is the plastic moment
- \( M_r \) is the limiting elastic buckling moment
- \( \lambda_p \) is the yield limit (analogous to \( \lambda_c \))
- \( \lambda_r \) is the elastic buckling limit (analogous to \( \lambda_u \))

in which the product of \( C_b \) and the basic strength may exceed the maximum strength \( (M_p) \), which is given as an upper bound on \( M_n \). In order to avoid the need for this conditional terminology, the proposed specification guarantees that the inelastic curve defines the stress range between \( F_{hc} \) and \( F_{bu} \), which are fixed values depending only on the material.

The upper bound \( \lambda_c \) is the slenderness at which the full plastic moment is reached. Lay and Galambos [46] detailed a method for approximating this bound by assuming the tangent modulus is equal to the strain hardening modulus \( E_{st} \) and neglecting the St. Venant torsional stiffness. Assuming the critical moment equals the plastic
moment \((M_p = F_y Z_x)\), then solving equation 4.7 for the corresponding span \(L_{c0}\) gives:

\[
\lambda_{c0} = \frac{L_{c0}}{r_y} = \pi \sqrt{\frac{E_{st} h_c A}{2 F_y Z_x}} \approx 107 \sqrt{\frac{1}{F_y}}
\]

where: \(E_{st}\) is taken as 900 ksi (6200 MPa)

\(Z_x\) is the strong-axis plastic section modulus

a typical value of 1.6 is taken for \(r_c A / Z_x\)

Further refinements to this basic value allow corrections for the average contribution of:

- St. Venant torsional stiffness,

- Variation of the tangent modulus within the depth of the cross-section (i.e. \(\bar{E}_t > E_{st}\)),

- Additional torsional rigidity beyond the “pinned end” assumption, and

- Moment gradient.

The first two items can be considered using the database of wide flange sections and the finite element solutions to compute a representative correction factor, \(\zeta\). The latter two are accounted for as in previous specifications (see Salmon and Johnson [64], Chapter 9), using a factor of \(1.25 \sqrt{C_b}\) so that the required slenderness is:

\[
\lambda_c = 133 \zeta \sqrt{\frac{C_b}{F_y}}
\]

Since the slenderness \(\lambda_c\) is associated with plastic yield, only compact sections (see Appendix A) are included in the tabulation. The correction factor

\[
\zeta = \frac{\lambda_c^{FEM}}{\lambda_{c0}}
\]
is computed using the finite element analysis for the uniform moment case \((C_b = 1)\). Its range is quite large, from 1.26 to over 30 (statistics are given in Table 4.1). It would be conservative to take the minimum value. However, it is found that 95% of the sections considered surpass the value \(\zeta = 2.22\), which is taken as a reasonable alternative. The resulting lower bound slenderness is then:

\[
\lambda_c = 300 \sqrt{\frac{C_b}{F_y}}
\]

(4.17)

Note again that this bound differs from the LRFD [4] by considering \(C_b\) directly. As discussed above, this is done to avoid conditional terminology in the specification of allowable stresses.

<table>
<thead>
<tr>
<th>Minimum</th>
<th>1.26</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum</td>
<td>30.3</td>
</tr>
<tr>
<td>Median</td>
<td>3.50</td>
</tr>
<tr>
<td>Mean</td>
<td>4.92</td>
</tr>
</tbody>
</table>

Table 4.1: Values of correction factor, \(\zeta\), for LTB slenderness at yield.

### 4.2.6 Iterative Solution

Since the global stiffness, \(K\), is a function of the average tangent modulus, \(\overline{E_t}\) (and therefore the strong-axis bending stresses and moments), equation 4.15 must be solved iteratively. This is done by assuming a moment \(M_0\), then forming the global stiffness and solving for \(M_{cr}\) until:

\[
\left| \frac{M_{cr}}{M_0} - 1 \right| \leq \tau
\]

where: \(\tau\) is the desired accuracy in the solution

It is reasonable to assume \(M_0\) is equal to the tangent modulus moment as a starting point in the iterations.
Further, to compute the average tangent modulus, each iteration requires the determination of the strain profile at the centroid of each element:

$$E_t = \frac{\int_A E_t(\epsilon(x,y)) dA}{A}$$

where $E_t(\epsilon)$ is defined in equation 4.1, and the normal strain is assumed to vary linearly from the neutral axis:

$$\epsilon(y) = \frac{2\epsilon_{max}}{d} y$$

Within the major iterations (described above), the maximum strain $\epsilon_{max}$ at each element centroid must be determined iteratively according to the local moment. This is done by assuming a maximum strain for the element, then computing the resultant moment until:

$$\left| \frac{M_R}{M_e} - 1 \right| \leq \tau$$

where: $M_e = \frac{M_0}{M_{max}} M(0.5)$ is the element centroid moment

$$M_R = \int_A y\sigma(y)dA$$

is the resultant moment

$\sigma(y)$ is the normal stress defined by the stress-strain relationship in Section 4.1

When the solution is inelastic, the maximum strain is bounded by $\epsilon_p$ and $\epsilon_u$, so that a reasonable starting value (in the absence of any other information) is $\frac{1}{2}(\epsilon_p + \epsilon_u)$. 
4.3 Flange Local Buckling

The flange local buckling of flexural members (FLB) is primarily a function of the flange slenderness, or the ratio of width-to-thickness. Over the range of slenderness, failure is controlled by yield for the lowest values, and by elastic buckling for the highest. In the intermediate range, when the stress exceeds the proportional limit, inelastic buckling theory predicts the failure load. It will be seen in this section that flange local buckling has a secondary dependence on web slenderness, or ratio of the web’s width-to-thickness. Figure 4.6 shows the buckled configuration associated with flange local buckling. In this figure the flange is shown buckling simultaneously with the web.

![Flange local buckling behavior](image)

Figure 4.6: Flange local buckling behavior (original figure by ET-Global [29]).

A general solution for elastic buckling exists with some limitations; however, the solution for inelastic buckling must be obtained numerically. Because of the complexity of the web-flange interaction, the current study uses experimental data rather than finite element analysis to study the inelastic behavior. This is used to parameterize a simple smooth equation for design of flange local buckling. The experimental data is presented in Section 5.1.3.2 along with the results predicted by that model.
The general expression (see [73]) for critical elastic buckling stress of a rectangular plate uniformly compressed on two opposite edges is:

\[ \sigma_{cr} = \frac{\pi^2 k E}{12(1 - \nu^2)} \left( \frac{t}{b} \right)^2 \]  

(4.18)

where: \( b, t \) are the width and thickness of the plate

\( k \) is a coefficient depending on the boundary conditions

Critical buckling of the flange is a special case and may be obtained by multiplying equation 4.18 by \( F_y/F_y \), then substituting the appropriate flange properties, and utilizing the specific constant \( k_c \) for \( k \):

\[ F_{crFLB} = \frac{\pi^2 F_y}{12(1 - \nu^2)} \frac{1}{\lambda_{FLB}^2} \]  

(4.19)

where: \( \lambda_{FLB} = \frac{b_f}{2t_f} \sqrt{\frac{F_y}{k_c E}} \)  

(4.20)

\( k_c \) is a coefficient depending on restraint provided by the web

The theoretical limits on \( k_c \) are 0.425 for no web restraint (simply supported) and 1.277 for maximum restraint (fixed) [14]. It has been shown experimentally [41] that \( k_c < 0.425 \) is possible and occurs when the web adjacent to the compression flange becomes unstable, decreasing the overall buckling capacity.

For \( F_{crFLB} \) greater than the proportional limit, the elastic modulus is replaced by a reduced value \( \eta E, \eta < 1 \). Bleich [14] suggests using \( \eta = \sqrt{E_t/E} \) but there is no closed form solution for the inelastic buckling stress due to the nature of the boundary conditions.
4.3.1 Bounds on Inelastic Behavior

The lower bound is established by setting the critical stress equal to the proportional limit and solving equation 4.19 for $\lambda_{F_{LB}}$. The resulting parameter is:

$$\lambda_p = \pi \sqrt{\frac{1}{12(1-\nu^2)} \frac{F_y}{F_p}} = 1.34$$

The bound for strain hardening is established by assuming the critical stress equal to the yield stress, and taking a reduced value for the elastic modulus to account for inelastic behavior:

$$\lambda_0 = \alpha_0 \pi \sqrt{\frac{1}{12(1-\nu^2)}}$$

where $\alpha_0 = 0.46$ [34] so that $\lambda_0 = 0.44$. When $k_c = 0.763$ is assumed\(^2\), this reduces to the familiar $65/\sqrt{E} = 0.38$ from the ASD and LRFD specifications [3], [4].

\(^2\)This is the upper bound on $k_c$ in the LRFD code [4]. It is arbitrary, but is chosen to correspond with $20000/26200 = 0.763$, implicitly used in earlier codes.
4.4 Web Local Buckling

The web local buckling of flexural members (WLB) is primarily a function of the web slenderness, or the ratio of width-to-thickness. Over the range of slenderness, failure is controlled by yield for the lowest values, and by elastic buckling for the highest. In the intermediate range, when the stress exceeds the proportional limit, inelastic buckling theory predicts the failure load. Figure 4.7 shows the buckled configuration associated with web local buckling for a long unstiffened plate. The angle of wave formation is at a skew with respect to the primary axes, as predicted by the theory in the following sections.

Figure 4.7: Web local buckling behavior.

Ilyushin [40] presented the solution of elasto-plastic stability for (i) strain at all fibers plastic and (ii) concave face strains plastic, convex face strains elastic. Stowell [72] followed up on the work of Ilyushin to present a streamlined theory, which concentrates only on case (i). The difference in the two theories is in the stiffness associated with the overall section at the buckling load. Once bending occurs due to buckling, the strain reversal would increase the net stiffness because the strains on the convex face would be elastic. However, this potential increase in stiffness would give rise to a higher buckling load, increasing the plastic strains on both faces, superseding any actual strain reversal. This argument is analogous to Shanley’s theory [67] for column buckling, which assumes that bending proceeds simultaneously with increasing axial load.

A large number of laboratory tests of the ultimate shear strength of webs were done in the 1960s and earlier. Some of the data in the inelastic region (see for example Okumura, et al. [60]) is used in this study to show the validity of the inelastic theory
and to verify the finite element modeling. The present LRFD code bases the strength of webs in the inelastic region on the work of Basler [10]. That study presented a curve fit to some earlier experiments by Lyse & Godfrey [53] in the post yield region. The expression gives conservative values for strength in the inelastic region, but it is nonsmooth at both the boundaries of yield and elastic buckling.

The current study uses finite element analysis to solve the differential equations posed by Ilyushin [40] and Stowell [72]. The intent is to study the validity of the theoretical solution in the inelastic region and determine a simple, smooth equation (presented in 5.1.4), which can then be used for design.

### 4.4.1 Elastic Stress Range

The theoretical buckling stress, or critical shear stress, for a simply supported rectangular plate in pure shear is given by (see [73] article 9.7 and refer to Figure 4.8):

$$
\tau_{cr} = \frac{\pi^2 k_v E}{12(1 - \nu^2)} \left( \frac{t_w}{h} \right)^2
$$

(4.21)

where: $t_w$ is the plate thickness

$$
k_v = \begin{cases} 
5.34 + \frac{4}{(a/h)^2}, & a/h \geq 1 \\
4 + \frac{5.34}{(a/h)^2}, & a/h < 1
\end{cases}
$$

(4.22)

$h$ is the web or plate height

$a$ is the transverse stiffener spacing

Note that for unstiffened plates, $k_v = 5.34$. A more useful expression for design may be obtained by multiplying equation 4.21 by $F_y/F_y$, and substituting the slenderness parameter $\lambda_v$:

$$
F_{crv} = \frac{\pi^2 F_y}{12(1 - \nu^2)} \frac{1}{\lambda_v^2}
$$

(4.23)

where: $\lambda_v = \frac{h}{t_w} \sqrt[3]{\frac{F_y}{k_v E}}$

(4.24)
4.4.2 Inelastic Stress Range

When $\tau_{cr}$ exceeds $\tau_p$, the proportional limit in shear, equation 4.21 overestimates the critical stress because of the reduction in modulus, $E$. The buckling stress in the inelastic region is:

$$\tau_{cr} = \frac{\pi^2 k_w E \eta}{12(1 - \nu^2)} \left( \frac{t_w}{h} \right)^2$$  \hspace{1cm} (4.25)

which differs from equation 4.21 by a factor $\eta$, the ratio of the actual critical stress to the critical stress for an elastic material. If the tangent modulus theory is used:

$$\eta = \frac{E_t}{E}$$

Unlike columns, the tangent modulus theory gives overly conservative results for plates. A comparison of theoretical elastic and inelastic curves can be found at the end of this section (Figure 4.9).

Stowell used his theory of plastic buckling [72] to determine a more rational expression of $\eta$ for infinitely long plates [71]. Several of the assumptions in the procedure were in error, so the derivation is repeated here (using the method originally given by Timoshenko [73], article 9.7). To begin, an expression for the deflected shape is
assumed:

\[ w(x, y) = w_0 \sin \frac{\pi y}{h} \sin \frac{\pi}{a}(x + \alpha y) \]  

(4.26)

where: \( \alpha = \tan \phi \)

\( \phi \) is the angle of buckling wave formation

\( \hat{a} \) is the half-wavelength

and satisfying: \( w = 0 \) at \( y = 0 \) and \( y = h \)

\( w = 0 \) at \( x + \alpha y = 0, \hat{a}, 2\hat{a}, \ldots \), i.e. along nodal lines

but not satisfying the natural boundary conditions (i.e. using equation 4.26, \( M_y \neq 0 \) at \( y = 0 \) and \( y = h \)). Substituting equation 4.26 in the expression for total energy (equation 4.49, Section 4.4.4) results in an approximate expression for shear stress:

\[ \tau_{xy}(\hat{a}, \alpha) = \frac{\pi^2 D_s}{2\alpha h^2 t_w} \left[ 6 \alpha^2 + 2c_3 + \left( \frac{\hat{a}}{h} \right)^2 + (\alpha^4 + 2c_3 \alpha^2 + 1) \left( \frac{h}{\hat{a}} \right)^2 \right] \]  

(4.27)

where: \( D_s = \frac{E_s v_s^3}{12(1 - v_s^2)} \)

\( E_s = \frac{\sigma}{\epsilon} \) is the secant modulus

\( c_3 \) is the plasticity coefficient (see Section 4.4.4)

A stationary point, \( [\hat{a}_*, \alpha_*]^T \) of equation 4.27, is found by solving the equations \( \nabla \tau_{xy} = 0 \):

\[ \left( \frac{\hat{a}_*}{h} \right)^2 - \sqrt{\alpha_*^4 + 2c_3 \alpha_*^2 + 1} = 0 \]  

(4.28)

\[ 6\alpha_*^2 - 2c_3 - \left( \frac{\hat{a}_*}{h} \right)^2 + (3\alpha_*^4 + 2c_3 \alpha_*^2 - 1) \left( \frac{h}{\hat{a}_*} \right)^2 = 0 \]  

(4.29)

By making the following practical assumptions:

\( D_s, h, \hat{a}, \alpha > 0 \)

\( \alpha^2 > 0 \)

\( \frac{1}{2} \leq c_3 \leq 1 \)
then without loss of generality:

\[
\det \nabla^2 \tau_{xy} = \frac{\pi^4 D_s^2}{h^4 a^2 \alpha^4} \left\{ 2c_3 \alpha^2 + 6 + 2c_3 \left( \frac{\dot{a}}{h} \right)^2 + 6c_3 \left( \frac{h}{\dot{a}} \right)^2 (\alpha^4 + 2c_3 \alpha^2 + 1) \right. \\
\left. + 2 \left( \frac{h}{\dot{a}} \right)^4 [3c_3 \alpha^6 + (9 - 2c_3^2) \alpha^4 + 5c_3 \alpha^2 + 1] \right\} > 0
\]

or, equivalently, the Hessian is positive definite and \([\dot{a}, \alpha]^T\) is a minimizer of equation 4.27. Substituting the solution of equations 4.28 and 4.29 in equation 4.27, the minimum shear stress is:

\[
\tau_{\text{min}} = \frac{\pi^2 D_s}{2\alpha_s h^2 l_w} \left( \alpha_s^2 + 2c_3 + 2\sqrt{\alpha_s^4 + 2c_3 \alpha_s^2 + 1} \right)
\]

Ultimately, the desired parameter is calculated as:

\[
\eta = \frac{\tau_{\text{min}}}{\tau_0} = \frac{1}{2\alpha_s k_v E(1 - \nu^2)} \left( 6\alpha^2_s + 2c_3 + 2\sqrt{\alpha_s^4 + 2c_3 \alpha_s^2 + 1} \right) \tag{4.30}
\]

where \(\tau_0\) is the critical elastic shear stress from equation 4.21. Since the shape function (equation 4.26) taken for \(w(x, y)\) is inexact, the results using 4.30 with the usual value of \(k_v\) (5.34) are off by 5-10%. However since \(\eta\) is defined in terms of a baseline elastic shear stress \(\tau_0\), the error can be reduced by introducing the same assumptions into the calculation of \(\tau_0\), which gives a value of \(k_v = 4\sqrt{2} \approx 5.7\) for an infinitely long plate. Thus a more accurate expression for the inelastic buckling factor is:

\[
\eta = \frac{1}{8\sqrt{2}\alpha_s} E_s(1 - \nu^2) \left( 6\alpha^2_s + 2c_3 + 2\sqrt{\alpha_s^4 + 2c_3 \alpha_s^2 + 1} \right) \tag{4.31}
\]

When the material is elastic, \(E_s = E\), \(c_3 = 1\), \(\nu_s = \nu\), \(\alpha_s = \frac{1}{\sqrt{2}}\) and using equation 4.31, \(\eta = 1\). Note that dividing the minimum shear stress by a baseline elastic shear stress also allows Stowell’s method to give reasonably good results (see also [14], article 87).

Using the approximate inelastic solution (equations 4.25 and 4.31), we can compute the strength coefficient:

\[
C_v = \frac{\tau_{\text{cr}}}{\tau_y} \tag{4.32}
\]
for various values of web slenderness $\lambda_w$. A relatively straightforward procedure can be used wherein a value of shear stress $\tau_p \leq \tau_{xy} \leq \tau_y$ is assumed. The corresponding value of normal stress $\sigma$ is then computed$^3$. The strain $\epsilon$, tangent modulus $E_t$ (from equation 4.1), secant modulus $E_s$, and "secant" Poisson’s ratio $\nu_s$ (from equation 4.3) are determined. The coefficient $c_3$ is computed using equation 4.45 (Section 4.4.4). The value of $\alpha_*$ is determined by iteratively solving equations 4.28 and 4.29. The reduction parameter $\eta$ is then computed with equation 4.31. Rearranging equation 4.25 and substituting the expression for $\lambda_u$ (note that $F_y$ and $\sigma_y$ are used interchangeably):

$$\tau_{xy} = \frac{\pi^2 \eta \sigma_y}{12(1-\nu^2)} \left(\frac{t_w}{h}\right)^2 \frac{k_w E}{F_y}$$

$$= \frac{\pi^2 \eta \sigma_y}{12(1-\nu^2)} \frac{1}{\lambda_u^2}$$

so that: $\lambda_u = \pi \sqrt{\frac{\eta \sigma_y}{12(1-\nu^2)} \frac{\sigma_y}{\tau_{xy}}}$

Figure 4.9 shows the theoretical strength curves from elastic, inelastic, and tangent modulus theory.

![Figure 4.9: Shear strength—Theoretical curves.](image)

$^3$The relationship between normal and shear stress is taken as $\tau = 0.6\sigma$ using the constant 0.6 instead of the exact value $\frac{1}{\sqrt{3}} \approx 0.577$ to agree with the LRFD code [4]
4.4.3 Experimental Results of Plate Shear Failure

A large number of full scale testing programs have amassed data for the ultimate strength of steel webs. Over 150 experiments in eighteen published studies were found while searching for data, but the majority of these were slender webs, beyond the inelastic region. Only 21 experiments were conducted in the region of interest \( (0.95 \leq \lambda_v \leq 1.37) \) and of these, only nine were considered reliable\(^4\) and are included in Table 4.2 below). These data points are shown overlayed on theoretical curves in this chapter in Figure 4.10, Section 4.4.6 and later on in Section 5.1.4, Figure 5.4.

<table>
<thead>
<tr>
<th>Ref</th>
<th>Mark</th>
<th>(h) (in)</th>
<th>(t_w) (in)</th>
<th>(a) (in)</th>
<th>(F_y) (ksi)</th>
<th>(V_u^{ex}) (k)</th>
<th>(\lambda_v)</th>
<th>(C_v^{ex})</th>
</tr>
</thead>
<tbody>
<tr>
<td>[53]</td>
<td>WB-3</td>
<td>16.00</td>
<td>0.2720</td>
<td>48.0</td>
<td>49.6</td>
<td>139.0</td>
<td>1.01</td>
<td>1.07</td>
</tr>
<tr>
<td>[53]</td>
<td>WB-6</td>
<td>17.60</td>
<td>0.2510</td>
<td>50.0</td>
<td>33.1</td>
<td>96.3</td>
<td>0.98</td>
<td>1.10</td>
</tr>
<tr>
<td>[60]</td>
<td>G3</td>
<td>22.05</td>
<td>0.315</td>
<td>58.0</td>
<td>62.6</td>
<td>218.3</td>
<td>1.34</td>
<td>0.84</td>
</tr>
<tr>
<td>[60]</td>
<td>G4</td>
<td>22.05</td>
<td>0.315</td>
<td>78.7</td>
<td>62.6</td>
<td>213.8</td>
<td>1.37</td>
<td>0.82</td>
</tr>
<tr>
<td>[60]</td>
<td>G5</td>
<td>22.05</td>
<td>0.315</td>
<td>59.1</td>
<td>62.6</td>
<td>235.9</td>
<td>1.34</td>
<td>0.90</td>
</tr>
<tr>
<td>[60]</td>
<td>G6</td>
<td>22.05</td>
<td>0.315</td>
<td>27.6</td>
<td>62.6</td>
<td>264.6</td>
<td>1.16</td>
<td>1.01</td>
</tr>
<tr>
<td>[60]</td>
<td>G7</td>
<td>22.05</td>
<td>0.315</td>
<td>59.1</td>
<td>62.6</td>
<td>235.9</td>
<td>1.34</td>
<td>0.90</td>
</tr>
<tr>
<td>[18]</td>
<td>C-AH1</td>
<td>17.96</td>
<td>0.261</td>
<td>99.0</td>
<td>48.4</td>
<td>130.0</td>
<td>1.20</td>
<td>0.95</td>
</tr>
<tr>
<td>[52]</td>
<td>A1</td>
<td>5.25</td>
<td>0.056</td>
<td>6.8</td>
<td>33.4</td>
<td>5.8</td>
<td>1.14</td>
<td>0.98</td>
</tr>
</tbody>
</table>

Table 4.2: Experimental shear strength data.

The web slenderness \(\lambda_v\) was computed from the data given using equations 4.24 and the code [3] equation for \(k_v\). The yield stress, \(F_y\), was based on a coupon test in all cases. However, in some studies, a single coupon was used for a set of beams.\(^4\)

\(^4\)Only those experiments that gave an ultimate strength at least equal to that predicted by the current LRFD code were used.
The strength coefficient was computed using equation 4.32:

\[
C_u^{ex} = \frac{\tau_u^{ex}}{\tau_y} = \frac{V_u^{ex}/A_w}{0.6F_y} = \frac{V_u^{ex}}{0.6F_yA_w}
\]

where \(A_w = ht_w\), is the web area.

**4.4.4 Finite Element Solution of Plate Shear Buckling**

Rather than using Stowell’s result [72] directly, the weak form of the differential equation is developed based on his method but in terms of \(E_t\), \(E_s\), and Poisson’s ratio. Here the more appropriate elasto-plastic value \(\nu_s\) given by equation 4.3 is used.

In accordance with the previous analysis the following assumptions for inelastic plate buckling are made:

- Deflections are small so that the curvatures in the \(x\) and \(y\) direction are proportional to the second derivatives of the deflection \(w\).
- Flexural stresses due to outward buckling are negligible compared to the membrane stresses.
- The out-of-plane shear strains \(\gamma_{xz}\) and \(\gamma_{yz}\) are negligible.
- Since bending occurs simultaneously with an increase in load, there is no strain reversal.
- All four edges are simply supported (i.e., rotational restraint from the flanges is neglected).
- Prior to loading, the plate is straight and stress-free (i.e., imperfections and residual stresses are not directly considered).
The differential equation for solution using the finite element method will be developed from the strong form of the differential equation using Galerkin’s method, and integrating by parts to get the weak form.

The variational plate moments are:

\[
\delta M_x = \int_{-\frac{h}{2}}^{\frac{h}{2}} \delta \sigma_x z dz
\]
\[
\delta M_y = \int_{-\frac{h}{2}}^{\frac{h}{2}} \delta \sigma_y z dz
\]
\[
\delta M_{xy} = \int_{-\frac{h}{2}}^{\frac{h}{2}} \delta \tau_{xy} z dz
\]

where:

\[
\delta \sigma_x = \frac{1}{1 - \nu_s^2} (\delta S_x + \nu_s \delta S_y)
\]
\[
\delta \sigma_y = \frac{1}{1 - \nu_s^2} (\delta S_y + \nu_s \delta S_x)
\]

\[
\delta \tau_{xy} = \frac{E_s \delta \gamma_{xy} + (z - z_0) \frac{E_s - E_t}{2(1 + \nu_s)}}{\frac{\gamma_{xy}}{\sigma \varepsilon}} \left( \frac{\sigma_x}{\partial x^2} + 2 \tau_{xy} \frac{\partial^2 w}{\partial x \partial y} + \frac{\sigma_y}{\partial y^2} \right)
\]
\[
\delta S_x = \frac{E_s \delta \varepsilon_x + (z - z_0)(E_s - E_t)}{\frac{\varepsilon_x}{\sigma \varepsilon}} \left( \frac{\sigma_x}{\partial x^2} + 2 \tau_{xy} \frac{\partial^2 w}{\partial x \partial y} + \frac{\sigma_y}{\partial y^2} \right)
\]
\[
\delta S_y = \frac{E_s \delta \varepsilon_y + (z - z_0)(E_s - E_t)}{\frac{\varepsilon_y}{\sigma \varepsilon}} \left( \frac{\sigma_x}{\partial x^2} + 2 \tau_{xy} \frac{\partial^2 w}{\partial x \partial y} + \frac{\sigma_y}{\partial y^2} \right)
\]

\[
G_s = \frac{E_s}{2(1 + \nu_s)}
\]
\[
\delta \varepsilon_x = \varepsilon_x - \frac{\partial^2 w}{\partial x^2}
\]
\[
\delta \varepsilon_y = \varepsilon_y - \frac{\partial^2 w}{\partial y^2}
\]
\[
\delta \gamma_{xy} = \gamma_{xy} - 2z \frac{\partial^2 w}{\partial x \partial y}
\]
\[
\bar{\sigma} = \sqrt{\sigma_x^2 - \sigma_x \sigma_y + \sigma_y^2 + 3 \tau_{xy}^2}
\]
\[
\bar{\varepsilon} = \frac{\sigma}{E_s}
\]
\[
\epsilon_x = \frac{1}{E_s} (\sigma_x - \nu_s \sigma_y)
\]
\[
\epsilon_y = \frac{1}{E_s} (\sigma_y - \nu_s \sigma_x) \\
\gamma_{xy} = \frac{1}{G_s} \tau_{xy}
\]

and: \( z \) is the coordinate out of the plane of the plate

\( z_0 \) is the coordinate of the neutral surface for which \( \delta \bar{e} = 0 \)

Noting that:

\[
\frac{\epsilon_x + \nu_s \epsilon_y}{\bar{\ddot{e}}} = (1 - \nu_s^2) \frac{\sigma_x}{\bar{\ddot{e}}} \\
\frac{\epsilon_y + \nu_s \epsilon_x}{\bar{\ddot{e}}} = (1 - \nu_s^2) \frac{\sigma_y}{\bar{\ddot{e}}} \\
\frac{\gamma_{xy}}{\bar{\ddot{e}}} = 2(1 + \nu_s) \frac{\tau_{xy}}{\bar{\ddot{e}}}
\]

then substituting terms and evaluating the integrals in equations 4.33, 4.34, and 4.35:

\[
\delta M_x = -D_s \left[ \left( \frac{\partial^2 w}{\partial x^2} + \nu_s \frac{\partial^2 w}{\partial y^2} \right) \\
- (1 - \nu_s^2) \left( 1 - \frac{E_t}{E_s} \right) \left( \frac{\sigma_x}{\bar{\ddot{e}}} \right) \left( \frac{\partial^2 w}{\partial x^2} + 2 \frac{\tau_{xy}}{\partial x \partial y} + \frac{\sigma_y}{\partial y^2} \right) \right] \tag{4.37}
\]

\[
\delta M_y = -D_s \left[ \left( \frac{\partial^2 w}{\partial y^2} + \nu_s \frac{\partial^2 w}{\partial x^2} \right) \\
- (1 - \nu_s^2) \left( 1 - \frac{E_t}{E_s} \right) \left( \frac{\sigma_y}{\bar{\ddot{e}}} \right) \left( \frac{\partial^2 w}{\partial x^2} + 2 \frac{\tau_{xy}}{\partial x \partial y} + \frac{\sigma_y}{\partial y^2} \right) \right] \tag{4.38}
\]

\[
\delta M_{xy} = -D_s (1 - \nu_s) \left( \frac{\partial^2 w}{\partial x \partial y} \right) \\
- (1 + \nu_s) \left( 1 - \frac{E_t}{E_s} \right) \left( \frac{\tau_{xy}}{\bar{\ddot{e}}} \right) \left( \frac{\partial^2 w}{\partial x^2} + 2 \frac{\tau_{xy}}{\partial x \partial y} + \frac{\sigma_y}{\partial y^2} \right) \tag{4.39}
\]

The equation of equilibrium of an element of the plate is:

\[
\frac{\partial^2 (\delta M_x)}{\partial x^2} + 2 \frac{\partial^2 (\delta M_{xy})}{\partial x \partial y} + \frac{\partial^2 (\delta M_y)}{\partial y^2} = -\frac{t_w}{D_s} \left( \frac{\sigma_x}{\partial x^2} + 2 \frac{\tau_{xy}}{\partial x \partial y} + \frac{\sigma_y}{\partial y^2} \right) \tag{4.40}
\]
Equations 4.37, 4.38, and 4.39 can then be substituted into equation 4.40. Then differentiating:

\[
\frac{\partial^4 w}{\partial x^4} - c_1 \frac{\partial^4 w}{\partial x^2 \partial y^2} + 2c_3 \frac{\partial^4 w}{\partial x \partial y^3} + c_4 \frac{\partial^4 w}{\partial y^4} = -\frac{t_w}{D_s} \left( \sigma_x \frac{\partial^2 w}{\partial x^2} + 2\tau_{xy} \frac{\partial^2 w}{\partial x \partial y} + \sigma_y \frac{\partial^2 w}{\partial y^2} \right)
\]  

(4.41)

where:

\[
c_1 = 1 - (1 - \nu_s^2) \left( 1 - \frac{E_t}{E_s} \right) \left( \frac{\sigma_x}{\sigma} \right)^2
\]

\[
c_2 = 4(1 - \nu_s^2) \left( 1 - \frac{E_t}{E_s} \right) \frac{\sigma_x \tau_{xy}}{\sigma^2}
\]

\[
c_3 = 1 - (1 - \nu_s^2) \left( 1 - \frac{E_t}{E_s} \right) \frac{\sigma_y \tau_{xy}}{\sigma^2}
\]

\[
c_4 = 4(1 - \nu_s^2) \left( 1 - \frac{E_t}{E_s} \right) \frac{\sigma_y}{\sigma^2}
\]

\[
c_5 = 1 - (1 - \nu_s^2) \left( 1 - \frac{E_t}{E_s} \right) \left( \frac{\sigma_y}{\sigma} \right)^2
\]

The boundary conditions for a simply supported plate include:

for \( x = 0 \) and \( x = a \) : \( w = 0 \), \( M_x = 0 \) \( \Rightarrow \frac{\partial w}{\partial y} = \frac{\partial^2 w}{\partial x^2} = \frac{\partial^2 w}{\partial y^2} = 0 \)

(4.42)

for \( y = 0 \) and \( y = h \) : \( w = 0 \), \( M_y = 0 \) \( \Rightarrow \frac{\partial w}{\partial x} = \frac{\partial^2 w}{\partial x^2} = \frac{\partial^2 w}{\partial y^2} = 0 \)

(4.43)

For pure shear \( \sigma_x = \sigma_y = 0 \). In this case the strong form (equation 4.41) reduces to:

\[
\frac{\partial^4 w}{\partial x^4} + 2c_3 \frac{\partial^4 w}{\partial x^2 \partial y^2} + \frac{\partial^4 w}{\partial y^4} = -\frac{t_w}{D_s} \tau_{xy} \frac{\partial^2 w}{\partial x \partial y}
\]

(4.44)

since \( c_1 = c_5 = 1 \), \( c_2 = c_4 = 0 \) and \( c_3 = 1 - 2(1 - \nu_s^2) \left( 1 - \frac{E_t}{E_s} \right) \left( \frac{\tau_{xy}}{\sigma} \right)^2 \). Further, the von Mises stress \( \sigma = \sqrt{3}\tau_{xy} \) (from equation 4.36) so that:

\[
c_3 = 1 - \frac{2}{3}(1 + \nu_s)(1 - \nu_s) \left( 1 - \frac{E_t}{E_s} \right) \left( \frac{\tau_{xy}}{\sigma} \right)^2
\]

(4.45)

Using Galerkin’s method, equation 4.44 is rewritten as the product of zero and a
test function $\psi(x, y)$, chosen to satisfy the essential boundary conditions:

for $x = 0$ and $x = a$ : $\psi = 0 \Rightarrow \frac{\partial \psi}{\partial y} = 0$ \hspace{1cm} (4.46)

for $y = 0$ and $y = h$ : $\psi = 0 \Rightarrow \frac{\partial \psi}{\partial x} = 0$ \hspace{1cm} (4.47)

and integrated over the domain of the plate:

$$
\int_0^h \int_0^a \psi(x, y) \left[ D_s \left( \frac{\partial^4 w}{\partial x^4} + c_3 \frac{\partial^4 w}{\partial x^2 \partial y^2} + c_3 \frac{\partial^4 w}{\partial x^2 \partial y^4} + \frac{\partial^4 w}{\partial y^4} \right) + N_{xy} \frac{\partial^2 w}{\partial x \partial y} + N_{xy} \frac{\partial^2 w}{\partial x \partial y} \right] dxdy = 0
$$

where: $N_{xy} = t_w \tau_{xy}$

Note that the cross derivative terms are intentionally split into two, to be integrated separately. Integrating once:

$$
D_s \psi(x, y) \left[ \frac{\partial^3 w}{\partial x^3} + c_3 \left( \frac{\partial^3 w}{\partial x \partial y^2} + \frac{\partial^3 w}{\partial x^2 \partial y} \right) + \frac{\partial^3 w}{\partial y^3} \right] \bigg|_0^a \bigg|_0^h
- D_s \int_0^h \int_0^a \left[ \frac{\partial \psi}{\partial x} \left( \frac{\partial^3 w}{\partial x^3} + c_3 \frac{\partial^3 w}{\partial x \partial y^2} + \frac{\partial \psi}{\partial y} \left( c_3 \frac{\partial^3 w}{\partial x^2 \partial y} + \frac{\partial^3 w}{\partial y^3} \right) \right) + N_{xy} \psi(x, y) \left( \frac{\partial w}{\partial y} + \frac{\partial w}{\partial x} \right) \bigg|_0^a \bigg|_0^h - N_{xy} \int_0^h \int_0^a \left( \frac{\partial \psi}{\partial x} \frac{\partial w}{\partial y} + \frac{\partial \psi}{\partial y} \frac{\partial w}{\partial x} \right) dxdy = 0
$$

The first and third terms are zero as a result of the boundary conditions in equations 4.46 and 4.47. Integrating a second time:

$$
- D_s \left[ \frac{\partial \psi}{\partial x} \left( \frac{\partial^2 w}{\partial x^2} + c_3 \frac{\partial^2 w}{\partial x^2 \partial y} + \frac{\partial \psi}{\partial y} \left( c_3 \frac{\partial^2 w}{\partial x^2} + c_3 \frac{\partial^2 w}{\partial x \partial y^2} + \frac{\partial^2 w}{\partial y^2} \right) \right) \bigg|_0^h \bigg|_0^a
+ D_s \int_0^h \int_0^a \left[ \frac{\partial^2 \psi}{\partial x^2} \left( \frac{\partial^2 w}{\partial x^2} + c_3 \frac{\partial^2 w}{\partial x^2 \partial y} \right) + c_3 \frac{\partial^2 \psi}{\partial x \partial y \partial y} + \frac{\partial^2 \psi}{\partial y^2} \left( c_3 \frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} \right) \right] dxdy
- N_{xy} \int_0^h \int_0^a \left( \frac{\partial \psi}{\partial x} \frac{\partial w}{\partial y} + \frac{\partial \psi}{\partial y} \frac{\partial w}{\partial x} \right) dxdy = 0
$$

The first term is zero as a result of the boundary conditions in equations 4.46 and 4.47 so that:

$$
\int_0^h \int_0^a \left\{ D_s \left[ \frac{\partial^2 \psi}{\partial x^2} \frac{\partial^2 w}{\partial x^2} + c_3 \left( \frac{\partial^2 \psi}{\partial x^2 \partial y} \frac{\partial^2 w}{\partial x^2 \partial y} + \frac{\partial^2 \psi}{\partial x \partial y \partial y} \frac{\partial^2 w}{\partial x \partial y \partial y} + \frac{\partial^2 \psi}{\partial y^2} \frac{\partial^2 w}{\partial x^2 \partial y} \right) + \frac{\partial^2 \psi}{\partial y^2} \frac{\partial^2 w}{\partial y^2} \right] - N_{xy} \left( \frac{\partial \psi}{\partial x} \frac{\partial w}{\partial y} + \frac{\partial \psi}{\partial y} \frac{\partial w}{\partial x} \right) \right\} dxdy = 0 \hspace{1cm} (4.48)
$$
This is the weak form of equation 4.44 with boundary conditions given in equations 4.42 and 4.43.

If \( \psi(x, y) \) is taken to be equal to \( w(x, y) \), then equation 4.48 becomes:

\[
\frac{D_2}{2} \int_0^h \int_0^a \left\{ \left( \frac{\partial^2 w}{\partial x^2} \right)^2 + c_3 \left[ \frac{\partial^2 w}{\partial x^2} \frac{\partial^2 w}{\partial y^2} + \left( \frac{\partial^2 w}{\partial x \partial y} \right)^2 \right] + \left( \frac{\partial^2 w}{\partial y^2} \right)^2 \right\} dx dy \\
= N_{xy} \int_0^h \int_0^a \frac{\partial w}{\partial x} \frac{\partial w}{\partial y} dx dy \quad (4.49)
\]

which equates the strain energy with the external work done by the load \( N_{xy} \).

The weak form (equation 4.48) can be discretized to form the nonlinear eigenvalue system:

\[
K(\tau_{xy})\Delta = N_{xy}G\Delta 
\]

where: \( K \) is the plate bending stiffness matrix

\( G \) is the plate geometric stiffness matrix

\( N_{xy} \) is the shearing force per unit length of the plate

\( \Delta \) is the modeshape associated with \( N_{xy} \)

The critical buckling load is \( (N_{xy})_{cr} \), the minimum positive eigenvalue of 4.50. The critical buckling stress is then \( \tau_{cr} = (N_{xy})_{cr}/t_w \). Note that since the plate is inelastic, the bending stiffness is a function of the shear stress, \( \tau_{xy} \). Thus, the critical buckling stress must be obtained by an iterative solution (see Section 4.4.6).

The plate buckling problem was modeled using triangular Reissner-Mindlin plate bending elements. These include a stiffness term to account for deformation due to out-of-plane (beam) shears and are thus suitable for thin or thick plates. The element used has been shown by patch tests to be robust. The element does not suffer from the common problem of shear locking\(^5\) and is sufficiently stable for general use without restrictive caveats [84].

\(^5\)This refers to the very stiff behavior seen in thin plates when an erroneous shear strain energy is present in the finite elements. Any element designed for both thick and thin plates must consider this problem.
Since a finite length model was necessary, an aspect ratio of $a/h = 10$ was chosen, anticipating that the plate would buckle in 8 half waves of approximately $1.25h$. The web slenderness $\lambda_w$ was varied by selecting different values of $t_w$ with fixed values of $k_w$, $F_y$, and $E$.

The mesh was automatically generated, then refined until the mesh quality was sufficiently uniform. For verification, the model was used to solve for the critical stress at the proportional limit and compared to the elastic solution (the relative error was less than 0.1%).

### 4.4.5 Bounds on Inelastic Behavior

The lower bound is established by setting the critical stress equal to the proportional limit and solving equation 4.23 for $\lambda_w$. When the proportional limit $\tau_p$ is taken equal to $0.6(0.8F_y)$, the resulting parameter is:

$$
\lambda_p = \pi \sqrt{\frac{F_y}{12(1 - \nu^2)\tau_p}} = 1.37
$$

It is also important to establish the value of $\lambda_w$ at yield. The tangent modulus theory predicts this value to be zero, but this is known to be too low from experience with other theories and experimental results (e.g., inelastic theory predicts a value of about 1).

In order to find the value of $\lambda_y$, an iterative method was used to solve the fixed point problem:

$$
\lambda_w \frac{(N_{xy})_{cr}}{\tau_y h} \sqrt{\frac{k_w E}{F_y}} - 1 = 0
$$

where: $(N_{xy})_{cr}$ is the minimum positive eigenvalue of

$$
K(\tau_y)\Delta = N_{xy} G\Delta
$$

The solution of this equation resulted in a value of $\lambda_y = 1.0$. 

4.4.6 Iterative Solution

The remainder of the analysis was done to compute the critical buckling stress associated with different values of the web slenderness $\lambda_y \leq \lambda_v \leq \lambda_p$ and the strength coefficient, $C_v$. An iterative method was used to solve the fixed point problem:

$$\frac{\tau_{xy} t_w}{(N_{xy})_{cr}} - 1 = 0$$

where $(N_{xy})_{cr}$ is the minimum positive eigenvalue of equation 4.50. Results from this analysis are shown in Figure 4.10. Notice in this figure that the inelastic solution from Section 4.4.2 is shown, and is in good agreement (the maximum variation is of the same order of magnitude as the accuracy of the finite element model). Both curves fall below all of the experimental data. Unfortunately, the outlined method for the calculation of the inelastic solution is inconvenient for design.

![Figure 4.10: Shear strength—finite element results.](image)
4.5 Summary

The theoretical background for elastic and inelastic buckling of flexural members (LTB, FLB and WLB) has been presented for reference. A stress-strain model was specified and used in analysis of these types of buckling failure. In all cases, the elastic theory was first presented, followed by the more general inelastic problem. Bounds on the inelastic region of behavior at the proportional limit and yield stresses have been specified for each mode of failure.

The information in this chapter will be used in Chapter 5 to justify the use of smooth alternative allowable stress functions for shear and bending.
Chapter 5

Development of a Smooth Formulation

The development of a smooth formulation of the structural optimization problem requires that the stress constraints are smooth, or nearly so. This chapter will use a systematic approach to identify and eliminate the sources of nonsmoothness in the AISC ASD [3] specification. Nonsmooth stress interaction ratios (see Appendix A) will first be reformulated to remove envelope functions and absolute values (Section 5.3). The remaining nonsmoothness will be due to (a) discontinuities in the interaction ratios at transitions between different axial stress levels, and (b) nonsmoothness of the allowable stresses. The first of these will be addressed by substituting approximate “smoothing” functions (discussed further in Section 6.1). The smoothing functions (see Appendix B) will be cubic polynomials, which match the function and its first derivative at each end. The second source of nonsmoothness will be addressed by the proposal of alternative, smooth allowable stress functions (Section 5.1).

The idea of checking all possible constraint combinations for a flexural member has been addressed [37] by enumeration of the failure modes in each range of web slenderness. Earlier work where all possible combinations were exhausted dealt with a single member, rather than an entire structure. In the general case, the number
of combinations is prohibitive, so we wish to impose all constraints simultaneously (see the discussion in Section 1.1). Consequently, we seek a set of constraints that are not mutually exclusive. That is, the constraints should be posed so that valid conditions for any one constraint do not invalidate the others. The stress constraints will be written so that each becomes inactive when not applicable. This will allow simultaneous checking of all possible modes of failure without enumeration.

Note: In this chapter the symbol “$\lambda$” refers to a slenderness parameter (rather than to a Lagrange multiplier). The symbol “$\Omega$” refers to a factor of safety (rather than to the feasible domain). The allowable stress ($F_a$), interaction ratio ($IR$), and unity check ($UC$) functions depend on the optimization variables $u$, and sometimes the state variables $y$. This dependency is clearly stated in each section where the functions are first defined but is omitted for clarity thereafter.

5.1 Allowable Stresses

Besides the discontinuous interaction ratios, the allowable stress functions themselves have regions of nonsmoothness. Referring to Appendix A, these are due to:

- Use of $F_t = \min \ldots$ for tension members.
- Use of $F_b = \min \ldots$ for laterally unbraced members.
- Use of $F_{LTB} = \min (\max \ldots, \ldots)$.
- A linear transition of $Q_s$ in the inelastic buckling region.
- A linear transition of $F_{bFLB}$ in the inelastic buckling region.
- A nonlinear transition of $F_v$ in the inelastic buckling region.
- Discontinuity of $k_c$ at the boundary between compact and non-compact sections ($h/t_w = 70$).
• Nonsmoothness of $k_v$ at $a/h = 1$.

The idea of providing continuity in the code equations has been addressed [30], but beyond this we require continuity of the first derivatives (smoothness). Several strategies will be put to use in order to provide this. The most extensive will be aimed at the allowable bending stress. The latitude to use alternatives "if justified by a more precise analysis" is a fundamental part of most design codes, including the AISC ASD [3]. The background material, presented in Chapter 4, justifies the use of the proposed alternatives.

To eliminate the "min" function from the definition of $F_t$, the check for tension on the net effective area will be omitted (see Section 5.1.1). In the case of $F_b$, the "min" function will be eliminated by checking both cases of flange local buckling (FLB) and lateral-torsional buckling (LTB) for all values of $L_b$ (see Section 5.1.3). The LTB section of the ASD code uses a great many simplifications in the theory that are no longer necessary because of the ease of performing these calculations on a computer. The newer LRFD code [4] takes a different approach that is more exact and is the model for the proposed alternate equations. A single equation is needed for the elastic buckling region, and a nearly smooth transition function will be used for the inelastic buckling region. The FLB section of the code will be addressed by replacing the curve for $Q_s$ in the inelastic buckling region with a smooth alternative.

The nonlinear transition in $F_t$ will be replaced by a nearly smooth transition in the inelastic buckling region. Theoretical values indicate that the current specification is overly conservative. The proposed curve, which gives slightly higher allowable stresses, is still conservative. Data to verify this claim is presented in Section 4.4.3.

The $k_v$ curve will be replaced with a smooth alternative. The $k_v$ curve will be replaced with a nearly smooth alternative. All of the functions which are "nearly smooth" will be further refined by the use of smoothing transition functions (see Section 6.1).
5.1.1 Tensile Stress

Since a general expression for $A_e$ is not available, the check for tension on the net effective area is omitted. This eliminates the nonsmoothness due to the use of the "min" function in the general allowable tensile stress equation (see Appendix A).

If we assume $0.5F_u A_e/A \geq 0.6F_y$, the resulting value of $F_t$ is a constant. Accordingly, the basic equation is not changed, but the parameter $\Omega_t$ is defined for uniformity and completeness. The allowable tensile stress is:

$$F_t = \frac{F_y}{\Omega_t}$$  \hspace{1cm} (5.1)

where:

$$\Omega_t = \frac{5}{3}$$ is the factor of safety for tension

5.1.2 Compressive Stress

The transition in $F_a$ is actually smooth due to the use of a quadratic polynomial in the inelastic buckling region (see Appendix A). This curve is chosen to match the function and its first derivative at the inelastic transition. Accordingly, the basic equations are not changed, but the parameters $\lambda_a$, $C_a$ and $\Omega_a$ are defined for uniformity and completeness. The allowable compression stress is:

$$F_a(\lambda_a(u), \lambda_{FLB}(u)) = \frac{Qs(\lambda_{FLB}(u))F_y C_a(\lambda_a(u))}{\Omega_a(\lambda_a(u))}$$  \hspace{1cm} (5.2)

where:

$$C_a(\lambda_a(u)) = \begin{cases} 
1 - \frac{1}{2} \lambda_a(u)^2, & \lambda_a(u) < 1 \\
\frac{1}{2\lambda_a(u)^2}, & \lambda_a(u) \geq 1 
\end{cases}$$  \hspace{1cm} (5.3)
\[ \lambda_a(u) = \frac{K_y L_y}{\pi r_y(u)} \sqrt{\frac{Q_s(\lambda_{FLB}(u))F_y}{2E}} \]  

(5.4)

\[ \lambda_{FLB}(u) \] is given by equation 4.20

\[ Q_s(\lambda_{FLB}(u)) \] is given by equation 5.16

\[ \Omega_a(\lambda_a(u)) = \begin{cases} 
\frac{5}{3} + \frac{3}{8} \lambda_a(u) - \frac{1}{8} \lambda_a(u)^3, & \lambda_a(u) < 1 \\
\frac{23}{12}, & \lambda_a(u) \geq 1 
\end{cases} \]  

(5.5)

Figure 5.5(a) at the end of Section 5.1 shows the factor of safety \( \Omega_a \) over a range of slenderness parameters \( \lambda_a \).

### 5.1.3 Bending Stress

It is assumed that lateral-torsional buckling of the beam is independent of flange local buckling. In general, both cases must be checked so that the usual bending stress constraint is:

\[ f_b \leq F_b \]

\[ F_b = \min \{ F_{bLTB}, F_{bFLB} \} \]

However, \( F_b \) is a nonsmooth curve at the point where \( F_{bFLB} = F_{bLTB} \). Without any loss of generality, it is preferable to enforce two constraints:

\[ f_b \leq F_{bLTB} \]

\[ f_b \leq F_{bFLB} \]

The next two sections discuss the functions for lateral-torsional buckling and flange local buckling.

### 5.1.3.1 Lateral-Torsional Buckling

The critical bending stress \( F_{\sigma_{LTB}} \) is normalized by the yield stress \( F_y \), which results in a unitless parameter \( C_{LTB} \). A transition curve of the form:

\[ C_{LTB}(\lambda_{LTB}; p) = \frac{5}{3} \left[ \frac{F_{bc}}{F_y} + \frac{F_{lu} - F_{bc}}{F_y} \left( \frac{\lambda_{LTB} - \lambda_c}{\lambda_u - \lambda_c} \right)^p \right], \lambda_c < \lambda_{LTB} < \lambda_u \]  

(5.6)
where: $F_{bc} = 0.66F_y$, is the allowable stress at yield

$$F_{bu} = 0.6F_p = 0.6(0.55F_y) = 0.33F_y$$, is the proportional limit divided by a factor of safety of $\frac{5}{3} = \frac{1}{0.6} \approx 1.67$

$\lambda_{LTB}$ is the beam slenderness (equation 4.10)

$\lambda_c$ is the slenderness at which yielding controls (equation 4.17)

$\lambda_u$ is the slenderness at the onset of inelastic behavior (equation 4.16)

$p$ is a parameter to be determined

is proposed to model the behavior of lateral-torsional buckling in the inelastic region.

Its first derivative is:

$$\frac{dC_{LTB}}{d\lambda_{LTB}} = p \left( \frac{0.55}{\lambda_u - \lambda_c} \right) \left( \frac{\lambda_{LTB} - \lambda_c}{\lambda_u - \lambda_c} \right)^{p-1}$$

The curve satisfies the following for all values of $p > 1$:

$$C_{LTB}(\lambda_c) = \frac{5}{3} \frac{F_{bc}}{F_y} = 1.1$$

$$C_{LTB}(\lambda_u) = \frac{5}{3} \frac{F_{bu}}{F_y} = 0.55$$

$$\left. \frac{dC_{LTB}}{d\lambda_{LTB}} \right|_{\lambda_c} = 0$$

so that the curve is smooth at $\lambda_c$ and continuous for all $\lambda_{LTB}$ when:

$$0.6F_yC_{LTB} = \begin{cases} 
F_{bc}, & 0 \leq \lambda_{LTB} \leq \lambda_c \\
F_{bu}, & \lambda_{LTB} = \lambda_u 
\end{cases}$$

The parameter $p$ is to be determined so that the proposed curve:

- Agrees with the derivative of the elastic curve at $\lambda_u$ (smoothness).

- Agrees with the finite element solution (accuracy).

- Provides a sufficient factor of safety.
A database of rolled wide flange sections is taken from the latest AISC code \([4]\). The finite element solution (see Section 4.2.4) is determined at five equally spaced points in the interval \([\lambda_c, \lambda_u]\) for each shape, for values of \(F_y = 36\) and 50 ksi (250 and 340 MPa), and for three cases of loading including:

a. Uniform moment \((C_b = 1, C_a = 0)\).

b. Uniform transverse load at top flange \((C_b = 1.1, C_a = 0.45)\).

c. Unequal end moments \((C_b = 2, C_a = 0)\).

The finite element solutions provide numerical data for the following discussion.

For smoothness, we note that the derivatives of the inelastic curve must match the yield and the elastic curve at the boundaries. The derivative is matched at the lower bound for any value of \(p > 1\). By setting the derivatives equal at \(\lambda_u\) and solving for \(p\) within the database of wide flange sections, values range from 0.74 to 1.2 with a mean value of 0.91. From this point of view, a value of \(p\) slightly greater than 1 would be preferred.

For accuracy, the residual between the finite element solution and the predicted allowable stress should be minimized. The nonlinear least squares problem:

\[
\min_{p > 1} \frac{1}{2} \| \mathbf{F}_{\text{crLTB}}^{\text{FEM}} - F_y C_{\text{LTB}}(p; \lambda_{\text{LTB}}^{\text{FEM}}) \|^2_2
\]

(5.7)

where: \(\lambda_{\text{LTB}}^{\text{FEM}}\) is a vector of slenderness values for which the finite element solution is known

\(\mathbf{F}_{\text{crLTB}}^{\text{FEM}}\) is a vector of critical stresses based on the finite element solution of equation 4.15 and \(F_{\text{crLTB}} = M_{\text{cr}}/S_2\)

is solved resulting in values of \(p = 1.2, 1.3\), and 1.6 for loading cases a, b, and c respectively, and \(p = 1.3\) considering all cases simultaneously.

After considering the requirements of accuracy and smoothness, a value of \(1 < p \leq 1.3\) is determined. It is also desirable to gain a sufficient factor of safety against
failure using the finite element solution as an ultimate stress. The factor of safety is defined as:

\[ \Omega_{LTB}^{FEM} = \frac{F_{\sigma_{LTB}}^{FEM}}{0.6F_yC_{LTB}} \]

which when computed for \( p = 1 \) and 1.3, is found to be greater than 1.63 and 1.55 respectively, for all finite element solutions. The usual factor of safety for lateral-torsional buckling is 1.67, so \( p < 1.3 \) is desirable. A value of \( p = 1.1 \) is a good compromise, providing some smoothness and a safety factor of at least 1.60 (over 80% of cases have a safety factor of 1.67 or greater). The apparent lack of sufficient safety factor is offset by the conservativeness built into the finite element solution, which assumes pinned ends in all cases.

The allowable stress for lateral-torsional buckling of wide flange beams is given by:

\[ F_{bLTB}(\lambda_{LTB}(u), \lambda(u)) = \frac{F_yC_{LTB}(\lambda_{LTB}(u), \lambda(u))}{\Omega_{LTB}} \] (5.8)

where:

\[ C_{LTB}(\lambda_{LTB}(u), \lambda(u)) = \begin{cases} 1.1, & 0 \leq \lambda_{LTB}(u) \leq \lambda_c \\ 1.1 - 0.55 \left( \frac{\lambda_{LTB}(u) - \lambda_u}{\lambda_u(u) - \lambda_c} \right)^{1.1}, & \lambda_c < \lambda_{LTB}(u) < \lambda_u(u) \\ \frac{F_{\sigma_{LTB}}(\lambda_{LTB}(u))}{F_y}, & \lambda_{LTB}(u) \geq \lambda_u(u) \end{cases} \] (5.9)

\( \lambda_{LTB}(u) \) is given by equation 4.10
\( \lambda_c \) is given by equation 4.17
\( \lambda_u(u) \) is given by equation 4.16
\( \Omega_{LTB} = \frac{5}{3} \) is the factor of safety for LTB
\( F_{\sigma_{LTB}}(\lambda_{LTB}(u)) \) is given by equation 4.9

Figures 5.1(a), 5.1(b), and 5.1(c) show the proposed curve for a W6X20 beam (\( F_y = 50 \) ksi (340 MPa)), along with the finite element results and nonsmooth ASD and LRFD curves. The LRFD “allowable stress” is the nominal moment \( M_n \) divided
by the elastic section modulus $S_x$ and divided by a factor of safety of $\frac{5}{3}$. In general, the allowable stress agrees between the proposed curve and the ASD code curve. But, because of the discontinuity in the ASD curve as $F_{LTP}$ transitions between $0.6F_y$ and $0.66F_y$, the two specifications may disagree by roughly 10%.
Figure 5.1: Proposed LTB curve (W6X20, $F_y = 50ksi$).
5.1.3.2 Flange Local Buckling

The critical bending stress is normalized by the yield stress $F_y$, which results in a unitless parameter $Q_s$. A transition curve of the form:

$$Q_s(\lambda_{FLB}; p; \lambda_y) = 1 + \frac{F_p - F_y}{F_y} \left( \frac{\lambda_{FLB} - \lambda_y}{\lambda_p - \lambda_y} \right)^p, \lambda_y < \lambda_{FLB} < \lambda_p$$  \hspace{1cm} (5.10)

where: $F_p = 0.5F_y$, is the proportional limit for uniform compression

$\lambda_{FLB}$ is the flange slenderness (equation 4.20)

$\lambda_p$ is the slenderness at the onset of inelastic behavior (1.34)

$\lambda_y$ is the slenderness at which yielding controls (to be determined)

$p$ is a parameter to be determined

is proposed to model the behavior of flange local buckling in the inelastic region (see [79]). Its first derivative is:

$$\frac{dQ_s}{d\lambda_{FLB}} = p \left( \frac{-0.5}{\lambda_p - \lambda_y} \right) \left( \frac{\lambda_{FLB} - \lambda_y}{\lambda_p - \lambda_y} \right)^{p-1}$$

The curve satisfies the following for all values of $p > 1$:

$$Q_s(\lambda_p) = 1$$

$$Q_s(\lambda_y) = 1 + \frac{F_p - F_y}{F_y} = 0.5$$

$$\left. \frac{dQ_s}{d\lambda_{FLB}} \right|_{\lambda_y} = 0$$

so that the curve is smooth at $\lambda_y$ and continuous for all $\lambda_{FLB}$ when:

$$Q_s = \begin{cases} 
1, & 0 \leq \lambda_{FLB} \leq \lambda_y \\
0.5, & \lambda_{FLB} = \lambda_p 
\end{cases}$$

The parameter $p$ is to be determined so that the proposed curve:

---

1The naming convention for the unitless strength parameter is inconsistent with that used in the sections on lateral-torsional buckling and web local buckling (shear). This is because historically, the ratio for flange local buckling has been known as $Q_s$. 

• Agrees with the derivative of the elastic curve at \( \lambda_p \) (smoothness).

• Provides a sufficient factor of safety.

Experimental data (from [41] and [12]) will be used to determine the factor of safety achieved. This will consider only experiments in which failure was a result of flange local buckling, and in which the critical stress was less than the yield stress (i.e. without any strain hardening).

For smoothness, we note that the derivatives of the inelastic curve must match the yield and the elastic curve at the boundaries. The derivative is matched at the lower bound for any value of \( p > 1 \). By setting the derivatives equal at \( \lambda_p \) and solving for \( p \):

\[
p_s = \frac{\pi^2}{3(1-\nu^2)} \frac{\lambda_p - \lambda_y}{\lambda_p^3}
\]

(5.11)

which depends on the choice of \( \lambda_y \). This is examined further in the remainder of this section.

To determine the parameters in equation 5.10, the residual between the experimental data and predicted allowable stress should be minimized. Here the parameter \( k_e \) is assumed to vary with the ratio \( h/t_w \) (as in [41]). The constrained nonlinear least squares problem:

\[
\min_x \frac{1}{2} \| Q_s^{ex} - Q_s(x; \lambda_{FLD}^{ex}) \|_2^2
\]

s.t. \( x^L \leq x \leq x^U \),

\( 0 < k_e \leq 1.277 \)

\[
\text{where: } x = \begin{bmatrix} a \\ q \\ \lambda_y \\ p \end{bmatrix}, \quad x^L = \begin{bmatrix} 0 \\ -\infty \\ \lambda_0 \\ 1 \end{bmatrix}, \quad x^U = \begin{bmatrix} \infty \\ 0 \\ \lambda_p \\ \infty \end{bmatrix}
\]

\( (k_e)_i = a \left( \frac{h}{l_{w,i}} \right)^q, \quad i = 1, \ldots, N_{ex} \)

\( a, q \) are parameters to be determined.
\( \lambda_{FLB}^{ex} \) is a vector of FLB slenderness parameters from experiments

\( Q_s^{ex} \) is a vector of normalized critical stresses from experiments = \( \frac{F_{crFLB}^{ex}}{F_y} \)

\( N_{ex} \) is the number of experimental data points

\( h = d - 2t_f \) is the clear distance between flanges

\( t_w \) is the web thickness

is solved resulting in values of \( a = 5.4, q = -0.49, \lambda_y = 0.56 \), and \( p = 1 \), with a maximum residual of about 3%. Using equation 5.11, \( p_s = 1.2 \) would result in a smooth curve at the elastic transition, \( \lambda_p \). Note that the least squares value of \( \lambda_y \) (0.56 = 95/\( \sqrt{E} \)) is the same as the bound in the current ASD specification [3]. The least squares problem (Problem 5.12) is solved again adding the constraints:

\[
p = p_s
\]

\[
q = -0.5
\]

This results in values of \( a = 5.4, q = -0.5, \lambda_y = 0.56 \), and \( p = 1.2 \), with a maximum residual of about 5%. This is a more desirable solution as long as a sufficient factor of safety is provided. Table 5.1 shows the critical stress predicted for the experimental data set using the ASD [3], LRFD [4], and proposed method. The factor of safety achieved by the proposed method:

\[
\Omega_{FLB}^{ex} = \frac{F_{crFLB}^{ex}}{0.6F_yQ_s}
\]

is tabulated, and it ranges from 1.65 to 1.91. This is within 1% of the desired value of 1.67.

The parameter \( k_c \) has a maximum value of 1.277 (see [14]) when the flange is fully restrained by the web (fixed). In order to provide this at \( h/t_w = 0 \), a quadratic polynomial is fit between 0 and 25 where \( k_c = 1 \). The coefficients are chosen so that
<table>
<thead>
<tr>
<th>Ref</th>
<th>Mark</th>
<th>$\frac{h}{t_w}$</th>
<th>$\lambda_{FLB}$</th>
<th>$F_y$ (ksi)</th>
<th>$F_{cr,FLB}^x$ (ksi)</th>
<th>ASD$^a$</th>
<th>LRFD$^b$</th>
<th>Proposed$^a$</th>
<th>$\Omega_{FLB}^{xx}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>[41]</td>
<td>2</td>
<td>245</td>
<td>216</td>
<td>62.1</td>
<td>39.63</td>
<td>35.20</td>
<td>38.07</td>
<td>34.53</td>
<td>1.91</td>
</tr>
<tr>
<td>[41]</td>
<td>4</td>
<td>236</td>
<td>207</td>
<td>59.0</td>
<td>38.49</td>
<td>36.42</td>
<td>38.72</td>
<td>35.12</td>
<td>1.83</td>
</tr>
<tr>
<td>[41]</td>
<td>5</td>
<td>203</td>
<td>207</td>
<td>62.0</td>
<td>40.06</td>
<td>38.11</td>
<td>37.82</td>
<td>36.95</td>
<td>1.81</td>
</tr>
<tr>
<td>[41]</td>
<td>6</td>
<td>201</td>
<td>208</td>
<td>63.4</td>
<td>40.67</td>
<td>38.45</td>
<td>37.95</td>
<td>37.43</td>
<td>1.81</td>
</tr>
<tr>
<td>[41]</td>
<td>1A</td>
<td>176</td>
<td>173</td>
<td>55.1</td>
<td>41.57</td>
<td>41.76</td>
<td>41.97</td>
<td>40.50</td>
<td>1.71</td>
</tr>
<tr>
<td>[41]</td>
<td>2A</td>
<td>174</td>
<td>171</td>
<td>54.0</td>
<td>41.10</td>
<td>41.30</td>
<td>41.48</td>
<td>40.19</td>
<td>1.70</td>
</tr>
<tr>
<td>[41]</td>
<td>3A</td>
<td>154</td>
<td>166</td>
<td>54.4</td>
<td>42.00</td>
<td>42.29</td>
<td>41.73</td>
<td>41.43</td>
<td>1.69</td>
</tr>
<tr>
<td>[41]</td>
<td>4A</td>
<td>154</td>
<td>166</td>
<td>54.7</td>
<td>41.39</td>
<td>42.52</td>
<td>41.95</td>
<td>41.66</td>
<td>1.66</td>
</tr>
<tr>
<td>[41]</td>
<td>5A</td>
<td>111</td>
<td>151</td>
<td>54.3</td>
<td>44.82</td>
<td>44.57</td>
<td>43.99</td>
<td>44.50</td>
<td>1.68</td>
</tr>
<tr>
<td>[41]</td>
<td>6A</td>
<td>109</td>
<td>153</td>
<td>55.3</td>
<td>44.56</td>
<td>45.13</td>
<td>44.45</td>
<td>45.01</td>
<td>1.65</td>
</tr>
<tr>
<td>[41]</td>
<td>7A</td>
<td>73.6</td>
<td>139</td>
<td>55.1</td>
<td>47.90</td>
<td>47.08</td>
<td>47.83</td>
<td>47.57</td>
<td>1.63</td>
</tr>
<tr>
<td>[41]</td>
<td>8A</td>
<td>73.2</td>
<td>140</td>
<td>55.9</td>
<td>49.76</td>
<td>47.67</td>
<td>48.42</td>
<td>48.15</td>
<td>1.72</td>
</tr>
<tr>
<td>[41]</td>
<td>10A</td>
<td>149</td>
<td>141</td>
<td>61.6</td>
<td>52.23</td>
<td>52.74</td>
<td>54.50</td>
<td>52.86</td>
<td>1.65</td>
</tr>
<tr>
<td>[12]</td>
<td>A1</td>
<td>21.3</td>
<td>128</td>
<td>122</td>
<td>111.0</td>
<td>107.7</td>
<td>111.7</td>
<td>110.0</td>
<td>1.68</td>
</tr>
<tr>
<td>[12]</td>
<td>A2</td>
<td>21.3</td>
<td>128</td>
<td>122</td>
<td>112.0</td>
<td>107.6</td>
<td>111.6</td>
<td>109.9</td>
<td>1.70</td>
</tr>
</tbody>
</table>

Table 5.1: Experimental and predicted critical FLB stresses.

$^aF_yQ_s$ (ksi)

$^bM_n/S_x$ (ksi)

$k_c$ is smooth at $h/t_w = 25$:

$$k_c(u) = \begin{cases} 
1.3 - 0.1 \left( \frac{h(u)/t_w}{25} \right) - 0.2 \left( \frac{h(u)/t_w}{25} \right)^2, & 0 \leq \frac{h(u)}{t_w} < 25 \\
\frac{5}{\sqrt{h(u)/t_w}}, & \frac{h(u)}{t_w} \geq 25 
\end{cases} \quad (5.13)$$

The derivative at $h/t_w = 0$ is arbitrary. Figure 5.2 shows the proposed curve as well as the discontinuous ASD curve. The experimental data from [41] and [12] is overlaid for reference. Note that the inclusion of stocky web data from Beg and Hladnik [12] qualifies the use of $k_c > 1$.

The transition from yield to strain hardening is assumed to provide about 10% of additional strength when the flanges are compact (i.e., at $\lambda_0$). Since the proposed curve for $Q_s$ only provides a transition from $\lambda_y$ to $\lambda_p$, a variable factor of safety will be used between $\lambda_0$ and $\lambda_y$. This is chosen in accordance with the current ASD specification [3] so that the allowable stress is increased from $0.6F_y$ at initial yield to
0.66F_y at strain hardening. However, since we require smoothness at \( \lambda_0 \) and \( \lambda_y \), a linear transition as in the ASD is insufficient, and a cubic polynomial is needed. The coefficients are chosen so that the curve is continuous and smooth at \( \lambda_0 \) and \( \lambda_y \):

\[
\Omega_{FLB}(\lambda_{FLB}(u)) = \begin{cases} 
\frac{3}{2}, & 0 \leq \lambda_{FLB}(u) \leq 0.44 \\
\frac{3}{2} + \left(1 - \frac{\lambda_{FLB}(u) - 0.44}{0.36}\right) \left(\frac{\lambda_{FLB}(u) - 0.44}{0.12}\right)^2, & 0.44 < \lambda_{FLB}(u) < 0.56 \\
\frac{5}{3}, & \lambda_{FLB}(u) \geq 0.56 
\end{cases}
\]

where:

\( \lambda_{FLB}(u) \) is given by equation 4.20

Figure 5.5(b) at the end of Section 5.1 shows the factor of safety \( \Omega_{FLB} \) over a range of flange slenderness parameters \( \lambda_{FLB} \).

The allowable stress for flange local buckling of wide flange beams is given by:

\[
F_{bFLB}(\lambda_{FLB}(u)) = \frac{F_yQ_x(\lambda_{FLB}(u))}{\Omega_{FLB}(\lambda_{FLB}(u))}
\]

(5.15)
where:

\[ Q_s(\lambda_{FLB}(u)) = \begin{cases} 
1, & 0 \leq \lambda_{FLB}(u) \leq 0.56 \\
1 - 0.5 \left( \frac{\lambda_{FLB}(u) - 0.56}{0.78} \right)^{1.2}, & 0.56 < \lambda_{FLB}(u) < 1.34 \\
\frac{F_{crFLB}(\lambda_{FLB}(u))}{F_y}, & \lambda_{FLB}(u) \geq 1.34 
\end{cases} \]  

(5.16)

\( \lambda_{FLB}(u) \) is given by equation 4.20

\( k_c(u) \) (in \( \lambda_{FLB} \)) is given by equation 5.13

\( \Omega_{FLB}(\lambda_{FLB}(u)) \) is given by equation 5.14

\( F_{crFLB}(\lambda_{FLB}(u)) \) is given by equation 4.19

Figure 5.3 shows the proposed curve along with the nonsmooth ASD and LRFD curves. The experimental data from [41] and [12] is overlaid for reference.

Figure 5.3: Proposed FLB curve.
5.1.4 Shear Stress

The critical shear stress is normalized by the shear yield stress, $\tau_y$, which results in a unitless parameter $C_v$. A curve of the form:

$$C_v(\lambda_v; p) = 1 + \frac{F_p - F_y}{F_y} \left( \frac{\lambda_v - \lambda_y}{\lambda_p - \lambda_y} \right)^p, \quad \lambda_y < \lambda_v < \lambda_p \quad (5.17)$$

where: $F_p = 0.8F_y$, is the proportional limit for uniform shear

- $\lambda_v$ is given by equation 4.24
- $\lambda_y$ is the slenderness at which yielding controls
- $\lambda_p$ is the slenderness at the onset of inelastic behavior
- $p$ is a parameter to be determined

is proposed to model the shear strength coefficient in the inelastic region. Its first derivative is:

$$\frac{dC_v}{d\lambda_v} = p \left( -0.2 \right) \left( \frac{\lambda_v - \lambda_y}{\lambda_p - \lambda_y} \right)^{p-1}$$

The curve satisfies the following for all values of $p > 1$:

$$C_v(\lambda_y) = 1$$
$$C_v(\lambda_p) = 1 + \frac{F_p - F_y}{F_y} = 0.8$$
$$\left. \frac{dC_v}{d\lambda_v} \right|_{\lambda_y} = 0$$

so that the curve is smooth at $\lambda_y$ and continuous for all $\lambda_v$ when:

$$C_v = \begin{cases} 1, & 0 \leq \lambda_v \leq \lambda_y \\ 0.8, & \lambda_v = \lambda_p \end{cases}$$

The parameter $p$ is to be determined so that the proposed curve:

- Agrees with the derivative of the elastic curve at $\lambda_p$ (smoothness).
• Agrees with the finite element solution (accuracy).

The finite element solution (see Section 4.4.4 and [80]) is determined at equally spaced points in the interval \([\lambda_y, \lambda_p]\). This solution provides numerical data for the following discussion.

For smoothness, we note that the derivatives of the inelastic curve must match the yield and elastic curve at the boundaries. The derivative is matched at the lower bound for any value of \(p > 1\). By setting the derivatives equal at \(\lambda_p\) and solving for \(p\):

\[
p_s = \frac{\pi^2}{0.72(1 - \nu^2)} \frac{\lambda_p - \lambda_y}{\lambda_p^3}
\]

which gives a value of \(p_s = 2.2\).

For accuracy, the residual between the finite element solution and predicted allowable stress should be minimized. The nonlinear least squares problem:

\[
\min_{1 < p < 2.2} \frac{1}{2} \left\| C^{\text{FEM}}_v - C_v(p; \lambda^{\text{FEM}}_v) \right\|_2^2 \tag{5.18}
\]

where: \(\lambda^{\text{FEM}}_v\) is a vector of web slenderness values for which the finite element solution is known

\(C^{\text{FEM}}_v\) is a vector of strength coefficients based on the finite element solution

is solved resulting in a value of \(p = 1.7\) with residual error in \(C_v\) of the same order of magnitude as the accuracy of the finite element model. Thus, a sufficient factor of safety is automatically achieved if \(p \leq 1.7\). Using the least squares value of \(p\) in Equation 5.17 is adequate for design.

The allowable shear stress at yield is historically assumed to be about 10% beyond \(0.6F_y/(5/3) = 0.36F_y\), thus a variable factor of safety will be used between \(\lambda_y\) and \(\lambda_p\). This is chosen in accordance with the current ASD specification [3] so that the allowable stress is increased to \(0.4F_y\) at yield. Since we require smoothness at \(\lambda_y\) and
\( \lambda_p \), a cubic polynomial is needed. The coefficients are chosen so that the curve is continuous and smooth at \( \lambda_y \) and \( \lambda_p \):

\[
\Omega_v(\lambda_v(u)) = \begin{cases} 
\frac{3}{2}, & 0 \leq \lambda_v(u) \leq 1 \\
\frac{3}{2} + \left( \frac{1}{2} - \frac{\lambda_v(u) - 1}{0.11} \right) \left( \frac{\lambda_v(u) - 1}{0.37} \right)^2, & 1 < \lambda_v(u) < 1.37 \\
\frac{5}{3}, & \lambda_v(u) \geq 1.37
\end{cases}
\] (5.19)

where:

\( \lambda_v(u) \) is given by equation 4.24

Figure 5.5(c) at the end of Section 5.1 shows the factor of safety \( \Omega_v \) over a range of web slenderness parameters \( \lambda_v \).

The allowable shear stress for unstiffened members is given by:

\[
F_v(\lambda_v(u)) = \frac{0.6F_y C_v(\lambda_v(u))}{\Omega_v(\lambda_v(u))}
\] (5.20)

where:

\[
C_v(\lambda_v(u)) = \begin{cases} 
1, & 0 \leq \lambda_v(u) \leq 1 \\
1 - 0.2 \left( \frac{\lambda_v(u) - 1}{0.37} \right)^{1.7}, & 1 < \lambda_v(u) < 1.37 \\
\frac{F_{crv}(\lambda_v(u))}{F_y}, & \lambda_v(u) \geq 1.37
\end{cases}
\] (5.21)

\( \lambda_v(u) \) is given by equation 4.24

\( k_v = 5.34 \) (constant for unstiffened members)

\( \Omega_v(\lambda_v(u)) \) is given by equation 5.19

\( F_{crv}(\lambda_v(u)) \) is given by equation 4.21

The allowable shear stress for stiffened members is given by (also see above):

\[
F_v(u, \lambda_v(u)) = \frac{0.6F_y}{\Omega_v(\lambda_v(u))} \left( C_v(\lambda_v(u)) + \frac{1 - C_v(\lambda_v(u))}{1.2\sqrt{1 + (a/h(u))^2}} \right)
\] (5.22)

where:

\( k_v(u) \) (in \( \lambda_u \)) is given by equation 4.22

\( a \) is the distance between transverse stiffeners
Figure 5.4 shows the proposed curve along with the nonsmooth ASD curve (the LRFD curve is similar to the ASD curve). The experimental data from [53], [60], [18], and [52] is overlaid for reference.

Figure 5.4: Proposed WLB curve.
Figure 5.5: Proposed, smooth factor of safety curves.
5.2 Sample Calculations with Proposed Allowable Stresses

Some sample calculations are presented herein to illustrate the use of the proposed allowable stress equations. These are compared to the AISC [3] allowable stresses for verification.

5.2.1 Lateral-Torsional Buckling

Given a W6X20 beam, subjected to uniform moment,

\[ F_y = 50 \text{ ksi}, \]
\[ L_b = 15 \text{ ft (180 in), } C_b = 1, \text{ and } C_a = 0. \]

From section property tables:
\[ b_f = 6.02 \text{ in, } r_y = 1.5 \text{ in,} \]
\[ X_1 = 3600 \text{ ksi, } X_2 = 805 \times 10^{-6} \text{ ksi}^{-2}, \]
\[ r_T = 1.64 \text{ in, and } \frac{d}{A_f} = 2.82 \text{ in}^{-1}. \]

(a) ASD Design Criteria

Calculate the maximum unbraced length for full plastic yield:

\[
L_c = \min \left\{ \frac{76b_f}{\sqrt{F_y}} \frac{20,000}{(d/A_f)F_y} \right\} \\
= \min \left\{ \frac{76 \times 6.02}{\sqrt{50}} \frac{20,000}{1.64 \times 2.82 \times 50} \right\} \\
= \min \{64.7, 142\} = 64.7 < L_b
\]
Use either equations F1-6 or F1-7, and equation F1-8. Determine if inelastic:

\[
\frac{L_b}{r_T} = \frac{180}{1.64} = 110
\]

\[
\sqrt{\frac{102,000C_b}{F_y}} = \sqrt{\frac{102,000 \times 1}{50}} = 45.2 < 110
\]

\[
\sqrt{\frac{510,000C_b}{F_y}} = \sqrt{\frac{510,000 \times 1}{50}} = 101 < 110
\]

Use equation F1-7 for warping torsional stiffness:

\[
F_{bw} = \frac{170,000C_b}{(L_b/r_T)^2} \leq 0.6F_y
\]

\[
= \frac{170,000 \times 1}{110^2} = 14.0 \text{ ksi} = 0.28F_y
\]

Use equation F1-8 for St. Venant torsional stiffness:

\[
F_{bSV} = \frac{12,000C_b}{L_b(d/A_T)} \leq 0.6F_y
\]

\[
= \frac{12,000 \times 1}{180 \times 2.82} = 23.6 \text{ ksi} = 0.47F_y
\]

Calculate the allowable bending stress:

\[
F_{bLTB} = \min \{0.6F_y, \max \{F_{bw}, F_{bSV}\}\}
\]

\[
= \min \{0.6 \times 50, \max \{14.0, 23.6\}\}
\]

\[
= \min \{30, 23.6\} = 23.6 \text{ ksi}
\]

(b) Proposed Design Criteria
Calculate the slenderness and lower and upper bounds:

$$\lambda_{LTB} = \frac{L_b}{r_y} = \frac{180}{1.5} = 120$$

$$\lambda_c = 300\sqrt{\frac{C_b}{F_y}} = 300\sqrt{\frac{1}{50}} = 42.4 < 120$$

$$\lambda_u = \frac{C_bX_1}{0.55F_y}\sqrt{1 + \sqrt{1 + X_2\left(\frac{0.55F_y}{C_b}\right)^2}}$$

$$= \frac{1 \times 3600}{0.55 \times 50}\sqrt{1 + \sqrt{1 + 0.000805\left(\frac{0.55 \times 50}{1}\right)^2}}$$

$$= 197 > 120$$

Inelastic buckling controls. Calculate the allowable stress using equation 5.8:

$$C_{LTB} = 1.1 - 0.55\left(\frac{\lambda_{LTB} - \lambda_c}{\lambda_u - \lambda_c}\right)^{1.1}$$

$$= 1.1 - 0.55\left(\frac{120 - 42.4}{197 - 42.4}\right)^{1.1} = 0.842$$

$$F_{bLTB} = \frac{F_yC_{LTB}}{\Omega_{LTB}}$$

$$= \frac{50 \times 0.842}{5/3} = 25.3 \text{ ksi}$$

Note that the allowable stress using this method differs from the ASD by 1.7 ksi (about 7%).

### 5.2.2 Flange Local Buckling

Given a welded, doubly symmetric wide flange,

$$d = 36 \text{ in}, \ t_w = 0.25 \text{ in}, \ b_f = 12 \text{ in}, \ t_f = 0.5 \text{ in}, \text{ and}$$

$$F_y = 50 \text{ ksi}.$$  

Calculate the web and flange slenderness:
\[ h = d - 2t_f = 36 - 2 \times 0.5 = 35, \]
\[ \frac{h}{t_w} = \frac{35}{0.25} = 140, \text{ and} \]
\[ \frac{b_f}{2t_f} = \frac{12}{2 \times 0.5} = 12 \]

(a) ASD Design Criteria

Calculate the flange restraint parameter:

\[
k_c = \frac{4.05}{(h/t_w)^{0.46}}
\]
\[
= \frac{4.05}{(140)^{0.46}} = 0.417
\]

Calculate the flange slenderness parameter:

\[
\frac{b_f}{2t_f} \sqrt{\frac{F_y}{k_c}} = 12 \sqrt{\frac{50}{0.417}} = 131
\]

This falls between 95 and 195. Thus, the flange is slender but below the elastic buckling range. Calculate the allowable stress:

\[
Q_s = 1.293 - 0.00309 \frac{b_f}{2t_f} \sqrt{\frac{F_y}{k_c}}
\]
\[
= 1.293 - 0.00309 \times 131 = 0.887
\]

\[
F_{bFLB} = 0.6F_yQ_s
\]
\[
= 0.6 \times 50 \times 0.887 = 26.6 \text{ ksi}
\]

(b) Proposed Design Criteria

Calculate the flange restraint parameter:

\[
k_c = \frac{5}{\sqrt{h/t_w}}
\]
\[
= \frac{5}{\sqrt{140}} = 0.423
\]
Calculate the flange slenderness parameter:

\[
\frac{b_f}{2t_f} \sqrt{\frac{F_y}{k_cE}} = 12 \sqrt{\frac{50}{0.423 \times 29,000}} = 0.77
\]

This falls between 0.56 and 1.34. Thus, the flange is in the inelastic buckling range.

Calculate the allowable stress using equation 5.15:

\[
Q_s = 1 - 0.5 \left( \frac{\lambda_{FLB} - 0.56}{0.78} \right)^{1.2}
\]

\[
= 1 - 0.5 \left( \frac{0.77 - 0.56}{0.78} \right)^{1.2} = 0.896
\]

\[
\Omega_{FLB} = \frac{5}{3}
\]

\[
F_{bFLB} = \frac{F_yQ_s}{\Omega_{FLB}}
\]

\[
= \frac{50 \times 0.896}{5/3} = 26.9 \text{ ksi}
\]

Note that the allowable stress using this method differs from the ASD by 0.3 ksi (about 1%).

### 5.2.3 Web Local Buckling

Given an unstiffened, welded, doubly symmetric wide flange,

\[d = 36 \text{ in}, \ t_w = 0.5 \text{ in}, \ b_f = 12 \text{ in}, \ t_f = 0.5 \text{ in},\]

\[F_y = 50 \text{ ksi}, \text{ and } k_v = 5.34.\]

Calculate the web slenderness:

\[h = d - 2t_f = 36 - 2 \times 0.5 = 35,\]

\[\frac{h}{t_w} = \frac{35}{0.5} = 70\]
(a) ASD Design Criteria

The ratio of web width-to-thickness \( h/t_w \) exceeds the yield limit of \( 380/\sqrt{F_y} = 54 \), so that equation F4-2 is used. Calculate the shear coefficient (assume inelastic):

\[
C_v = \frac{190}{h/t_w} \sqrt{\frac{k_v}{F_y}}
\]

\[
= \frac{190}{70} \sqrt{\frac{5.34}{50}} = 0.887 > 0.8
\]

This value is more than 0.8, so the web is in the inelastic buckling range. Calculate the allowable stress:

\[
F_v = \frac{F_y}{2.89 C_v}
\]

\[
= \frac{50}{2.89 \times 0.887} = 15.3 \text{ ksi}
\]

(b) Proposed Design Criteria

Calculate the flange slenderness parameter:

\[
\frac{h}{t_w \sqrt{\frac{F_y}{k_v E}}} = 70 \sqrt{\frac{50}{5.34 \times 29,000}} = 1.26
\]

This falls between 1 and 1.37, thus the web is in the inelastic buckling range. Calculate
the allowable stress using equation 5.20:

\[
C_v = 1 - 0.2 \left( \frac{\lambda_v - 1}{0.37} \right)^{1.7}
\]

\[
= 1 - 0.2 \left( \frac{1.26 - 1}{0.37} \right)^{1.7} = 0.890
\]

\[
\Omega_v = \frac{3}{2} + \left( \frac{1}{2} - \frac{\lambda_v - 1}{1.11} \right) \left( \frac{\lambda_v - 1}{0.37} \right)^2
\]

\[
= \frac{3}{2} + \left( \frac{1}{2} - \frac{1.26 - 1}{1.11} \right) \left( \frac{1.26 - 1}{0.37} \right)^2 = 1.63
\]

\[
F_v = \frac{0.6F_yC_v}{\Omega_v}
\]

\[
= \frac{0.6 \times 50 \times 0.890}{1.63} = 16.4 \text{ ksi}
\]

Note that the allowable stress using this method differs from the ASD by 1.1 ksi (about 7%).

### 5.3 Interaction Ratios

Reviewing the unity check requirements from the AISC ASD [3] specification (see also Appendix A):

\[
UC = \max \left( IR, \frac{|f_a|}{F_v} \right)
\]

where:

\[
IR = \begin{cases} 
\frac{f_a}{F_v} + \frac{|f_a|}{F_v}, & f_a \geq 0 \\
\frac{f_a}{F_a} + \frac{|f_a|}{F_b}, & -0.15F_a \leq f_a < 0 \\
\max \left( -\frac{f_a}{F_a} + B_1 \frac{|f_a|}{F_b}, \cdots \right), & f_a < -0.15F_a
\end{cases}
\]

we see that the combined axial and bending stress interaction ratios (IRs) depend primarily on the axial stress, \(f_a\). By concentrating on the distinct axial stress regions (tension, small compression, and large compression), the three equations are rearranged into two more general equations for yield and buckling failure. These involve
the generalized allowable stress equations $F_e, F_a$, and $F_b$, which vary with the state as well as the design variables. After the addition of the required individual check on axial stress, the basic member unity check (UC) expression becomes:

$$UC(u, y) = \left[ \frac{f_a(u,y)}{F_a(u,y)} + \frac{f_b(u,y)}{F_b(u,y)}, \frac{f_a(u,y)}{F_a(u,y)} + \frac{|f_b(u,y)|}{F_b(u,y)}, -\frac{|f_a(u,y)|}{F_a(u)} \right]^T$$  \hspace{1cm} (5.23)

where:

$$\bar{F}_t(u, y) = \begin{cases} F_t, & f_a(u, y) \geq 0 \\ -F_t, & -0.15F_a(u) \leq f_a(u, y) < 0 \\ -0.6Q_t(u)f_y, & f_a(u, y) < -0.15F_a(u) \end{cases}$$  \hspace{1cm} (5.24)

$$\bar{F}_a(u, y) = \begin{cases} F_t, & f_a(u, y) \geq 0 \\ -F_a(u), & f_a(u, y) < 0 \end{cases}$$  \hspace{1cm} (5.25)

$$\bar{F}_b(u, y) = \frac{F_b(u)}{B_1(u, y)}$$  \hspace{1cm} (5.26)

$$B_1(u, y) = \begin{cases} 1, & f_a(u, y) \geq F_a(u)\lambda_{1B_1}(u) \\ \frac{C_m}{1 + \frac{f_b(u,y)}{F_b(u,y)}}, & f_a(u, y) < F_a(u)\lambda_{1B_1}(u) \end{cases}$$  \hspace{1cm} (5.27)

$$\lambda_{1B_1}(u) = \min \left( -0.15, (C_m - 1)\frac{F_a(u)}{F_a(u)} \right)$$  \hspace{1cm} (5.28)

The factor $B_1$ models the effect of secondary moments when axial force and moment are both present (see [9]). Its upper bound $\lambda_{1B_1}$ is chosen so that $B_1 \leq 1$. To account for absolute values of bending and shear stress, we consider combinations ($\pm f_b, \pm f_a$) of each of the first three equations in 5.23, resulting in seven equations:

$$UC(u, y) = \left[ IR_t(u, y) \pm \frac{f_b(u,y)}{F_b(u)}, IR_a(u, y) \pm IR_b(u, y), \pm IR_a(u, y), -\frac{|f_a(u,y)|}{F_a(u)} \right]^T$$  \hspace{1cm} (5.29)
where the generalized interaction ratios are defined by:

\[
IR_t(u, y) = \frac{f_a(u, y)}{F_t(u, y)} \tag{5.30}
\]

\[
IR_a(u, y) = \frac{f_a(u, y)}{F_a(u, y)} \tag{5.31}
\]

\[
IR_b(u, y) = \frac{f_b(u, y)}{F_b(u, y)} \tag{5.32}
\]

\[
IR_v(u, y) = \frac{f_v(u, y)}{F_v(u, y)} \tag{5.33}
\]

Section 5.1.3 requires that we check the bending stress against both LTB and FLB. This increases the number of member constraint checks (equation 5.29) by four, resulting in the complete unity check expression (the dependencies on \( u \) and \( y \) are omitted for clarity):

\[
UC(u, y) = \begin{bmatrix}
IR_t + \frac{f_b}{F_bFLB} \\
IR_t - \frac{f_b}{F_bFLB} \\
IR_t + \frac{f_b}{F_bLTB} \\
IR_t - \frac{f_b}{F_bLTB} \\
IR_a + IR_b|_{F_a=F_bFLB} \\
IR_a - IR_b|_{F_a=F_bFLB} \\
IR_a + IR_b|_{F_a=F_bLTB} \\
IR_a - IR_b|_{F_a=F_bLTB} \\
IR_v \\
-IR_v \\
-\frac{f_b}{F_a}
\end{bmatrix} \tag{5.34}
\]

This is a minimum set of mutually exclusive constraints that need to be applied in order to impose all of the code allowable stress criteria. The allowable stress functions used in calculation of the \( UC \) refer to the “smooth” alternative functions detailed in Section 5.1. These are \( F_t \) (equation 5.1), \( F_a \) (equation 5.2), \( F_bLTB \) (equation 5.8), \( F_bFLB \) (equation 5.15), and \( F_v \) (equations 5.20 and 5.22).

Note that the absolute value of the axial stress is not used because depending on its value, the generalized allowable compression (\( F_a \)) and tension (\( F_t \)) stresses take on the appropriate sign. Checking these constraints will accommodate all of the
code requirements but will add some inactive (redundant) constraints. Among the ± combinations, half will result in a smaller value and will be inactive. When $F_{b\text{FLB}} < F_{b\text{LTB}}$, all 4 constraints involving FLB are inactive (similarly when $F_{b\text{LTB}} < F_{\text{FLB}}$). When the axial stress is tensile, the constraints involving $F_t$ and $F_a$ give the same value, so one set is redundant. When $\lambda_a > 0$, then $F_a < F_t$ and the first set of constraints are inactive\(^2\). Finally, for tension, the last constraint is always inactive.

### 5.4 Comparison Between Proposal and Code Constraints

The study discussed in Section 3.6 used nonsmooth allowable stresses from the AISC ASD code [3]. This approach was shown to create difficulties in solving certain problems when the constraint near the solution was discontinuous, or just nonsmooth. This section presents the results of solving a modified problem, using the interior point method. The allowable stress modifications detailed in this section are put to use, namely:

- The absolute values of stresses are not used, instead applying twice as many constraints to check the ± combinations of the bending stress.

- The allowable bending stress for lateral-torsional buckling is based on the nearly smooth equation 5.8.

Once these changes are in place, the only nonsmoothness is due to a discontinuous first derivative of $C_{LTB}$ at the elastic buckling transition. Recall from Section 3.6 that the cross-section is always compact so that the allowable bending stress for flange local buckling is $0.66F_y$ and does not control ($F_{b\text{FLB}} \geq F_{b\text{LTB}}$). Also note that the

\(^2\)This is because only gross section properties are considered. Allowable tensile strength is in general the smaller of the yield strength of the gross section and the ultimate strength of the effective net section. See also Section 5.1.1.
axial stress for the model test19x is zero, greatly simplifying the constraints. Using Knitro, all the problem solutions were found in 10 function evaluations or less. The optimal solutions of this problem (\( \bar{u}^* \)) were near the solution of the original problem (\( u^* = [d_1^*, d_2^*]^T \)). In most cases, the solution of the modified problem takes advantage of the conservative allowable bending stress in the ASD and results in a lower objective value. Table 5.2 summarizes the results, tabulating the old (nonsmooth) and new (smooth) solution and objective. The normalized distance between the two solutions is shown in the last column and at most varies by 8.5%.

<table>
<thead>
<tr>
<th>ID</th>
<th>Smooth</th>
<th></th>
<th>Nonsmooth</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( f(\bar{u}^*) )</td>
<td>( \bar{u}^* ) (in)</td>
<td>( f(u^*) )</td>
<td>( u^* ) (in)</td>
<td>( |\bar{u}^* - u^<em>| / |u^</em>| )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>c</td>
<td>2.675</td>
<td>29.58</td>
<td>26.83</td>
<td>2.675</td>
<td>29.58</td>
<td>26.83</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>d1</td>
<td>1.954</td>
<td>29.74</td>
<td>25.13</td>
<td>1.954</td>
<td>29.74</td>
<td>25.13</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>d1a</td>
<td>1.364</td>
<td>24.85</td>
<td>21.00</td>
<td>1.365</td>
<td>24.84</td>
<td>21.06</td>
<td>0.002</td>
<td></td>
<td></td>
</tr>
<tr>
<td>d2</td>
<td>1.986</td>
<td>30.03</td>
<td>25.01</td>
<td>2.075</td>
<td>30.82</td>
<td>24.64</td>
<td>0.022</td>
<td></td>
<td></td>
</tr>
<tr>
<td>d2a</td>
<td>1.520</td>
<td>26.32</td>
<td>21.48</td>
<td>1.553</td>
<td>26.66</td>
<td>21.32</td>
<td>0.011</td>
<td></td>
<td></td>
</tr>
<tr>
<td>d2b</td>
<td>3.758</td>
<td>41.24</td>
<td>34.85</td>
<td>3.894</td>
<td>42.12</td>
<td>34.47</td>
<td>0.018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>d3</td>
<td>2.143</td>
<td>31.41</td>
<td>24.32</td>
<td>2.169</td>
<td>31.64</td>
<td>24.19</td>
<td>0.006</td>
<td></td>
<td></td>
</tr>
<tr>
<td>d4</td>
<td>2.827</td>
<td>36.74</td>
<td>22.10</td>
<td>2.920</td>
<td>37.57</td>
<td>20.06</td>
<td>0.052</td>
<td></td>
<td></td>
</tr>
<tr>
<td>d4a</td>
<td>0.609</td>
<td>17.08</td>
<td>10.00</td>
<td>0.695</td>
<td>18.16</td>
<td>11.47</td>
<td>0.085</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 5.2: Test model 19 solution using smooth and nonsmooth formulation.

5.5 Summary

In this chapter, smooth, alternative allowable stress functions and interaction ratios have been presented. The proposed allowable stresses are based on theory and analysis presented in Chapter 4. The details of the proposed allowable stresses were presented in Section 5.1.3.1 for lateral-torsional buckling, Section 5.1.3.2 for flange local buckling, and Section 5.1.4 for web local buckling. Sample calculations to clarify the use of the proposed allowable stress equations have been given in section 5.2.
The proposed interaction ratios are equivalent to the nonsmooth AISC ASD [3] combined stress checks, but have been reformulated so that they are smooth. Elimination of envelope functions and absolute values resulted in additional (redundant) stress constraint functions. Rearrangement of the interaction ratios into two sets of generalized equations (for yield and buckling failure) provided piecewise defined stress constraints manageable by the Two-Stage algorithm (see Sections 6.1 and 6.2).
Chapter 6

Two-Stage Algorithm

Note: In this chapter the symbol “λ” refers to a generic design parameter (rather than to a Lagrange multiplier). The symbol “Ω” refers to the feasible domain (rather than to a factor of safety).

The Two-Stage algorithm is a proposed method of solving optimization problems that are characterized by nonsmooth constraints, defined piecewise on the domain of a design parameter, λ. Such a problem can be described as a particular case of Problem 2.1, given by:

\[
\begin{align*}
\min_x f(x) \\
\text{s.t. } g(\lambda(x)) &\leq 0 \\
\end{align*}
\]

where: \( g(\lambda(x)) = \begin{cases} 
    g_1(\lambda(x)), & 0 \leq \lambda(x) \leq \lambda_1 \\
    g_2(\lambda(x)), & \lambda_1 < \lambda(x) \leq \lambda_2 \\
    \vdots \\
    g_m(\lambda(x)), & \lambda_{m-1} < \lambda(x) \leq \lambda_m 
\end{cases} \)  

\( g_1 \in C^2[0, \lambda_1], \ g_2 \in C^2(\lambda_1, \lambda_2], \ldots, \ g_m \in C^2(\lambda_{m-1}, \lambda_m] \)

\( 0 \leq \lambda(x) \leq \lambda_m \) is a design parameter
\( \lambda_i, i = 1, \ldots, m - 1 \) are the boundaries between segments of the domain of \( g \)

but:

\[
\begin{align*}
g'_1(\lambda_1) & \neq g'_2(\lambda_1) \\
g'_2(\lambda_2) & \neq g'_3(\lambda_2) \\
& \vdots \\
g'_{m-1}(\lambda_{m-1}) & \neq g'_m(\lambda_{m-1})
\end{align*}
\]

Since the nonsmoothness exists only at the boundaries of the piecewise segmented domain, this algorithm seeks to limit the use of these constraints to a single segment wherein the constraints are smooth. For example, if \( x^* \) is a local minimizer of Problem 6.1, and \( \lambda_j \leq \lambda(x^*) \leq \lambda_{j+1} \) for some \( j \in [0, m - 1] \), then the problem can be easily restated as a reduced problem, using a secondary constraint on \( \lambda \):

\[
\min_x f(x) \\
\text{s.t. } g(\lambda(x)) \leq 0, \\
\lambda_j \leq \lambda(x) \leq \lambda_{j+1}
\]

which is a smooth problem. Since the solution of Problem 6.1 is difficult, the first stage of the algorithm will seek to find a solution, \( \tilde{x}^* \), of a nearby smooth problem instead. The first stage is a continuation method where the smoothness gradually decreases as intermediate solutions are obtained, until a sufficiently accurate solution to a relatively smooth approximation of the original problem is found. The solution from each iteration is used as the starting value for the next. A parameter \( (\alpha_s) \) that controls the amount of smoothness in the optimization problem is varied to decrease the smoothness as the iterations progress. The required tolerance on feasibility and optimality is initially relaxed but becomes stricter as the iterations progress. This reduces the work done in the earlier part of the solution, in which the problems are
easier to solve but give less accurate results. A specification of the algorithm is given in Figure 6.1.

Stage I Algorithm

**Initialization:** Set $(\alpha_s)_0 = 0$. Select values for $x_0$, $\eta_0 > 0$, $\omega_0 > 0$, $\tau_\eta \in (0,1)$, $\tau_\omega \in (0,1)$, $\alpha^*_s \in ((\alpha_s)_0, 1)$, $\delta^* > 0$, $\eta^* > 0$, $\varphi^* > 0$, $\omega^* > 0$

For $k = 0, 1, 2, \ldots$, do the following:

1. **Find** $x_k^* = \arg \min_{x \in \Omega} f(x)$
   with starting point $x_k$, smoothness according to $(\alpha_s)_k$, and within feasibility tolerance $\eta_k$ and optimality tolerance $\omega_k$.

2. **Set** $x_{k+1} = x_k^*$,
   
   $(\alpha_s)_{k+1} = \frac{1}{2}((\alpha_s)_k + 1)$,
   
   $\eta_{k+1} = \tau_\eta \eta_k \geq \eta^*$,
   
   $\omega_{k+1} = \tau_\omega \omega_k \geq \omega^*$.

3. If $\frac{\|x_{k+1} - x_k\|}{\|x_{k+1}\|} \leq \delta^*$, or $\frac{|f(x_{k+1}) - f(x_k)|}{|f(x_{k+1})|} \leq \varphi^*$, or $(\alpha_s)_{k+1} \geq \alpha_s^*$, STOP.

Figure 6.1: Stage I Algorithm for the approximate solution of Problem 6.1.

The feasible domain, $\Omega$, is the set of optimization variables that are feasible with respect to the problem constraints. For example in the SAND formulation of structural optimization Problem 3.2:

$$\Omega = \{x : g(x) \leq 0, \ x^L \leq x \leq x^U\}$$

which includes the nonlinear stress and geometric inequality constraints $g$, and the box constraints on $x$. The feasibility tolerance, $\eta$, generally refers to some measure of the violation of the constraints at the solution. The optimality tolerance, $\omega$, generally
refers to some measure of the rate of change of the Lagrangian function, $\mathcal{L}$, at the solution. Section 3.5 gives specific feasibility and optimality criteria used in each of the four optimization methods found in this study.

The following assumption is made to motivate stage II:

**Assumption 6.0.1.** If $x^*$ is a local minimizer of a given optimization problem written in the form of Problem 6.1, and $\tilde{x}^*$ is a solution of a reasonably close approximation to the given problem, and if for some $j \in [0, m - 1]$, $\lambda_j \leq \lambda(\tilde{x}^*) \leq \lambda_{j+1}$, then $\lambda_j \leq \lambda(x^*) \leq \lambda_{j+1}$.

Here the term “reasonably close approximation” refers to the solution provided by the stage I algorithm. If Assumption 6.0.1 is true, the approximate solution from stage I can be successfully used to formulate the reduced problem as described above.

Stage II consists of the process of identification of the relevant limits on the design parameters and formulation of the reduced feasible domain, $\hat{\Omega}$. See Figure 6.2 for the specification of the stage II algorithm.

No optimization takes place in stage II; it is simply a bookkeeping procedure. For example, in structural optimization Problems 3.1 or 3.2, once $f_a$ (or a reasonably close approximation) is known for a specific member cross-section, the generalized allowable stress equations 5.24, 5.25, and 5.26 are limited to a single axial stress range (tension, small compression, or large compression). As in Problem 6.3, these equations then become smooth as they are limited in scope at a particular member cross-section. All of the other nonsmooth components of the constraints are similarly handled, making the resulting problem smooth.

Finally, when both stages have been completed, the solution of Problem 6.3 is calculated, using some appropriate method for smooth problems (e.g., sequential quadratic programming, or the interior point method). Thus, the Two-Stage algorithm (see Figure 6.3) consists of first executing stage I, then stage II, and finally solving the original problem with the reduced feasible domain from stage II.

Sections 6.1 and 6.2 give the implementation details that allow Problems 3.1 and
Stage II Algorithm

Initialization: Choose an appropriate approximate solution \( \hat{x}^* \) (e.g. using the Stage I algorithm). Define the set:

\[ \mathcal{I} = \{ i : g_i \text{ is of a form such as equation 6.2} \} \]

1. For all \( i \in \mathcal{I} \), do the following:
   
   (a) **Find** \( j \) such that \( \lambda^i_j \leq \lambda^i(\hat{x}^*) \leq \lambda^i_{j+1} \)
   
   (b) **Let** \( (\lambda^i)^L = \lambda^i_j \), and \( (\lambda^i)^U = \lambda^i_{j+1} \)

2. **Define** \( \hat{\Omega} = \{ x : x \in \Omega, (\lambda^i)^L \leq \lambda^i(x) \leq (\lambda^i)^U, \forall i \in \mathcal{I} \} \subset \Omega \).

Figure 6.2: Stage II Algorithm for the formulation of Problem 6.3.

Two-Stage Algorithm

1. **Calculate** an approximate solution, \( \hat{x}^* \), using the Stage I algorithm.

2. **Define** the secondary constraints on design parameters using the Stage II algorithm.

3. **Set** \( \alpha_s = 1 \).

4. **Solve** the reduced problem,

\[ \min_{x \in \hat{\Omega}} f(x) \]

Figure 6.3: Two-Stage Algorithm for the solution of Problem 6.1.

3.2 to use the Two-Stage algorithm. In Section 6.3, a two-dimensional prototype optimization problem is presented and solved first graphically, then using the Two-Stage algorithm proposed for nonsmooth problems. Finally in Section 6.4, the structural
optimization test set problems (presented in Section 3.4) are solved using the Two-Stage algorithm. The test set is included to validate the algorithm’s robustness and show that Assumption 6.0.1 holds in a variety of cases. Once the test set has been solved, the results are compared to a direct search method to evaluate the efficiency of the Two-Stage algorithm.
6.1 Implementation of Stage I

The generalized allowable stress equations (5.25, 5.24, and 5.26) are discontinuous due to transitions in $f_a$ (at $-0.15F_a$ and zero), where their values change abruptly. In particular:

- $\overline{F}_t$ is discontinuous at $f_a = -0.15F_a$ when $Q_s < 1$.
- $\overline{F}_t$ is always discontinuous at $f_a = 0$.
- $\overline{F}_a$ is always discontinuous at $f_a = 0$.
- $\overline{F}_b$ is discontinuous at $f_a = -0.15F_a$ when $\lambda_{1B_1} > -0.15$.

Inserting smoothing functions (see Appendix B) into equations 5.27, 5.30, and 5.31 at the transition points will alleviate these difficulties. The degree of nonsmoothness is controlled by a parameter $\alpha_s$. When $\alpha_s = 1$, all of the equations are used as presented in Section 5.1, but when $\alpha_s < 1$, transition functions are inserted locally at nonsmooth regions. A minimum value of $\alpha_s = 0$ provides the most smoothness. Figures 6.4, 6.5, and 6.6 show the generalized allowable stress functions for certain values of the design variables with $\alpha_s = 0$. The smooth and actual interaction ratios are plotted below the allowable stress in each figure. Figure 6.7 shows the smooth $B_1$ curve as well as the nonsmooth ASD curve. Note that there are both conservative and unconservative results due to the use of the smoothing functions.

The discontinuities in $\overline{F}_t$ at the transition points of 0 and $-0.15f_a/F_a$ are shown in Figure 6.4. The corresponding regions of nonsmoothness in the generalized interaction ratio $IR_t$ are due to the need for the absolute value of the axial stress, and the sudden drop in allowable stress when $Q_s$ is applied as a reduction factor. The proposed smooth curve approximates the interaction ratio but sacrifices accuracy for smoothness. In this case, the approximate curve is unconservative by at most 10%.

The discontinuity in $\overline{F}_a$ at the zero transition point is shown in Figure 6.5. The corresponding region of nonsmoothness in the generalized interaction ratio $IR_a$ is
Figure 6.4: Proposed, smooth tensile interaction ratio ($IR_t$) curve.

Figure 6.5: Proposed, smooth compressive interaction ratio ($IR_c$) curve.
essentially due to the need for the absolute value of the axial stress, \( f_a \). The proposed smooth curve approximates the interaction ratio but sacrifices accuracy for smoothness. In this case the approximate curve is conservative by nearly 40%.

![Figure 6.6: Proposed, smooth bending interaction ratio (\( IR_b \)) curve.](image)

The discontinuity in \( \overline{F}_b \) at the transition point \(-0.15 f_a / F_a\) is shown in Figure 6.6. The corresponding region of nonsmoothness in the second-order moment magnifier \( B_1 \) and the generalized interaction ratio \( IR_b \) is due to the sudden drop in allowable stress when \( B_1 \) is applied. The proposed smooth curve approximates the interaction ratio but sacrifices accuracy for smoothness. In this case the approximate curve is unconservative by only about 5%.

Following is a summary of the smoothing transition functions inserted at points of discontinuity. The transition functions for tension are defined by:

\[
\lambda_{t1} < \lambda < \lambda_{t0} \Rightarrow IR_t = s_{t1}(\lambda) \\
0 < \lambda < \lambda_{t2} \Rightarrow IR_t = s_{t2}(\lambda)
\]
where: \( \lambda = \frac{f_a}{F_a} \)

\[
\begin{align*}
\lambda_{t1} &= (\alpha_s - 1) - 0.15\alpha_s \in [-1, -0.15] \\
\lambda_{t0} &= -0.15\alpha_s \in [-0.15, 0] \\
\lambda_{t2} &= 1 - \alpha_s \in [0, 1] \\
s_{t1} &\text{ is a polynomial smoothing function at } \lambda = -0.15F_a \\
s_{t2} &\text{ is a polynomial smoothing function at } \lambda = 0
\end{align*}
\]

The transition function for compression is defined by:

\[
\lambda_{a1} < \lambda < \lambda_{a2} \Rightarrow IR_a = s_a(\lambda)
\]

where: \( \lambda = \frac{f_a}{F_a} \)

\[
\begin{align*}
\lambda_{a1} &= \alpha_s - 1 \in [-1, 0] \\
\lambda_{a2} &= 1 - \alpha_s \in [0, 1] \\
s_a &\text{ is a polynomial smoothing function at } \lambda = 0
\end{align*}
\]
The transition function for bending \((B_1)\) is defined by:

\[ \lambda_{b1} < \lambda < \lambda_{1B1} \Rightarrow B_1 = s_b(\lambda) \]

where: \( \lambda = \frac{f_a}{F_a} \)

\[ \lambda_{b1} = (\alpha_s - 1) + \alpha_s \lambda_{1B1}, \quad \lambda_{1B1} \in [-1, \lambda_{1B1}] \]

\(s_b\) is a polynomial smoothing function at \(\lambda = \lambda_{1B1}\)

Note when \(\alpha_s = 1\) the intervals vanish, so that the interaction ratios revert to the proposed nonsmooth values.

Figures 6.8(a), 6.8(b), and 6.8(c) show the proposed interaction ratios for an intermediate value of smoothness \((\alpha_s = 0.5)\). Recall that the interaction ratios shown in Figures 6.4, 6.5, and 6.6 are of maximum smoothness \((\alpha_s = 0)\) and compare the second set of figures: in all cases, the accuracy of the proposed, approximate curve is increased as its smoothness is decreased. When \(\alpha_s = 1\) the proposed interaction ratios are equal to the code [3] interaction ratios but are of maximum nonsmoothness.
Figure 6.8: Interaction ratios of intermediate smoothness.
The final matter involves nonsmoothness at some transitions of miscellaneous functions. The expressions given in Section 5.1 for $C_{LTB}$ (thus $F_{hLTB}$), $C_v$ (thus $F_v$), and $k_v$ are all continuous but nonsmooth functions. Smoothing functions (see Appendix B) will be inserted locally to address the nonsmoothness of these curves:

- $C_{LTB}$ at $\lambda_{LTB} = \lambda_u$,
- $C_v$ at $\lambda_v = 1.37$, and
- $k_v$ at $a/h = 1$.

The two stress ratio curves, $C_{LTB}$ and $C_v$, behave similarly, requiring a single curve to bridge the transition from inelastic buckling to elastic buckling. Each of these is smoothed by inserting a cubic polynomial locally at the value of the slenderness parameter associated with the proportional limit. The web buckling parameter curve, $k_v$, requires a curve to bridge the transition between short rectangular and long rectangular plates. The transition point of $a/h = 1$ for square plates has a known function value of $k_v = 9.34$, which is used as the upper bound for the transition curve. Figure 6.9 shows the smooth $k_v$ curve along with nonsmooth the ASD curve. Table 6.1 gives the upper and lower bounds of the transition intervals for each of the miscellaneous curves, and the range of the lower bound as $\alpha_s$ varies from 0 to 1. The generic term $\lambda$ in the table refers to the independent variable of the particular function. Superscripts $L$ and $U$ in the table refer to the lower and upper bounds on the transition region ($\lambda^L \leq \lambda^U$).

The choice of a transition interval is somewhat arbitrary, but it is important that the functions do not change too abruptly. When $\alpha_s = 0$ the transition interval should be relatively large, but when $\alpha_s = 1$ the interval must vanish. The size of the interval when $\alpha_s = 0$ was chosen to utilize half of the available range of the independent variable for transition curves controlled by geometric parameters only (i.e., dependent only on $u$). For the curves that depend on $u$ and $y$, the interval
in which $\alpha_s = 0$ more or less consists of the full available range of the independent variable.

<table>
<thead>
<tr>
<th>Function</th>
<th>$\lambda$</th>
<th>$\lambda^L(\alpha_s)$</th>
<th>$\lambda^U$</th>
<th>$\lambda^L(0)$</th>
<th>$\lambda^L(1)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{LTB}$</td>
<td>$\lambda_{LTB}$</td>
<td>$\frac{1}{2}[(1 - \alpha_s)\lambda_c + (1 + \alpha_s)\lambda_u]$</td>
<td>$\lambda_u$</td>
<td>$\frac{1}{2}(\lambda_c + \lambda_u)$</td>
<td>$\lambda_u$</td>
</tr>
<tr>
<td>$C_v$</td>
<td>$\lambda_v$</td>
<td>$\frac{1}{2}[(1 - \alpha_s) + (1 + \alpha_s)1.37]$</td>
<td>1.37</td>
<td>1.185</td>
<td>1.37</td>
</tr>
<tr>
<td>$k_v$</td>
<td>$\frac{a}{h}$</td>
<td>$\frac{1}{2}(1 + \alpha_s)$</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 6.1: Interval data for miscellaneous transition functions.
6.2 Implementation of Stage II

The second stage reduced feasible domain, $\hat{\Omega}$, is determined based on the values of the governing parameters, $f_a/F_a$, $\lambda_a$, $\lambda_{LTB}$, $\lambda_{FLB}$, $\lambda_v$, $h/t_w$, and $a/h$ from stage I. These parameters are used to pose secondary constraints determining the range of each non-smooth equation. Figures 6.10(a), 6.10(b), and 6.10(c) show the dependence of each quantity on its design parameter(s). In those figures, $(\lambda)^U$ and $(\lambda)^L$ are the indicated upper and lower bounds on each of the governing parameters. The dependency is built up from the parameters $\lambda_{LTB}$, $\lambda_{LTB}$, $h/t_w$, $\lambda_v$, and $a/h$ so that miscellaneous functions can be computed (Figure 6.10(a)). From these and the parameter $\lambda_a$, the allowable stresses can be computed (Figure 6.10(b)). Finally, from these and the parameter $f_a/F_a$, the interaction ratios can be computed (Figure 6.10(c)).

The secondary constraints supersede the original limit constraints (see Figure 3.4), which are redefined as:

$$g_{lim}(u, y) = \begin{bmatrix}
(\lambda_a)^L - \lambda_v(u) \\
\lambda_a(u) - (\lambda_a)^U \\
(\lambda_{LTB})^L - \lambda_{LTB}(u) \\
\lambda_{LTB}(u) - (\lambda_{LTB})^U \\
(\lambda_{FLB})^L - \lambda_{FLB}(u) \\
\lambda_{FLB}(u) - (\lambda_{FLB})^U \\
\frac{(\lambda_{h/t_w})^L - \frac{h(u)}{t_w}}{\frac{h(u)}{t_w}} - \frac{(\lambda_{h/t_w})^U}{(\lambda_{h/t_w})^U} \\
\frac{(\lambda_a)^L - \lambda_v(u)}{\frac{a}{h(u)}} - \frac{(\lambda_a)^U}{(\lambda_a)^U} \\
\frac{(\lambda_{a/h})^L - \frac{a}{h(u)}}{\frac{a}{h(u)}} - \frac{(\lambda_{a/h})^U}{(\lambda_{a/h})^U} \\
\frac{(\lambda_{FLB})^L - \frac{F_{a(u)}}{F_a(u)}}{\frac{F_{a(u)}}{F_a(u)}} - \frac{(\lambda_{FLB})^U}{(\lambda_{FLB})^U}
\end{bmatrix}$$

The last two constraints supersede the axial stress check in the unity check equations 5.34 (i.e., the equation can be thought of as providing the constraint $-1 \leq f_a/F_a \leq \infty$). Table 6.2 summarizes the possible bounds for the design parameters used to determine allowable stresses. Table 6.3 summarizes the possible bounds for the design
parameters used to determine plate buckling coefficients. Intermediate bounds are labeled according to their physical meaning, if applicable. Maximum values are based on user limits, with the exception of $h/t_w$. This is limited so that the web is not slender, guaranteeing that the reduction factor for stiffened elements, $Q_{a1}$, is 1. The criteria is based on the ASD [3] non-compact section limit (given in Appendix A as $760/\sqrt{F_b}$, and assuming $F_b \leq 0.6F_y$). With this web slenderness limit, and using the minimum value of $k_v = 5.34$, the upper bound on the web local buckling parameter
$\lambda_v$ is 2.5.

<table>
<thead>
<tr>
<th>$\lambda$</th>
<th>Minimum</th>
<th>Yield</th>
<th>Elastic</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\lambda_a$</td>
<td>0</td>
<td>n/a</td>
<td>1</td>
<td>$(KL_e)^U \frac{1}{E}$</td>
</tr>
<tr>
<td>$\lambda_{LTB}$</td>
<td>0</td>
<td>$\lambda_c$</td>
<td>$\lambda_u$</td>
<td>$(L_e/r_p)^U$</td>
</tr>
<tr>
<td>$\lambda_{FLB}$</td>
<td>0</td>
<td>0.44, 0.56</td>
<td>1.34</td>
<td>$(b_t/2t_f)^U \sqrt{F_y/k_sE}$</td>
</tr>
<tr>
<td>$\lambda_v$</td>
<td>0</td>
<td>1</td>
<td>1.37</td>
<td>2.5</td>
</tr>
</tbody>
</table>

Table 6.2: Design parameter bounds for allowable stresses.

<table>
<thead>
<tr>
<th>$\lambda$</th>
<th>Minimum</th>
<th>Transition</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>$h/w$</td>
<td>0</td>
<td>25</td>
<td>$980/\sqrt{F_y}$</td>
</tr>
<tr>
<td>$a/h$</td>
<td>0</td>
<td>1</td>
<td>$\infty$</td>
</tr>
</tbody>
</table>

Table 6.3: Design parameter bounds for plate buckling coefficients.

Once these parameters are limited to a certain range, for feasible designs only one of the failure modes in each allowable stress equation will be possible for a member group. Similarly, depending on the axial stress, only one of the equations for each of the generalized allowable stresses will be possible for a member. The upper and lower bounds for $f_a/F_a$ in the context of these equations are $-1$ and $F_t/F_a$. The transition points are:

$-0.15$ and $0$ for $F_t$

$0$ for $F_a$

$-0.15$ for $F_b$

$\lambda_{1B_1}$ (from equation 5.28) for $B_1$

Any of these may be a bound on the ratio $f_a/F_a$ in stage II. The two most restrictive bounds will determine the secondary constraints for the member. For example, say
that for a particular member, the following quantities are determined at the approximate solution from stage I:

\[
\begin{align*}
\lambda_a &= 1.2 \\
\lambda_{LTB} &= 100, \quad \lambda_c = 42, \quad \lambda_u = 200 \\
\lambda_{FLB} &= 0.5, \quad \frac{h}{t_w} = 70 \\
\lambda_v &= 1.1, \quad \frac{a}{h} = 3 \\
f_a &= -0.91, \quad F_a = 9.13, \quad \frac{f_a}{F_a} = -0.1 \\
C_m &= 0.85, \quad F_e' = 10.1, \quad \lambda_{1B1} = -0.17
\end{align*}
\]

so that the bounds are:

\[
1 \leq \lambda_a \leq \left(\frac{KL}{r}\right)^U \frac{1}{C_c} \text{ (elastic)}
\]

42 \leq \lambda_{LTB} \leq 200 \text{ (inelastic)}

0.44 \leq \lambda_{FLB} \leq 0.56 \text{ (inelastic)}

\[
25 \leq \frac{h}{t_w} \leq \frac{980}{\sqrt{F_y}}
\]

1 \leq \lambda_v \leq 1.37 \text{ (inelastic)}

1 \leq \frac{a}{h} \leq \infty

\[
-0.15 \leq \frac{f_a}{F_a} \leq 0 \text{ (small compression)}
\]

Even though \(f_a/F_a\) may require different bounds for the tension, compression, and bending interaction ratios, the values \(-0.15\) and \(0\) are the most restrictive and ensure the axial stress is within the desired range for all three.

One further note is needed here for the ASD [3] specification on lateral-torsional buckling. The goal of the Two-Stage algorithm is to solve the original problem, by first solving a nearby smooth problem. The commentary of the specification allows for the elimination of separate checks for flexural members controlled by St. Venant or warping torsion (see Appendix A), making the allowable stress acceptable from the
point of view of smoothness, assuming the proper secondary constraints are applied. When using the ASD specification in the final step of the Two-Stage algorithm, the allowable stress for lateral-torsional buckling will be taken as:

\[ F_{LTB} = \begin{cases} \frac{2}{3} \frac{F_y (I_{y}/r_{Tequiv})^{2}}{1,350,000C_y}, & \sqrt{\frac{102,000C_y}{F_y}} \leq \frac{I_y}{r_{Tequiv}} \leq \sqrt{\frac{510,000C_y}{F_y}} \\ \frac{170,000C_y}{(I_{y}/r_{Tequiv})^{2}}, & \frac{I_y}{r_{Tequiv}} > \sqrt{\frac{510,000C_y}{F_y}} \end{cases} \]

where: \( r_{Tequiv} = \sqrt{\frac{I_y}{2S_x}} \sqrt{d^2 + \frac{0.156L_b^2J}{I_y}} \)

When referring to lateral-torsional buckling in the ASD code, the slenderness parameter, \( \lambda_{LTB} \), uses the above equivalent radius of gyration, \( r_{Tequiv} \), rather than \( r_y \) as in equation 4.10. The remaining specifications in Appendix A apply for beam slenderness below \( \sqrt{102,000C_y/F_y} \). One further secondary constraint arises to limit the choice of \( L_c \), the allowable unbraced length at the transition between yield and inelastic buckling. Referring to Appendix A, one of:

\[ \frac{76b_f}{\sqrt{f_y}} \leq \frac{20,000 A_f}{F_y \ d} \]

or

\[ \frac{20,000 A_f}{F_y \ d} \leq \frac{76b_f}{\sqrt{f_y}} \]

must hold. As with the other secondary constraints, the choice of which constraint to apply in stage II depends on which constraint holds in the solution from stage I.
6.3 Prototype Example

A prototype example problem has been created to demonstrate the effectiveness of the proposed technique for nonsmooth optimization. The form of the discontinuous constraint is similar to that in the ratio of actual to allowable bending stress when $(Cm - 1)F_c'/F_a > -0.15$. The problem posed is:

\[
\begin{align*}
\min_{x} & \quad x_1x_2 \\
\text{s.t.} & \quad x_2 + \frac{1}{g_a(\lambda(x))x_1^2x_2} \leq 1, \\
& \begin{bmatrix} 1 \\ 0.1 \end{bmatrix} \leq \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \leq \begin{bmatrix} 10 \\ 1 \end{bmatrix}
\end{align*}
\]

where: 

\[
g_a(\lambda(x)) = \begin{cases} 
1, & \lambda(x) \leq 1 \\
0.9, & \lambda(x) > 1 
\end{cases}
\]

\[
\lambda(x) = \frac{x_1}{33x_2}
\]

Figure 6.11(a) shows the design space with the constraint function (solid) overlaid on contours of the objective. The line $\lambda = 1$ is shown (dashed). By inspection, the solution lies near $[3.3, 0.1]$. Zooming into this region (Figure 6.11(b)) the discontinuity is apparent and two local solutions can be identified.

The first local solution is when $\lambda = 1$, and the inequality constraint is active. The second local solution is when $\lambda > 1$, the inequality constraint is active, and the lower bound constraint on $x_2$ is active. Table 6.4 summarizes the local solutions and the algebraic equations used to solve for them. The first local solution is the global minimizer.

The augmented Lagrange pattern search algorithm (with gradient pruning) finds the global minimizer in 148 function evaluations. Attempting to use the SQP method fails with an error message after the iteration limit is reached. The problem is modified
Figure 6.11: Nonsmooth design space of Problem 6.4

<table>
<thead>
<tr>
<th>Solution</th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$\lambda(x)$</th>
<th>$f(x)$</th>
<th>Solved by:</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3.323</td>
<td>0.1007</td>
<td>1.000</td>
<td>0.3346</td>
<td>$(33)^2x_1^4 - (33)^2x_2^2 + 1 = 0$</td>
</tr>
<tr>
<td>2</td>
<td>3.514</td>
<td>0.1</td>
<td>1.065</td>
<td>0.3514</td>
<td>$0.1(0.9)^2x_1^2 - 1 = 0$</td>
</tr>
</tbody>
</table>

Table 6.4: Local solutions of Problem 6.4.

slightly so that:

$$g_\alpha(\lambda(x); \alpha_s) = \begin{cases} 1, & \lambda(x) \leq \lambda_1(\alpha_s) \\ s(\lambda(x)), & \lambda_1(\alpha_s) < \lambda(x) \leq 1 \\ 0.9, & \lambda(x) > 1 \end{cases}$$

where: $\alpha_s \in [0, 1]$ is a smoothness parameter

$$\lambda_1(\alpha_s) = \alpha_s \in [0, 1]$$

$s(\lambda(x))$ is a polynomial smoothing function (see Appendix B),

with $s(\lambda_1) = 1$, $s(1) = 0.9$, $s'(\lambda_1) = s'(1) = 0$

The addition of a smoothing function to straddle the discontinuity makes the problem smooth. The stage I algorithm is applied to the modified problem and produces an approximate solution $x = [3.514, 0.1]^T$, with $\lambda(x) = 1.065$ in 20 function evaluations. Table 6.5 summarizes the stage I results. The algorithm terminates after three it-
erations when \( \alpha_s = 0.875 \) and the step size is nearly zero. The required feasibility (\( \eta \)) and optimality (\( \omega \)) tolerance are small enough so that a very good solution is obtained by the third iteration.

| \( k \) | Function Eval’ns | \( (\alpha_s)_k \) | \( \eta_k \) | \( \omega_k \) | \( \lambda(x_k) \) | \( f(x_k) \) | \( \|x_k-x_{k-1}\| \) | \( \|x_k\| \) | \( |f(x_k)-f(x_{k-1})| \) |
|-----|------------------|-----------------|---------|---------|-------------|--------|----------------|--------|----------------|
| 0   | 1                | 0               | 0.01    | 0.01    | 0.3030      | 0.1    | –              | –      | –              |
| 1   | 14               | 0.5             | 0.001   | 0.001   | 1.061       | 0.3501 | 0.7141         | 0.7143 | 0.0037         |
| 2   | 17               | 0.75            | 0.0001  | 0.0001  | 1.065       | 0.3514 | 0.00037        | 0.0037 | 0.00002        |
| 3   | 20               | 0.875           | 0.0001  | 0.0001  | 1.065       | 0.3514 | 0.00002        | 0.00002 |

Table 6.5: Stage I data for Problem 6.4.

Figures 6.12(a), 6.12(b), 6.13(a), and 6.13(b)) show the design space and auxiliary function \( g_a \) as the smoothness parameter, \( \alpha_s \), varies from 0 to 0.875. Figure 6.12(a) \( (\alpha_s = 0) \) shows a very smooth constraint which poses an easily solvable problem. In contrast, Figure 6.13(b) \( (\alpha_s = 0.875) \) shows a relatively nonsmooth constraint and auxiliary function. The successive problems, although increasingly nonsmooth are easily solved because of the accurate initial guess from the first iteration.

In the second stage, the inequality constraint \( \lambda > 1 \) is added to Problem 6.4, since we know the solution we seek is in this range. Solving the nonsmooth problem again but with a constraint on \( \lambda \), the SQP method finds the solution \( x = [3.514, 0.1]^T \) in 9 function evaluations. A starting value of \( x_0 = [1, 0.1]^T \) was used in all cases. Had this been a different value, the first stage of the Two-Stage algorithm might have found a solution with \( \lambda \leq 1 \). When the second stage adds this inequality constraint to Problem 6.4, the solution \( x = [3.323, 0.1007]^T \) is found in 15 function evaluations. Whether or not the global solution is found is a matter of pure chance. Note that the number of function evaluations for the Two-Stage algorithm is at most 35 compared to 148 for the pattern search algorithm. Since both methods use derivatives, fewer function evaluations indicates the Two-Stage method is more efficient in this case.
Figure 6.12: Design space of Problem 6.4 during stage 1 solution (part 1).
Figure 6.13: Design space of Problem 6.4 during stage I solution (part 2).
6.4 Comparison of Methods on Test Set

This section details the analysis performed to verify the Two-Stage algorithm, then to compare its performance with direct search methods, which are appropriate for nonsmooth problems. The optimization problems in the Two-Stage algorithm are solved using the SQP method (see Sections 2.2.1 and 3.5). Two direct search methods are used: the augmented Lagrange with pattern search, and the filter pattern search (see Sections 2.2.1, 2.2.2, and 3.5). The augmented Lagrange algorithm employs gradient pruning (see Section 2.2.2), making it a first-order method. Gradient pruning in the filter method was considered unreliable using the software available\textsuperscript{1}, making the filter method a zero-order method. The Two-Stage algorithm uses BFGS Hessian approximations, putting it somewhere in between a first- and second-order method. This is the real strength of the Two-Stage algorithm, because the convergence of the subproblems is very fast. The implicit state formulation (NAND) was used in all cases. Feasibility and optimality criteria were uniform for all methods ($\eta = \omega = 10^{-4}$). A limit of 10,000 function evaluations was imposed. The augmented Lagrange iterations were limited to 10. The following stage I algorithm parameters were used in testing (and are recommended in general):

$$\alpha_0^* = 1 - 2^{-5}$$
$$\delta^* = 10^{-4}, \eta^* = 10^{-4}, \varphi^* = 10^{-4}, \omega^* = 10^{-4},$$
$$\eta_0 = \sqrt{\eta^*}, \omega_0 = \sqrt{\omega^*}$$
$$\tau_\eta = 0.1, \tau_\omega = 0.1$$

Initial sizes were based on engineering judgement. In all test cases, standard

\textsuperscript{1}The original paper on gradient pruning [2] refers only to linearly constrained problems, and the extension to nonlinear constraints is difficult. The program NOMADm attempts to use the gradient of the objective in conjunction with the active constraint gradient to find a reasonable descent direction. The software implementation is somewhat slow in practice due to the computation involved.
rolled shapes were chosen to approximately satisfy the constraints. Some, but not all of the initial designs are feasible. Only the sizes for the model bigframe have been optimized to any extent: this was done using a method to redistribute structural weight to minimize the tip nodal displacement [62]. Table 6.6 lists the initial sizes used for each model. The values of the design variables corresponding to these sizes are given in the ASD [3] code, Part 1. Table 6.7 lists the objective function $f(u_0)$ and aggregate constraint violation $h(u_0)$ at the starting value for each test model. None of the methods assumes a feasible starting point.

<table>
<thead>
<tr>
<th>Group</th>
<th>test19d1a</th>
<th>test20a</th>
<th>test21a</th>
<th>test23</th>
<th>bigframe</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>W24X76</td>
<td>W14X26</td>
<td>W14X30</td>
<td>W10X60</td>
<td>W24X146</td>
</tr>
<tr>
<td>2</td>
<td>W24X62</td>
<td>W8X21</td>
<td>W12X16</td>
<td>W12X65</td>
<td>W24X146</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td>W24X68</td>
<td>W12X19</td>
<td>W24X117</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td>W24X55</td>
<td>W12X19</td>
<td>W24X117</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td>W27X84</td>
<td>W12X19</td>
<td>W24X117</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td>W21X44</td>
<td></td>
<td>W24X68</td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>W24X104</td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>W24X104</td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>W24X68</td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>W24X94</td>
</tr>
<tr>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>W24X84</td>
</tr>
<tr>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>W24X62</td>
</tr>
<tr>
<td>13</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>W24X68</td>
</tr>
<tr>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>W24X68</td>
</tr>
<tr>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>W24X62</td>
</tr>
<tr>
<td>16</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>W24X62</td>
</tr>
<tr>
<td>17</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>W24X55</td>
</tr>
<tr>
<td>18</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>W24X55</td>
</tr>
</tbody>
</table>

Table 6.6: Nearest rolled shapes corresponding to initial sizes for test models.

Both pattern search methods were used to solve all of the smaller test cases. It was assumed that the model bigframe with $n_u = 72$ was beyond the scope of the pattern search method. The augmented Lagrange algorithm was much more reliable,
<table>
<thead>
<tr>
<th>ID</th>
<th>$f(u_0)$</th>
<th>$h(u_0)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>test19d1a</td>
<td>2.585</td>
<td>0.000</td>
</tr>
<tr>
<td>test20a</td>
<td>0.922</td>
<td>0.211</td>
</tr>
<tr>
<td>test21a</td>
<td>7.655</td>
<td>2.511</td>
</tr>
<tr>
<td>test23</td>
<td>17.04</td>
<td>0.102</td>
</tr>
<tr>
<td>bigframe</td>
<td>203.8</td>
<td>0.000</td>
</tr>
</tbody>
</table>

Table 6.7: Initial objective and aggregate constraint violation for test models.

and it outperformed the filter method in all cases. The filter method convergence was exceedingly slow, and it used over 10,000 function evaluations in all cases. See Figure 6.14(a) for its performance on the model test20a. In this case after about 2500 function evaluations, the algorithm stalled without ever meeting the stopping criteria. Figure 6.14(b) shows the points in the filter. This plot is instructive, as it shows there exists a broad range of infeasible points near the optimum.
Figure 6.14: NOMAD (GPS filter method) run statistics for model test20a.
Since the objective from the filter method was generally far from the optimum at termination, the results from the Two-Stage algorithm are only compared to the augmented Lagrange pattern search method. The filter method would have been more successful with relaxed stopping criteria, but since the augmented Lagrange algorithm performed so well, this was not explored. The results for both methods are shown in Table 6.8. Cases where the method failed to converge within the function evaluation limit are noted in the table. In one case (model test23), the augmented Lagrange failed to converge in 10 iterations, indicating that the method had stalled, although the solution nearly meets the stopping criteria\(^2\) and is superior to the result of the filter method. This case is also noted in the table. Note that the two methods do not necessarily converge to the same local minimizer.

<table>
<thead>
<tr>
<th>ID</th>
<th>AL (Pattern Search)</th>
<th>Filter (Pattern Search)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ID</td>
<td>f(u*)</td>
</tr>
<tr>
<td></td>
<td>Eval'ns</td>
<td>f(u*)</td>
</tr>
<tr>
<td>test19d1a</td>
<td>438</td>
<td>1.247</td>
</tr>
<tr>
<td>test20a</td>
<td>738</td>
<td>0.664</td>
</tr>
<tr>
<td>test21a</td>
<td>569</td>
<td>6.637</td>
</tr>
<tr>
<td>test23</td>
<td>4134(^1)</td>
<td>9.743</td>
</tr>
</tbody>
</table>

Table 6.8: Pattern Search convergence results for test models.

\(^*\)Failed to converge.

\(^1\)Did not meet required stopping criteria.

The stage I algorithm (see Figure 6.1) was used on each test model to find an approximate solution to the problem posed in structural optimization Problem 3.1 using the allowable stresses given in Chapter 5. In all cases, the algorithm stopped when \(\alpha_s \geq \alpha_s^*\) was met. In the model test21a, the objective increases in the first iteration, but decreases thereafter. Since this test model is initially infeasible, the first iteration primarily serves to achieve feasibility. In all cases, the objective was changed by at most a few percent in the final iteration. In the models test23 and bigframe (see

\(^2\)Optimality is met, and the aggregate constraint violation (feasibility measure) is \(10^{-3}\).
Tables 6.12 and 6.13), the norm of the step decreased with each iteration, possibly indicating convergence to a single solution. In the models test19d1a, test20a, and test21a (see Tables 6.9, 6.10 and 6.11), the norm of the step increases at the third iteration, then decreases to the end. It is possible that in these cases, a nearby local solution was found in the first few iterations, then abandoned before the final iteration. It should be emphasized that the final Two-Stage solution is completely dependent on the local solution found at the end of stage I.
<table>
<thead>
<tr>
<th>$k$</th>
<th>Function Eval'ns</th>
<th>$(\alpha_s)_k$</th>
<th>$\eta_k$</th>
<th>$\omega_k$</th>
<th>$f(u_k)$</th>
<th>$|u_k-u_{k-1}|$</th>
<th>$|f(u_k)-f(u_{k-1})|$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0.0000</td>
<td>0.0100</td>
<td>0.0100</td>
<td>0.922</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>17</td>
<td>0.5000</td>
<td>0.0010</td>
<td>0.0010</td>
<td>0.918</td>
<td>0.3661</td>
<td>0.0048</td>
</tr>
<tr>
<td>2</td>
<td>25</td>
<td>0.7500</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.741</td>
<td>0.0078</td>
<td>0.2384</td>
</tr>
<tr>
<td>3</td>
<td>38</td>
<td>0.8750</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.669</td>
<td>0.0951</td>
<td>0.1073</td>
</tr>
<tr>
<td>4</td>
<td>55</td>
<td>0.9375</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.628</td>
<td>0.0521</td>
<td>0.0655</td>
</tr>
<tr>
<td>5</td>
<td>60</td>
<td>0.9688</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.628</td>
<td>0.0006</td>
<td>0.0008</td>
</tr>
</tbody>
</table>

Table 6.9: Stage I data for model test19d1a.

<table>
<thead>
<tr>
<th>$k$</th>
<th>Function Eval'ns</th>
<th>$(\alpha_s)_k$</th>
<th>$\eta_k$</th>
<th>$\omega_k$</th>
<th>$f(u_k)$</th>
<th>$|u_k-u_{k-1}|$</th>
<th>$|f(u_k)-f(u_{k-1})|$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0.0000</td>
<td>0.0100</td>
<td>0.0100</td>
<td>7.655</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>37</td>
<td>0.5000</td>
<td>0.0010</td>
<td>0.0010</td>
<td>9.717</td>
<td>0.3619</td>
<td>0.2122</td>
</tr>
<tr>
<td>2</td>
<td>65</td>
<td>0.7500</td>
<td>0.0001</td>
<td>0.0001</td>
<td>7.472</td>
<td>0.0666</td>
<td>0.3004</td>
</tr>
<tr>
<td>3</td>
<td>97</td>
<td>0.8750</td>
<td>0.0001</td>
<td>0.0001</td>
<td>6.851</td>
<td>0.0965</td>
<td>0.0907</td>
</tr>
<tr>
<td>4</td>
<td>135</td>
<td>0.9375</td>
<td>0.0001</td>
<td>0.0001</td>
<td>6.590</td>
<td>0.0644</td>
<td>0.0396</td>
</tr>
<tr>
<td>5</td>
<td>146</td>
<td>0.9688</td>
<td>0.0001</td>
<td>0.0001</td>
<td>6.474</td>
<td>0.0035</td>
<td>0.0179</td>
</tr>
</tbody>
</table>

Table 6.10: Stage I data for model test20a.

<table>
<thead>
<tr>
<th>$k$</th>
<th>Function Eval'ns</th>
<th>$(\alpha_s)_k$</th>
<th>$\eta_k$</th>
<th>$\omega_k$</th>
<th>$f(u_k)$</th>
<th>$|u_k-u_{k-1}|$</th>
<th>$|f(u_k)-f(u_{k-1})|$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0.0000</td>
<td>0.0100</td>
<td>0.0100</td>
<td>7.655</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>37</td>
<td>0.5000</td>
<td>0.0010</td>
<td>0.0010</td>
<td>9.717</td>
<td>0.3619</td>
<td>0.2122</td>
</tr>
<tr>
<td>2</td>
<td>65</td>
<td>0.7500</td>
<td>0.0001</td>
<td>0.0001</td>
<td>7.472</td>
<td>0.0666</td>
<td>0.3004</td>
</tr>
<tr>
<td>3</td>
<td>97</td>
<td>0.8750</td>
<td>0.0001</td>
<td>0.0001</td>
<td>6.851</td>
<td>0.0965</td>
<td>0.0907</td>
</tr>
<tr>
<td>4</td>
<td>135</td>
<td>0.9375</td>
<td>0.0001</td>
<td>0.0001</td>
<td>6.590</td>
<td>0.0644</td>
<td>0.0396</td>
</tr>
<tr>
<td>5</td>
<td>146</td>
<td>0.9688</td>
<td>0.0001</td>
<td>0.0001</td>
<td>6.474</td>
<td>0.0035</td>
<td>0.0179</td>
</tr>
</tbody>
</table>

Table 6.11: Stage I data for model test21a.
<table>
<thead>
<tr>
<th>$k$</th>
<th>Function Eval'ns</th>
<th>$(\alpha_s)_k$</th>
<th>$\eta_k$</th>
<th>$\omega_k$</th>
<th>$f(u_k)$</th>
<th>$|u_k-u_{k-1}|$</th>
<th>$|f(u_k)-f(u_{k-1})|$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0.0000</td>
<td>0.0100</td>
<td>0.0100</td>
<td>17.04</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>1</td>
<td>36</td>
<td>0.5000</td>
<td>0.0010</td>
<td>0.0010</td>
<td>9.148</td>
<td>0.3219</td>
<td>0.8631</td>
</tr>
<tr>
<td>2</td>
<td>85</td>
<td>0.7500</td>
<td>0.0001</td>
<td>0.0001</td>
<td>8.806</td>
<td>0.1658</td>
<td>0.0389</td>
</tr>
<tr>
<td>3</td>
<td>148</td>
<td>0.8750</td>
<td>0.0001</td>
<td>0.0001</td>
<td>8.737</td>
<td>0.0751</td>
<td>0.0079</td>
</tr>
<tr>
<td>4</td>
<td>158</td>
<td>0.9375</td>
<td>0.0001</td>
<td>0.0001</td>
<td>8.723</td>
<td>0.0170</td>
<td>0.0016</td>
</tr>
<tr>
<td>5</td>
<td>171</td>
<td>0.9688</td>
<td>0.0001</td>
<td>0.0001</td>
<td>8.718</td>
<td>0.0046</td>
<td>0.0005</td>
</tr>
</tbody>
</table>

Table 6.12: Stage I data for model test23.

<table>
<thead>
<tr>
<th>$k$</th>
<th>Function Eval'ns</th>
<th>$(\alpha_s)_k$</th>
<th>$\eta_k$</th>
<th>$\omega_k$</th>
<th>$f(u_k)$</th>
<th>$|u_k-u_{k-1}|$</th>
<th>$|f(u_k)-f(u_{k-1})|$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0.0000</td>
<td>0.0100</td>
<td>0.0100</td>
<td>203.8</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>1</td>
<td>59</td>
<td>0.5000</td>
<td>0.0010</td>
<td>0.0010</td>
<td>164.1</td>
<td>0.2154</td>
<td>0.2417</td>
</tr>
<tr>
<td>2</td>
<td>160</td>
<td>0.7500</td>
<td>0.0001</td>
<td>0.0001</td>
<td>142.0</td>
<td>0.1418</td>
<td>0.1558</td>
</tr>
<tr>
<td>3</td>
<td>285</td>
<td>0.8750</td>
<td>0.0001</td>
<td>0.0001</td>
<td>136.1</td>
<td>0.0775</td>
<td>0.0437</td>
</tr>
<tr>
<td>4</td>
<td>376</td>
<td>0.9375</td>
<td>0.0001</td>
<td>0.0001</td>
<td>134.0</td>
<td>0.0459</td>
<td>0.0158</td>
</tr>
<tr>
<td>5</td>
<td>450</td>
<td>0.9688</td>
<td>0.0001</td>
<td>0.0001</td>
<td>133.1</td>
<td>0.0207</td>
<td>0.0063</td>
</tr>
</tbody>
</table>

Table 6.13: Stage I data for model bigframe.
The number of function evaluations used in each iteration is fairly consistent. The algorithm is designed to calculate a coarser solution in the earlier iterations, when the problem solved is further from the original problem. This strategy attempts to prevent unnecessary work early on. Once the algorithm begins to converge to a solution, more precision is required. At the same time, the starting value is nearer to the solution of the subproblem for that iteration, requiring relatively fewer steps. In the model bigframe, which takes the most function evaluations, the stage I iterations require between 59 and 125 function evaluations for a total of 450 and an average of 90. The other problems behave similarly.

It is important to note that stage I uses at most 5 major iterations for all of the test models, regardless of the problem size. The Two-Stage algorithm has subproblems which can be counted as the number of major iterations in stage I plus the final major iteration to solve the reduced problem. Since the number of iterations in stage I can be controlled by the value of the stopping parameter, \( \alpha^* \), the number of subproblems is size independent, provided that a sufficiently accurate solution is provided for the second stage.

Information from the stage I solution is to be used for secondary constraints in the stage II algorithm (see Figure 6.2). Summaries of the behavior of the structure in the stage I solution for the model test20a are given in Table 6.14. The upper and lower slenderness bounds are to be applied so that each design group is limited to a certain failure mode. The failure modes are summarized under “notes” and include the full range: yield, inelastic buckling, and elastic buckling. Note that the upper bound for web local buckling is the limit applied for non-compact sections. The members are all in compression with only one column exceeding 15% of the allowable axial stress.

With Assumption 6.0.1, we suppose that the design parameters (\( \lambda \)) in the optimal solution have the same bounds as those in the approximate solution from stage I. When this is true, the solution of the reduced problem with secondary constraints is the same as the solution of the original problem. When the assumption is false, the
<table>
<thead>
<tr>
<th>Group</th>
<th>Criteria</th>
<th>$\lambda^L$</th>
<th>$\lambda$</th>
<th>$\lambda^U$</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>KLR</td>
<td>0.0</td>
<td>0.91</td>
<td>1.0</td>
<td>inelastic</td>
</tr>
<tr>
<td>1</td>
<td>LTB</td>
<td>0.0</td>
<td>0.0</td>
<td>50.0</td>
<td>yield</td>
</tr>
<tr>
<td>1</td>
<td>FLB</td>
<td>0.44</td>
<td>0.46</td>
<td>0.56</td>
<td>inelastic</td>
</tr>
<tr>
<td>1</td>
<td>WLB</td>
<td>1.37</td>
<td>2.10</td>
<td>2.5</td>
<td>elastic</td>
</tr>
<tr>
<td>2</td>
<td>KLR</td>
<td>0.0</td>
<td>0.0</td>
<td>1.0</td>
<td>inelastic</td>
</tr>
<tr>
<td>2</td>
<td>LTB</td>
<td>50.0</td>
<td>65.4</td>
<td>154</td>
<td>inelastic</td>
</tr>
<tr>
<td>2</td>
<td>FLB</td>
<td>0.44</td>
<td>0.48</td>
<td>0.56</td>
<td>inelastic</td>
</tr>
<tr>
<td>2</td>
<td>WLB</td>
<td>1.37</td>
<td>2.08</td>
<td>2.5</td>
<td>elastic</td>
</tr>
</tbody>
</table>

Notes:

- KLR  Member global buckling
- LTB  Lateral-torsional buckling
- FLB  Flange local buckling
- WLB  Web local buckling

<table>
<thead>
<tr>
<th>Member</th>
<th>$\lambda^L$</th>
<th>$\lambda$</th>
<th>$\lambda^U$</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-0.150</td>
<td>-0.100</td>
<td>0.000</td>
<td>small compression</td>
</tr>
<tr>
<td>2</td>
<td>-0.150</td>
<td>-0.095</td>
<td>0.000</td>
<td>small compression</td>
</tr>
<tr>
<td>3</td>
<td>-1.000</td>
<td>-0.180</td>
<td>-0.150</td>
<td>large compression</td>
</tr>
</tbody>
</table>

Table 6.14: Governing slenderness and axial stress parameters for model test20a.

The solution of the reduced problem is feasible, but nonoptimal. This situation is avoided by solving stage I with sufficiently restrictive stopping criteria. In particular, as long as the smoothness parameter, $\alpha_s$, is nearly 1 at the end of stage I, the chance of such a mishap is greatly reduced. In cases where stage I terminates because of other criteria, such as a small step, it is unlikely that the solution will continue to change much in the final solution either.

To validate Assumption 6.0.1, each test model was reviewed to determine whether the premise was violated. The final solution and design parameters from the Two-Stage algorithm were compared to a “free” solution started at the initial point provided from stage I. Stage II secondary constraints were not applied so that the solution algorithm would be free to explore more optimal points, if they existed locally. In the
smaller test cases, all of the design parameters were within the bounds determined from stage I, and the solutions of the free problem proved to be the same as the solution from the Two-Stage algorithm. However, in the model bigframe, the assumption was violated in five instances (see Table 6.15). As expected, the Two-Stage algorithm gave a nonoptimal, but feasible solution (i.e., the objective was higher than the minimum, but only by 0.01%). Note that three of the five parameters in the table were axial stresses of members with little or no axial force. In these cases, the sign of a small axial load changed due to roundoff in the finite element solution of the state equations, once indicating compression and once indicating tension. Table 6.16 summarizes the results of the final step of the Two-Stage optimization and tabulates the difference in the solution from the starting point (i.e., the approximate solution from stage I). The norm of the step in the free solution is identical for the smaller test models, but is 0.0185 (compared to 0.0123 for the Two-Stage solution) for the model bigframe.

<table>
<thead>
<tr>
<th>Group</th>
<th>Criteria</th>
<th>Stage I Bounds</th>
<th>Free Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>$\lambda_L$</td>
<td>$\lambda$</td>
</tr>
<tr>
<td>1</td>
<td>FLB</td>
<td>0.56</td>
<td>0.5602</td>
</tr>
<tr>
<td>5</td>
<td>FLB</td>
<td>0.56</td>
<td>0.5603</td>
</tr>
</tbody>
</table>

Notes:
FLB  Flange local buckling

<table>
<thead>
<tr>
<th>Member</th>
<th>Stage I Bounds</th>
<th>Free Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\lambda_L$</td>
<td>$\lambda$</td>
</tr>
<tr>
<td>19</td>
<td>-0.15</td>
<td>$-10^{-6}$</td>
</tr>
<tr>
<td>106</td>
<td>0</td>
<td>$10^{-4}$</td>
</tr>
<tr>
<td>110</td>
<td>0</td>
<td>$10^{-4}$</td>
</tr>
</tbody>
</table>

Table 6.15: Selected slenderness and axial stress parameters for model bigframe.

As a further assessment of the flexibility of the Two-Stage method, stage I results were also used to attempt to find a local solution to Problem 3.1 with the ASD [3]
Table 6.16: Final step data for the Two-Stage algorithm.

| ID     | Function | $\|u^* - \bar{u}^*\|$ | $\|u^*\|$ | $|f(u^*) - f(\bar{u}^*)|$ | $|f(u^*)|$ |
|--------|----------|------------------------|---------|---------------------|---------|
| test19d1a | 7        | 0.0023                | 0.0169  | 0.00001             | 0.0167  |
| test20a  | 3        | 0.0001                | 0.00001 | 0.00001             | 0.0167  |
| test21a  | 8        | 0.0032                | 0.0001  | 0.00001             | 0.0167  |
| test23   | 3        | 0.0001                | 0.00001 | 0.000001            | 0.0044  |
| bigframe | 35       | 0.0123                | 0.0001  | 0.000001            | 0.0044  |

specifications. In order for this to be successful, there must be a feasible solution to the stage II constrained problem with the ASD stress constraints. This goes beyond Assumption 6.0.1, which states that the stage I solution solves a “reasonably close approximation” to the given problem. In this case, the stage I optimization uses the proposed allowable stresses, so that the solution of the given problem is that posed with similar, but nonequivalent constraints. However, it was found that in all but one case (test model bigframe), a feasible and optimal solution was found in the final step of the Two-Stage optimization using both the proposed and the ASD specifications. By relaxing the optimality and constraint tolerances slightly (to $10^{-3}$), a Two-Stage solution for the model bigframe was found as well using the ASD [3] specifications.

A comparison between the pattern search and Two-Stage methods is given in Table 6.17. In all cases, the method used the ASD specifications for stress constraints. That is, it solved the original problem. Cases where relaxed tolerances were required are noted in the table. Note that the two methods do not necessarily converge to the same local minimizer.

The objective is reduced nearly the same by both methods, so they are both effective from that point of view. The number of function evaluations is 10 times more on average using the pattern search method. Considering that both methods compute first derivatives, the Two-Stage method is clearly more efficient. As long as the stopping criteria can be relaxed, both methods are reliable in all cases. The Two-
<table>
<thead>
<tr>
<th>ID</th>
<th>$n_u$</th>
<th>AL (Pattern Search)</th>
<th>Two-Stage</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Function $f(u^*)$</td>
<td>Function $f(u^*)$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Eval'ns $f(u^*)$</td>
<td>Eval'ns $f(u^*)$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$f(u_0)$</td>
<td>$f(u_0)$</td>
</tr>
<tr>
<td>test19d1a</td>
<td>8</td>
<td>438 1.247 0.482</td>
<td>99 1.264 0.489</td>
</tr>
<tr>
<td>test20a</td>
<td>8</td>
<td>738 0.664 0.720</td>
<td>78 0.637 0.690</td>
</tr>
<tr>
<td>test21a</td>
<td>24</td>
<td>569 6.637 0.867</td>
<td>181 6.501 0.849</td>
</tr>
<tr>
<td>test23</td>
<td>20</td>
<td>3987* 8.769 0.515</td>
<td>192 8.761 0.514</td>
</tr>
<tr>
<td>bigframe</td>
<td>72</td>
<td>–     –</td>
<td>455* 134.3 0.659</td>
</tr>
</tbody>
</table>

Table 6.17: Final comparison of results of two methods for test models.

*Did not meet required stopping criteria.

Stage algorithm was able to solve all of the problems with the proposed specifications using the original optimality and feasibility tolerances ($10^{-4}$).

The solution convergence for the small test models is plotted in Figure 6.15. These figures show the smaller model data from Table 6.17 graphically. In all cases, the Two-Stage algorithm obtains the minimum objective considerably sooner than the pattern search method. In many cases, the pattern search method nearly obtains the minimum objective early on, but needs many more function evaluations to meet the required optimality criteria.
Figure 6.15: Convergence history using Pattern Search and Two-Stage methods.
6.5 Summary

The type of nonsmooth optimization problem intended for the Two-Stage algorithm has been presented, and the algorithm has been specified. Implementation details for Problems 3.1 and 3.2 for use in the Two-Stage algorithm have been given in Sections 6.1 and 6.2. Finally, comparisons for efficiency and overall effectiveness of the Two-Stage algorithm with direct search methods were made. Solution convergence using the Two-Stage algorithm is shown for the model bigframe in Figure 6.16. This figure suggests a linear convergence rate.

![Graph showing convergence history](image)

Figure 6.16: Convergence history of model bigframe.

The actual convergence rates at each major iteration of the Two-Stage algorithm are given in Table 6.18. In this table, $u^*$ is taken as the final Two-Stage solution using the proposed specifications. The convergence for all cases appears to be linear, although there are not a sufficient number of terms to judge the asymptotic behavior.

As a final note, the SQP method was applied to all of the models directly (without any smoothness considerations). The results were mixed: the model test20a was
<table>
<thead>
<tr>
<th></th>
<th>test19d1a</th>
<th></th>
<th>test20a</th>
<th></th>
<th>test21a</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$|u_k-u^*|$</td>
<td>$|u_{k-1}-u^*|$</td>
<td>$|u_k-u^*|$</td>
<td>$|u_{k-1}-u^*|$</td>
<td>$|u_k-u^*|$</td>
<td>$|u_{k-1}-u^*|$</td>
</tr>
<tr>
<td>$k$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>8.97</td>
<td>9.82</td>
<td>13.0</td>
<td>16.7</td>
<td>0.694</td>
<td>0.807</td>
</tr>
<tr>
<td>1</td>
<td>8.75</td>
<td>0.975</td>
<td>3.12</td>
<td>0.318</td>
<td>10.5</td>
<td>0.439</td>
</tr>
<tr>
<td>2</td>
<td>8.54</td>
<td>0.976</td>
<td>3.00</td>
<td>0.961</td>
<td>10.5</td>
<td>0.439</td>
</tr>
<tr>
<td>3</td>
<td>0.50</td>
<td>0.058</td>
<td>1.36</td>
<td>0.452</td>
<td>4.60</td>
<td>0.439</td>
</tr>
<tr>
<td>4</td>
<td>0.19</td>
<td>0.391</td>
<td>0.02</td>
<td>0.013</td>
<td>0.48</td>
<td>0.104</td>
</tr>
<tr>
<td>5</td>
<td>0.09</td>
<td>0.488</td>
<td>0.00</td>
<td>0.160</td>
<td>0.23</td>
<td>0.482</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>test23</th>
<th></th>
<th>bigframe</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$|u_k-u^*|$</td>
<td>$|u_{k-1}-u^*|$</td>
<td>$|u_k-u^*|$</td>
<td>$|u_{k-1}-u^*|$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$k$</td>
<td></td>
<td></td>
<td>$k$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>16.0</td>
<td></td>
<td>28.9</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>10.1</td>
<td>0.634</td>
<td>23.3</td>
<td>0.805</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>3.61</td>
<td>0.356</td>
<td>15.2</td>
<td>0.655</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0.83</td>
<td>0.228</td>
<td>8.90</td>
<td>0.584</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>0.19</td>
<td>0.226</td>
<td>3.66</td>
<td>0.411</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>0.00</td>
<td>0.014</td>
<td>1.51</td>
<td>0.412</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 6.18: Convergence rates for Two-Stage algorithm

solved easily and more quickly than with the Two-Stage algorithm. Of the remaining problems, the model test23 was solved in 342 function evaluations (more than with the Two-Stage algorithm), while the models test19d1a, test21a, and bigframe all stalled during solution and were terminated after 2000 function evaluations.
Chapter 7

Application to Tug Load Minimization

In this chapter, a second optimization problem is posed and solved using the concept of smoothed constraints (see also [81]). In this case the tug load required to stabilize an oil tanker while connected to a moored Floating Production Storage and Offloading (FPSO) vessel is minimized. The equilibrium constraints in this problem have environmental loads that are defined piecewise linearly as a function of the relative heading angle of the vessel. This is a nonsmooth problem that can be reformulated as a smooth one and is, thus, easily solved.

A schematic configuration of the offloading procedure for a tanker and a moored FPSO is shown in Figure 7.1. Compass north is used as a baseline direction for headings. The FPSO and tanker are assumed rigidly linked at a single point on each vessel. Similarly, the tug is connected to the tanker at a single point. The FPSO mooring provides sufficient support so that there are no large horizontal or rotational displacements. A Hawser that is considered axially and flexurally rigid\(^1\), but provides no restraint against rotation at its ends, connects the tanker to the FPSO. The tanker is able to maintain a certain heading and may reduce the force in the Hawser through

\(^1\)The Hawser and tug line are assumed to be in tension only.
the exerted tug load. The tanker is acted upon by external environmental loads (see Section 7.1.1), which tend to push it out of the optimal position. The resistance of the Hawser and the exerted tug load offset the external loads on the tanker. The mooring lines stabilize the FPSO.

![Diagram of offloading procedure](image)

Figure 7.1: Schematic diagram of offloading procedure.

Because of the assumed support conditions of the FPSO it is assumed free to rotate and translate only in the horizontal plane. Because of the rigidity of the Hawser and the action of the tug, the tanker is assumed to translate with the FPSO, but rotate independently as a rigid body about its center of gravity in the plane. The Hawser and tug forces are described by their magnitude and direction. The system is thus described by 8 variables:

1. Tanker heading $\Psi$,
2. Tug load, $T$,
3. Local Tug angle, $\tau$,
4. Hawser force, $H$,
5. Hawser angle, $\eta$, 

6. FPSO yaw, $\Delta \Phi$,

7. FPSO $X$-direction drift, $\Delta X$, and

8. FPSO $Y$-direction drift, $\Delta X$.

**Note:** In this chapter the uppercase symbols "X" and "Y" refer to directions or displacements in the engineering sense. The lowercase symbol "x" refers to the optimization and state variables. The symbol "$\lambda$" refers to a generic parameter (rather than to a Lagrange multiplier).

### 7.1 Problem Formulation

The tug load, $T$, and Hawser force, $H$, are optimization variables, while the remaining 6 unknowns are state variables. The equilibrium equations are nonlinear (see Section 7.1.2) so it is more convenient to use an all-at-once (SAND) formulation with independent optimization and state variables. Furthermore, if a change of variables is made from $(T, \tau)$ and $(H, \eta)$ to $(T_X, T_Y)$ and $(H_X, H_Y)$ (i.e., $X$ and $Y$-components) the equilibrium equations are less nonlinear.

The tug load minimization problem is:

$$\min_x T_X^2 + T_Y^2 \tag{7.1}$$

s.t. $c(x) = 0$

$$T_X \leq 0 \tag{7.2}$$

$$|\tau(x) - 180| \leq \tau_{max} \tag{7.3}$$

$$H_X \geq 0 \tag{7.4}$$

$$H_{min} \leq H(x) \leq H_{max} \tag{7.5}$$

$$|\eta(x)| \leq \eta_{max} \tag{7.6}$$

$$|\eta_{FPSO}(x)| \leq \eta_{max} \tag{7.7}$$
|Ψ - Φ(x) - 180| ≤ ψ_{max}, \quad (7.8)

|ΔΦ| ≤ ϕ_{max}, \quad (7.9)

Δ_{FPSO}(x) ≤ r_{max} \quad (7.10)

where: \( x = \begin{bmatrix} Ψ & T_X & T_Y & H_X & H_Y & ΔΦ & ΔX & ΔY \end{bmatrix}^T \)

c(x) are the equilibrium constraints (see Section 7.1.2)

\( T_X, T_Y \) are components of the tug load, \( T(x) = \sqrt{T_X^2 + T_Y^2} \)

\( τ(x) = \tan^{-1} \left( \frac{T_Y}{T_X} \right) \)

\( H_X, H_Y \) are components of the Hawser force, \( H(x) = \sqrt{H_X^2 + H_Y^2} \)

\( η(x) = \tan^{-1} \left( \frac{H_Y}{H_X} \right), \quad η_{FPSO}(x) = Ψ - 180 - Φ(x) + η(x) \)

\( Φ(x) = Φ_0 + ΔΦ, \quad Φ_0 \) is the mean FPSO heading

\( Δ_{FPSO}(x) = \sqrt{ΔX^2 + ΔY^2} \)

\( H_{min}, H_{max} \) are bounds on the Hawser force

\( τ_{max} < 90 \) and \( η_{max} < 90 \) are bounds on tug and Hawser motion

\( ψ_{max} < 90 \) and \( ϕ_{max} < 90 \) are bounds on tanker and FPSO motion

\( r_{max} \) is the watch circle radius (a bound on FPSO drift)

Equality constraints impose equilibrium (see Section 7.1.2). Inequality constraints impose operational limits on the system. The equilibrium equations, c, are nonsmooth because the environmental loads are nonsmooth (see Section 7.1.1).

### 7.1.1 Environmental Loads

The net environmental loads acting on the tanker (or FPSO) depend on vessel dimensions, seasonal parameters and the position of the tanker (or FPSO) with respect to the prevailing weather conditions. For example, the wind loads are smallest when
the vessel is facing straight into the wind because the windward surface area is at a minimum. The net environmental load is the sum of wind, current, swell, and local (wind-generated) waves. Once the vessel dimensions and seasonal parameters are fixed, the environmental loads are a function only of the relative vessel heading. Environmental loads are described by 3 components: two forces and a moment.

The net environmental loads were experimentally determined at 10° intervals of the relative headings (referred to by the generic parameter, λ). In the X direction:

\[
X_{ENV}(\lambda_1) = X_{ENV_1}
\]
\[
X_{ENV}(\lambda_2) = X_{ENV_2}
\]
\[
\vdots
\]
\[
X_{ENV}(\lambda_{37}) = X_{ENV_{37}}
\]

where: \(\lambda = 0°, 10°, \ldots, 360°\)

Note that the environmental loads are periodic (with period 2\(\pi\)) so that \(X_{ENV_{37}} = X_{ENV_1}\). For headings in between the known values, the loads are approximated using linear interpolation:

\[
X_{ENV}(\lambda(x)) = \begin{cases} 
X_{ENV_1} + \frac{1}{10} (X_{ENV_2} - X_{ENV_1}) \lambda(x), & 0 \leq \lambda(x) \leq 10 \\
X_{ENV_2} + \frac{1}{10} (X_{ENV_3} - X_{ENV_2}) (\lambda(x) - 10), & 10 \leq \lambda(x) \leq 20 \\
\vdots \\
X_{ENV_{36}} + \frac{1}{10} (X_{ENV_{37}} - X_{ENV_{36}}) (\lambda(x) - 350), & 350 \leq \lambda(x) \leq 360 
\end{cases}
\]

(7.11)

where: \(\lambda(x) = \theta_X - \Psi\) for tanker loads

\(\lambda(x) = \theta_X - \Phi(x)\) for FPSO loads

\(\theta_X\) is the heading of the net X-direction environmental load

The Y-direction force and Z-direction moment are similar.
7.1.2 Equilibrium and Feasibility

We study local equilibrium by considering free bodies of the FPSO and tanker. Forces on the FPSO include the environmental loads, Hawser force, and the mooring reactions (see Figure 7.2(a)). Equilibrium of the FPSO is defined by:

\[
\begin{bmatrix}
X_{ENV\,FPSO}(x) + H_{X\,FPSO}(x) \\
Y_{ENV\,FPSO}(x) + H_{Y\,FPSO}(x) \\
M_{ENV\,FPSO}(x) + \frac{1}{2}L_{FPSO}H_{Y\,FPSO}(x)
\end{bmatrix} =
\begin{bmatrix}
R_{Fx}(x) \\
R_{Fy}(x) \\
R_{Mc}(x)
\end{bmatrix}
\]

where: \( L_{FPSO} \) is the FPSO length

\( X_{ENV\,FPSO}, Y_{ENV\,FPSO}, \) and \( M_{ENV\,FPSO} \) are the FPSO environmental loads

\[
H_{X\,FPSO}(x) = H(x) \cos \eta_{FPSO}(x), \quad H_{Y\,FPSO}(x) = H(x) \sin \eta_{FPSO}(x)
\]

\[
\begin{bmatrix}
R_{Fx}(x) \\
R_{Fy}(x) \\
R_{Mc}(x)
\end{bmatrix} = K_{FPSO}
\begin{bmatrix}
\Delta X \\
\Delta Y \\
\Delta \phi
\end{bmatrix}
\]

\( K_{FPSO} \) is the 3 by 3 linearized mooring stiffness

Forces on the tanker include the environmental loads, Hawser force, and tug load (see Figure 7.2(b)). Equilibrium of the tanker is defined by:

\[
\begin{bmatrix}
X_{ENV\,t}(x) + T_{X} + H_{X} \\
Y_{ENV\,t}(x) + T_{Y} + H_{Y} \\
M_{ENV\,t}(x) - \frac{1}{2}L_{t}T_{Y} + \frac{1}{2}L_{t}H_{Y}
\end{bmatrix} =
\begin{bmatrix}
0 \\
0 \\
0
\end{bmatrix}
\]

where: \( L_{t} \) is the tanker length

\( X_{ENV\,t}, Y_{ENV\,t}, \) and \( M_{ENV\,t} \) are the tanker environmental loads

After grouping the FPSO and tanker equilibrium equations into 6 system equations and subtracting the right hand sides, the resulting state equations are obtained:

\[
c(x) =
\begin{bmatrix}
X_{ENV\,t}(x) + T_{X} + H_{X} \\
Y_{ENV\,t}(x) + T_{Y} + H_{Y} \\
M_{ENV\,t}(x) - \frac{1}{2}L_{t}T_{Y} + \frac{1}{2}L_{t}H_{Y} \\
X_{ENV\,FPSO}(x) + H_{X\,FPSO}(x) \\
Y_{ENV\,FPSO}(x) + H_{Y\,FPSO}(x) \\
M_{ENV\,FPSO}(x) + \frac{1}{2}L_{FPSO}H_{Y\,FPSO}(x)
\end{bmatrix} -
\begin{bmatrix}
0 \\
0 \\
0 \\
R_{Fx}(x) \\
R_{Fy}(x) \\
R_{Mc}(x)
\end{bmatrix} \quad (7.12)
\]
7.2 Smooth Formulation

This section discusses application of the concepts developed in Chapters 5 and 6 to Problem 7.1. The sources of nonsmoothness in Problem 7.1 are:

- Nonsmooth environmental loads in equation 7.12.
- Use of absolute values in inequality constraints.

Instead of linear interpolation as in equation 7.11, cubic splines (see [26], Chapter 4), are used to interpolate the environmental load data. The environmental loads are then represented by 36 piecewise cubic polynomials. Each cubic polynomial has 4 coefficients to be determined (see Appendix B) so that there are a total of 144 available conditions. Equality of the function at each of 37 known points is required. For smoothness, the function and its first and second derivatives should be continuous at 35 of the interior known points. These requirements amount to a total of 142
conditions, leaving 2 free conditions. Since the environmental load functions are periodic, the remaining 2 conditions are taken so that first and second derivatives match at $0^\circ$ and $360^\circ$. The function itself is defined so that it is already continuous at these points. Use of splines is as accurate as linear interpolation since the known data points are interpolated exactly. More importantly, the resulting environmental loads are smooth because the splines are twice-continuously differentiable.

Inequality constraints with absolute values (equations 7.3, 7.6, 7.7, 7.8, and 7.9) are each replaced with two equivalent constraints:

\[
-\tau_{max} \leq \tau(x) - 180 \leq \tau_{max} \\
-\eta_{max} \leq \eta(x) \leq \eta_{max} \\
-\eta_{max} \leq \eta_{FPSO}(x) \leq \eta_{max} \\
-\psi_{max} \leq \Psi - \Phi(x) - 180 \leq \psi_{max} \\
-\phi_{max} \leq \Delta \Phi \leq \phi_{max}
\]  

(7.13) (7.14) (7.15) (7.16) (7.17)

A further refinement is made to reduce nonlinearities in the inequality constraints. Constraints involving arctangents and square roots (equations 7.13, 7.14, 7.15, 7.5, and 7.10) are equivalently written as:

\[
T_X \tan (180 - \tau_{max}) \leq T_Y \leq T_X \tan (180 + \tau_{max}) \\
H_X \tan (-\eta_{max}) \leq H_Y \leq H_X \tan (\eta_{max}) \\
H_X \tan (180 - \Psi + \Phi(x) - \eta_{max}) \leq H_Y \leq H_X \tan (180 - \Psi + \Phi(x) + \eta_{max}) \\
H_{min}^2 \leq H_X^2 + H_Y^2 \leq H_{max}^2 \\
\Delta X^2 + \Delta Y^2 \leq r_{max}^2
\]

(7.18) (7.19) (7.20) (7.21) (7.22)

Note that the $X$-direction components of the tug load and Hawser force are moved from the denominator to avoid division by zero. In the main formulation, equations 7.2 and 7.4 are constraints on the $X$-direction components of tug load and Hawser force. These, along with limitations on $\tau_{max}$ and $\eta_{max}$ are imposed to avoid certain numerical difficulties that may arise in equations 7.18 7.19, and 7.20 when either $T_X$
or $H_X$ is small. For example, if $\eta$ is $90^\circ$ then $H_X$ must be zero. In this case, the bounds on $H_Y$ shrink to zero as well. However, if $\eta_{\text{max}} = 85^\circ$, then $H_X$ (similarly $T_X$) will be bounded away from zero and these situations are prevented.

Problem 7.1 is less complex than the structural optimization problem, and it has the appealing quality that the smooth formulation using cubic splines is exact at the known data points. Consequently, it is unnecessary to solve subproblems using a smoothness parameter, $\alpha_s$ (as in Chapter 6). It is also unnecessary to apply secondary constraints and solve a reduced problem. Instead, a simple strategy of smoothing is employed. The next section presents the results of optimizing the resulting smooth problem using an SQP method.

### 7.3 Comparison of Smooth and Nonsmooth Formulations

A test set of mooring configurations was created to compare the solution using the smooth and nonsmooth equilibrium constraints. These system parameters were used:

\[
\begin{align*}
H_{\text{min}} &= 1 \text{ tf}, \quad H_{\text{max}} = 800 \text{ tf} \\
\tau_{\text{max}} &= 85^\circ, \quad \eta_{\text{max}} = 85^\circ \\
\psi_{\text{max}} &= 85^\circ, \quad \phi_{\text{max}} = 15^\circ \\
\tau_{\text{max}} &= 10^4 \text{ m}, \quad \Phi_0 = 180^\circ \\
L_t &= 263 \text{ m}, \quad L_{\text{FPSO}} = 300 \text{ m} \\
K_{\text{FPSO}} &= \begin{bmatrix}
3.08 \text{ tf/m} & 0 & 0 \\
0 & 8.91 \text{ tf/m} & 0 \\
0 & 0 & 10^4 \text{ tf-m/}^\circ
\end{bmatrix}
\end{align*}
\]
These staring value were used:

\[
x_0 = \begin{bmatrix}
\Phi_0 + 180 \\
-\frac{1}{2} \\
0 \\
\frac{1}{2}H_{max} \\
0 \\
0 \\
0
\end{bmatrix}
\]

The environmental parameters for wind, current, wave, and swell from each test case are shown in Table 7.1.

<table>
<thead>
<tr>
<th>Case</th>
<th>Wind</th>
<th>Current</th>
<th>Swell</th>
<th>Wave</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Heading (°)</td>
<td>Speed (m/sec)</td>
<td>Heading (°)</td>
<td>Speed (m/sec)</td>
</tr>
<tr>
<td>1</td>
<td>180</td>
<td>8.0</td>
<td>180</td>
<td>0.75</td>
</tr>
<tr>
<td>2</td>
<td>180</td>
<td>8.0</td>
<td>180</td>
<td>0.75</td>
</tr>
<tr>
<td>3</td>
<td>180</td>
<td>0.0</td>
<td>180</td>
<td>0.75</td>
</tr>
<tr>
<td>4</td>
<td>180</td>
<td>8.0</td>
<td>180</td>
<td>0.00</td>
</tr>
<tr>
<td>5</td>
<td>180</td>
<td>0.0</td>
<td>180</td>
<td>0.75</td>
</tr>
<tr>
<td>6</td>
<td>180</td>
<td>8.0</td>
<td>180</td>
<td>0.75</td>
</tr>
<tr>
<td>7</td>
<td>180</td>
<td>8.0</td>
<td>180</td>
<td>0.75</td>
</tr>
<tr>
<td>8</td>
<td>180</td>
<td>25.0</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>9</td>
<td>180</td>
<td>25.0</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>10</td>
<td>180</td>
<td>25.0</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>11</td>
<td>180</td>
<td>25.0</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>12</td>
<td>180</td>
<td>15.0</td>
<td>180</td>
<td>0.00</td>
</tr>
<tr>
<td>13</td>
<td>180</td>
<td>15.0</td>
<td>180</td>
<td>1.50</td>
</tr>
<tr>
<td>14</td>
<td>180</td>
<td>15.0</td>
<td>180</td>
<td>1.50</td>
</tr>
<tr>
<td>15</td>
<td>180</td>
<td>15.0</td>
<td>180</td>
<td>0.00</td>
</tr>
<tr>
<td>16</td>
<td>180</td>
<td>0.0</td>
<td>180</td>
<td>1.50</td>
</tr>
<tr>
<td>17</td>
<td>180</td>
<td>15.0</td>
<td>180</td>
<td>1.50</td>
</tr>
<tr>
<td>18</td>
<td>180</td>
<td>15.0</td>
<td>180</td>
<td>1.50</td>
</tr>
<tr>
<td>19</td>
<td>180</td>
<td>0.0</td>
<td>180</td>
<td>1.50</td>
</tr>
<tr>
<td>20</td>
<td>270</td>
<td>15.0</td>
<td>180</td>
<td>1.50</td>
</tr>
<tr>
<td>21</td>
<td>180</td>
<td>8.0</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>22</td>
<td>0</td>
<td>0.0</td>
<td>180</td>
<td>0.75</td>
</tr>
<tr>
<td>23</td>
<td>180</td>
<td>8.0</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>24</td>
<td>180</td>
<td>8.0</td>
<td>180</td>
<td>0.75</td>
</tr>
</tbody>
</table>

Table 7.1: Tug load optimization test data.

There is no supposition that the exact environmental loads (discussed in Section 7.1.1 vary linearly between the known data points, thus the smooth equations are as good as the nonsmooth equations from the point of view of problem accuracy. They are superior from the point of view of optimizing Problem 7.1, as shown in Table 7.2. This table gives the number of function evaluations required to solve each of the test problems and \( \Psi, H, \) and \( \Phi \) from the solution of each. Note that the nonsmooth
<table>
<thead>
<tr>
<th>Case</th>
<th>Smooth</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th>Nonsmooth</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Function</td>
<td>$\Psi$</td>
<td>$H$</td>
<td>$\Phi$</td>
<td>Function</td>
<td>$\Psi$</td>
<td>$H$</td>
<td>$\Phi$</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Eval’ns</td>
<td>(°)</td>
<td>(tf)</td>
<td>(°)</td>
<td>Eval’ns</td>
<td>(°)</td>
<td>(tf)</td>
<td>(°)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>8</td>
<td>0.00</td>
<td>6.40</td>
<td>0.00</td>
<td>16</td>
<td>0.00</td>
<td>6.40</td>
<td>0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>8</td>
<td>0.00</td>
<td>10.13</td>
<td>0.00</td>
<td>23</td>
<td>0.00</td>
<td>10.13</td>
<td>0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>8</td>
<td>0.00</td>
<td>2.47</td>
<td>0.00</td>
<td>12</td>
<td>0.00</td>
<td>2.47</td>
<td>0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>8</td>
<td>0.00</td>
<td>3.93</td>
<td>0.00</td>
<td>26</td>
<td>0.00</td>
<td>3.93</td>
<td>0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>8</td>
<td>0.00</td>
<td>2.47</td>
<td>0.00</td>
<td>12</td>
<td>0.00</td>
<td>2.47</td>
<td>0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>8</td>
<td>0.00</td>
<td>6.40</td>
<td>0.00</td>
<td>16</td>
<td>0.00</td>
<td>6.40</td>
<td>0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>8</td>
<td>0.00</td>
<td>10.13</td>
<td>0.00</td>
<td>23</td>
<td>0.00</td>
<td>10.13</td>
<td>0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>0.00</td>
<td>38.41</td>
<td>0.00</td>
<td>16</td>
<td>0.00</td>
<td>38.41</td>
<td>0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>8</td>
<td>0.00</td>
<td>31.94</td>
<td>0.00</td>
<td>23</td>
<td>0.00</td>
<td>31.94</td>
<td>0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>8</td>
<td>0.00</td>
<td>31.94</td>
<td>0.00</td>
<td>23</td>
<td>0.00</td>
<td>31.94</td>
<td>0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>8</td>
<td>0.00</td>
<td>42.13</td>
<td>0.00</td>
<td>16</td>
<td>0.00</td>
<td>42.13</td>
<td>0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>8</td>
<td>0.00</td>
<td>13.83</td>
<td>0.00</td>
<td>19</td>
<td>0.00</td>
<td>13.83</td>
<td>0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>8</td>
<td>0.00</td>
<td>23.70</td>
<td>0.00</td>
<td>16</td>
<td>0.00</td>
<td>23.70</td>
<td>0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>8</td>
<td>0.00</td>
<td>37.00</td>
<td>0.00</td>
<td>16</td>
<td>0.00</td>
<td>37.00</td>
<td>0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>8</td>
<td>0.00</td>
<td>13.83</td>
<td>0.00</td>
<td>19</td>
<td>0.00</td>
<td>13.83</td>
<td>0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>8</td>
<td>0.00</td>
<td>9.88</td>
<td>0.00</td>
<td>16</td>
<td>0.00</td>
<td>9.88</td>
<td>0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>8</td>
<td>0.00</td>
<td>23.70</td>
<td>0.00</td>
<td>16</td>
<td>0.00</td>
<td>23.70</td>
<td>0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>8</td>
<td>0.00</td>
<td>37.00</td>
<td>0.00</td>
<td>16</td>
<td>0.00</td>
<td>37.00</td>
<td>0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>8</td>
<td>0.00</td>
<td>9.88</td>
<td>0.00</td>
<td>16</td>
<td>0.00</td>
<td>9.88</td>
<td>0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>12</td>
<td>37.30</td>
<td>13.37</td>
<td>0.48</td>
<td>14</td>
<td>37.30</td>
<td>13.34</td>
<td>0.48</td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>8</td>
<td>0.00</td>
<td>3.93</td>
<td>0.00</td>
<td>26</td>
<td>0.00</td>
<td>3.93</td>
<td>0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>22</td>
<td>38.98</td>
<td>8.08</td>
<td>0.11</td>
<td>22</td>
<td>39.02</td>
<td>8.04</td>
<td>0.11</td>
<td></td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>26</td>
<td>41.71</td>
<td>8.01</td>
<td>0.12</td>
<td>*</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>12</td>
<td>26.56</td>
<td>13.65</td>
<td>0.04</td>
<td>12</td>
<td>26.30</td>
<td>13.58</td>
<td>0.04</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 7.2: Smooth and nonsmooth optimization of Problem 7.1.

*Failed to converge.

formulation failed to converge in one case. Also note that the solutions are nearly the same in all other cases and that the smooth formulation took roughly one-half as many function evaluations to converge to a solution.
7.4 Summary

A nonsmooth optimization problem, unrelated to the structural optimization problem has been presented. Details of the problem formulation have been given. Sufficient details have been presented so that the underlying sources of nonsmoothness can be understood.

With this background, a smooth alternative formulation was proposed using methods presented earlier for the reformulation of the structural optimization problem. A comparison was made between the solution obtained when the formulation was both smooth and nonsmooth, demonstrating the potential savings in algorithm iterations by using these simple techniques to reformulate the nonsmooth problem as a smooth one.
Chapter 8

Conclusions and Recommendations

A new algorithm has been presented for the solution of structural optimization problems in which the stress constraints are nonsmooth. The goal of the algorithm was to significantly improve the existing design. The proposed method is not guaranteed to return optimal points, but it will result in an improved feasible solution. The proposed method was developed because it was anticipated that it would outperform direct search methods by using derivative information more effectively. It outperformed two pattern search methods in the testing presented in Section 6.4. Also, it was demonstrated in Section 1.1 that reformulation of the problem to a smooth mixed integer problem would require the solution of too many subproblems, since this number depends on the problem size. The number of subproblems in the proposed method is determined algorithmically, and it has been shown to be independent of the problem size. The merit of the proposed algorithm is thus in its relative efficiency compared to these two alternatives.

It has been shown in Section 6.4 that the Two-Stage algorithm is more efficient than the direct search methods. It has also been shown that it does not fail to give feasible designs in the test cases when the proposed constraints are used throughout, but that it may fail to find an optimal design when some of the design parameters are near constraint boundaries. The Two-Stage algorithm can give less than ideal
results when the solution from stage I is near a design parameter boundary ($\lambda \approx \lambda^L$ or $\lambda \approx \lambda^U$). When this is the case:

- It may find a nonoptimal solution if the wrong secondary constraints are chosen. This is only a slight problem because the resulting solution will be much better than the initial solution and probably not far from optimal.
- It may result in an infeasible solution using the code constraints if the wrong secondary constraints are chosen. By relaxing the constraint tolerance, a nearly feasible solution can be found.

These possible pitfalls must be taken in the context of the overall problem. In particular, the solution from the continuous variable problem must be mapped to a discrete domain before it can be used. This step introduces other difficulties with respect to feasibility (see Section 8.2).

### 8.1 Thesis Summary

The Two-Stage algorithm has been shown to work well for the class of problems that arise in structural engineering in which the allowable stresses are defined piecewise over the domain of their design variables. It is anticipated that it will be effective for other similar optimization problems, provided that a smooth approximation to the problem is available (see for example, the tug load minimization problem in Chapter 7).

Following are the contributions of this work:

- Determination of the sources of nonsmoothness in the current implementation of the AISC ASD [3] code.
- Study of these effects using optimality criteria and direct search methods.
Formulation of a minimum set of mutually exclusive constraints that need be applied in order to impose all of the code allowable stress criteria.

Proposal of acceptable alternative allowable stress functions for the inelastic behavior of lateral-torsional buckling, flange local buckling, and web local buckling; and the proposal of acceptable alternative miscellaneous functions (plate buckling coefficients, etc.).

Development of the Two-Stage algorithm for the solution of nonsmooth structural optimization problems; and the development of software to implement the algorithm, and verification of the proposed algorithm on a set of test models using the sequential quadratic programming method.

Comparison of the proposed algorithm with a direct search method on the test models.

Contributions have been made in optimization as well as structural engineering. The main contribution is the specification of the Two-Stage algorithm, which is general enough to be applied to other nonsmooth problems that have the appropriate characteristics. Contributions in structural engineering have been made in general through the advancement of the automated design problem, and specifically through the proposal of smooth allowable stress specifications for lateral-torsional buckling, flange local buckling, and web local buckling, which have been shown to be sufficiently accurate for use in design.

Future work using the ideas presented includes:

Formulation of the optimization problem using alternate codes, such as the AISC LRFD [4]. It is fully anticipated that the method presented will apply to most other steel design codes.

Extension of the allowable stress equations to include sections besides wide flanges (e.g. channels, tubes, angles).
8.2 Future Areas of Research

Several areas must be further researched before the proposed algorithm is fully useful as a design tool. The original goal of the research was to develop software for the automated design of steel structures. A major boundary was crossed, and now readily available, efficient software can be applied to the general structural optimization problem.

- The question of performance is a major issue. Optimization of structures with a large number of variables and constraints is slow. This can be addressed by studying the best choices of surrogate constraint functions.

- The method discussed requires an initial solution, and only finds a nearby, local minimum. Design space exploration methods are of interest to explore for alternative, "novel" designs. The limitations of large design spaces currently dictate that systematic exploration of the design space is impractical.

- The solution of the design problem using continuous variables is done for convenience because discrete methods are less attractive. Once a continuous solution has been obtained, a nearly equivalent discrete solution must be determined before the design becomes useful to the engineer (see below). A method is needed for extracting the most critical information from the continuous design so that the discrete solution retains as much of the optimal characteristics as possible.

- Consideration of other cost issues besides raw steel tonnage are of interest. For example, a comparison between rolled and built-up shapes requires consideration of the added cost in connections, such as welding, and the addition of stiffener and doubler plates. Another item is the cost savings associated with ordering certain items in bulk. A structure that has many repeated member sizes is easier to fabricate and generally costs less.
The results from Chapter 6 provide useful information regarding the viability of the optimization of structures with nonsmooth stress constraints. The results are good, and it appears that the proposed algorithm is effective for this purpose. To motivate further study and validate the results, the optimal sizes for "bigframe," the large test model, are further studied here. It was mentioned in Chapter 2 that the solution of the continuous variable problem would somehow need to be converted to its discrete equivalent. Here, a simple method of rounding the design variables demonstrates the pitfalls of this procedure. If all of the variables are rounded to some nearby discrete size, one can assume that a design consisting of built-up members might be produced that is nearly equivalent in weight to the optimal design. This is true; however, the resulting solution may not be feasible. If this simplistic procedure is used on the solution from Chapter 6, the results for the objective and aggregate constraint function for the model bigframe are \( f = 130.0 \) and \( h = 5.661 \) (about 3% lighter than the optimal solution, but highly infeasible). A somewhat more sophisticated method is employed here wherein the depth and flange width are rounded to the nearest 3 and \( \frac{1}{2} \) inch, respectively, while the web and flange thickness are rounded up to the nearest \( \frac{1}{16} \) and \( \frac{1}{8} \) inch, respectively. This yields a feasible design (\( h = 0 \)), at the expense of optimality (\( f = 150.8 \), about 12% heavier than optimal). This design is shown in Table 8.1 along with the optimal solution.

Another issue that requires further study is the added cost of producing built-up members versus rolled shapes. No rolled shapes exist corresponding to many of the sizes in Table 8.1. Each of these items must be further addressed before a useful design tool can be produced.
<table>
<thead>
<tr>
<th>Group</th>
<th>Optimal</th>
<th>Rounded (discrete)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$d$ (in)</td>
<td>$t_w$ (in)</td>
</tr>
<tr>
<td>1</td>
<td>34.2</td>
<td>0.2332</td>
</tr>
<tr>
<td>2</td>
<td>40.3</td>
<td>0.2784</td>
</tr>
<tr>
<td>3</td>
<td>30.5</td>
<td>0.2160</td>
</tr>
<tr>
<td>4</td>
<td>27.7</td>
<td>0.1870</td>
</tr>
<tr>
<td>5</td>
<td>25.5</td>
<td>0.1745</td>
</tr>
<tr>
<td>6</td>
<td>31.1</td>
<td>0.2194</td>
</tr>
<tr>
<td>7</td>
<td>24.9</td>
<td>0.1716</td>
</tr>
<tr>
<td>8</td>
<td>23.3</td>
<td>0.1603</td>
</tr>
<tr>
<td>9</td>
<td>30.4</td>
<td>0.2152</td>
</tr>
<tr>
<td>10</td>
<td>23.0</td>
<td>0.1616</td>
</tr>
<tr>
<td>11</td>
<td>21.1</td>
<td>0.1423</td>
</tr>
<tr>
<td>12</td>
<td>27.8</td>
<td>0.2085</td>
</tr>
<tr>
<td>13</td>
<td>21.6</td>
<td>0.1526</td>
</tr>
<tr>
<td>14</td>
<td>17.9</td>
<td>0.1250</td>
</tr>
<tr>
<td>15</td>
<td>25.2</td>
<td>0.2016</td>
</tr>
<tr>
<td>16</td>
<td>24.9</td>
<td>0.1733</td>
</tr>
<tr>
<td>17</td>
<td>10.6</td>
<td>0.1250</td>
</tr>
<tr>
<td>18</td>
<td>25.3</td>
<td>0.2041</td>
</tr>
</tbody>
</table>

Table 8.1: Optimal section properties for model bigframe.
Bibliography


Appendix A

AISC ASD Allowable Stresses

The ASD [3] code gives the following specifications for doubly symmetric non-hybrid, unstiffened wide flange members with non-slender webs subjected to axial compression or tension, shear, and bending about their strong-axis. Refer to Figure 3.3 for steel design parameter details. The chapter reference(s) at the beginning of each set of equations refer to the ASD [3] code.

Local Buckling (ASD Chapters B5 and A-B5)

\[
\begin{align*}
\text{compact} & \iff \\
& \begin{cases}
\frac{b_f}{2t_f} & \leq \frac{65}{\sqrt{F_y}} \\
\frac{d}{t_w} & \leq \frac{640}{\sqrt{F_y}} \left(1 + 3.74 \frac{f_a}{F_y}\right), & f_a \geq 0 \\
\frac{h}{t_w} & \leq 70, & -0.16F_y \leq f_a < 0 \\
& \frac{257}{\sqrt{F_y}}, & f_a < -0.16F_y
\end{cases}
\end{align*}
\]

\[
\begin{align*}
\text{slender} & \iff \\
& \begin{cases}
\frac{b_f}{2t_f} & > 95 \sqrt{\frac{k_c}{F_y}} \\
\frac{h}{t_w} & > 760 \sqrt{\frac{f_t}{F_y}}
\end{cases}
\end{align*}
\]

non-compact \iff not compact or slender

205
$$Q_s = \begin{cases} 1, & \frac{b_f}{2t_f} \leq 95\sqrt{\frac{k_c}{F_y}} \\ 1.293 - 0.00309 \left( \frac{b_f}{2t_f} \right)^2 \sqrt{\frac{F_y}{k_c}}, & 95\sqrt{\frac{k_c}{F_y}} < \frac{b_f}{2t_f} < 195\sqrt{\frac{k_c}{F_y}} \\ \frac{26,200k_c}{F_y(b_f/2t_f)^2}, & \frac{b_f}{2t_f} \geq 195\sqrt{\frac{k_c}{F_y}} \end{cases}$$

$$Q_o = 1 \text{ (see below)}$$

where: $$k_c = \begin{cases} 1, & \frac{h}{t_w} \leq 70 \\ \frac{4.05}{(h/t_w)^{0.45}}, & \frac{h}{t_w} > 70 \end{cases}$$

It is assumed that all stiffened elements (i.e. the web) are non-slender, thus $$Q_o = 1$$. This is enforced by appropriate constraints on $$h/t_w$$.

Limiting Slenderness Ratios (ASD Chapter B7)

$$\frac{K_y L_y}{r_y} \leq \begin{cases} 300, & \text{for tension members} \\ 200, & \text{for compression members} \end{cases}$$

Tension Members (ASD Chapter D)

$$F_t = \min \left( 0.6F_y, 0.5F_w \frac{A_e}{A} \right)$$

Columns and Other Compression Members (ASD Chapter E)

$$F_a = \begin{cases} \frac{1 - (K_y L_y/r_y)^2}{2r_y^2} Q_o F_y, & \frac{K_y L_y}{r_y} < C_c \\ \frac{5}{3} + \frac{3K_y L_y/r_y}{8A_c} - \frac{(K_y L_y/r_y)^3}{8A_c}, & \frac{K_y L_y}{r_y} < C_c \\ \frac{12\pi^2 E}{23(K_y L_y/r_y)^2}, & \frac{K_y L_y}{r_y} \geq C_c \end{cases}$$
where: \( C_c = \pi \sqrt{\frac{2E}{Q_s F_y}} \)

Beams and Other Flexural Members (ASD Chapter F)

\[
F_b = \begin{cases} 
F_{bFLB}, & L_b \leq L_c \\
\min (F_{bFLB}, F_{bLTB}), & L_b > L_c 
\end{cases}
\]

\[
F_y = \begin{cases} 
0.4F_y, & \frac{380}{\sqrt{F_y}} \leq \frac{h}{t_w} \\
\frac{190}{h/t_w} \sqrt{\frac{b_y}{F_y}} \frac{F_y}{2.59}, & \frac{380}{\sqrt{F_y}} < \frac{h}{t_w} < \frac{550}{\sqrt{F_y}} \\
\frac{45,000}{(h/t_w)^2} \frac{b_y}{F_y} \frac{F_y}{2.89}, & \frac{h}{t_w} \geq \frac{550}{\sqrt{F_y}} 
\end{cases}
\]

where: \( L_c = \min \left( \frac{76b_f}{\sqrt{F_y}}, \frac{20,000}{(d/A_f)F_y} \right) \)

\[
F_{bFLB} = \begin{cases} 
0.66F_y, & \text{if compact} \\
(0.79 - 0.002 \frac{b_y}{2f} \sqrt{\frac{F_y}{k_c}}), & \text{if non-compact} \\
0.6Q_sF_y, & \text{if slender} 
\end{cases}
\]

\[
F_{bLTB} = \min (0.6F_y, \max (F_{bw}, F_{bSV}))
\]

\[
F_{bw} = \begin{cases} 
\left[ \frac{2}{3} - \frac{F_y(L_b/r_T)^2}{1,530,000C_b} \right], & \sqrt{\frac{102,000C_b}{F_y}} \leq \frac{L_b}{r_T} \leq \sqrt{\frac{510,000C_b}{F_y}} \\
\left[ \frac{170,000C_b}{(L_b/r_T)^2}, & \frac{L_b}{r_T} > \sqrt{\frac{510,000C_b}{F_y}} \right] 
\end{cases}
\]

\[
F_{bSV} = \frac{12,000C_b}{L_b(d/A_f)}
\]

Combined Stresses (ASD Chapter H)

Members must satisfy \( UC \leq 1 \)
where: $UC = \max \left( IR, \left| \frac{f_v}{F_v} \right| \right)$

$$IR = \begin{cases} 
\frac{f_a}{F_a} + \frac{|f_a|}{F_b}, & f_a \geq 0 \\
\frac{f_a}{F_a} + \frac{|f_a|}{F_b}, & -0.15F_a \leq f_a < 0 \\
\max \left( -\frac{f_a}{F_a} + B_1 \frac{|f_a|}{F_b}, \ldots \right. & \\
\left. -\frac{f_a}{0.6Q_v F_b} + \frac{|f_a|}{F_b} \right), & f_a < -0.15F_a 
\end{cases}$$

$$B_1 = \frac{C_m}{1 + f_a/F_e} \geq 1$$

$$F_e' = \frac{12\pi^2 E}{23(K_L L_x/r_x)^2}$$
Appendix B

Polynomial Smoothing Functions

If a piecewise continuous function \( f(\lambda) \) is defined as:

\[
f(\lambda) = \begin{cases} 
  f_1(\lambda), & \lambda \leq \lambda_1 \\
  f_2(\lambda), & \lambda > \lambda_1
\end{cases}
\]

and has known values \( f_1(a) = f_a, f_2(b) = f_b \), and first derivatives \( f_1'(a) = f'_a, f_2'(b) = f'_b \) for \( a < \lambda_1 < b \), then a transition function between \( a \) and \( b \) is defined by:

\[
s : [0, 1] \rightarrow \mathbb{R}
\]

\[
s(z) = c_0 + c_1 z + c_2 z^2 + c_3 z^3
\]

\[
s'(z) = c_1 + 2c_2 z + 3c_3 z^2
\]

\[
s''(z) = 2c_2 + 6c_3 z
\]

where: \( z = \frac{\lambda - a}{b - a} \)

To create a smooth transition function, the polynomial coefficients are determined so that the function \( s \) and its first derivative match the function \( f \) at the endpoints \( a \) and \( b \).
and $b$. Thus $c$ solves $Ac = d$, where:

$$
c = \begin{bmatrix} c_3 \\ c_2 \\ c_1 \\ c_0 \end{bmatrix}, \quad A = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 3 & 2 & 1 & 0 \end{bmatrix}, \quad d = \begin{bmatrix} s(0) \\ s'(0) \\ s(1) \\ s'(1) \end{bmatrix} = \begin{bmatrix} f_a \\ (b - a)f'_a \\ f_b \\ (b - a)f'_b \end{bmatrix}
$$

In general, if $\lambda = \lambda(x)$, $a = a(x)$, $b = b(x)$ for $x \in \mathbb{R}^n$ then:

$$
f_a(x) = f_1(a(x)), \quad f_b(x) = f_2(b(x))
$$

$$
f'_a(x) = f'_1(a(x)), \quad f'_b(x) = f'_2(b(x))
$$

$$
c = c(x), \quad d = d(x)
$$

$$
z(x) = \frac{\lambda(x) - a(x)}{b(x) - a(x)}
$$

$$
s(x, z(x)) = c_0(x) + c_1(x)z(x) + c_2(x)z(x)^2 + c_3(x)z(x)^3
$$

The gradients of $z$ and $s$ with respect to $x$ are:

$$
\nabla_x z = \frac{1}{b - a} [\nabla_x \lambda - \nabla_x a - z (\nabla_x b - \nabla_x a)]
$$

$$
\nabla_x s(x, z(x)) = \nabla_x c_0 + z \nabla_x c_1 + z^2 \nabla_x c_2 + z^3 \nabla_x c_3 + s'(z)\nabla_x z
$$

The Hessians of $z$ and $s$ with respect to $x$ are:

$$
\nabla_{xx}^2 z = \frac{1}{b - a} \left[ \nabla_{xx}^2 \lambda - \nabla_{xx}^2 a - z (\nabla_{xx}^2 b - \nabla_{xx}^2 a) \right]
$$

$$
- (\nabla_x b - \nabla_x a) \nabla_x z^T - \nabla_x z (\nabla_x b - \nabla_x a)^T \right]
$$

$$
\nabla_{xx}^2 s(x, z(x)) = \nabla_{xx}^2 c_0 + z \nabla_{xx}^2 c_1 + z^2 \nabla_{xx}^2 c_2 + z^3 \nabla_{xx}^2 c_3 + s''(z)\nabla_x z \nabla_x z^T + s'(z)\nabla_{xx}^2 z
$$

$$
+ (\nabla_x c_1 + 2z \nabla_x c_2 + 3z^2 \nabla_x c_3) \nabla_x z^T + \nabla_x z (\nabla_x c_1 + 2z \nabla_x c_2 + 3z^2 \nabla_x c_3)^T
$$

Since the coefficients are related to the function and derivative values by a linear relationship $c(x) = A^{-1}d(x)$, the derivatives of the coefficients are $c_x = A^{-1}d_x$ and $c_{xx} = A^{-1}d_{xx}$, where:

$$
d_x = \begin{bmatrix} \nabla_x f_a^T \\ (b - a)\nabla_x f_a^T + f'_a (\nabla_x b - \nabla_x a)^T \\ \nabla_x f_b^T \\ (b - a)\nabla_x f_b^T + f'_b (\nabla_x b - \nabla_x a)^T \end{bmatrix}
$$
$$d_{xx} = \begin{bmatrix}
\Delta_{xx}^2 f_a \\
(b - a)\Delta_{xx}^2 f'_a + \Delta_{xx}^2 f'_a (\nabla_{xx} b - \nabla_{xx} a)^T \ldots \\
+ (\nabla_{xx} b - \nabla_{xx} a) \nabla_{xx} f'_a T + f'_a (\nabla_{xx}^2 b - \nabla_{xx}^2 a)
\end{bmatrix}
$$

$$
\begin{bmatrix}
\Delta_{xx}^2 f_b \\
(b - a)\Delta_{xx}^2 f'_b + \Delta_{xx}^2 f'_b (\nabla_{xx} b - \nabla_{xx} a)^T \ldots \\
+ (\nabla_{xx} b - \nabla_{xx} a) \nabla_{xx} f'_b T + f'_b (\nabla_{xx}^2 b - \nabla_{xx}^2 a)
\end{bmatrix}
$$