Abstract

FISHER, SETH TYSON. Development of a Simplified Procedure to Predict Dead Load Deflections of Skewed and Non-Skewed Steel Plate Girder Bridges. (Under the direction of Emmett Sumner PhD., P.E.)

Many of today’s steel bridges are being constructed with longer spans and higher skew. As a result, the North Carolina Department of Transportation (NCDOT) has experienced numerous problems in predicting the dead load deflections of steel plate girder bridges. In response to these problems, the NCDOT has funded this research project (Project Number 2004-14). Common dead load deflection prediction methods, which traditionally utilize single girder line (SGL) analysis, have been shown to over predict the dead load deflections; the inaccuracy can result in various costly construction delays and maintenance and safety issues. The primary objective of this research is to develop a simplified procedure to predict dead load deflections of skewed and non-skewed steel plate girder bridges. In developing the simplified procedure, ten steel plate girder bridges were monitored during placement of the concrete deck to observe the deflection of the girders. Detailed three-dimensional finite element models of the bridge structures were generated in the commercially available finite element analysis program ANSYS, and correlations were made between the simulated deflections and the field measured deflections. With confidence in the ability of the developed finite element models to capture bridge deflection behavior, a preprocessor program was written to automate the finite element generation. Subsequently, a parametric study was conducted to investigate the effect of skew angle, girder spacing, span length, cross frame stiffness, number of girders within the span, and exterior to interior girder load ratio on the girder deflection behavior. The results from the parametric were used to
develop a simplified procedure, which modifies traditional SGL predictions with empirical
equations to account for skew angle, girder spacing, span length, and exterior to interior
girder load ratio. Predictions of the deflections from the simplified procedure and from SGL
analyses were compared to the deflections predicted from finite element models (ANSYS)
and the field measured deflections to validate the procedure. It was concluded that the
simplified procedure may be utilized to predict dead load deflections for simple span, steel
plate girder bridges. Additionally, an alternative prediction method has been proposed to
predict deflections in continuous span, steel plate girder bridges with equal exterior girder
loads, and supplementary comparisons were made to validate this method.
DEVELOPMENT OF A SIMPLIFIED PROCEDURE TO PREDICT DEAD LOAD DEFLECTIONS OF SKEWED AND NON-SKEWED STEEL PLATE GIRDER BRIDGES

by

SETH TYSON FISHER

A thesis submitted to the Graduate Faculty of North Carolina State University in partial fulfillment of the requirements for the Degree of Master of Science

CIVIL ENGINEERING

Raleigh

2006

APPROVED BY:

Sami H. Rizkalla, Ph.D., P.E.  Mervyn J. Kowalsky, Ph.D.

Emmett A. Sumner, Ph.D., P.E., Chair
for my sunshine...
Biography

Seth Tyson Fisher was born on January 6, 1981 and raised in Winston Salem, NC. He entered North Carolina State University (NCSU) in the fall of 1999 in pursuit of an engineering degree. After graduating with a Bachelor of Science degree in Civil Engineering in December of 2003, he re-entered NCSU in pursuit of a Master of Science degree in Civil Engineering. Upon completion of his Master’s degree, the author will begin work at HNTB in Raleigh, NC as a bridge design engineer.
Acknowledgements

First, I would like to thank the North Carolina Department of Transportation for providing the funding and support of this project.

I would like to thank my advisor and committee chair, Dr. Emmett A. Sumner, for his excellent guidance over the past two years. He was steadfast in answering any question that arose, professionally or personally. It is an honor to call him friend. I would also like to thank Dr. Sami Rizkalla and Dr. Mervyn Kowalsky for kindly serving as members of my thesis committee.

I would like to thank fellow graduate student Todd Whisenhunt for showing me the ropes, and especially for being a true inspiration to the definition of dedication and hard work. I respect him as an engineer and as a great friend. Thanks also to Nuttapone Paoinchantara for always being there as a friend and never hesitating to help out, whether in the field or in the office.

I would like to thank my parents for all their unconditional love and support throughout my 25 years. I will consider myself fortunate to become half the parent either of them has been.

I thank the Lord through Whom all things are possible.
Table of Contents

List of Tables ........................................................................................................................................ viii
List of Figures......................................................................................................................................... ix
Chapter 1 - Introduction .......................................................................................................................... 1
  1.1 Background ..................................................................................................................................... 1
  1.1.1 General ...................................................................................................................................... 1
  1.1.2 Current Analysis and Design ................................................................................................. 2
  1.1.3 Bridge Components ................................................................................................................. 4
  1.1.4 Equivalent Skew Offset ......................................................................................................... 9
  1.2 Objective and Scope .................................................................................................................. 11
  1.3 Outline of Thesis ....................................................................................................................... 12
Chapter 2 - Literature Review ................................................................................................................ 14
  2.1 Introduction ............................................................................................................................... 14
  2.2 Phase I Research ...................................................................................................................... 14
    2.2.1 Construction Issues ............................................................................................................. 14
    2.2.2 Parameters.......................................................................................................................... 15
    2.2.3 Finite Element Modeling ..................................................................................................... 16
  2.3 Phase II Research ....................................................................................................................... 18
    2.3.1 Reviews of Whisenhunt (2004) and Paoinchantara (2005) ............................................. 18
    2.3.2 Parametric Studies ............................................................................................................. 19
    2.3.3 Preprocessor Programs ...................................................................................................... 20
  2.4 Need for Research ...................................................................................................................... 21
Chapter 3 - Field Measurement Procedure and Results ........................................................................ 23
  3.1 Introduction ............................................................................................................................... 23
  3.2 General ........................................................................................................................................ 23
  3.3 Bridge Selection ........................................................................................................................... 23
  3.4 Bridges Studied ............................................................................................................................ 24
    3.4.1 General Characteristics ....................................................................................................... 24
    3.4.2 Specific Bridges ................................................................................................................ 25
  3.5 Field Measurement .................................................................................................................... 33
    3.5.1 Overview ............................................................................................................................ 33
    3.5.2 Conventional Method ........................................................................................................ 33
    3.5.3 Alternate Method: Wilmington St Bridge .......................................................................... 37
  3.6 Summary of Measured Deflections ............................................................................................ 39
  3.7 Summary ...................................................................................................................................... 40
Chapter 4 - Finite Element Modeling and Results .................................................................................. 42
  4.1 Introduction ............................................................................................................................... 42
  4.2 General ....................................................................................................................................... 42
  4.3 Bridge Components ....................................................................................................................... 43
    4.3.1 Plate Girders ..................................................................................................................... 43
    4.3.2 Cross Frames ..................................................................................................................... 47
    4.3.3 Stay-in-Place Metal Deck Forms ..................................................................................... 51
    4.3.4 Concrete Deck and Rigid Links ....................................................................................... 53
  4.4 Modeling Procedure ................................................................................................................... 54
    4.4.1 Automated Model Generation Using MATLAB ............................................................. 55
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.4.2</td>
<td>MATLAB Limitations</td>
<td>58</td>
</tr>
<tr>
<td>4.4.3</td>
<td>Additional Modeling and Consistency Checks</td>
<td>59</td>
</tr>
<tr>
<td>4.4.4</td>
<td>Specific Modeling Adjustments</td>
<td>60</td>
</tr>
<tr>
<td>4.4.5</td>
<td>Validation of ANSYS Models Generated with MATLAB</td>
<td>60</td>
</tr>
<tr>
<td>4.5</td>
<td>Modeling Assumptions</td>
<td>62</td>
</tr>
<tr>
<td>4.6</td>
<td>Deflection Results of ANSYS Models</td>
<td>63</td>
</tr>
<tr>
<td>4.6.1</td>
<td>No SIP Forms</td>
<td>63</td>
</tr>
<tr>
<td>4.6.2</td>
<td>Including SIP Forms</td>
<td>65</td>
</tr>
<tr>
<td>4.7</td>
<td>Summary</td>
<td>66</td>
</tr>
<tr>
<td>5.1</td>
<td>Introduction</td>
<td>68</td>
</tr>
<tr>
<td>5.2</td>
<td>General</td>
<td>68</td>
</tr>
<tr>
<td>5.3</td>
<td>Parametric Study</td>
<td>69</td>
</tr>
<tr>
<td>5.3.1</td>
<td>Number of Girders</td>
<td>70</td>
</tr>
<tr>
<td>5.3.2</td>
<td>Cross Frame Stiffness</td>
<td>71</td>
</tr>
<tr>
<td>5.3.3</td>
<td>Exterior-to-Interior Girder Load Ratio</td>
<td>74</td>
</tr>
<tr>
<td>5.3.4</td>
<td>Skew Offset</td>
<td>75</td>
</tr>
<tr>
<td>5.3.5</td>
<td>Girder Spacing-to-Span Ratio</td>
<td>77</td>
</tr>
<tr>
<td>5.3.6</td>
<td>Conclusions</td>
<td>79</td>
</tr>
<tr>
<td>5.4</td>
<td>Simplified Procedure Development</td>
<td>80</td>
</tr>
<tr>
<td>5.4.1</td>
<td>Exterior Girder Deflections</td>
<td>81</td>
</tr>
<tr>
<td>5.4.2</td>
<td>Differential Deflections</td>
<td>86</td>
</tr>
<tr>
<td>5.4.3</td>
<td>Example</td>
<td>95</td>
</tr>
<tr>
<td>5.4.4</td>
<td>Conclusions</td>
<td>96</td>
</tr>
<tr>
<td>5.5</td>
<td>Additional Considerations</td>
<td>96</td>
</tr>
<tr>
<td>5.5.1</td>
<td>Continuous Span Bridges</td>
<td>97</td>
</tr>
<tr>
<td>5.5.2</td>
<td>Unequal Exterior-to-Interior Girder Load Ratios</td>
<td>100</td>
</tr>
<tr>
<td>5.6</td>
<td>Summary</td>
<td>102</td>
</tr>
<tr>
<td>5.1</td>
<td>Introduction</td>
<td>104</td>
</tr>
<tr>
<td>5.2</td>
<td>General</td>
<td>105</td>
</tr>
<tr>
<td>5.3</td>
<td>Comparisons of Field Measured Deflections to Predicted Single Girder Line and ANSYS Deflections</td>
<td>106</td>
</tr>
<tr>
<td>6.3.1</td>
<td>Predicted Single Girder Line Deflections vs. Field Measured Deflections</td>
<td>106</td>
</tr>
<tr>
<td>6.3.2</td>
<td>ANSYS Predicted Deflections vs. Field Measured Deflections</td>
<td>110</td>
</tr>
<tr>
<td>6.3.3</td>
<td>Single Girder Line Predicted Deflections vs. ANSYS Predicted Deflections</td>
<td>114</td>
</tr>
<tr>
<td>6.4</td>
<td>Comparisons of ANSYS Predicted Deflections to Simplified Procedure Predictions and SGL Predictions for Simple Span Bridges with Equal Exterior-to-Interior Girder Load Ratios</td>
<td>118</td>
</tr>
<tr>
<td>6.4.1</td>
<td>General</td>
<td>118</td>
</tr>
<tr>
<td>6.4.2</td>
<td>Comparisons</td>
<td>119</td>
</tr>
<tr>
<td>6.4.3</td>
<td>Summary</td>
<td>126</td>
</tr>
</tbody>
</table>
6.5 Comparisons of ANSYS Predicted Deflections to Alternative Simplified Procedure Predictions and SGL Predictions for Simple Span Bridges with Unequal Exterior-to-Interior Girder Load Ratios

6.5.1 General

6.5.2 Comparisons

6.5.3 Summary

6.6 Comparisons of ANSYS Deflections to SGL Straight Line Predictions and SGL Predictions for Continuous Span Bridges with Equal Exterior-to-Interior Girder Load Ratios

6.6.1 General

6.6.2 Comparisons

6.6.3 Summary

6.7 Comparisons of Prediction Methods to Field Measured Deflections

6.7.1 General

6.7.2 Simplified Procedure Predictions vs. Field Measured Deflections

6.7.3 Alternative Simplified Procedure Predictions vs. Field Measured Deflections

6.7.4 SGL Straight Line Predictions vs. Field Measured Deflections

6.8 Summary

Chapter 7 - Observations, Conclusions, and Recommendations

7.1 Summary

7.2 Observations

7.3 Conclusions

7.4 Recommended Simplified Procedures

7.4.1 Simple Span Bridges with Equal Exterior-to-Interior Girder Load Ratios

7.4.2 Simple Span Bridges with Unequal Exterior-to-Interior Girder Load Ratios

7.4.3 Continuous Span Bridges with Equal Exterior-to-Interior Girder Load Ratios

7.5 Future Considerations

References

Appendices

Appendix A - Simplified Procedure Flow Chart

Appendix B - Sample Calculations of the Simplified Procedure

Appendix C - Deflection Summary for Bridge 8

Appendix D - Deflection Summary for the Wilmington ST Bridge

Appendix E - Deflection Summary for Bridge 14

Appendix F - Deflection Summary for Bridge 10

Appendix G - Deflection Summary for Bridge 1

Appendix H - MATLAB Files
List of Tables

Table 3.1: Targeted Range of Geometric Properties .............................................................. 24
Table 3.2: Summary of Bridges Measured ............................................................................. 26
Table 3.3: Total Measured Vertical Deflection (inches) ........................................................ 39
Table 4.1: Midspan Deflections and Ratios Comparing Eno River Bridge ANSYS Models. 61
Table 4.2: ANSYS Predicted Deflections (No SIP Forms, Inches)........................................ 64
Table 4.3: ANSYS Predicted Deflections (Including SIP Forms, Inches) ............................. 65
Table 5.1: Girder Spacing-to-Span Ratios .............................................................................. 78
Table 5.2: Parametric Study Matrix........................................................................................ 80
Table 6.1: Ratios of SGL Predicted Deflections to Field Measured Deflections for Simple  
   Span Bridges at Midspan .............................................................................................. 108
Table 6.2: Ratios of SGL Predicted Deflections to Field Measured Deflections for  
   Continuous Span Bridges........................................................................................... 110
Table 6.3: Ratios of ANSYS Predicted Deflections to Field Measured Deflections for Simple 
   Span Bridges at Midspan ........................................................................................... 112
Table 6.4: Ratios of ANSYS Predicted Deflections to Field Measured Deflections for  
   Continuous Span Bridges........................................................................................... 114
Table 6.5: Statistical Analysis of Deflection Ratios at Midspan for Simple Span Bridges .. 115
Table 6.6: Statistical Analysis of Deflection Ratios for Continuous Span Bridges.......... 116
Table 6.7: Statistical Analysis Comparing SP Predictions to SGL Predictions at Various  
   Skew Offsets ............................................................................................................. 120
Table 6.8: Statistical Analysis Comparing ASP Predictions to SGL Predictions............ 127
Table 6.9: Statistical Analysis Comparing SGL Predictions to SGLSL Predictions........ 131
Table 6.10: Midspan Deflection Ratios of SP Predictions to Field Measured Deflections .. 134
Table 6.11: Statistical Analysis Comparing SP Predictions to SGL Predictions .......... 135
Table 6.12: Midspan Deflection Ratios of ASP Predictions to Field Measured Deflections 137
Table 6.13: Statistical Analysis Comparing ASP Predictions to SGL Predictions........... 138
Table 6.14: Deflection Ratios of SGLSL Predictions to Field Measured Deflections ...... 141
Table 6.15: Statistical Analysis Comparing SGLSL Predictions to SGL Predictions........ 141
Table 6.16: Complete Comparison of Deflection Ratios ...................................................... 145
Table 6.17: Complete Comparison of Differences in Deflection Magnitudes ................. 146
List of Figures

Figure 1.1: Traditional Single Girder Line Prediction Technique ............................................... 3
Figure 1.2: Misaligned Concrete Deck Elevations in Staged Construction ............................... 4
Figure 1.3: Steel Plate Girders, Intermediate Cross Frames and Intermediate Web Stiffeners .... 5
Figure 1.4: End Bent Diaphragm .......................................................................................... 6
Figure 1.5: SIP Metal Deck Forms ..................................................................................... 7
Figure 1.6: SIP Metal Deck Form Connection Detail .............................................................. 7
Figure 1.7: Pot Bearing Support .......................................................................................... 8
Figure 1.8: Elastomeric Bearing Pad Support ......................................................................... 8
Figure 1.9: Skew Angle and Bridge Orientation (Plan View) .................................................. 10
Figure 3.1: Typical Concrete Placement on Skewed Bridge .................................................. 24
Figure 3.2: Bridge 8 in Knightdale, North Carolina ............................................................... 27
Figure 3.3: Plan View Illustration of Bridge 8 (Not to Scale) ................................................ 27
Figure 3.4: Wilmington St Bridge in Raleigh, North Carolina ............................................... 28
Figure 3.5: Plan View Illustration of the Wilmington St Bridge (Not to Scale) ..................... 28
Figure 3.6: Bridge 14 in Knightdale, North Carolina ............................................................ 29
Figure 3.7: Plan View Illustration of Bridge 14 (Not to Scale) ............................................... 29
Figure 3.8: Bridge 10 in Knightdale, North Carolina ............................................................. 30
Figure 3.9: Plan View Illustration of Bridge 10 (Not to Scale) .............................................. 31
Figure 3.10: Bridge 1 in Raleigh, North Carolina ................................................................. 32
Figure 3.11: Plan View Illustration of Bridge 1 (Not to Scale) ............................................... 32
Figure 3.12: Instrumentation: String Potentiometer, Extension Wire, and Weight ............... 34
Figure 3.13: Instrumentation: Perforated Steel Angle, C-clamps, and Extension Wire ......... 34
Figure 3.14: Instrumentation: Switch & Balance, Power Supply, and Multimeter ............... 35
Figure 3.15: Instrumentation: Dial Gage .............................................................................. 36
Figure 3.16: Instrumentation: Tell-Tail (Weight, Extension Wire, and Wooden Stake) ......... 38
Figure 3.17: Plot of Non-composite Deflections ................................................................... 40
Figure 4.1: Single Plate Girder Model ................................................................................... 44
Figure 4.2: Bearing and Intermediate Web Stiffeners .......................................................... 46
Figure 4.3: Intermediate Cross Frames ................................................................................ 48
Figure 4.4: Finite Element Model with Cross Frames ............................................................ 49
Figure 4.5: End Bent Diaphragm ....................................................................................... 50
Figure 4.6: Plan View Illustration of SIP Form Truss System ............................................... 52
Figure 4.7: Schematic of Applied Method to Model the Concrete Slab ................................. 53
Figure 4.8: Finite Element Model Including a Segment of Concrete Deck Elements ......... 54
Figure 4.9: Midspan Deflections of Eno River Bridge Models ............................................ 62
Figure 4.10: ANSYS Deflection Plot (No SIP Forms) .......................................................... 65
Figure 4.11: ANSYS Deflection Plot (Including SIP Forms) ................................................ 66
Figure 5.1: Exterior Girder Deflection and Differential Deflection ........................................ 69
Figure 5.2: Bridge 8 at 0 Degree Skew Offset – Number of Girders Investigation ............... 70
Figure 5.3: Bridge 8 at 50 Degrees Skew Offset – Number of Girders Investigation ........... 70
Figure 5.4: Bridge 8 at 0 Degree Skew Offset – Cross Frame Stiffness Investigation .......... 72
Figure 5.5: Bridge 8 at 50 Degrees Skew Offset – Cross Frame Stiffness Investigation ....... 72
Figure 5.6: Eno at 0 Degree Skew Offset – Cross Frame Stiffness Investigation ................. 73
Figure 5.7: Eno at 50 Degrees Skew Offset – Cross Frame Stiffness Investigation ............... 73
Figure 6.12: Simplified Procedure Predictions vs. SGL Predictions ........................................ 124
Figure 6.13: ANSYS Deflections vs. Simplified Procedure and SGL Predictions for the
Camden SB Bridge .............................................................................................................. 125
Figure 6.14: ASP Predictions vs. SGL Predictions for Simple Span Bridges with Unequal
Exterior-to-Interior Girder Load Ratios .............................................................................. 128
Figure 6.15: ANSYS Deflections vs. ASP and SGL Predictions for the Eno and Wilmington
St Bridges ......................................................................................................................... 129
Figure 6.16: SGL Predictions vs. SGLSL Predictions for Continuous Span Bridges with
Equal Exterior-to-Interior Girder Load Ratios .................................................................. 132
Figure 6.17: ANSYS Deflections vs. SGL and SGLSL Predictions for Bridge 10 .............. 133
Figure 6.18: SP Predictions vs. SGL Predictions for Comparison to Field Measured
Deflections .......................................................................................................................... 136
Figure 6.19: Field Measured Deflections vs. SP and SGL Predictions for US-29 ............. 137
Figure 6.20: ASP Predictions vs. SGL Predictions for Comparison to Field Measured
Deflections .......................................................................................................................... 139
Figure 6.21: Field Measured Deflections vs. ASP and SGL Predictions for the Wilmington St
Bridge ................................................................................................................................. 140
Figure 6.22: SGLSL Predictions vs. SGL Predictions for Comparison to Field Measured
Deflections .......................................................................................................................... 142
Figure 6.23: Field Measured Deflections vs. SGLSL and SGL Predictions for Bridge 10
(Span B) ............................................................................................................................. 143
Figure 6.24: Field Measured Deflections vs. Predicted Deflections for Bridge 8 ............. 147
Figure 6.25: Field Measured Deflections vs. Predicted Deflections for the Avondale Bridge
........................................................................................................................................... 147
Figure 6.26: Field Measured Deflections vs. Predicted Deflections for the US-29 Bridge... 148
Figure 6.27: Field Measured Deflections vs. Predicted Deflections for the Camden NB
Bridge ................................................................................................................................. 148
Figure 6.28: Field Measured Deflections vs. Predicted Deflections for the Camden SB Bridge
........................................................................................................................................... 149
Figure 6.29: Field Measured Deflections vs. Predicted Deflections for the Eno Bridge..... 149
Figure 6.30: Field Measured Deflections vs. Predicted Deflections for the Wilmington St
Bridge ................................................................................................................................. 150
Figure 6.31: Field Measured Deflections vs. Predicted Deflections for Bridge 14 (Span B)150
Figure 6.32: Field Measured Deflections vs. Predicted Deflections for Bridge 10 (Span B)151
Figure 6.33: Field Measured Deflections vs. Predicted Deflections for Bridge 1 (Span B). 151
Figure 7.1: Simplified Procedure (SP) Application ................................................................ 156
Figure 7.2: Steps 1 and 2 of the Alternative Simplified Procedure (ASP) .............................. 158
Figure 7.3: Step 4 of the Alternative Simplified Procedure (ASP) ........................................ 159
Figure 7.4: Step 6 of the Alternative Simplified Procedure (ASP) ........................................ 160
Figure 7.5: SGL Straight Line (SGLSL) Application ............................................................... 161
Chapter 1

Introduction

1.1 Background

1.1.1 General

Many current and upcoming bridge construction projects in North Carolina incorporate steel plate girder bridges. Due to currently increasing site constraints, many of these bridges are being designed for longer spans at higher skews than in the past. In addition, they are being constructed in stages to maintain traffic flow on existing roadways. The development of higher strength steel allows for the design of longer spans with more slender cross-sections. As a result, the deflection of the girder is a more significant factor in the design. Therefore, it is important to accurately predict girder deflections so that desired vertical elevations are met.

Specifically, designers must accurately predict non-composite girder dead load deflections to produce the girder camber tables accordingly. The non-composite girder deflection is the deflection resulting from loads occurring during construction, prior to the curing of the concrete deck (i.e. prior to composite action between the steel girders and concrete deck). They include: girder self weight, other structural steel (cross frames, end bent diaphragms, connector plates, bearing stiffeners and web stiffeners), stay-in-place (SIP)
metal deck forms, deck reinforcement (rebar), and concrete deck slab. Additional dead loads during construction consist of the overhang falsework, deck concrete screeding machine, and construction live load (personnel). Some of these loads are temporary and the resulting elastic deflections are assumed to recover after unloading.

The North Carolina Department of Transportation (NCDOT) has experienced numerous problems in accurately predicting the non-composite girder deflections, resulting in many costly construction delays and maintenance and safety issues. As a result, the NCDOT has funded this research project (Project Number 2004-14). The primary goal of the research project is to develop a method to more accurately predict non-composite girder deflections of skewed and non-skewed steel plate girder bridges.

1.1.2 Current Analysis and Design

Typically, non-composite dead load deflections are predicted using single girder line (SGL) analysis. This method does not account for any transverse load distribution transmitted through intermediate cross frames and/or the SIP forms. The predicted deflection is directly dependent on the calculated dead load, which is determined according to the tributary width of the deck slab. If the girders are equally spaced, the interior girders are predicted to deflect the same and the exterior girders are predicted to deflect accordingly with the respective slab overhang dimension. A typical cross-section with girders, connector plates, cross frames, SIP forms, and the concrete deck is illustrated in Figure 1.1. Note that the tributary width used for prediction of an interior and exterior girder is dimensioned.
Figure 1.1: Traditional Single Girder Line Prediction Technique

Various construction issues may result from the use of traditional SGL analysis. When girders deflect less than expected, the deck slab and/or concrete covering the top layer of rebar may be too thin, resulting in rapid deck deterioration. When the girders deflect more than expected, dead loads are greater than accounted for in design.

Additionally, unexpected girder deflections may cause misaligned bridge decks during stage construction. During the first stage of construction, one half of the bridge superstructure is constructed while traffic is maintained on the existing structure. During the second stage, traffic is directed onto the first stage structure while the second half is being constructed. In the final stage, a closure strip is poured between the two stages. Figure 1.2 illustrates the differential deflection between construction phases as a result of inaccurate deflection predictions.
Misaligned bridge decks can cause numerous construction delays. For instance, the deck surface may require grinding to smooth the deck surface, which reduces the slab thickness and the cover concrete. The grinding maintenance could prove costly if the thinner deck causes a premature deterioration of the bridge deck.

1.1.3 Bridge Components

There are bridge components common to each of the bridges incorporated into this study. The bridges are comprised of steel plate girders, steel intermediate cross frames, steel end and interior bent diaphragms, reinforced concrete decks, and SIP metal deck forms. A discussion of each bridge component is included herein.

1.1.3.1 Steel Plate Girders and Intermediate Cross Frames

Steel plate girders consist of steel plates for each of the following: top flange, bottom flange, web, bearing stiffeners, intermediate web stiffeners, connector plates. Additionally, shear studs are welded to the top flange. Intermediate cross frames are steel members (typically structural tees or angles) utilized to laterally brace the plate girders along the span.
The steel plate girders, intermediate cross frames and intermediate web stiffeners are displayed in Figure 1.3.

![Figure 1.3: Steel Plate Girders, Intermediate Cross Frames and Intermediate Web Stiffeners](image)

1.1.3.2 End and Interior Bent Diaphragms

End and interior bent diaphragms consist of structural steel members utilized to laterally brace steel plate girders at supports. The diaphragm members are typically steel channels, structural tees and angles. An end bent diaphragm is presented in Figure 1.4. Note: interior bent diaphragms are commonly detailed identical to intermediate cross frames.
1.1.3.3  *SIP Metal Deck Forms*

SIP metal deck forms support wet concrete loads between adjacent girders during deck construction. The forms remain a bridge component throughout its lifespan, but are assumed to not provide vertical load support subsequent to the concrete curing. SIP forms are pictured in Figure 1.5 and Figure 1.6 illustrates a typical connection detail of the SIP forms to the top girder flange.

*Figure 1.4: End Bent Diaphragm*
1.1.3.4 Girder Bearing Types

Girder bearing supports are located between the bottom girder flange and the supporting abutment at the ends of the girders. Pot bearings and elastomeric bearing pads were utilized by the bridges in this study. Pot bearings (see Figure 1.7) can allow girder end
rotations, restrain all lateral movements, or allow lateral translation in one direction (along the length of the girder). Elastomeric bearing pads (see Figure 1.8) are capable of similar restrictions.

Figure 1.7: Pot Bearing Support

Figure 1.8: Elastomeric Bearing Pad Support
1.1.4 *Equivalent Skew Offset*

Skewed bridges are defined as bridges with support abutments constructed at angles other than 90 degrees (in plan view) from the longitudinal centerline of the girders. Depending on the direction of stationing, a bridge may be defined with an angle less than, equal to, or greater than 90 degrees (see Figure 1.9).
Figure 1.9: Skew Angle and Bridge Orientation (Plan View)
An equivalent skew offset has been defined for this research so that bridges defined with skews less than 90 degrees may be compared directly to bridges defined with skews greater than 90 degrees. The equivalent skew offset, $\theta$, is calculated by Equation 1.1 and the result defines the skew severity (i.e. the larger the number, the more severe the skew). Note that if the skew angle (via the bridge construction plans) was equal to 90, the equivalent skew offset would be equal to zero.

$$\theta = |\text{skew} - 90|$$  \hspace{1cm} (eq 1.1)

where: $\text{skew}$ = skew angle defined in bridge plans

1.2 **Objective and Scope**

The primary objective of this research is to develop a simplified method to predict dead load deflections of skewed and non-skewed steel plate girder bridges by completing the following tasks:

- Measure girder deflections in the field during the concrete deck placement.
- Develop three-dimensional finite element models to simulate deflections measured in the field.
- Utilize finite element models to conduct a parametric study for evaluating key parameters and their effect on non-composite deflection behavior.
- Develop the simplified procedure from the results of the parametric study.
- Verify the method by comparing all predicted deflection to those measured in the field.
The research project has been completed in two phases, the first of which was conducted by Whisenhunt (2004). During the first phase, deflections were measured for five, simple span, steel plate girder bridges and the finite element modeling technique was developed.

In this thesis, the second phase of the research project is reported. Field measured deflections have been recorded for three additional simple span bridges, two two-span continuous bridges and one three-span continuous bridge. A preprocessor program was developed in MATLAB to automate the generation of finite element models and to provide the means necessary to conduct an extensive parametric study. The parametric study investigated skew angle, exterior-to-interior girder load ratio, girder spacing, span length, cross frame stiffness, and number of girders to establish their effects on bridge deflection behavior. Finally, the simplified procedure was developed to predict steel plate girder dead load deflections.

1.3 Outline of Thesis

The following is a brief outline of the major topics covered in this thesis:

- Chapter 2 presents a literature review that summarizes previous research regarding the first research phase, bridge construction issues as related to bridge parameters, parametric studies and preprocessor programs for automated finite element generation.

- Chapter 3 presents descriptions of the bridges included in the study, the field measurement procedures implemented to monitor the bridges during construction, and a summary of the field measured deflections.
• Chapter 4 presents the detailed finite element modeling procedure, the development of the preprocessor program, and a summary of the simulated deflection results.

• Chapter 5 presents the parametric study, its results, and the development of the simplified procedure for simple span bridges with equal exterior-to-interior girder load ratios, simple span bridges with unequal exterior-to-interior girder load ratios, and continuous span bridges with equal exterior-to-interior girder load ratios.

• Chapter 6 presents the comparisons of field measured deflections to SGL predictions, ANSYS modeling predictions, and predictions from the developed simplified procedure.

• Chapter 7 presents observations and conclusions drawn from the research and recommendations made for predicting dead load deflections of skewed and non-skewed steel plate girder bridges.

• Appendix A presents a flow chart outlining the simplified procedure.

• Appendix B presents sample calculations utilizing the simplified procedure to predict girder deflections.

• Appendices C-G present the following for the five bridges monitored in this second research phase: a detailed description of the bridge components, elevation and plan view illustrations, a summary of non-composite field measured deflections, a description of the finite element model, and a summary of the deflections predicted by the finite element models.

• Appendix H presents the MATLAB source files along with input files for the ten studied bridges.
Chapter 2

Literature Review

2.1 Introduction

The research presented in this thesis is a continuation of the initial research conducted by Whisenhunt (2004). During this first phase by Whisenhunt (2004), an extensive literature review was completed regarding construction issues, bridge parameters and bridge modeling techniques. During this second phase, additional literature sources have been reviewed regarding the conclusions reached in Whisenhunt’s thesis, parametric studies and preprocessor programs utilized to automate the generation of finite element models. Additionally, supplemental research conducted by Paoinchantara (2005) is summarized.

2.2 Phase I Research

A detailed discussion of the subjects researched during the first phase of this project is included in Whisenhunt (2004); a summary is discussed herein.

2.2.1 Construction Issues

Hilton (1972) stated that traditional assumptions made to predict non-composite dead load deflections do not consider load sharing capabilities of the intermediate cross frames. Resulting predictions tend to be larger than what is measured in the field on account of the bridge superstructure deflecting as a unit, rather than individual girders.

Swett (1998) and Swett et al. (2000) concluded that the combination of twisting and vertical displacement in skewed bridges may cause conflicting final deck elevations during stage construction. AASHTO/NSBA (2002) stated that, when girders are braced with cross frames, transverse web rotations are a bigger problem on account of the increased use of
lighter weight, higher strength steel. As a result, AASHTO/NSBA (2002) states that girders installed vertically out of plumb may compensate for the rotation during construction, but the effects of differential deflections and girder rotations in skewed, curved and stage constructed bridges should be considered.

Norton (2001) and Norton et al. (2003) completed a study on a skewed, simple span bridge in which the girders were erected out of plumb prior to construction. This was to compensate for the expected rotation during construction, but the results revealed non-vertical webs at completion.

Staged construction problems were presented in ACI (1992), Swett (1998) and Swett et al. (2000). ACI (1992) stated that stages should remain separate prior to the closure strip pour as cross frame connections could result in the overloading of the stage I structure. Swett (1998) and Swett et al. (2000) analyzed stresses and deflections for six proposed stage construction methods by correlating a finite element model to the field measured deflections of a steel girder bridge. Three methods connected the two stages via cross frames prior to stage II construction and three did not. They concluded that either method type may be applicable, but when the stages are connected prior to stage II construction, unwanted stresses are introduced to the stage I girders and when the stages are not connected, the differential deflections between stages are usually undesirable.

2.2.2 Parameters

Gupta and Kumar (1983) concluded that bridges with equivalent skew offsets (see Chapter 1) of greater than 30 degrees should be carefully analyzed. Similarly, Bakht (1988), Bishara (1993), and Bishara and Elmir (1990) concluded the same for bridges with equivalent skew offsets greater than 20 degrees. Further, Bakht (1988) proposed that bridges
having \((S \tan \Phi/L)\) less than 0.05 can be analyzed as a non-skewed bridge, where \(S\), \(L\), and \(\Phi\) are the girder spacing, bridge span, and angle of skew, respectively. Additionally, Bishara and Elmir (1990) stated that increased differential deflections between adjacent girders increases the internal forces in the cross frames, all of which is caused by increasing the skew offset.

Bishara (1993), Chen et al. (1986) stated that intermediate cross frames provide load-sharing capabilities, which are usually unaccounted for during design. According to Helwig and Wang (2003) and Keating and Alan (1992), oversized cross frame bracing attracts large live load forces, and, thus, leads to fatigue problems at the cross frame to girder connections.

Currah (1993), Soderberg (1994), Helwig (1994) and Jetann et al. (2002) studied the lateral bracing ability of stay-in-place (SIP) metal deck forms. Currah (1993) concluded that the flexibility of supporting angles (used to connect the SIP forms to the girders) must be considered if the SIP forms are utilized as bracing elements. Soderberg (1994) and Jetann et al. (2002) focused on improving connection details between the girders and SIP forms. Helwig (1994) stated that SIP forms provide continuous lateral bracing and his finite element results proved that the presence of SIP forms allows for less required cross frames along a span.

2.2.3 Finite Element Modeling

SAP90 was utilized by Hays et al. (1986), Brockenbrough (1986), Tarhini et al. (1995), Mabsout et al. (1997a, 1997b), and Mabsout et al. (1998). Concrete slabs were modeled with quadrilateral shell elements, girders with space frame elements or shell elements, and the connection between the slab and girders with rigid link elements.
Bishara and Elmir (1990) Bishara et al. (1993) utilized the finite element program ADINA to investigate simple span steel bridges. The models consisted of triangular plate elements for the concrete deck, beam elements for the girder flanges, rigid links, and cross frames, and shell elements for the girder web.

Tarhini and Frederick (1992) used ICES-STRUDL II to model the concrete slab with eight node, isotropic brick elements and the girder web and flanges with quadrilateral shell elements. Tarhini et al. (1995) and Mabsout et al. (1997a) adopted the modeling method to evaluate wheel load distribution factors of steel girder bridges.

Finite element models were generated by Ebeido and Kennedy (1996) in ABAQUS. The concrete deck was modeled utilizing shell elements and the girders and diaphragms were modeled using 3-D beam elements. The shear connection between the slab and girders was modeled by employing a multipoint constraint equation.

ANSYS has been utilized for finite element analysis by various researchers, including but not limited to: Schilling (1982), Helwig (1994), Sahajwani (1995), Tabsh and Sahajwani (1997), Shi (1997), Helwig and Yura (2003), Helwig and Wang (2003), and Egilmez et al. (2003). In the finite element models, beam elements or shell elements were utilized for the girder web and flanges. Similarly, triangular isoparametric plate elements or rectangular shell elements were utilized for the concrete deck, beam elements for rigid links between the slab and girders, beam and truss elements for the cross frames, and shell, truss or beam elements for SIP forms.

incorporated by Berglund and Shultz (2001) included shell elements (girder web and concrete deck), frame elements (girder flanges and diaphragms), and rigid elements (composite shear connection between girders and deck). Finally, Norton et al. (2003) employed frame elements for the girder flanges, stiffeners and cross frames and shell elements for the girder web and concrete deck.

2.3 Phase II Research

Whisenhunt (2004) completed the first project phase and Paoinchantara (2005) completed a supplementary phase and reviews of their works are discussed herein. Additionally, works on parametric studies and preprocessor programs for finite element analysis were researched during this second phase and are included.

2.3.1 Reviews of Whisenhunt (2004) and Paoinchantara (2005)

Whisenhunt (2004) measured non-composite dead load deflections for five simple span, steel plate girder bridges during concrete deck construction. The deflection results were compared to traditional single girder line (SGL) predictions and it was concluded that SGL models do not accurately predict non-composite deflections. Further, he concluded that SGL analysis over predicts the interior and exterior girders of steel plate girder bridges of any skew angle by approximately 39 and 6 percent, respectively.

Next, finite element models were generated in ANSYS to more accurately predict the actual non-composite deflections of steel plate girder bridges. The simulated deflections from the models were compared to the field measured deflections and he concluded that (a) the created finite element models predicted a very comparable behavior, and (b) stay-in-place (SIP) metal deck forms can have a significant effect on the non-composite behavior of skewed plate girder bridges. Further, the finite element models predicted deflections within
6-14 percent of the field measured deflections for the five studied bridges and the models were considered adequate to be utilized in the simplified procedure development of phase II.

Paoinchantara (2005) implemented a finite element modeling technique in the commercially available structural analysis program SAP2000. The objective of his research was to develop a simplified modeling method to predict dead load deflections of skewed and non-skewed steel plate girder bridges. Paoinchantara (2005) concluded that girder deflection predictions from the simplified models correlated well with field measured deflections of simple span bridges, but not with continuous span bridges.

2.3.2 Parametric Studies

A large portion of this project has been dedicated to a parametric study, completed to determine which bridge parameters affect non-composite deflection behavior in steel plate girder bridges. A few related sources have been reviewed.

Bishara (1993) conducted a parametric study to evaluate internal cross frame forces in simple span, steel girder bridges. He investigated 36 finite element models of various configurations by varying skew angle, span length, deck width and cross frame spacing. As a result, Bishara (1993) developed a procedure to analyze internal cross frame forces with acceptable accuracy.

Bishara and Elmir (1990) investigated the interaction between cross frames and girders by generating multiple finite element models and varying skew angles and cross frame member sizes. He concluded: in skewed bridge models, the maximum compression force developed in a cross frame occurs at the exterior girder near the obtuse angle, and vertical deflections were insensitive to the size of the cross frame members.
Ebeido and Kennedy (1996) studied the influence of bridge skew on moment and shear distribution factors for simple span, skewed steel composite bridges. A finite element scheme was then implemented to derive expressions for the distribution factors.

Martin et al. (2000) conducted a parametric study to investigate the relative effects of various design parameters on the dynamic response of bridges. In the study, bridge characteristics (stiffness and mass) and loading parameters (magnitude, frequency, and vehicle speed) were varied. Martin et al. (2000) concluded that the most important factors affecting dynamic response are the basic flexibility (mass and stiffness) of the structure.

Buckler et al. (2000) investigated the effect of girder spacing on bridge deck response by varying the girder spacing and span length in finite element bridge models. It was concluded that increasing girder spacing can significantly increase both tensile and compressive stresses in the deck.

2.3.3 Preprocessor Programs

Manually generating or revising finite element bridge models can be a very time consuming task. It is beneficial to incorporate a preprocessor program to automate the model generation, especially when several models must be analyzed (as in a parametric study). The subsequent review includes sources related to this issue.

Austin et al. (1993) presents preprocessor software for generating three-dimensional finite element meshes, applying truck loadings, and specifying boundary conditions for straight, non-skewed highway bridges. The software, “XBUILD,” is written in the C programming language and creates input files in a format acceptable to the finite element analysis program ANSYS.
Barefoot et al. (1997) discusses a preprocessor program developed to model bridges with steel I-section girders and concrete deck slabs. The program is an ASCII batch file written in the ANSYS Parametric Design Language (ADPL) and allows efficient generation, and modification, of detailed finite element models in ANSYS.

Padur et al. (2002) describes a preprocessor program, “UCII Bridge Modeler,” that has been developed to automate the generation of steel stringer bridges in SAP90 or SAP2000. The program is written in Microsoft Visual Basic and is designed to accept user-defined input through a graphical user interface and to output a file formatted as input to SAP90.

2.4 Need for Research

Researchers have documented observed discrepancies to predicted behavior during bridge deck construction and some have recommended erection techniques as solutions. Research has produced numerous studies on bridge deflection behavior as affected by various parameters, such as skew angle and girder spacing. Additionally, there is a significant amount of research regarding various modeling techniques, parametric studies and preprocessor programs for finite element analysis. Overall, though, there is a limited amount of research related to predicting non-composite dead load deflections in steel plate girder bridges.

As part of this research, skew angle, cross frame stiffness, girder spacing, span length, number of girders and girder overhang will be investigated to establish relationships between them and non-composite dead load deflection behavior in steel plate girder bridges. The finite element modeling methods established by Helwig (1994), Egilmez et al. (2003), and
Helwig and Wang (2003) will utilized for analysis and a preprocessor program developed in MATLAB will automate the model generation.
Chapter 3

Field Measurement Procedure and Results

3.1 Introduction

As part of a combined research effort to study girder deflection behavior, five steel plate girder bridges were monitored during the concrete deck construction phase. The bridges include: two simple span bridges, two two-span continuous bridges, and one three-span continuous bridge. This chapter discusses the measured bridges, the field measurement procedure, and the measurement results.

3.2 General

Whisenhunt (2004) measured five simple span bridges; specific details and descriptions of each are included therein. The five additional bridges presented in this thesis increased the variance in both bridge type and geometry, and provided additional validation of the finite element models (see Chapter 4). Whisenhunt’s (2004) approach to discuss bridge descriptions, field measurement techniques, and measurement results were followed in this chapter.

3.3 Bridge Selection

The bridges selected for this project met certain criteria. The first obvious requirement was that the bridges were under construction during the field data collection phase of the project. Also, a range of geometric properties was desirable in order to observe different deflection behaviors during construction. Table 3.1 summarizes the targeted range of the geometric bridge properties considered in the bridge selection process.
Table 3.1: Targeted Range of Geometric Properties

<table>
<thead>
<tr>
<th>Bridge Property</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Span Type</td>
<td>Simple, 2-Span Cont., 3-Span Cont.</td>
</tr>
<tr>
<td>Equivalent Skew Offset</td>
<td>0 - 75 degrees</td>
</tr>
<tr>
<td>Number of Girders</td>
<td>4 - 12</td>
</tr>
<tr>
<td>Span Length</td>
<td>50 - 250 feet</td>
</tr>
<tr>
<td>Girder Spacing</td>
<td>6 - 12 feet</td>
</tr>
</tbody>
</table>

3.4 Bridges Studied

3.4.1 General Characteristics

There are characteristics common to all five bridges measured for this research phase. They are as follows:

- The steel plate girders are straight and connected by intermediate cross frames.
- Stay-in-place (SIP) metal deck forms were used to support fresh concrete during the deck placement.
- All structural steel (girders, cross frames, etc) is American Association of State Highway and Transportation Officials (AASHTO) M270 grade 50W steel.
- The concrete was cast parallel to the support abutment centerline (see Figure 3.1).

![Figure 3.1: Typical Concrete Placement on Skewed Bridge](image)

Figure 3.1: Typical Concrete Placement on Skewed Bridge
Four of the five bridges incorporated elastomeric bearing pads at the girder support locations. The settlements at these bearings were monitored and subtracted from the measured deflections within the span for direct correlation to the finite element analysis, which restrains vertical translation due to the modeled boundary conditions. Pot bearing supports were not monitored as deflections at this type of support are minimal.

Atypical to simple span bridges, sequence concrete pours are utilized for deck construction on most continuous span bridges (including all three in this study), in which the deck placement is completed in two or more separate pours. Specific characteristics of the five bridges, including pour sequence details, are included in Appendices C-G.

3.4.2 Specific Bridges

A complete list of the ten bridges included in the combined study is presented in Table 3.2, which includes the key parameters of each. The first seven bridges are simple span, listed by increasing equivalent skew offset, whereas the last three are continuous span, listed accordingly. Descriptions of the five bridges measured in part of this thesis are included herein.
Table 3.2: Summary of Bridges Measured

<table>
<thead>
<tr>
<th>Span Type</th>
<th>Number of Girders</th>
<th>Span Length (ft)</th>
<th>Girder Spacing (ft)</th>
<th>Nominal Skew Angle (deg)</th>
<th>Equivalent Skew Offset (deg)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eno*</td>
<td>Simple</td>
<td>5</td>
<td>236</td>
<td>9.6</td>
<td>90</td>
</tr>
<tr>
<td>Bridge 8</td>
<td>Simple</td>
<td>6</td>
<td>153</td>
<td>11.3</td>
<td>60</td>
</tr>
<tr>
<td>Avondale*</td>
<td>Simple</td>
<td>7</td>
<td>144</td>
<td>11.2</td>
<td>53</td>
</tr>
<tr>
<td>US-29*</td>
<td>Simple</td>
<td>7</td>
<td>124</td>
<td>7.8</td>
<td>46</td>
</tr>
<tr>
<td>Camden NB*</td>
<td>Simple</td>
<td>6</td>
<td>144</td>
<td>8.7</td>
<td>150</td>
</tr>
<tr>
<td>Camden SB*</td>
<td>Simple</td>
<td>7</td>
<td>144</td>
<td>8.7</td>
<td>150</td>
</tr>
<tr>
<td>Wilmington St</td>
<td>Simple</td>
<td>5</td>
<td>150</td>
<td>8.3</td>
<td>152</td>
</tr>
<tr>
<td>Bridge 14</td>
<td>2-Span Cont.</td>
<td>5</td>
<td>102, 106</td>
<td>10.0</td>
<td>66</td>
</tr>
<tr>
<td>Bridge 10</td>
<td>2-Span Cont.</td>
<td>4</td>
<td>156, 145</td>
<td>9.5</td>
<td>147</td>
</tr>
<tr>
<td>Bridge 1</td>
<td>3-Span Cont.</td>
<td>7</td>
<td>164, 234, 188</td>
<td>9.7</td>
<td>58</td>
</tr>
</tbody>
</table>

* from Whisenhunt (2004)

3.4.2.1 Bridge 8 (US 64 Bypass Eastbound over Smithfield Rd, Project # R-2547C)

Bridge 8 (see Figure 3.2) is located in Knightdale, North Carolina and is one of the two simple span structures included in this thesis. The site included two completely separate (but close to symmetric) bridges, one eastbound over Smithfield Rd and the other westbound. Only the eastbound structure was monitored and included in this study. Deflections were measured on this six girder bridge at three positions along the girder span, including the one-quarter point and three-quarter point locations. The third location was about 16.5 feet offset from the accurate mid-point location, due to traffic limitations on Smithfield Rd. The single deck placement lasted approximately 5 hours. Bridge 8 is illustrated in Figure 3.3.
3.4.2.2  Wilmington St Bridge (Wilmington St over Norfolk Southern Railroad, Project # B-3257)

The Wilmington St Bridge is a five girder, simple span bridge near downtown Raleigh, North Carolina (see Figure 3.4). The entire structure consists of three simple spans built in staged construction across the Norfolk Southern Railroad. The middle, southbound simple span was monitored for this investigation. Deflections were measured at the one-quarter point, the three-quarter point and at a location about 15 feet offset from the mid-span,
due to railway clearance restrictions. The deck placement for this bridge lasted approximately 5 hours. The Wilmington St Bridge is illustrated in Figure 3.5.

Figure 3.4: Wilmington St Bridge in Raleigh, North Carolina

Figure 3.5: Plan View Illustration of the Wilmington St Bridge (Not to Scale)

3.4.2.3 Bridge 14 (Bridge on Ramp RPBDY1 over US 64 Business, Project # R-2547CC)

Bridge 14 is a five girder, two-span continuous structure, also located in Knightdale, North Carolina (see Figure 3.6). For this structure, deflections were measured for all five girders at the following locations: the four-tenths point of Span A (predicted maximum
deflection point), the three-tenths point of Span B and the six-tenths point of Span B (predicted maximum deflection point). A two sequence concrete deck pour was utilized. The first pour lasted about 4 hours, whereas the second lasted close to 5 hours. Bridge 14 is illustrated in Figure 3.7.

![Figure 3.6: Bridge 14 in Knightdale, North Carolina](image1)

![Figure 3.7: Plan View Illustration of Bridge 14 (Not to Scale)](image2)
3.4.2.4  Bridge 10 (Knightdale Eagle Rock Rd over US 64 Bypass, Project # R-2547CC)

Bridge 10 is a four girder, two-span continuous structure located in Knightdale, North Carolina (see Figure 3.8). During construction, deflections were measured on all four girders at four separate locations along the span. These locations included the four-tenths (predicted maximum deflection point) and seven-tenths points of span B along with the two-tenths and six-tenths (predicted maximum deflection point) points of span C. The construction process involved a sequenced deck placement, the first and second pours taking about 2 and 7 hours to complete, respectively. Figure 3.9 is an illustration of Bridge 10.

Figure 3.8: Bridge 10 in Knightdale, North Carolina
3.4.2.5  Bridge 1 (Rogers Lane Extension over US 64 Bypass, Project # R-2547BB)

Bridge 1, in Raleigh NC (pictured in Figure 3.10), is unique to the study in that it is the only three-span continuous bridge monitored. The desired measurement locations were at the predicted maximum deflection points of all three spans; these were the four-tenths point of Span A, the mid-point of Span B and the six-tenths point of Span C. Due to Crabtree Creek below Span B and the Norfolk Southern Railroad below Span C, measurement points were offset from those locations. Span B was monitored at its four-tenths point, 23 feet from the mid-point and Span C was monitored at its thirty five-hundredths point, some 66 feet from the six-tenths point. The deck construction involved three separate concrete pours. Pours 1, 2 and 3 lasted about 4, 7 and 9 hours respectively. Figure 3.11 is an illustration of Bridge 1.
Figure 3.10: Bridge 1 in Raleigh, North Carolina

Girder Centerline: [Diagram showing girder centerline]
Measurement Location: [Diagram showing measurement location]

Span A = 164.09 ft (50.015 m)
Span B = 233.61 ft (71.205 m)
Span C = 188.28 ft (57.388 m)

Girder Spacing = 9.68 ft (2.95 m)

Figure 3.11: Plan View Illustration of Bridge 1 (Not to Scale)
3.5 Field Measurement

3.5.1 Overview

The instrumentation and measurement procedure utilized in this study was very similar to the procedure used by Whisenhunt (2004). Four of the bridges were monitored using the conventional technique while the Wilmington St Bridge was monitored using an alternate technique. Both measurement procedures are described herein.

3.5.2 Conventional Method

3.5.2.1 Instrumentation

String potentiometers were used to measure girder deflections during the concrete deck placement. The potentiometers were calibrated in the laboratory to establish the linear relationship between the output voltage and the distance traveled by the string. Utilizing this relationship, voltage readings recorded in the field were readily converted to deflections.

The string potentiometers were placed on a firm surface directly beneath measurement locations and connected to the bottom flange of the girder by way of steel extension wire. The wire was adjoined to the girder by securing it to a perforated steel angle clamped to the bottom flange with c-clamps. Also, small weights were tied to the wire between the girder and potentiometer to keep constant tension in the system. The string potentiometer, extension wire, and small weight are pictured in Figure 3.12. The perforated steel angle, c-clamps, and extension wire are pictured in Figure 3.13.
The potentiometers were connected to a switch and balance unit and a constant voltage power supply. A multimeter was used to read the voltage for each potentiometer.
connected to the switch and balance unit. The switch and balance units, power supply, and multimeter are pictured in Figure 3.14.

![Figure 3.14: Instrumentation: Switch & Balance, Power Supply, and Multimeter](Image)

Dial gages were positioned next to the girder bearings of each girder to monitor bearing settlements (see Figure 3.15). The dial gages are accurate to 0.0001 inches, well within the desired accuracy of this project.
3.5.2.2 Procedure

Voltage readings for each string potentiometer were recorded before, during and after the concrete deck placement, and the dial gage readings were typically recorded only before and after the deck placement. To ensure dependable readings, the calibration of each string potentiometer was checked against approximate manual tape measurements both before and after the concrete pour.

3.5.2.3 Potential Sources of Error

The string potentiometers used in this research are very sensitive and can relay very small variances in voltage. Small wind gusts or vibrations from nearby traffic may have caused such variances, though they were considered insignificant to the measured deflections.

During the hydration process, concrete in contact with the top flanges can reach temperatures much greater than that of the surrounding environment. It is possible for the
temperature gradient between the top and bottom flanges to decrease the dead load deflection as the top flange attempts to expand. Such variations are not accounted for in this research.

3.5.3 Alternate Method: Wilmington St Bridge

3.5.3.1 Instrumentation

Due to construction overlap with Bridge 1, the Wilmington St Bridge was monitored using an alternate method. Similar to the conventional method, the tell-tail method utilized a steel extension wire attached to a perforated steel angle, which was clamped to the bottom flanges with c-clamps (as pictured in Figure 3.13). Again, small weights were tied to the bottom of the extension wire to keep constant tension in the system. The weights themselves additionally served as elevation markers to measure the girder deflection. Wooden stakes were driven next to each suspended weight as a stationary measurement reference. The tell-tail setup including the suspended weight and the wooden stake is pictured in Figure 3.16.
3.5.3.2 Procedure

Deflections were measured by marking the wooden stakes at the bottom of the suspended weights as the bridge girders deflected. Measurements were recorded immediately prior to the concrete deck placement, at three instances during the pour, and after the entire deck had been cast. After gathering the wooden stakes, manual measurements were made in the laboratory to determine the magnitude of deflection each girder experienced. Note: the steel plate girders rested on pot bearings, thus, bearing settlements were not monitored during construction of the Wilmington St Bridge.
3.6 Summary of Measured Deflections

Table 3.3 summarizes the field measured deflections recorded for the five bridges included in this phase of the research. Deflections from the sequenced concrete pours were super-imposed for the continuous span structures and all tabulated deflections are in inches. Note that the girders are generically labeled A-G. Each bridge incorporates the appropriate labels depending on its number of girders. For instance, Bridge 10 only has four girders and they are labeled A-D, with girders A and D representing the exterior girders. Similarly, Bridge 1 has seven girders labeled A-G, with girders A and G now representing the exterior girders. For a given bridge, the dashed entries correspond to girders not monitored in the field and the boxes labeled “na” refer to nonexistent girders. As previously discussed, continuous span bridges have more than one location of predicted maximum deflection. Therefore, Table 3.3 includes the deflections at each of the predicted maximum deflection locations for all three continuous span bridges. A detailed deflection summary is available in Appendices C-G; included are deflection measurements of each pour sequence for the continuous span structures.

Table 3.3: Total Measured Vertical Deflection (inches)

<table>
<thead>
<tr>
<th>Span Location</th>
<th>Girder A</th>
<th>Girder B</th>
<th>Girder C</th>
<th>Girder D</th>
<th>Girder E</th>
<th>Girder F</th>
<th>Girder G</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bridge 8</td>
<td>Mid-Span</td>
<td>2.89</td>
<td>3.14</td>
<td>3.17</td>
<td>3.26</td>
<td>3.30</td>
<td>3.24</td>
</tr>
<tr>
<td>Wilmington St</td>
<td>Mid-Span</td>
<td>5.04</td>
<td>4.19</td>
<td>3.78</td>
<td>3.70</td>
<td>3.80</td>
<td>na</td>
</tr>
<tr>
<td>Bridge 14</td>
<td>4/10 Span A</td>
<td>0.87</td>
<td>0.79</td>
<td>0.97</td>
<td>0.85</td>
<td>0.51</td>
<td>na</td>
</tr>
<tr>
<td></td>
<td>6/10 Span B</td>
<td>1.55</td>
<td>1.45</td>
<td>1.66</td>
<td>1.50</td>
<td>1.64</td>
<td>na</td>
</tr>
<tr>
<td>Bridge 10</td>
<td>4/10 Span B</td>
<td>1.97</td>
<td>1.91</td>
<td>1.74</td>
<td>2.02</td>
<td>na</td>
<td>na</td>
</tr>
<tr>
<td></td>
<td>6/10 Span C</td>
<td>2.07</td>
<td>1.64</td>
<td>1.66</td>
<td>1.64</td>
<td>na</td>
<td>na</td>
</tr>
<tr>
<td>Bridge 1</td>
<td>4/10 Span A</td>
<td>1.99</td>
<td>1.73</td>
<td>-</td>
<td>1.53</td>
<td>-</td>
<td>1.77</td>
</tr>
<tr>
<td></td>
<td>4/10 Span B</td>
<td>4.59</td>
<td>4.38</td>
<td>-</td>
<td>4.18</td>
<td>-</td>
<td>3.99</td>
</tr>
<tr>
<td></td>
<td>35/100 Span C</td>
<td>1.27</td>
<td>1.13</td>
<td>-</td>
<td>1.21</td>
<td>-</td>
<td>1.41</td>
</tr>
</tbody>
</table>
The deflections from Table 3.3 were plotted and displayed in Figure 3.17. For clarity, only the “span B” deflections have been plotted for each continuous span bridge. It is apparent that there are five different bridge deflection behaviors for each of the five structures. The Wilmington St Bridge is the only bridge with unequal overhangs, thus unequal exterior girder loads. The inequality justifies the general slope from left to right, but not the “hat” shape observed. The other four deflected shapes appear essentially flat, with minor slopes for Bridge 8 and Bridge 1.

![Figure 3.17: Plot of Non-composite Deflections](image)

3.7 Summary

During this second research phase, five additional steel plate girder bridges have been monitored during the concrete deck placement. Of the five, two are simple span, two are two-span continuous and one is three-span continuous. The additional bridges increased the
variance of the parameters believed to directly contribute to each bridge’s deflection behavior during construction. Likewise, the additional data was used to further validate the finite element modeling, which is addressed in Chapter 4 along with details of the finite element modeling procedure and the automated generation process.
Chapter 4

Finite Element Modeling and Results

4.1 Introduction

Detailed finite element models of steel plate girder bridges have been created using the commercially available finite element analysis program ANSYS (ANSYS 2003). Initially, the models were developed to predict the bridge girder deflections which were compared to field measured values. Whisenhunt (2004) concluded that a complete finite element simulation predicts deflections better than the traditional single girder line (SGL) analysis.

With newfound confidence in the ability of ANSYS models to accurately predict non-composite girder deflections, a preprocessor program was developed in MATLAB to automate the procedure of processing detailed bridge information and generating commands to create the finite element models. The preprocessor program greatly reduced the time and effort spent generating the models and allowed for the administration of an extensive parametric study to determine which bridge components affect deflection behavior (see Chapter 5).

This chapter will discuss: the finite element models, the modeling procedure, the MATLAB preprocessor program, and modeling assumptions. Also included are the deflection results, predicted by the ANSYS models, for all five bridges measured in this second research phase.

4.2 General

Static analysis is used to determine structural displacements, stresses, strains, and forces caused by loads that do not generate significant inertia and damping effects (ANSYS
Therefore, without the presence of non-linear effects, the finite element bridge models of this research implement a static and linear analysis.

There are two linear elastic material property sets defined in each model, one for the structural steel and the other for the concrete deck. All structural steel elements are defined with an elastic modulus of 29,000 ksi (200,000 MPa) and a Poisson’s ratio of 0.3. The concrete elements are defined with an elastic modulus, \( E_c \), calculated by,

\[
E_c = 57,000 \sqrt{f'_{c}}
\]  \hspace{1cm} (eq 4.1)

where \( f'_{c} \) is the compressive strength of the concrete (in psi). The Poisson’s ratio for the concrete elements is defined as 0.2 as Whisenhunt (2004) concluded the models to be insensitive to adjustments of this ratio for concrete.

MATLAB is a matrix-based, high-level computing language commonly used to solve technical computing problems. MATLAB was chosen for this facet of the research project for the author’s familiarity of both MATLAB and the C programming language, which is closely related to the computing language incorporated into MATLAB. Statistically, output files are commonly between 2,000 and 6,000 lines of commands, while the MATLAB files programmed to generate the output consist of about 5,000 lines of code.

4.3 Bridge Components

The finite element models developed in this research include specifically detailed bridge components. Generally, these components include facets of the plate girders, the cross frames, the stay-in-place (SIP) metal deck forms and the concrete deck, each of which will be addressed in the following subsections. Note that in the subsequent discussion, a
centerline distance refers to the distance from the centerline of the top flange of the girder section to the centerline of the girder bottom flange.

4.3.1 Plate Girders

4.3.1.1 Girder

The plate girders are modeled by creating six keypoints to outline the geometric cross-section (web and flanges), according to actual centerline dimensions. To establish the entire girder framework, the keypoints are then copied to desired locations along to span and areas are generated between the keypoints. Figure 4.1 displays perspective and cross-section views of a single girder modeled in ANSYS.

Along a typical span, girder cross-sections vary in size. In developing a model, the centerline dimension is kept constant and defined by the section with the highest moment of
inertia. The section properties are then adjusted by applying real constant sets appropriately within ANSYS, i.e. changing the plate thicknesses. The constant centerline assumption differs from reality in that web depths are typically constant along the span. Therefore, the centerline dimension fluctuates as the flange thickness is changed. Whisenhunt (2004) conducted a sensitivity study and resolved that the centerline assumption has minimal effect on the bridge deflection behavior.

Whisenhunt (2004) also performed a sensitivity study to verify the ability of a single girder modeled in ANSYS to capture theoretically true deflections. Results affirmed that the girder model can accurately capture deflections for both simple and continuous span bridges.

4.3.1.2 Bearing Stiffeners, Intermediate Web Stiffeners, and Connector Plates

Bearing stiffeners, intermediate web stiffeners, and connector plates are typical of the ten studied bridges. Bearing stiffeners are present to stiffen the web at support bearing locations, intermediate web stiffeners are utilized for web stiffening along the span, and connector plates are used doubly as links between the intermediate cross frames and girder, and as additional web stiffeners.

The bearing stiffeners, intermediate web stiffeners, and connector plates are modeled by creating areas between web keypoints and keypoints at the flange edge. On the actual girders, stiffeners and plates are of constant width and rarely extend to the flange edge. It was confirmed by Whisenhunt (2004) that the finite element models are insensitive to this modeling assumption, which essentially fully welds the stiffeners and plates to the girder at the web and both flanges. Figure 4.2 displays oblique and cross-sectional views of bearing and intermediate web stiffeners.
4.3.1.3 *Finite Elements*

Eight-node shell elements (SHELL93) are utilized for each of the plate girder components, including: the girder, bearing stiffeners, intermediate web stiffeners and connector plates. The SHELL93 element has six degrees of freedom per node and includes shearing deformations (ANSYS 2003). Actual plate thicknesses are attained directly from the bridge construction plans and applied appropriately in the finite element models. Whisenhunt (2004) deemed a finite element mesh of approximately one foot square to be viable for convergence. Aspect ratios were checked and considered acceptable at values less than five; values greater than three are rarely present in the models. Element representations are available in Figures 4.1 and 4.2.
4.3.2 Cross Frames

4.3.2.1 General

Three different cross frames are common to bridges in the study: intermediate cross frames, end bent diaphragms and interior bent diaphragms. According to the AASHTO LRFD Bridge Design Specifications (2004), the aforementioned cross frames must: transfer lateral wind loads from the bottom of the girder to the deck to the bearings, support bottom flange in negative moment regions, stabilize the top flange before the deck has cured, and distribute the all vertical dead and live loads applied to the bridge.

Each cross frame is modeled by creating lines between the girder keypoints existing at the intersection of the web and flange centerlines. On the actual girders, the cross frame connections are offset from the flange to web intersection to allow for the connection bolts. This simplifying assumption has been shown to have little effect on the predicted girder deflection. The other assumption is that the cross frame member stiffnesses are very small relative to the girders themselves; therefore, the member connections are modeled as pins and are free to rotate about the joint.

In the finite element models, each cross frame member is modeled as a single line element. The cross frame member section properties were acquired from the AISC Manual of Steel Construction and applied directly into ANSYS.

4.3.2.2 Intermediate Cross Frames

Intermediate cross frames are utilized on all ten measured bridges and were erected perpendicular to the girder centerlines. The intermediate cross frame members are typically steel angles or structural tees between three and five inches in size and are bolted to the connector plates. X- and K-type cross frames are the two types associated with the studied
bridges and are illustrated in Figures 4.3a and 4.3b respectively. Whisenhunt (2004) conducted a small parametric study and determined that the type of cross frame utilized on each bridge has minimal effect on the bridge deflection behavior.

![Figure 4.3: Intermediate Cross Frames]

Intermediate cross frames are modeled with three-dimensional truss (LINK8) elements and three-dimensional beam (BEAM4) elements. LINK8 elements have two nodes
with three degrees of freedom at each, whereas BEAM4 elements are defined with two or three nodes and have six degrees of freedom at each (ANSYS 2003). LINK8 elements are utilized for each member of the X-type intermediate cross frame. For the K-type intermediate cross frame, BEAM4 elements are utilized for the bottom horizontal members and LINK8 elements are utilized for the diagonals. Figure 4.4 displays a characteristic ANSYS model with X-type intermediate cross frames.

4.3.2.3 End and Interior Bent Diaphragms

End bent diaphragms are utilized on nine of the ten measured bridges and were erected parallel to the abutment centerline. Bridge 14 includes integral bents and, therefore,
does not require end bent diaphragms. Figure 4.5 illustrates a typical end bent diaphragm with a large, horizontal steel channel section at the top and smaller steel angles or structural tees elsewhere. The other observed configuration included a short vertical member between the bottom horizontal member and central gusset plate (as was the case for Bridge 10 and the Wilmington St Bridge). The end bent diaphragms brace the girder ends, at or near the bearing stiffeners.

![Figure 4.5: End Bent Diaphragm](image)

Interior bent diaphragms are present on two of the three continuous span bridges (Bridge 14 and Bridge 1) and were also assembled parallel to the abutment centerline. In both cases, the diaphragms are exact duplicates of the intermediate cross frames, except that they are oriented differently and exist only at the interior supports. The other continuous span bridge (Bridge 10) utilizes intermediate cross frames directly at the interior bearing, perpendicular to the girder centerline; therefore, it does not utilize interior bent diaphragms.

End and interior bent diaphragms are modeled with LINK8 and BEAM4 elements. Typically, BEAM4 elements are utilized for horizontal members and LINK8 elements are
utilized for diagonal and vertical members. Figure 4.4 illustrates an end bent diaphragm for a
typical ANSYS finite element model.

4.3.3 Stay-in-Place Metal Deck Forms

4.3.3.1 General

Whisenhunt (2004) adopted a method to model the stay-in-place (SIP) metal deck
forms, previously developed by Helwig and Yura (2003). The method employs truss
members (diagonal and chord members) between the girders to represent the SIP form’s axial
stiffness. The approach allows the models to capture the true ability of the SIP forms to
transmit loads between girders.

Two small adjustments were made by Whisenhunt (2004) to the previously developed
method. First, a sensitivity study indicated small deflection variances in using two truss
diagonals instead of one. It was determined, however, that an x-brace system, with two
diagonals, serves better to represent both the SIP form shear stiffness and the direction of in-
plane shear transfer (Whisenhunt 2004). Second, truss elements are coupled with nodes
along the flange edge, rather than nodes at the web and flange intersection. The adjustment
is preferential and believed to more accurately depict the geometry of the SIP form
connection. The following includes a detailed discussion of the latter modeling modification.

4.3.3.2 Modeling Procedure

The SIP metal deck forms are modeled by direct generation. First, nodes are created
directly along the girder flange edge. Next, truss elements are appropriately generated
between the nodes to form the aforementioned x-braces. Finally, the generated nodes are
coupled with existing flange edge nodes to restrain lateral translations in all three global
directions. Rotational degrees of freedom are not restrained so that the truss elements can
rotate freely, thus accurately representing the flexible connection between the SIP forms and the girder flange.

LINK8 elements are utilized to model the SIP metal deck forms. Properties attributed to the truss elements were calculated following the extensive procedure described in Chapter 4 and Appendix G of Whisenhunt (2004). Figure 4.6 illustrates a close-up plan view of an ANSYS finite element model including the SIP form truss system.
4.3.4 Concrete Deck and Rigid Links

4.3.4.1 General

Finite element bridge models with concrete decks are required for composite analysis. During this research phase, composite analysis was only conducted on structures with sequenced pours, i.e. continuous span bridges.

4.3.4.2 Modeling Procedure

The concrete deck is modeled utilizing the same procedure as used for the plate girders. First, keypoints are created an offset distance above the top girder flange, at the centerline of the concrete slab. Areas are then generated to join the keypoints and create the simulated slab. Rigid link elements are then created between the existing keypoints of the slab and the existing keypoints of the girder (at the intersection of the web and top flange). The modeling approach is presented in Figure 4.7.

![Figure 4.7: Schematic of Applied Method to Model the Concrete Slab](image)

Four node shell (SHELL63) elements are utilized for the entire slab in the bridge models and two node rigid beam (MPC184) elements represent the links between the girders.
and slab to simulate composite behavior. For both element types, each node has six degrees of freedom. The thickness properties applied to the slab elements are attained directly from the bridge construction plans. The resulting shell element stiffness bears no consideration to the steel reinforcement or its possible bond development with the concrete. Figure 4.8 depicts a finite element model in which a bridge segment has been modeled as a composite section, complete with concrete slab elements. Note that the SIP forms are absent for clarity.

Figure 4.8: Finite Element Model Including a Segment of Concrete Deck Elements

4.4 Modeling Procedure

During the second phase of this research project, the large majority of finite element bridge models were created utilizing the MATLAB preprocessor program. The following discussion includes: automated model generation utilizing the MATLAB preprocessor program, MATLAB limitations, additional modeling performed manually, model adjustments specific to individual bridges, and MATLAB modeling validation. Note: the
4.4.1  Automated Model Generation Using MATLAB

4.4.1.1  General

The MATLAB preprocessor program is comprised of thirty eight files designed to collect data from bridge input files and generate two corresponding output files. A complete collection of these files is included in Appendix H. For each program run, the user modifies the single input file per detailed bridge plans and changes the output file names in ‘main.m’; both tasks are completed within MATLAB’s file editor window. Consequently, two MATLAB (*.m) output files are created and the user copies the commands in the output files and pastes them into the ANSYS command prompt window.

Two things were required to ensure an appropriate transition from MATLAB to ANSYS. First, it was imperative that the program adequately “write” commands to the output files so that ANSYS could process them. Second, the code files were programmed to output a specifically ordered command list to ensure the proper modeling technique (see Section 4.4.1.3).

4.4.1.2  Required Input

The MATLAB program requires many specific characteristics of each bridge as input, including:

- Skew angle
- Number of girders
- Girder spacing
• Slab overhang lengths (separately for each side)
• Girder span length (one for each span for continuous span bridges)
• Bridge type (simple span, two-span continuous, three-span continuous)
• Build-up concrete thickness
• Slab thickness
• Elastic Modulus and Poisson’s ratio of steel and concrete
• Field measurement locations
• Construction joint locations
• Number of girder sections and the z-coordinate location at which the section ends
• Width and thickness of the top and bottom flanges for each girder section
• Height and thickness of the web
• Number of bearing and intermediate web stiffeners, their thicknesses, and their z-coordinate location along the span
• Connector plate thickness, their spacing, and the z-coordinate location of the first one
• Type of intermediate cross frame, end bent diaphragm and interior bent diaphragm
• Areas and moments of inertia for all cross frame members
• SIP forms spacing
• SIP member areas
• SIP node couple tolerance

4.4.1.3 Generated Output

The MATLAB preprocessor program writes commands to two output files that are compatible with ANSYS. One output file is comprised of the commands to model the entire
bridge. The second includes the commands issued to model the SIP forms. As the output files are thousands of ANSYS command lines apiece, creating partitioned output files helped keep the information organized.

The output is generated to emulate a specific modeling procedure, listed as follows:

- Material property sets are defined for the steel and concrete.
- Finite element types are defined. (SHELL93, BEAM4, LINK8, etc)
- Real constant sets are defined, including: plate thicknesses, truss areas, beam moments of inertia, etc.
- Keypoints are created for the girders, web stiffeners and connector plates.
- Areas are generated between the keypoints to represent the girders, web stiffeners and connector plates.
- Keypoints and areas are created for the concrete slab.
- Attributes are applied to all of the modeled areas (attributes include the element type and real constant set); then they are sized appropriately and meshed to create the girder and slab elements.
- Rigid link lines are generated between the slab keypoints and the girder keypoints.
- Lines are created between existing and newly originated keypoints to generate all three cross frame types, as applicable.
- Attributes are applied to the modeled lines; then they are sized and meshed to create the rigid link and cross frame elements.
- SIP metal deck forms are directly generated with nodes and elements, thus requiring no sizing or meshing.
• Nodes of the SIP form are coupled laterally to existing top flange nodes and checked to ensure finite element compatibility.

4.4.2 MATLAB Limitations

As the MATLAB preprocessor program was developed exclusively for this project, the code was written only to handle variations in the geometric parameters found in the ten measured bridges. For instance, Bridge 1 is a long three-span continuous structure and the girder cross-section changes nine times along the entire span length. This is the maximum number of changes on any of the included bridges; therefore, the program can only manage nine cross-section variations. Other limitations are as follows:

• The bridge must have between four and ten girders.

• The structure must be either single span, two-span continuous or three-span continuous.

• Eight field measurement locations are allowed for deflection comparisons.

• The entire girder span must have a constant web depth.

• The intermediate cross frames must have an equal spacing for a given span.

• For structures with intermediate cross frames and/or interior bent diaphragms consisting of two horizontal members (top and bottom), both must be the same member.

• The entire program incorporates the metric system of units for all characteristic and dimensional properties, by preference of the author.
4.4.3 Additional Modeling and Consistency Checks

To complete each model, the loading conditions must be applied. First, the support boundary conditions are defined. The nodes along the bottom of the bearing stiffeners, which is the centerline of actual bearing, are restrained appropriately to simulate field boundary conditions. Pinned (or fixed) supports require restraints in all three translational directions and in rotational directions about the girder’s vertical and longitudinal axes. Roller (or expansion) supports are similarly modeled except that the nodes are allowed to translate along the girder’s longitudinal axis. In verifying this modeling technique, Whisenhunt (2004) analyzed the supports and found that although stress concentrations were present, they were below the yield stress. Additionally, girder dead loads are applied. To administer the user-defined loading, uniform pressures are applied to the top flange areas of the ANSYS model.

In addition to manually changing certain aspects of the finite element models, there is one component that must be inspected for consistency. As the SIP metal deck forms’ nodes are coupled to existing nodes of the top flanges, it is possible for other flange nodes to be located within the specified tolerance, resulting in a three-node couple rather than the desired two-node couple. If such coupled sets exist, a separate MATLAB file should be run to correct this problem. The generated output is copied from the MATLAB command window and pasted into the ANSYS command prompt window. Generally, an estimated 20 percent of the models created by the program require the coupled node sets to be revised.
4.4.4 Specific Modeling Adjustments

 Occasionally, the MATLAB program is unable to manage small anomalies between given bridges. As a result, minor modeling adjustments must be made manually to accurately represent the real structures. Two specific cases are subsequently discussed.

4.4.4.1 Bridge 10

 Bridge 10 is the only continuous span bridge without interior bent diaphragms. The MATLAB preprocessor program generates interior bent diaphragms for all continuous span structures; therefore, the interior bent diaphragms were manually deleted from the models.

4.4.4.2 Bridge 1

 Bridge 1 is the only bridge in the study with two different intermediate cross frame types along a given span. Both are X-type cross frames, but some have two horizontal members instead of one. The solution was to create all of the intermediate cross frames with two horizontal members and then manually delete the top member when not applicable.

4.4.5 Validation of ANSYS Models Generated with MATLAB

 To verify the ability of MATLAB generated ANSYS models to capture girder deflection behavior, comparisons were made to the Eno Bridge model created by Whisenhunt (2004). To ensure direct comparisons, girder loads applied to the MATLAB generated model were obtained directly from Appendix B in Whisenhunt (2004). After the ANSYS simulation, midspan deflections were correlated to those of Whisenhunt’s model, also obtained from Appendix B. Table 4.1 includes the deflection values of both models, as well as the ratio of the deflections from the Whisenhunt model to those of the model generated by
MATLAB. The deflection ratios establish that the deflections of the modeled girders differed by less than 3 percent.

### Table 4.1: Midspan Deflections and Ratios Comparing Eno River Bridge ANSYS Models

<table>
<thead>
<tr>
<th></th>
<th>Whisenhunt ANSYS Deflection (in)</th>
<th>MATLAB ANSYS Deflection (in)</th>
<th>Ratio (Whisenhunt/Matlab)</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>8.98</td>
<td>9.22</td>
<td>0.974</td>
</tr>
<tr>
<td>G2</td>
<td>8.67</td>
<td>8.89</td>
<td>0.975</td>
</tr>
<tr>
<td>G3</td>
<td>8.37</td>
<td>8.59</td>
<td>0.975</td>
</tr>
<tr>
<td>G4</td>
<td>8.06</td>
<td>8.28</td>
<td>0.973</td>
</tr>
<tr>
<td>G5</td>
<td>7.76</td>
<td>7.98</td>
<td>0.972</td>
</tr>
</tbody>
</table>

Similar ratios for each girder (see Table 4.1) signify matching deflected shapes between the two ANSYS models. Figure 4.9 presents the midspan girder deflections for both ANSYS models to illustrate the closely paralleled deflection behaviors, thus validating the MATLAB preprocessor program.
4.5 Modeling Assumptions

Whisenhunt (2004) includes an extensive and very detailed list of modeling assumptions accepted to allow for detailed bridge geometry while maintaining a practical modeling technique. Adapting the technique into the MATLAB preprocessor program required two additional modeling assumptions.

First, end bent diaphragms created by the program exist between keypoints located directly at the bearing stiffeners locations. On actual bridges, the bearing stiffeners are present at the location of bearing and the end bent diaphragms are bolted to nearby connector plates (which are slightly staggered in skewed bridges). This assumption removes the connector plates and attaches the end bent diaphragms directly to the bearing stiffeners. A sensitivity study was performed to establish the model’s responsiveness to the assumption; the results proved indifference – less than 1 percent.
The second assumption involves the intermediate cross frame spacing. Bridges occasionally have cross frames spaced unequally, usually due to girder splice constraints. To simplify the modeling procedure, it is assumed that the intermediate cross frames are always equally spaced. The spacing dimension is carefully defined to ensure that the model is closely correlated to the actual structure. As a result, the modeled cross frames are located at coordinates very near those in the real bridge and a sensitivity study resulted in extremely similar deflection behavior (again, less than 1 percent difference).

4.6 Deflection Results of ANSYS Models

The five bridges monitored during this research phase were initially modeled with and without the SIP metal deck forms. The model deflections were tabulated and graphed and are included in the following subsections. The tables incorporate total super-imposed deflections for the continuous span bridges at each location of predicted maximum deflection (see Section 3.3.2); only the midspan deflections are included for the simple span structures. A complete deflection summary for all ten models is available in Appendices C-G. Note that descriptions of generic girder labels and non-numerical table entries are addressed in Section 3.6.

4.6.1 No SIP Forms

Table 4.2 presents the girder deflection results for the ANSYS bridge models not including the SIP forms.
Table 4.2: ANSYS Predicted Deflections (No SIP Forms, Inches)

<table>
<thead>
<tr>
<th>Span Location</th>
<th>Bridge 8 Midspan</th>
<th>Girder A</th>
<th>Girder B</th>
<th>Girder C</th>
<th>Girder D</th>
<th>Girder E</th>
<th>Girder F</th>
<th>Girder G</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wilmington St</td>
<td>Midspan</td>
<td>2.74</td>
<td>3.41</td>
<td>3.60</td>
<td>3.56</td>
<td>3.72</td>
<td>na</td>
<td>na</td>
</tr>
<tr>
<td>Bridge 14</td>
<td>4/10 Span A</td>
<td>1.00</td>
<td>1.03</td>
<td>1.05</td>
<td>1.04</td>
<td>1.00</td>
<td>na</td>
<td>na</td>
</tr>
<tr>
<td></td>
<td>6/10 Span B</td>
<td>1.35</td>
<td>1.36</td>
<td>1.37</td>
<td>1.36</td>
<td>1.34</td>
<td>na</td>
<td>na</td>
</tr>
<tr>
<td>Bridge 10</td>
<td>4/10 Span B</td>
<td>1.79</td>
<td>1.78</td>
<td>1.82</td>
<td>1.92</td>
<td>na</td>
<td>na</td>
<td>na</td>
</tr>
<tr>
<td></td>
<td>6/10 Span C</td>
<td>1.26</td>
<td>1.18</td>
<td>1.14</td>
<td>1.15</td>
<td>na</td>
<td>na</td>
<td>na</td>
</tr>
<tr>
<td>Bridge 1</td>
<td>4/10 Span A</td>
<td>1.56</td>
<td>1.55</td>
<td>-</td>
<td>1.55</td>
<td>-</td>
<td>1.53</td>
<td>1.53</td>
</tr>
<tr>
<td></td>
<td>4/10 Span B</td>
<td>3.79</td>
<td>3.79</td>
<td>-</td>
<td>3.81</td>
<td>-</td>
<td>3.79</td>
<td>3.79</td>
</tr>
<tr>
<td></td>
<td>35/100 Span C</td>
<td>1.39</td>
<td>1.41</td>
<td>-</td>
<td>1.43</td>
<td>-</td>
<td>1.46</td>
<td>1.47</td>
</tr>
</tbody>
</table>

Midspan deflections of the simple span models and “span B” deflections of the continuous span models in Table 4.2 have been plotted in Figure 4.10. The deflected shapes of the continuous span models appear essentially straight. Contrastingly, the interior girders of Bridge 8 deflect more than the exterior girders. Unequal exterior girder loads on the Wilmington St Bridge model result in a slanted deflected shape, but the three leftmost girders (A-B-C) follow the general trend of Bridge 8.
4.6.2 Including SIP Forms

Table 4.3 presents ANSYS girder deflections for models including the SIP forms.

<table>
<thead>
<tr>
<th>Span Location</th>
<th>Girder A</th>
<th>Girder B</th>
<th>Girder C</th>
<th>Girder D</th>
<th>Girder E</th>
<th>Girder F</th>
<th>Girder G</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bridge 8</td>
<td>Midspan</td>
<td>4.11</td>
<td>4.14</td>
<td>4.17</td>
<td>4.17</td>
<td>4.14</td>
<td>4.11</td>
</tr>
<tr>
<td>Wilmington St</td>
<td>Midspan</td>
<td>3.20</td>
<td>3.02</td>
<td>3.03</td>
<td>3.28</td>
<td>3.86</td>
<td>na</td>
</tr>
<tr>
<td>Bridge 14</td>
<td>4/10 Span A</td>
<td>1.02</td>
<td>1.03</td>
<td>1.04</td>
<td>1.03</td>
<td>1.01</td>
<td>na</td>
</tr>
<tr>
<td></td>
<td>6/10 Span B</td>
<td>1.36</td>
<td>1.35</td>
<td>1.35</td>
<td>1.35</td>
<td>1.35</td>
<td>na</td>
</tr>
<tr>
<td>Bridge 10</td>
<td>4/10 Span B</td>
<td>1.74</td>
<td>1.72</td>
<td>1.80</td>
<td>1.97</td>
<td>na</td>
<td>na</td>
</tr>
<tr>
<td></td>
<td>6/10 Span C</td>
<td>1.30</td>
<td>1.17</td>
<td>1.11</td>
<td>1.11</td>
<td>na</td>
<td>na</td>
</tr>
<tr>
<td>Bridge 1</td>
<td>4/10 Span A</td>
<td>1.58</td>
<td>1.55</td>
<td>-</td>
<td>1.53</td>
<td>-</td>
<td>1.52</td>
</tr>
<tr>
<td></td>
<td>4/10 Span B</td>
<td>3.82</td>
<td>3.80</td>
<td>-</td>
<td>3.78</td>
<td>-</td>
<td>3.79</td>
</tr>
<tr>
<td></td>
<td>35/100 Span C</td>
<td>1.35</td>
<td>1.38</td>
<td>-</td>
<td>1.42</td>
<td>-</td>
<td>1.48</td>
</tr>
</tbody>
</table>

Once more, midspan and “span B” deflections in Table 4.3 have been plotted in Figure 4.11. Similar deflected shapes have remained for the continuous span models, but
differences exist in the deflected shapes of the simple span models. Although the interior girders of Bridge 8 continue to deflect more than the exterior girders, the deflected shape has flattened considerably. The most obvious deviation is apparent in the Wilmington St Bridge model as the shape has effectively flipped with the middle girder deflecting less than the exterior girders.

![Typical Cross Section](image)

**Figure 4.11: ANSYS Deflection Plot (Including SIP Forms)**

### 4.7 Summary

Finite element bridge models have been generated in ANSYS to simulate the dead load deflection response of skewed and non-skewed steel plate girder bridges. The modeling technique includes the following detailed bridge components: plate girders (girder, bearing stiffeners, intermediate web stiffeners, and connector plates), cross frames (intermediate cross frames, end bent diaphragms and interior bent diaphragms), SIP metal deck forms, and
the concrete deck. In generating the finite element models, several assumptions were made regarding the detailed bridge geometries in an effort to maintain a practical modeling technique. The resulting method was then applied to all five field measured bridges in this second research phase. Each bridge was created with and without the SIP forms and the results have been discussed.

To greatly reduce the time and effort spent modeling, a preprocessor program was developed in MATLAB and utilized to generate the finite element models in ANSYS. The program processes a single input file (modified by the user) and creates two individual output files. The output files contain model generation commands that are copied and pasted into ANSYS. Following a few additional adjustments, a detailed finite element model is ready for analysis.

Development of the MATLAB program to quickly generate finite element models proved very beneficial to the research project. Utilizing the preprocessor program, an extensive parametric study was conducted to analyze hundreds of very detailed finite element models. Chapter 5 presents a discussion on the parametric study and the development of the simplified method.
Chapter 5

Parametric Study and Development of the Simplified Procedure

5.1 Introduction

Utilizing the modeling technique and MATLAB preprocessor program described in Chapter 4, a parametric study was conducted to establish relationships between various bridge parameters and dead load deflections of skewed and non-skewed steel plate girder bridges. The controlling parameters were further investigated to develop a simplified procedure to predict the girder deflections. This chapter discusses detailed information of the parametric study and developing the simplified procedure. Despite the development’s focus on simple span bridges with equal exterior-to-interior girder load ratios, discussions on the deflection behavior of simple span bridges with unequal exterior-to-interior girder load ratios and continuous span bridges with equal exterior-to-interior girder load ratios are included.

5.2 General

Whisenhunt (2004) determined that stay-in-place (SIP) metal deck forms should be incorporated in the finite element models of this research project. It is reasoned that models with SIP form elements are more complete and better represent field measured deflections. Therefore, unless otherwise noted, bridge models discussed in the remainder of this thesis all incorporate SIP form elements.

Steel plate girder deflected shapes are described herein by the exterior girder deflection and the differential deflection between adjacent girders. Together, they can define the entire deflected shape at a given location along the span (i.e. deflections in cross-section). Figure 5.1 presents an example of the exterior girder deflection and differential deflection as defined in this thesis.
Also, the exterior-to-interior girder load ratio is defined in percent by dividing the exterior girder load by the interior girder load. For instance, the interior and exterior girders of Bridge 8 are loaded at 1.42 k/ft and 1.19 k/ft, respectively; thus, the exterior-to-interior girder load ratio is 84 percent. Last, a negative differential deflection between girders corresponds to an observed “hat” shape in cross-section (see deflections of the Wilmington St Bridge in Figure 4.11), whereas, a positive differential deflection corresponds to an observed “bowl” shape (see deflections of Bridge 8 in Figure 4.10).

5.3 Parametric Study

Five bridge parameters were investigated, either directly or indirectly, to help develop the simplified procedure for predicting dead load deflections of steel plate girder bridges. They are as follows: number of girders within the bridge span, cross frame stiffness, exterior-to-interior girder load ratio, skew offset of the bridge, and girder spacing-to-span ratio. Each parameter was investigated independently to discover any relationship that existed with the deflection of the girder.

![Figure 5.1: Exterior Girder Deflection and Differential Deflection](image)
5.3.1 Number of Girders

The number of girders within the span was investigated by creating ten finite element models using the Bridge 8 structure. Five girder arrangements were checked at two different skew offsets. The number of girders ranged from four to eight, whereas the skew offsets were set at 0 and 50 degrees. Figures 5.2 and 5.3 present the deflection results of the ANSYS models at the zero and fifty degree offsets, respectively.

![Figure 5.2: Bridge 8 at 0 Degree Skew Offset – Number of Girders Investigation](image)

![Figure 5.3: Bridge 8 at 50 Degrees Skew Offset – Number of Girders Investigation](image)
For models at the 0 degree skew offset, exterior girder deflections range from 4.38 to 4.44, a 1 percent difference of only 0.06 inches. At the 50 degree skew offset, the difference is 0.24 inches (from 4.06 to 4.30), which is an approximate 6 percent difference. For comparison, the differential deflection was averaged across the girders in each model. At the 0 degree skew offset, the differential deflection decreased only 0.07 inches as the number of girders was increased. Similarly, the decrease was 0.09 inches for the 50 degree skew offset models. Therefore, regardless of skew offset, the changes in exterior girder deflection and differential deflection are negligible.

5.3.2 Cross Frame Stiffness

Fourteen finite element models were generated to examine the effect of intermediate cross frame stiffness on deflection behavior. Bridge 8 was replicated with ten models: five select cross frame stiffnesses at 0 and 50 degree skew offsets. The cross frame stiffness was adjusted to represent one-tenth, one-quarter, one-half, one, and two times the original stiffness. Bridge 8 was chosen for this analysis because it has the maximum girder spacing, thus simulating the most extreme circumstances. Figures 5.4 and 5.5 represent the deflected shape of Bridge 8 at the 0 and 50 degree skew offsets, respectively, as cross frame stiffness is adjusted.
Additionally, the Eno Bridge was modeled four times, with stiffnesses adjusted to the extreme cases of one-tenth and two times the original stiffness at the 0 and 50 degree offsets. In this particular analysis, K-type intermediate cross frames replaced X-type cross frames (see Section 4.3.2.2) in the Eno Bridge models to verify the insignificance of cross frame type. Note that Eno was stage-constructed, thus unequal exterior-to-interior girder load ratios.
were present. Figures 5.6 and 5.7 represent the deflected shape of the Eno Bridge at the 0 and 50 degree offsets, respectively, for the two cross frame stiffnesses.

Figure 5.6: Eno at 0 Degree Skew Offset – Cross Frame Stiffness Investigation

Figure 5.7: Eno at 50 Degrees Skew Offset – Cross Frame Stiffness Investigation

The plotted results in all four figures indicate that variable cross frame stiffnesses have little effect on the non-composite deflection behavior of steel plate girder bridges. The maximum difference between exterior girder deflections at the two extreme cross frame
stiffnesses was 0.28 inches, a 6.5 percent difference (girder 6 of Bridge 8 at the 50 degree offset). The differential deflections appear to react slightly to stiffness adjustments, but not considerably enough. Note that in Figure 5.5, the differential deflection is positive for the 1/10 stiffness, whereas the other differentials are negative. In reality, steel angles are not manufactured small enough to achieve that cross frame stiffness.

### 5.3.3 Exterior-to-Interior Girder Load Ratio

Twenty-seven finite element models were generated to investigate how the exterior-to-interior girder load ratio affects steel plate girder deflection behavior. Three bridges (Camden SB, Eno, and Wilmington St) were modeled at 0, 50 and 60 degree skew offsets with equal exterior-to-interior girder load ratios of 50, 75 and 100 percent. The analysis revealed very similar results for all three bridges, therefore, only the Camden SB Bridge is discussed. Figures 5.8 and 5.9 represent the deflected shape of the Camden SB Bridge for the different exterior-to-interior girder load ratios at 0 and 50 degree skew offsets, respectively.

![Figure 5.8: Camden SB at 0 Degree Skew Offset – Exterior-to-Interior Girder Load Ratio Investigation](image-url)
It is apparent from the plots that exterior girder deflections and differential deflections between adjacent girders are both influenced by increased or decreased exterior-to-interior girder load ratios. For instance, doubling the exterior-to-interior girder load ratio in the 50 degree skew offset model causes girder 1 (an exterior girder) to deflect about 1 (0.98) additional inch and girder 4 (middle interior girder) to deflect an additional 0.33 inches (see Figure 5.8). The girder deflection behavior is affected because the exterior girders help carry the interior girder load by way of transverse load distribution. The relationship between exterior-to-interior girder load ratios and the deflection behavior required further investigation and a discussion is included.

5.3.4 Skew Offset

The skew offset parameter was analyzed by creating thirty-five finite element models. Each simple span bridge was modeled at skew offsets of 0, 25, 50, 60 and 75 degrees. After the analysis, it was evident that all seven bridges exhibited a common deflection behavior as
the skew offset was increased. To illustrate the effect of skew offset, deflections are displayed in Figure 5.10 for Bridge 8 and in Figure 5.11 for the Eno Bridge.

![Figure 5.10: Bridge 8 Midspan Deflections at Various Skew Offsets](image-url)
Figures 5.10 and 5.11 reveal a unique relationship between skew offset and girder deflection behavior. As the skew offset is increased, the exterior girders deflect less and the differential deflections become more negative. This relationship between skew offset and girder deflection behavior was further investigated.

5.3.5 Girder Spacing- to-Span Ratio

As four bridge parameters were investigated directly, an additional parameter was studied indirectly. The girder spacing-to-span ratio is a unitless parameter unique to each of the seven simple span bridges (see Table 5.1).
Table 5.1: Girder Spacing-to-Span Ratios

<table>
<thead>
<tr>
<th>Girder Spacing (ft)</th>
<th>Span Length (ft)</th>
<th>Spacing/Span Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eno</td>
<td>9.65</td>
<td>235.96</td>
</tr>
<tr>
<td>Wilmington St</td>
<td>8.25</td>
<td>149.50</td>
</tr>
<tr>
<td>Camden NB</td>
<td>8.69</td>
<td>144.25</td>
</tr>
<tr>
<td>Camden SB</td>
<td>8.69</td>
<td>144.25</td>
</tr>
<tr>
<td>US-29</td>
<td>7.74</td>
<td>123.83</td>
</tr>
<tr>
<td>Bridge 8</td>
<td>11.29</td>
<td>153.04</td>
</tr>
<tr>
<td>Avondale</td>
<td>11.19</td>
<td>142.96</td>
</tr>
</tbody>
</table>

To determine possible relationships between the girder spacing-to-span ratio and girder deflections, fourteen finite element models were generated: two models per bridge at 0 and 50 degree skew offsets, with an exterior-to-interior girder load ratio of 75 percent. As deflection magnitudes are primarily dependent on the magnitude of load, only differential deflections were compared to the girder spacing-to-span ratios. The results at 0 degree skew offset are plotted in Figure 5.12.
In Figure 5.12, the differential deflection value appears to increase in a linear fashion (displayed as a dashed line) as the girder spacing-to-span ratio is increased. The resulting relationship is considerable and investigated further.

5.3.6 Conclusions

Sections 5.3.1 – 5.3.5 present the results of a parametric study, conducted to determine the controlling bridge parameters affecting non-composite deflection behavior. Of the five parameters analyzed, the exterior-to-interior girder load ratio, skew offset, and the girder spacing-to-span ratio certainly influence girder deflections. Test results from the two studies involving number of girders within the span and cross frame stiffness did not produce significant changes in deflection behavior. Therefore, these two parameters are not included in the simplified procedure. Table 5.2 presents a matrix to summarize the entire parametric study and includes each parameter’s range of values. Note that checked cells indicate
referenced tests and shaded cells indicate repeated configurations. The number of girders within the span was not investigated against the girder spacing-to-span ratio as only one bridge (Bridge 8) was modeled with a varying number of girders. The results provided evidence that the number of girders within the span does not affect deflection behavior. Therefore, additional studies were not conducted for other bridge models.

<table>
<thead>
<tr>
<th>Table 5.2: Parametric Study Matrix</th>
</tr>
</thead>
<tbody>
<tr>
<td>Skew</td>
</tr>
<tr>
<td>Skew</td>
</tr>
<tr>
<td>Spacing/Span Ratio</td>
</tr>
<tr>
<td>Number of Girders</td>
</tr>
<tr>
<td>Exterior Girder Loading</td>
</tr>
<tr>
<td>Cross Frame Stiffness</td>
</tr>
<tr>
<td>Range of Values</td>
</tr>
</tbody>
</table>

5.4 Simplified Procedure Development

Developing the simplified procedure for predicting dead load steel plate girder deflections required a reasonable starting point. The traditional single girder line (SGL) prediction of an interior girder was deemed a reasonable base deflection on which to develop the simplified procedure for two reasons. First, SGL predictions involve simple calculations and are included in the majority of bridge design software. Second, an interior SGL prediction corresponds to an exterior SGL prediction with the exterior-to-interior girder load ratio at 100 percent and 0 degree skew offset, allowing for direct adjustments accordingly.
From the base prediction, a two-step approach was established to predict the deflection behavior. The first step is to predict the exterior girder deflections by adjusting the base prediction, while accounting for trends discovered in the parametric study. The second step is to utilize the predicted differential deflection, according to those same trends, to predict the interior girder deflections.

To implement this approach, specific relationships were established between the controlling parameters (skew offset, exterior-to-interior girder load ratio, and girder spacing-to-span ratio) and the girder deflection behavior by investigating the trends presented in Section 5.3. First, the effect of skew offset and exterior-to-interior girder load ratio on exterior girder deflections is addressed. Then, a discussion is presented regarding the differential deflection predictions, as influenced by all three key parameters.

5.4.1 Exterior Girder Deflections

5.4.1.1 Skew Offset

To investigate the skew offset, the exterior girder deflections at the 0 degree skew offset were divided by the corresponding deflection at the other skew offsets. The resulting ratio defined the change in deflection as the skew offset was increased. It is apparent in Figure 5.13 that plots of deflection ratio vs. skew offset followed a tangent function for each bridge. The A and B variables of the general tangent function, $A \tan(B\theta)$, were then adjusted to best fit the tangent function through the plots. Results indicated that values of 0.1 and 1.2 for A and B were appropriate up to around 65 degrees skew offset. Figure 5.13 includes plots of all seven simple span bridges and the fitted tangent function. Note that the tangent function is vertically offset one unit and aligned with the deflection ratio plots.
5.4.1.2 Exterior-to-Interior Girder Load Ratio

The exterior-to-interior girder load ratio was further studied by isolating the individual exterior girder deflections. Plotting the deflections vs. the exterior-to-interior girder load ratio revealed a linear relationship at all considered skew offset values (0, 50 and 60). Figure 5.14 presents the results for the Camden SB Bridge.
In the 0 degree skew offset model, the exterior girder deflection increases about 22 percent as the exterior-to-interior girder load ratio is increased from 50 to 100 percent, as shown in Figure 5.14. For Eno and Camden SB, the increase is 25 and 28 percent, respectively. It is apparent that the effect of exterior-to-interior girder load ratio on exterior girder deflections is dependent on additional variables.

To resolve the discrepancy, a multiplier variable was adapted into a spreadsheet analysis. The spreadsheet accounted for both the tangent relationship of the skew offset and the linear relationship of the exterior-to-interior girder load ratio. The multiplier was changed manually to match ANSYS modeling deflection results for every bridge, at various skew offsets, with a 75 percent exterior-to-interior girder load ratio. The multiplier values were tabulated and graphed vs. skew offset (see Figure 5.15).
Figure 5.15: Multiplier Analysis Results for Determining Exterior Girder Deflection

For the non-skewed models, the multiplier value averaged to 0.033 (labeled in Figure 5.15), and therefore, was set to 0.03 at 0 degree skew offset for all bridges. Because different behaviors transpired as the skew offset was increased, linear trend lines were plotted through each data set and their slopes were compared to other parameters. An applicable relationship exists between the trend line slope and girder spacing, as presented in Figure 5.16. The dashed line represents a fitted linear trend line between 2.5 and 3.5 meter girder spacing, with a slope of 0.001. The trend line slope value of 0.0002 is used at girder spacing less than or equal to 8.2 feet.
Therefore, the exterior girder deflection may be adjusted according the exterior-to-interior girder load ratio by also considering the girder spacing. The girder spacing determines the trend line slope value, which determines the multiplier value at a given skew offset. The multiplier is applied directly to the exterior-to-interior girder load ratio to restrain its effect on the exterior girder deflection.
5.4.1.3 Conclusive Results

A final equation to predict the exterior girder deflection was developed from the findings presented in the previous sections. The result is presented in Equation 5.1, accounting for skew offset and exterior-to-interior girder load ratio.

\[
\delta_{\text{EXT}} = [\delta_{\text{SGL_INT}} - \Phi(100 - L)][1 - 0.1\tan(1.2\theta)]
\]

(eq. 5.1)

where:

\[\delta_{\text{SGL_INT}} = \text{interior girder SGL predicted deflection at locations along the span (in)}\]

\[\Phi = 0.03 - a(\theta)\]

where:

\[a = 0.0002 \quad \text{if } (g \leq 8.2)\]

\[a = 0.0002 + 0.000305 (g - 8.2) \quad \text{if } (8.2 < g \leq 11.5)\]

where:

\[L = \text{exterior-to-interior girder load ratio (in percent, ex: 65 %)}\]

\[\theta = \text{skew offset (degrees)} = |\text{skew} - 90|\]

5.4.2 Differential Deflections

5.4.2.1 Skew Offset

The previously described procedure was repeated to determine the influence of skew offset on differential deflections. Instead of deflection ratios, the actual differential deflection values were reviewed; again, 0.1 and 1.2 for A and B were deemed appropriate for the tangent function up to a skew offset of about 65 degrees. The plot in Figure 5.17 displays the fitted tangent function (vertically offset down 0.05 units) and the differential deflections for all seven simple span bridges as the skew offset is increased.
5.4.2.2 Exterior-to-Interior Girder Load Ratio

Differential deflections were plotted vs. exterior-to-interior girder load ratios at various skew offsets to determine the relationship. Again, linear trends were observed in all three bridges (Eno, Wilmington St, and Camden SB), as shown for the Camden SB Bridge in Figure 5.18. As the exterior-to-interior girder load ratio is decreased, the differential deflection increases (i.e. produces more of a “bowl” shape).
For the three bridges, the change in differential deflection was analyzed vs. the girder spacing-to-span ratio, for 0 degree skew offset models, as the exterior-to-interior girder load ratio was decreased from 100 to 50 percent. Consequently, the differential deflection varied more for higher girder spacing-to-span ratios, following the trend displayed in Figure 5.12. Figure 5.19 presents the differential deflection increase vs. the girder spacing-to-span ratio for the three bridges, resulting from the decreased exterior-to-interior girder load ratio. Included is a linear trend line, fit to account for expected data point values for the other four simple span bridges (again, according to Figure 5.12). The slope value for the trend line was rounded up to ten (from about 9.3) because subsequent spreadsheet analysis revealed minimal change to the final differential deflection prediction as the slope value was varied.
Therefore, the amount of change in differential deflection, as the exterior-to-interior girder load ratio increases or decreases, is dependant upon the girder spacing-to-span ratio. Also, the minimal effect of changing the slope value applied in the equation reveals the minor, but considerable, influence of exterior-to-interior girder load ratio on differential deflection.

5.4.2.3 Girder Spacing-to-Span Ratio

Previously, Figure 5.12 presented the differential deflections vs. girder spacing-to-span ratios for 0 degree offset models. The results for the 50 degree offset models are displayed in Figure 5.20. The linear trend apparent in Figure 5.12 is no longer present in Figure 5.20, therefore, the effect of the girder spacing-to-span ratio is dependant on additional variables.
Figure 5.20: Differential Deflections at 50 Degrees Skew Offset as Related to the Girder Spacing-to-Span Ratio

Again, a multiplier variable was adapted into a spreadsheet analysis. Differential deflections were predicted in the spreadsheet, accounting for the skew offset and the exterior-to-interior girder load ratios, previously discussed. As previously described for the exterior girder deflection, the multiplier values were manually changed and the resulting multiplier values were graphed vs. skew offset (see Figure 5.21).
Figure 5.21: Multiplier Analysis Results for Determining Differential Deflection

Eno Bridge data is absent in Figure 5.21 on account of the inconsiderable effect of manually changing the multiplier value (i.e. small changes in differential deflection were observed for high ranges of multiplier values). For the non-skewed models of the remaining bridges, the multiplier value averaged to 2.98 (labeled in Figure 5.21), therefore set to 3.0 for all bridges. Distinct behaviors emerge as the skew offset is increased and, therefore, linear trend lines were plotted through the data sets and their slopes were set against other parameters. A useful relationship is present between the trend line slope and the girder spacing-to-span ratio, as presented in Figure 5.22. The dashed line represents a fitted linear trend line between the ratios of 0.05 and 0.08, with a slope of 8.0. The trend line slope value of -0.08 is used at ratio values less than or equal to 0.05.
Therefore, the differential deflection may account for the girder spacing-to-span ratio by reanalyzing the girder spacing-to-span ratio as the skew offset is increased. The ratio determines the trend line slope value, which determines the multiplier at a given skew offset, starting at 3.0 for non-skewed bridges. The multiplier is applied directly to the girder spacing-to-span ratio to determine its effect on the differential deflection.
5.4.2.4 Conclusive Results

A final equation to predict the differential deflection between adjacent girders was developed from the findings presented in the previous sections. The result is presented in Equation 5.2, accounting for skew offset, exterior-to-interior girder load ratio, and girder spacing-to-span ratio.

\[ D_{INT} = x[a(S - 0.04)(1 + z) - 0.1\tan(1.2\theta)] \]  

(eq. 5.2)

where:

\[ x = \frac{\delta_{SGL_{INT}}}{\delta_{SGL_M}} \]

where: \( \delta_{SGL_M} \) = SGL predicted girder deflection at midspan (in)

\[ \alpha = 3.0 - b(\theta) \]

where:

\[ b = \begin{cases} 0.08 & \text{if } S \leq 0.05 \\ -0.08 + 8(S - 0.05) & \text{if } 0.05 < S \leq 8.2 \end{cases} \]

where: \( S \) = girder spacing-to-span ratio

\[ z = (10(L - 0.04) + 0.02)(2 - L/50) \]

\[ \theta = \text{skew offset (degrees)} = |\text{skew} - 90| \]

The applied scalar variable, \( x \), scales the differential deflection by accounting for the location along the span. The maximum differential deflection occurs at the maximum deflection location (i.e. the midspan for simple span bridges). As the span approaches the support, the differential deflection is scaled proportional to the girder deflection at that location. For instance, the differential deflection at the quarter span is scaled by the ratio of quarter span deflection to midspan deflection. The deflections used to calculate the scalar, \( x \), should be obtained from simple SGL predictions.

To illustrate the scalar application, Figure 5.23 presents an example situation. Twentieth point deflections were calculated for a simple span bridge with a uniformly distributed load according to the AISC Manual of Steel Construction. The deflections were
divided by the midspan deflection and the ratios (i.e. the scalar variable) were plotted for half the span. Also included is an illustration of the span configuration. Note that the example is for girders with constant cross-section.

Figure 5.23: Scalar Values for Simple Span Bridge with Uniformly Distributed Load

A final note regarding the application of the differential deflection: through multiple spreadsheet analyses, it was apparent that the differential deflection should only be applied twice to adjacent girders. Therefore, in a seven girder bridge (girders labeled A-G), the girder A deflection is calculated with Equation 5.1, and then the deflections of girders B and C are calculated by adding the differential deflection predicted via Equation 5.2. Finally, the girder D deflection will simply equal that of girder C and the deflections of girders E, F, and G will equal to the deflections of girders C, B, and A respectively. The resulting predicted
deflected shape is symmetrical about a vertical axis through the middle of the cross-section. See the subsequent section and/or Appendix B for further explanation.

5.4.3 Example

To illustrate the entire simplified procedure, the deflections predicted by the simplified procedure were calculated and plotted for the US-29 Bridge in Figure 5.24, along with the SGL predicted deflections. First, the exterior girder deflection ($\delta_{\text{EXT}} = 4.73$ inches) was calculated according to the interior SGL deflection ($\delta_{\text{SGL},i} = 5.76$ inches). Next, the differential deflection ($D_{\text{INT}}$) was calculated as -0.085 inches and added twice to predict the adjacent girder deflections (as denoted in Figure 5.24). The predicted differential deflection is not added to the girder 3 prediction, and, therefore, the deflections of girders 3, 4 and 5 are equal (4.56 inches). Additionally, note that the deflected shape predicted by the simplified procedure is symmetrical about an imaginary vertical axis through girder 4. A more in depth example is presented in Appendix B with sample calculations.
5.4.4 Conclusions

The simplified development procedure involved generating two empirical equations. The first equation utilizes the traditional interior SGL prediction and adjusts the magnitude by considering the skew offset, the exterior-to-interior girder load ratio, and the girder spacing. The second equation predicts the differential deflection by accounting for the skew offset, the exterior-to-interior girder load ratio, the girder spacing-to-span ratio, and the span location. The detailed procedure is addressed in Chapter 7 and a flow chart is presented in Appendix A.

5.5 Additional Considerations

Thus far, the developed equations have exclusively accounted for simple span bridges with equal exterior-to-interior girder load ratios. Additional limited studies were conducted
to consider continuous span bridges with equal exterior-to-interior girder load ratios and simple span bridges with unequal exterior-to-interior girder load ratios.

### 5.5.1 Continuous Span Bridges

The effect of skew offset on deflection behavior was investigated for both two-span continuous bridges (Bridge 10 and Bridge 14) to determine if the developed equations are applicable to continuous span structures. Eight finite element models were generated: one model for each structure at 0, 25, 50, and 60 degree skew offsets. The resulting deflections were monitored at the locations of predicted maximum deflection (see Section 3.3.2). Figures 5.25 and 5.26 present deflections for Bridge 10, whereas Figures 5.27 and 5.28 present deflections for Bridge 14.

![Figure 5.25: Bridge 10 – Span B Deflections at Various Skew Offsets](image_url)
Figure 5.26: Bridge 10 – Span C Deflections at Various Skew Offsets

Figure 5.27: Bridge 14 – Span A Deflections at Various Skew Offsets
The illustrated behavior is dislike those observed for the simple span bridges, in which all girders deflected less as skew was increased (see Figure 5.10). For the continuous span bridges, one exterior girder deflects more as the skew offset is increased, while the other exterior girder deflects less. This behavior is caused by the interaction of a given span with the adjacent span.

Two prediction methods were investigated for two-span continuous bridges. They are: the traditional SGL method and an alternate SGL method. The alternate SGL method utilizes the exterior SGL deflections by connecting them with a straight line (i.e. the method predicts equal deflections for each girder, which is equal to the exterior SGL deflection); hence it is labeled the SGL straight line method (SGLSL method). A detailed procedure is addressed in Chapter 7 and a flow chart is presented in Appendix A. Note that the observed...
deflection behavior for continuous span bridges was inconsistent with simple span bridge behavior; therefore, the developed simplified procedure was not applicable.

5.5.2 Unequal Exterior-to-Interior Girder Load Ratios

Unequal exterior-to-interior girder load ratios were considered for the Eno Bridge and the Wilmington St Bridge. Eight finite element models were analyzed to check both bridges at skew offsets of 0, 25, 50, and 60 degrees. For the Eno Bridge, the exterior-to-interior girder load ratio for girders 1 and 5 were 94 percent and 74 percent, respectively (a 20 percent difference). For the Wilmington St Bridge, the ratio for girders 1 and 5 were 66 and 90 percent, respectively (a 24 percent difference). The results were graphed and it is apparent in Figures 5.29 (Eno) and 5.30 (Wilmington St) that an alternative procedure must be applied to aptly predict deflections for bridges with unequal exterior-to-interior girder load ratios.
Figure 5.29: Unequal Exterior-to-Interior Girder Load Ratio – Eno Bridge

Figure 5.30: Unequal Exterior-to-Interior Girder Load Ratio – Wilmington St Bridge
Several methods were investigated to predict deflections in bridges with unequal exterior-to-interior girder load ratios, all of which utilized the developed equations of the simplified procedure. The most appropriate technique involves calculating the exterior girder deflection (Equation 5.1) for the higher exterior-to-interior girder load ratio. Additionally, the exterior girder deflection and differential deflection (Equation 5.2) are calculated according the lower exterior-to-interior girder load ratio. The results are combined to predict a linear deflection behavior for simple span bridges with unequal exterior-to-interior girder load ratios. The procedure is tagged the alternative simplified procedure (ASP) and the details are discussed in Chapter 7 and a flow chart is presented in Appendix A.

5.6 Summary

An extensive parametric study was conducted to determine which bridge parameters influence steel plate girder deflections. During the study, about 200 finite element bridge models were built and analyzed, each with 200,000 – 250,000 degrees of freedom. It was discovered that skew offset, exterior-to-interior girder load ratio, and the girder spacing-to-span ratio all play key roles in the deflection behavior. Further investigation established relationships between the controlling parameters and the girder deflections. A bi-linear approach was developed to predict the non-composite dead load deflections for simple span bridges with equal exterior-to-interior girder load ratios (i.e. equal overhang dimensions). Additional limited studies were performed to account for continuous span bridges with equal exterior-to-interior girder load ratios and simple span bridges with unequal exterior-to-interior girder load ratios. Chapter 6 presents the results and comparisons of all observed deflection behaviors, including: field measurements, SGL analysis, ANSYS modeling, the
developed simplified procedure, alternative SGL analysis (for continuous span bridges) and the alternative simplified procedure (for unequal exterior-to-interior girder load ratios).
Chapter 6

Comparisons of Results

6.1 Introduction

The primary objective of this research is to develop a procedure to more accurately predict dead load deflections in skewed and non-skewed steel plate girder bridges. To show that this objective has been accomplished, multiple comparisons between field measured deflections, ANSYS predicted deflections, single girder line (SGL) predictions and other methods developed as a part of this research are presented. The detailed comparisons of the girder deflections presented in this chapter establish the necessity for an improved prediction method. The comparisons are presented in the following order:

- Field measured deflections are compared to SGL predicted deflections and ANSYS predicted deflections.
- ANSYS predicted deflections are compared to simplified procedure predictions and SGL predictions for simple span bridges with equal exterior-to-interior girder load ratios.
- ANSYS predicted deflections are compared to alternative simplified procedure (ASP) predictions and SGL predictions for simple span bridges with unequal exterior-to-interior girder load ratios.
- ANSYS predicted deflections are compared to SGL straight line (SGLSL) predictions and SGL predictions for continuous span bridges with equal exterior-to-interior girder load ratios.
• The newly developed predictions are compared to the field measured deflections for comparison, and to “close the loop”.

6.2 General

To compare deflection results, multiple statistical analyses have been performed on calculated deflection ratios throughout this chapter. The following statistics are included: average, minimum, maximum, standard deviation, and coefficient of variance. The latter two are included to evaluate the precision of the prediction methods. A low standard deviation and coefficient of variance signify a low variability in the data set (i.e. good precision). In the presented tables, the coefficient of variance is labeled COV and the standard deviation is St. Dev.

To illustrate the statistical analyses, several box plots have been incorporated. In the plots, the boxes represent the average ratio plus or minus one standard deviation; therefore, the darkest center band represents the average and standard deviation is expanded vertically up and down. The smaller (or tighter) the box, the better the precision in the data set. The plots also include ‘tails’ to designate the maximum and minimum ratio values.

In developing the simplified procedure to predict deflections, it was apparent that the deflection behavior of simple span bridges differs from that of continuous span bridges. Therefore, the results and comparisons are discussed individually for simple and continuous span bridges.

Finally, this chapter includes several deflection ratio tables with generic girder labels (‘Girders A’) and non-numeric data entries (‘-’ or ‘na’). A detailed discussion of these is included in Section 3.6.
6.3 Comparisons of Field Measured Deflections to Predicted Single Girder Line and ANSYS Deflections

Field measured deflections were compared to the predicted SGL and ANSYS deflections for all ten studied bridges. Initially, the field measured deflections were compared individually to the predicted SGL and ANSYS deflections by calculating the ratios of the predicted to measured deflections. The ratios were calculated for midspan deflections in the simple span bridges and at similar maximum deflection locations in the continuous span bridges. The ensuing statistical analysis contrasted the ratios to determine which deflections more accurately matched those measured in the field. The results are discussed herein.

6.3.1 Predicted Single Girder Line Deflections vs. Field Measured Deflections

Comparisons between the field measured deflections and predicted SGL deflections were made for all ten studied bridges. The details of the SGL predictions and the comparisons for simple and continuous span bridges are presented.

6.3.1.1 Single Girder Line Deflection Predictions

The structural analysis program SAP2000 was utilized to predict SGL deflections. Single girders were modeled with frame elements between nodes located at specific locations of cross-sectional variance, load bearing support, and field measurement location. Exact geometry was applied to the frame elements to accurately represent the bending properties of the steel plate girders. Additionally, the self weight of the frame elements was not included, and the effect of shearing deformation was included. Finally, non-composite dead loads were calculated from nominal dimensions presented in the construction plans, and applied to the SGL models for correlation. The deflection results confirmed the SGL models’ ability to
match the dead load deflections included in the bridge plans; thus, the models were deemed applicable for analysis.

6.3.1.2 Simple Span Bridges

Throughout the research study, it was apparent that the SGL predicted deflections were significantly greater than the field measured midspan deflections for simple span bridges. Figure 6.1 displays such an example for the Wilmington St Bridge. From the figure, the measured midspan deflection of G7 is approximately 3.5 inches less than predicted by the SGL method.

![Figure 6.1: SGL Predicted Deflections vs. Field Measured Deflections for the Wilmington St Bridge](image)

To gauge the amount of over prediction, the ratios of the predicted SGL deflections to field measured deflections were calculated for each girder of the seven simple span bridges
included in this study. The results are tabulated in Table 6.1; the bridges are listed in the order of increasing skew offset.

### Table 6.1: Ratios of SGL Predicted Deflections to Field Measured Deflections for Simple Span Bridges at Midspan

<table>
<thead>
<tr>
<th>Girder</th>
<th>Eno</th>
<th>Bridge 8</th>
<th>Avondale</th>
<th>US-29</th>
<th>Camden NB</th>
<th>Camden SB</th>
<th>Wilmington St</th>
</tr>
</thead>
<tbody>
<tr>
<td>Girder A</td>
<td>1.07</td>
<td>1.33</td>
<td>1.16</td>
<td>1.10</td>
<td>0.99</td>
<td>1.06</td>
<td>1.25</td>
</tr>
<tr>
<td>Girder B</td>
<td>1.19</td>
<td>1.46</td>
<td>1.30</td>
<td>1.39</td>
<td>1.53</td>
<td>1.62</td>
<td>1.94</td>
</tr>
<tr>
<td>Girder C</td>
<td>1.23</td>
<td>1.45</td>
<td>-</td>
<td>1.45</td>
<td>-</td>
<td>-</td>
<td>1.90</td>
</tr>
<tr>
<td>Girder D</td>
<td>1.29</td>
<td>1.41</td>
<td>1.26</td>
<td>-</td>
<td>1.45</td>
<td>1.45</td>
<td>1.71</td>
</tr>
<tr>
<td>Girder E</td>
<td>0.99</td>
<td>1.39</td>
<td>-</td>
<td>-</td>
<td>1.54</td>
<td>1.54</td>
<td>na</td>
</tr>
<tr>
<td>Girder F</td>
<td>na</td>
<td>1.19</td>
<td>1.20</td>
<td>1.34</td>
<td>1.02</td>
<td>1.62</td>
<td>na</td>
</tr>
<tr>
<td>Girder G</td>
<td>na</td>
<td>na</td>
<td>1.02</td>
<td>1.09</td>
<td>na</td>
<td>na</td>
<td>na</td>
</tr>
</tbody>
</table>

Only two data entries are slightly less than 1.0, revealing SGL deflections less than the field measured deflections (Girder A for Camden NB and Girder E of Eno). The deflection ratios tend to be greater for the interior girders than for the exterior girders. In Table 6.1, the average ratios are 1.12 and 1.46 for the exterior and interior girders respectively.

#### 6.3.1.3 Continuous Span Bridges

For the continuous span bridges, SGL models predict deflections greater and less than field measured deflections, with no clear trend (see Table 6.2). Figure 6.2 illustrates the SGL over prediction of span A and under prediction of span B in Bridge 1. The variance in behavior is likely due to the interaction of the adjacent continuous span.
The ratios of the predicted SGL deflections to field measured deflections were calculated for each girder of the three continuous span bridges. The results are tabulated in Table 6.2. For both two-span continuous bridges (Bridges 14 and 10), SGL deflections over predict the field measured deflections for one span and under predicts them for the other. For Bridge 1, Girders F and G are under predicted in all three spans, Girders A and B are under predicted in two of the three spans, and the middle girder (D) is under predicted only is Span B. Overall, the SGL deflections appear to predict deflections equally well for both the exterior and interior girders, with average ratios of 0.96 and 1.04 respectively.
Table 6.2: Ratios of SGL Predicted Deflections to Field Measured Deflections for Continuous Span Bridges

<table>
<thead>
<tr>
<th>Span Location</th>
<th>Girder A</th>
<th>Girder B</th>
<th>Girder C</th>
<th>Girder D</th>
<th>Girder E</th>
<th>Girder F</th>
<th>Girder G</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bridge 14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4/10 Span A</td>
<td>1.13</td>
<td>1.28</td>
<td>1.05</td>
<td>1.20</td>
<td>1.92</td>
<td>na</td>
<td>na</td>
</tr>
<tr>
<td>6/10 Span B</td>
<td>0.84</td>
<td>0.87</td>
<td>0.76</td>
<td>0.84</td>
<td>0.79</td>
<td>na</td>
<td>na</td>
</tr>
<tr>
<td>Bridge 10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4/10 Span B</td>
<td>1.10</td>
<td>1.27</td>
<td>1.40</td>
<td>1.07</td>
<td>na</td>
<td>na</td>
<td>na</td>
</tr>
<tr>
<td>6/10 Span C</td>
<td>0.69</td>
<td>0.98</td>
<td>0.96</td>
<td>0.87</td>
<td>na</td>
<td>na</td>
<td>na</td>
</tr>
<tr>
<td>Bridge 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4/10 Span A</td>
<td>0.80</td>
<td>0.99</td>
<td>-</td>
<td>1.11</td>
<td>-</td>
<td>0.96</td>
<td>0.78</td>
</tr>
<tr>
<td>4/10 Span B</td>
<td>0.78</td>
<td>0.88</td>
<td>-</td>
<td>0.92</td>
<td>-</td>
<td>0.97</td>
<td>0.91</td>
</tr>
<tr>
<td>35/100 Span C</td>
<td>1.05</td>
<td>1.24</td>
<td>-</td>
<td>1.16</td>
<td>-</td>
<td>0.99</td>
<td>0.77</td>
</tr>
</tbody>
</table>

6.3.2 ANSYS Predicted Deflections vs. Field Measured Deflections

ANSYS finite element models were generated for all ten studied bridges in an effort to improve predicted dead load deflections (the modeling technique is presented in Chapter 4). Comparisons of the field measured deflections to the ANSYS predicted deflections are discussed herein.

6.3.2.1 Simple Span Bridges

The predicted ANSYS deflections are greater than the field measured deflections at midspan in all but one of the simple span bridges. The under prediction is possibly due to partial composite behavior of the concrete deck slab during the concrete placement and/or temperature effects due to the curing of the concrete. Figure 6.3 presents the field measured deflections and the ANSYS predicted deflections at midspan for the US-29 Bridge.
A summary of the ratios of the ANSYS predicted deflections to field measured deflections is presented in Table 6.3. The ANSYS deflections for the Wilmington St Bridge under predict the field measured deflections by an average of 20 percent for the exterior and interior girders. Overall, the average deflection ratios for the exterior and interior girders are 1.11 and 1.07 respectively. Note that the bridges are listed in the order of increasing skew offset.
Table 6.3: Ratios of ANSYS Predicted Deflections to Field Measured Deflections for Simple Span Bridges at Midspan

<table>
<thead>
<tr>
<th></th>
<th>Girder A</th>
<th>Girder B</th>
<th>Girder C</th>
<th>Girder D</th>
<th>Girder E</th>
<th>Girder F</th>
<th>Girder G</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eno</td>
<td>1.08</td>
<td>1.11</td>
<td>1.14</td>
<td>1.17</td>
<td>1.22</td>
<td>na</td>
<td>na</td>
</tr>
<tr>
<td>Bridge 8</td>
<td>1.42</td>
<td>1.32</td>
<td>1.32</td>
<td>1.28</td>
<td>1.26</td>
<td>1.27</td>
<td>na</td>
</tr>
<tr>
<td>Avondale</td>
<td>1.12</td>
<td>1.09</td>
<td>-</td>
<td>1.08</td>
<td>-</td>
<td>1.04</td>
<td>1.04</td>
</tr>
<tr>
<td>US-29</td>
<td>1.02</td>
<td>1.07</td>
<td>-</td>
<td>1.12</td>
<td>-</td>
<td>1.04</td>
<td>0.97</td>
</tr>
<tr>
<td>Camden NB</td>
<td>1.24</td>
<td>1.10</td>
<td>-</td>
<td>1.01</td>
<td>1.11</td>
<td>1.28</td>
<td>na</td>
</tr>
<tr>
<td>Camden SB</td>
<td>1.14</td>
<td>1.01</td>
<td>-</td>
<td>0.94</td>
<td>-</td>
<td>1.00</td>
<td>1.15</td>
</tr>
<tr>
<td>Wilmington St</td>
<td>0.84</td>
<td>0.82</td>
<td>0.80</td>
<td>0.78</td>
<td>0.77</td>
<td>na</td>
<td>na</td>
</tr>
</tbody>
</table>

6.3.2.2 Continuous Span Bridges

For the continuous span bridges, the ANSYS predicted deflections were sometimes greater than and other times less than the field measured deflections. For instance, the ANSYS deflections were greater than the field measured deflections in span B of Bridge 14, and less in span A. Figure 6.4 includes the ANSYS predicted deflections and field measured deflections of spans B and C of Bridge 1.
The ratios of ANSYS deflections to field measured deflections were calculated for each girder in the three continuous span bridges. The results are tabulated in Table 6.4. Though the averages of the ratios are close to 1.0 for the exterior and interior girders (0.95 and 0.97 respectively), they alone are inadequate to assess the deflection correlations between ANSYS and the field measurements because the over predictions and under predictions, in effect, cancel each other out. A statistical analysis was performed to further investigate the correlations.
Table 6.4: Ratios of ANSYS Predicted Deflections to Field Measured Deflections for Continuous Span Bridges

<table>
<thead>
<tr>
<th>Span Location</th>
<th>Girder A</th>
<th>Girder B</th>
<th>Girder C</th>
<th>Girder D</th>
<th>Girder E</th>
<th>Girder F</th>
<th>Girder G</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bridge 14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4/10 Span A</td>
<td>1.17</td>
<td>1.30</td>
<td>1.08</td>
<td>1.22</td>
<td>1.97</td>
<td>na</td>
<td>na</td>
</tr>
<tr>
<td>6/10 Span B</td>
<td>0.88</td>
<td>0.93</td>
<td>0.82</td>
<td>0.90</td>
<td>0.82</td>
<td>na</td>
<td>na</td>
</tr>
<tr>
<td>Bridge 10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4/10 Span B</td>
<td>0.88</td>
<td>0.90</td>
<td>1.04</td>
<td>0.98</td>
<td>na</td>
<td>na</td>
<td>na</td>
</tr>
<tr>
<td>6/10 Span C</td>
<td>0.63</td>
<td>0.72</td>
<td>0.67</td>
<td>0.68</td>
<td>na</td>
<td>na</td>
<td>na</td>
</tr>
<tr>
<td>Bridge 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4/10 Span A</td>
<td>0.79</td>
<td>0.90</td>
<td>-</td>
<td>1.00</td>
<td>-</td>
<td>0.86</td>
<td>0.76</td>
</tr>
<tr>
<td>4/10 Span B</td>
<td>0.83</td>
<td>0.87</td>
<td>-</td>
<td>0.91</td>
<td>-</td>
<td>0.95</td>
<td>0.96</td>
</tr>
<tr>
<td>35/100 Span C</td>
<td>1.07</td>
<td>1.22</td>
<td>-</td>
<td>1.18</td>
<td>-</td>
<td>1.05</td>
<td>0.88</td>
</tr>
</tbody>
</table>

6.3.3 Single Girder Line Predicted Deflections vs. ANSYS Predicted Deflections

To thoroughly investigate the advantage of ANSYS modeling over traditional SGL analysis, statistical analyses were completed to compare the previously presented ratios. Box plots were created to illustrate a direct comparison of ANSYS and SGL deflection ratios. The results are presented first for simple span bridges and then for continuous span bridges.

6.3.3.1 Simple Span Bridges

The deflection ratios in Tables 6.1 and 6.3 were combined to conduct a statistical analysis for simple span bridges and the results are presented in Table 6.5. The results establish the advantage of ANSYS modeling over SGL analysis for the interior girders. The average ratio was lowered from 1.46 to 1.07 (39 percent more accurate) and the standard deviation was lowered from 0.20 to 0.15. It is apparent that the SGL analysis predicts exterior girder deflections more accurately than ANSYS. The average ratio was more accurate by 1 percent (from 1.12 to 1.11), and the SGL analysis exhibits better precision with a considerably lower standard deviation and coefficient of variance. A comparison is presented graphically in Figure 6.5 to confirm the observations.
Table 6.5: Statistical Analysis of Deflection Ratios at Midspan for Simple Span Bridges

<table>
<thead>
<tr>
<th></th>
<th>Exterior Girders</th>
<th>Interior Girders</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ANSYS/Measured</td>
<td>SGL/Measured</td>
</tr>
<tr>
<td>Average</td>
<td>1.11</td>
<td>1.12</td>
</tr>
<tr>
<td>Min</td>
<td>0.77</td>
<td>0.99</td>
</tr>
<tr>
<td>Max</td>
<td>1.42</td>
<td>1.33</td>
</tr>
<tr>
<td>St. Dev.</td>
<td>0.18</td>
<td>0.11</td>
</tr>
<tr>
<td>COV</td>
<td>0.16</td>
<td>0.10</td>
</tr>
<tr>
<td></td>
<td>1.07</td>
<td>1.46</td>
</tr>
<tr>
<td></td>
<td>0.78</td>
<td>1.19</td>
</tr>
<tr>
<td></td>
<td>1.32</td>
<td>1.94</td>
</tr>
<tr>
<td></td>
<td>0.15</td>
<td>0.20</td>
</tr>
<tr>
<td></td>
<td>0.14</td>
<td>0.14</td>
</tr>
</tbody>
</table>

Figure 6.5: ANSYS Predicted Deflections vs. SGL Predicted Deflections for Simple Span Bridges

6.3.3.2 Continuous Span Bridges

The deflection ratios in Tables 6.2 and 6.4 were combined to conduct a statistical analysis for continuous span bridges and the results are presented in Table 6.6. Comparable numbers in Table 6.6 reveal no clear advantage of one analysis over the other. For the exterior girders, the ANSYS and SGL average deflection ratios are 0.95 and 0.96.
respectively. Similarly, for the interior girders, the average deflection ratios are 0.97 and 1.04 respectively. Correspondingly, Figure 6.6 displays similar vertical spreads centered at similar average deflection ratios. Note that the large maximum deflection ratios for the exterior girders (1.97 and 1.92 for ANSYS and SGL respectively) result from small deflection magnitudes. For instance, the maximum deflection ratio for the ANSYS predicted deflections (1.97) correlates to an ANSYS prediction of 0.98 inches and a field measurement of 0.51 inches (a 0.47 inch difference).

<table>
<thead>
<tr>
<th></th>
<th>Exterior Girders</th>
<th>Interior Girders</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ANSYS/Measured</td>
<td>SGL/Measured</td>
</tr>
<tr>
<td>Average</td>
<td>0.95</td>
<td>0.96</td>
</tr>
<tr>
<td>Min</td>
<td>0.63</td>
<td>0.69</td>
</tr>
<tr>
<td>Max</td>
<td>1.97</td>
<td>1.92</td>
</tr>
<tr>
<td>St. Dev.</td>
<td>0.33</td>
<td>0.31</td>
</tr>
<tr>
<td>COV</td>
<td>0.34</td>
<td>0.32</td>
</tr>
</tbody>
</table>

Table 6.6: Statistical Analysis of Deflection Ratios for Continuous Span Bridges
Field measured deflections of the ten bridges included in this research were compared to SGL and ANSYS predicted deflections. Deflection plots quickly revealed the greater accuracy of ANSYS model predictions to the SGL analysis predictions in matching deflected shapes, for both simple and continuous span bridges. To compare the predictions, deflection ratios (SGL to field measured and ANSYS to field measured) were calculated for each bridge. A statistical analysis was performed on the ratios and the following conclusions were reached:

- ANSYS predicted deflections more closely match field measured deflections than SGL predicted deflections for the interior girders of the simple span bridges.

**Figure 6.6: ANSYS Predicted Deflections vs. SGL Predicted Deflections for Continuous Span Bridges**
• SGL predicted deflections more closely match field measured deflections than the ANSYS predicted deflections for the exterior girders of the simple span bridges.
• ANSYS modeling and the SGL method appear to predict field measured deflections equally well for the girders of the continuous span bridges.

6.4 Comparisons of ANSYS Predicted Deflections to Simplified Procedure Predictions and SGL Predictions for Simple Span Bridges with Equal Exterior-to-Interior Girder Load Ratios

6.4.1 General

The simplified procedure developed to predict dead load deflections utilizes two equations, as discussed in Chapter 5. The equations were derived from an extensive parametric study conducted to determine the key parameters affecting bridge deflection behavior. To ensure the equations’ ability to predict deflections, comparisons were made between the simplified procedure predictions and ANSYS predicted deflections at midspan. Additionally, SGL predictions were included to demonstrate the degree of improved accuracy.

For the comparisons discussed herein, the collection of ANSYS models included simple span bridges with equal exterior-to-interior girder load ratios (i.e. the two exterior girders were evenly loaded per bridge). These models incorporated multiple skew offsets, different of exterior-to-interior girder load ratios, and several girder spacing-to-span ratios. Girder loads were consistently altered during the parametric study; therefore, new SGL models were created for direct comparisons to the ANSYS predicted deflections and simplified procedure predictions.
6.4.2 Comparisons

Midspan deflection ratios were calculated to compare the ANSYS deflections to the simplified procedure predictions and the SGL predictions. The ratios were calculated as the prediction method’s deflections divided by the ANSYS predicted deflections. Accordingly, the ratios greater than 1.0 refer to an over prediction, and those less than 1.0 refer to an under prediction.

The calculated ratios were then broken down by various skew offsets to highlight the effect of skew offset on the behavior of the bridge. A statistical analysis was performed and the results are presented in Table 6.7 for both prediction methods at four skew offsets (0, 25, 50 and 60). Note that the results are presented individually for the exterior and interior girders and the simplified procedure reference is denoted as SP.
Table 6.7: Statistical Analysis Comparing SP Predictions to SGL Predictions at Various Skew Offsets

<table>
<thead>
<tr>
<th>Skew Offset</th>
<th>Exterior Girders</th>
<th>Interior Girders</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SP/ANSYS</td>
<td>SGL/ANSYS</td>
</tr>
<tr>
<td>0 Degree</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Average</td>
<td>1.00</td>
<td>0.86</td>
</tr>
<tr>
<td>Min</td>
<td>0.95</td>
<td>0.76</td>
</tr>
<tr>
<td>Max</td>
<td>1.05</td>
<td>0.96</td>
</tr>
<tr>
<td>St. Dev.</td>
<td>0.03</td>
<td>0.07</td>
</tr>
<tr>
<td>COV</td>
<td>0.03</td>
<td>0.08</td>
</tr>
<tr>
<td>25 Degree</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Average</td>
<td>1.03</td>
<td>0.89</td>
</tr>
<tr>
<td>Min</td>
<td>0.98</td>
<td>0.78</td>
</tr>
<tr>
<td>Max</td>
<td>1.09</td>
<td>0.99</td>
</tr>
<tr>
<td>St. Dev.</td>
<td>0.04</td>
<td>0.09</td>
</tr>
<tr>
<td>COV</td>
<td>0.04</td>
<td>0.10</td>
</tr>
<tr>
<td>50 Degree</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Average</td>
<td>1.08</td>
<td>1.01</td>
</tr>
<tr>
<td>Min</td>
<td>1.02</td>
<td>0.87</td>
</tr>
<tr>
<td>Max</td>
<td>1.15</td>
<td>1.13</td>
</tr>
<tr>
<td>St. Dev.</td>
<td>0.05</td>
<td>0.10</td>
</tr>
<tr>
<td>COV</td>
<td>0.05</td>
<td>0.10</td>
</tr>
<tr>
<td>60 Degree</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Average</td>
<td>1.00</td>
<td>1.10</td>
</tr>
<tr>
<td>Min</td>
<td>0.92</td>
<td>0.96</td>
</tr>
<tr>
<td>Max</td>
<td>1.08</td>
<td>1.25</td>
</tr>
<tr>
<td>St. Dev.</td>
<td>0.05</td>
<td>0.12</td>
</tr>
<tr>
<td>COV</td>
<td>0.05</td>
<td>0.11</td>
</tr>
</tbody>
</table>

As the skew offset is increased, it is apparent that the SGL predictions diminish, especially for the interior girders. For the interior girders, the average SGL deflection ratio
diverges from the ideal ratio of 1.0, while the average SP deflection ratio remains close to 1.0 (see Figure 6.7). For the interior and exterior girders, the average, standard deviation, and coefficient of variance all increase as the skew offset is increased. At the 60 degree skew offset, the average interior girder deflection ratio (1.68) signifies that the average interior SGL prediction is more than two-thirds greater than the corresponding ANSYS deflection. Additionally, the maximum interior girder deflection ratio is 1.96; this signifies an interior SGL prediction almost double that of the corresponding ANSYS deflection.

![Figure 6.7: Effect of Skew Offset on Deflection Ratio for Interior Girders of Simple Span Bridges](image)

Overall, the simplified procedure predictions more closely match the ANSYS predicted deflections than the SGL predictions. The standard deviations and coefficients of variance are less at all skew offsets, for the exterior and interior girders. Additionally, the
ratio averages at all four skew offsets are consistently close to 1.0 for the exterior and interior girders.

The results in Table 6.7 are displayed in the subsequent box plots to compare midspan deflection ratios of the SGL predictions to the simplified procedure predictions. Comparisons for the exterior girders are presented in Figures 6.8 and 6.9 and for the interior girders in Figures 6.10 and 6.11. Additionally, the midspan deflection ratios from the four skew offsets were combined to evaluate the overall prediction improvement and the resulting plot is presented in Figure 6.12.

![Figure 6.8: Exterior Girder SGL Predictions at Various Skew Offsets](image-url)
Figure 6.9: Exterior Girder Simplified Procedure Predictions at Various Skew Offsets

Figure 6.10: Interior Girder SGL Predictions at Various Skew Offsets
Figure 6.11: Interior Girder Simplified Procedure Predictions at Various Skew Offsets

Figure 6.12: Simplified Procedure Predictions vs. SGL Predictions
Figures 6.7 – 6.12 present further evidence that the simplified procedure predicts ANSYS deflections considerably better than traditional SGL predictions. In all cases, the vertical spreads are tighter and centered closer (or as close) to the ideal ratio of 1.0.

As an example to illustrate the improved predictions, Figure 6.13 presents the midspan deflection results for the Camden SB Bridge at 0 and 50 degree skew offsets. Again, SGL predictions do not change as skew offset is increased, as apparent in the figure. Note that in Figure 6.13, the number in parentheses beside the data set name refers to the skew offset and the simplified procedure prediction is denoted as ‘SP Prediction’. It is clear in the figure that the simplified procedure predicts ANSYS deflections significantly better than the SGL method. The deflected shapes predicted by the simplified procedure closely match the ANSYS deflected shapes for both skew offsets.

Figure 6.13: ANSYS Deflections vs. Simplified Procedure and SGL Predictions for the Camden SB Bridge
6.4.3 Summary

ANSYS predicted deflections were compared to simplified procedure predictions and SGL predictions for simple span bridges with equal exterior-to-interior girder load ratios. A statistical analysis was performed on midspan deflection ratios and the results were tabulated and plotted to demonstrate the improved accuracy of predicting dead load deflections by the simplified procedure. The primary conclusion is that deflections predicted by the simplified procedure are more accurate than SGL predicted deflections for exterior and interior girders at all skew offsets.

6.5 Comparisons of ANSYS Predicted Deflections to Alternative Simplified Procedure Predictions and SGL Predictions for Simple Span Bridges with Unequal Exterior-to-Interior Girder Load Ratios

6.5.1 General

The two equations developed for the simplified procedure are utilized for the alternative simplified procedure (ASP). The ASP method modifies the simplified procedure to predict deflections for simple span bridges with unequal exterior-to-interior girder load ratios. The result is a straight line prediction between the two exterior girder deflections.

To establish the ability of the ASP method to accurately capture deflection behavior, the predictions were compared to ANSYS predicted deflections at midspan. The Eno Bridge and the Wilmington St Bridge were modeled with unequal exterior-to-interior girder load ratios at skew offsets of 0, 25, 50 and 60 degrees. Additionally, SGL models of the two bridges were subjected to corresponding loads and analyzed for direct comparison with the ASP predictions and ANSYS predicted deflections. All comparisons are discussed herein.
6.5.2 Comparisons

The ASP and SGL predicted deflections were divided by the ANSYS predicted deflections at midspan for comparison. The corresponding ratios for all the models were combined and a statistical analysis was performed. It is apparent from the results (presented in Table 6.8) that the ASP predictions more closely match the exterior and interior ANSYS predicted deflections than the SGL predictions. For the interior girders, the average ASP deflection ratio (1.01) is closer than the SGL ratio (1.32) to the ideal ratio of 1.0 and better precision is exhibited. The average deflection ratios of the two prediction methods are comparable for the exterior girders, but the ASP method results in a lower standard deviation and coefficient of variance. The data in Table 6.8 is displayed graphically as box plots in Figure 6.14 to further validate the ASP prediction method.

Table 6.8: Statistical Analysis Comparing ASP Predictions to SGL Predictions

<table>
<thead>
<tr>
<th></th>
<th>Exterior Girders</th>
<th>Interior Girders</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ASP/ANSYS</td>
<td>SGL/ANSYS</td>
</tr>
<tr>
<td>Average</td>
<td>0.98</td>
<td>1.02</td>
</tr>
<tr>
<td>Min</td>
<td>0.83</td>
<td>0.83</td>
</tr>
<tr>
<td>Max</td>
<td>1.09</td>
<td>1.37</td>
</tr>
<tr>
<td>St. Dev.</td>
<td>0.07</td>
<td>0.15</td>
</tr>
<tr>
<td>COV</td>
<td>0.07</td>
<td>0.15</td>
</tr>
</tbody>
</table>
To illustrate the improved predictions, ANSYS predicted deflections at midspan were plotted against the corresponding ASP and SGL predictions for the Wilmington St Bridge at 50 degrees skew offset and for the Eno Bridge at 0 degree skew offset (see Figure 6.15). Note that the Wilmington St data sets are labeled ‘W’ in parentheses, whereas the Eno data sets are labeled ‘E’. The plots clearly display the ability of the ASP method to predict deflections for simple span bridges with unequal exterior-to-interior girder load ratios. The predictions are very accurate to the skewed and non-skewed ANSYS models, and the deflected shapes are much improved from the SGL predictions.
6.5.3 Summary

ANSYS deflections were compared to ASP and SGL predictions for simple span bridges with unequal exterior-to-interior girder load ratios by calculating deflection ratios at midspan. The ratios were subjected to a statistical analysis and the results pointed to significant advantages in utilizing ASP predictions. In direct comparison with SGL predictions, the ASP predictions were much more precise and deflected shapes more closely matched the ANSYS predicted deflections.
6.6 Comparisons of ANSYS Deflections to SGL Straight Line Predictions and SGL Predictions for Continuous Span Bridges with Equal Exterior-to-Interior Girder Load Ratios

6.6.1 General

Traditional SGL predictions are utilized for the SGL straight line (SGLSL) predictions. The SGLSL method simply predicts all girder deflections equal to the exterior SGL prediction. The SGLSL method is believed to more accurately predict ANSYS deflections for two reasons: exterior SGL predictions adequately match ANSYS predicted deflections, and deflected shapes for continuous span bridges are commonly flat (i.e. equal girder deflections in cross-section).

To establish the ability of the SGLSL method to accurately predict girder deflections, the predictions were compared to ANSYS predicted deflections and corresponding SGL predictions. Bridge 14 and Bridge 10 were modeled at skew offsets of 0, 25, 50 and 60 degrees, and the equal exterior-to-interior girder load ratios were 96 and 89 percent respectively. The comparisons are discussed herein.

6.6.2 Comparisons

SGLSL and SGL predicted deflections were divided by ANSYS predicted deflections to directly compare the methods. The corresponding ratios for all the models were combined and a statistical analysis was performed. Note that since the two methods predict identical exterior girder deflections, the exterior and interior girder ratios have been combined for this analysis. The results are presented in Table 6.9. It is apparent that SGLSL predictions are slightly more accurate than SGL predictions. The average is closer to 1.0 (1.02 compared to 1.06) and the standard deviation and coefficient of variance is lower for the SGLSL predictions. The data in Table 6.9 is displayed graphically in Figure 6.16 as a box plot.
Based on the behavior of simple span bridges, the SGL/ANSYS deflection ratios would likely deviate from 1.0 as the exterior-to-interior girder load ratio is decreased. In this analysis, both continuous span bridges have exterior-to-interior girder load ratios of 89 percent, or higher, resulting in relatively flat SGL predictions (see Figure 6.17). Further, it is likely that SGLSL/ANSYS deflection ratios would remain closer to 1.0 as the load is decreased as most ANSYS deflected shapes are essentially flat.

Table 6.9: Statistical Analysis Comparing SGL Predictions to SGLSL Predictions

<table>
<thead>
<tr>
<th></th>
<th>SGL Prediction/ ANSYS</th>
<th>SGLSL Prediction/ ANSYS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average</td>
<td>1.06</td>
<td>1.02</td>
</tr>
<tr>
<td>Min</td>
<td>0.86</td>
<td>0.86</td>
</tr>
<tr>
<td>Max</td>
<td>1.40</td>
<td>1.34</td>
</tr>
<tr>
<td>St. Dev.</td>
<td>0.10</td>
<td>0.08</td>
</tr>
<tr>
<td>COV</td>
<td>0.10</td>
<td>0.08</td>
</tr>
</tbody>
</table>
ANSYS predicted deflections, SGL predictions and SGLSL predictions have been plotted for Bridge 10 at 0 and 50 degrees skew offsets to further compare the prediction methods (see Figure 6.17). Note that the ANSYS data sets list the corresponding skew offsets (in degrees) in parentheses. The figure plainly illustrates the improved predictions of the SGLSL method. The SGLSL predicted deflected shape matches the ANSYS deflections better than the SGL prediction at both skew offsets. Additionally, the SGLSL interior girder predictions are closer to the ANSYS deflections at the skew offsets.
6.6.3 Summary

ANSYS deflections were compared to SGL and SGLSL predictions for continuous span bridges with equal exterior-to-interior girder load ratios. Deflection ratios were calculated and subjected to a statistical analysis. It was revealed that the SGLSL method appears to match ANSYS predicted deflections more closely than the traditional SGL method. Further, it is believed that the advantage of SGLSL over SGL would be more prevalent in models with smaller exterior-to-interior girder load ratios.

6.7 Comparisons of Prediction Methods to Field Measured Deflections

6.7.1 General

Sections 6.4 – 6.6 present comparisons of three developed prediction methods to ANSYS predicted deflections for various bridge configurations. In each case, the newly developed predictions were directly compared to the traditional SGL predictions, and in each
case, the new predictions matched ANSYS predicted deflections more closely than the SGL predictions. The final investigation compares the developed prediction methods back to deflections that were measured in the field. SGL predictions, addressed in Section 6.3, are included and all comparisons are discussed herein.

6.7.2 Simplified Procedure Predictions vs. Field Measured Deflections

Five studied bridges met the criterion for the simplified procedure, which was developed for simple span bridges with equal exterior-to-interior girder load ratios. The simplified procedure predictions at midspan were divided by the corresponding field measured deflections and the results are presented in Table 6.10. Note that the five bridges are listed in order of increasing skew offset and the simplified procedure is denoted as SP. It is apparent that the simplified procedure generally over predicts the field measured deflections. The five individual under predictions are restricted to various interior girders of seven-girder bridges.

<table>
<thead>
<tr>
<th>Girder</th>
<th>Bridge 8</th>
<th>Avondale</th>
<th>US-29</th>
<th>Camden NB</th>
<th>Camden SB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Girder A</td>
<td>1.49</td>
<td>1.24</td>
<td>1.07</td>
<td>1.03</td>
<td>1.10</td>
</tr>
<tr>
<td>Girder B</td>
<td>1.35</td>
<td>1.16</td>
<td>1.12</td>
<td>0.92</td>
<td>0.97</td>
</tr>
<tr>
<td>Girder C</td>
<td>1.31</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Girder D</td>
<td>1.27</td>
<td>1.09</td>
<td>1.15</td>
<td>0.80</td>
<td>0.85</td>
</tr>
<tr>
<td>Girder E</td>
<td>1.28</td>
<td>-</td>
<td>-</td>
<td>0.92</td>
<td>-</td>
</tr>
<tr>
<td>Girder F</td>
<td>1.33</td>
<td>1.07</td>
<td>1.08</td>
<td>1.06</td>
<td>0.97</td>
</tr>
<tr>
<td>Girder G</td>
<td>na</td>
<td>1.08</td>
<td>1.02</td>
<td>na</td>
<td>1.12</td>
</tr>
</tbody>
</table>

The ratios in Table 6.10 were combined with the related SGL ratios in Table 6.1 and a statistical analysis was performed. The results are tabulated in Table 6.11 and plotted in Figure 6.18. It is apparent that the simplified procedure predicts interior girder deflections more accurately than the SGL method. Although the standard deviation and coefficient of
variance is slightly higher, the average ratio is much closer to 1.0 (1.08 compared to 1.43). The SGL method more accurately predicts the exterior girder deflections; the average is closer to 1.0 (1.10 compared to 1.15) and the precision is better. Overall, the interior girder deflections are predicted significantly better by the simplified procedure, whereas the exterior girder deflections are approximately predicted equally as well.

Table 6.11: Statistical Analysis Comparing SP Predictions to SGL Predictions

<table>
<thead>
<tr>
<th></th>
<th>Exterior Girders</th>
<th>Interior Girders</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SP Prediction/</td>
<td>SGL Prediction/</td>
</tr>
<tr>
<td></td>
<td>Measured</td>
<td>Measured</td>
</tr>
<tr>
<td>Average</td>
<td>1.15</td>
<td>1.10</td>
</tr>
<tr>
<td>Min</td>
<td>1.02</td>
<td>0.99</td>
</tr>
<tr>
<td>Max</td>
<td>1.49</td>
<td>1.33</td>
</tr>
<tr>
<td>St. Dev.</td>
<td>0.15</td>
<td>0.10</td>
</tr>
<tr>
<td>COV</td>
<td>0.13</td>
<td>0.09</td>
</tr>
</tbody>
</table>
As an example to illustrate the prediction improvements made by the simplified procedure, the US-29 Bridge (skew offset = 44 degrees) has been plotted in Figure 6.19. Illustrated is the ability of the simplified procedure to accurately predict field measured deflections for the exterior and interior girders.
6.7.3 Alternative Simplified Procedure Predictions vs. Field Measured Deflections

The alternative simplified procedure (ASP) was developed for simple span bridges with unequal exterior-to-interior girder load ratios – only the Eno and Wilmington St Bridges met this criterion. The ASP predictions at midspan were divided by the corresponding field measured deflections and the results are presented in Table 6.12.

<table>
<thead>
<tr>
<th>Girder Number</th>
<th>Eno</th>
<th>Wilmington St</th>
</tr>
</thead>
<tbody>
<tr>
<td>Girder A</td>
<td>1.12</td>
<td>1.32</td>
</tr>
<tr>
<td>Girder B</td>
<td>1.13</td>
<td>1.43</td>
</tr>
<tr>
<td>Girder C</td>
<td>1.13</td>
<td>1.47</td>
</tr>
<tr>
<td>Girder D</td>
<td>1.14</td>
<td>1.39</td>
</tr>
<tr>
<td>Girder E</td>
<td>1.14</td>
<td>1.21</td>
</tr>
</tbody>
</table>

For the Eno and Wilmington St Bridges, the ASP predictions have over predicted the field measured deflections. The ratios in Table 6.12 were combined with the related SGL
ratios in Table 6.1 and a statistical analysis was performed. Table 6.13 and Figure 6.20 present the statistics results and it is apparent that the ASP method predicts deflections more accurately than the SGL method. The ratio averages are comparable for the exterior girders, but the ASP ratio is much closer to 1.0 for the interior girders (1.28 compared to 1.54). Additionally, the standard deviations and coefficients of variance of the exterior and interior girders are significantly lower for the ASP predictions.

**Table 6.13: Statistical Analysis Comparing ASP Predictions to SGL Predictions**

<table>
<thead>
<tr>
<th></th>
<th>Exterior Girders</th>
<th>Interior Girders</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ASP Prediction/</td>
<td>SGL Prediction/</td>
</tr>
<tr>
<td></td>
<td>Measured</td>
<td>Measured</td>
</tr>
<tr>
<td>Average</td>
<td>1.20</td>
<td>1.15</td>
</tr>
<tr>
<td>Min</td>
<td>1.12</td>
<td>0.99</td>
</tr>
<tr>
<td>Max</td>
<td>1.32</td>
<td>1.28</td>
</tr>
<tr>
<td>St. Dev.</td>
<td>0.09</td>
<td>0.14</td>
</tr>
<tr>
<td>COV</td>
<td>0.08</td>
<td>0.12</td>
</tr>
</tbody>
</table>

|                  | ASP Prediction/  | SGL Prediction/  |
|                  | Measured         | Measured         |
| Average          | 1.28             | 1.54             |
| Min              | 1.13             | 1.19             |
| Max              | 1.47             | 1.94             |
| St. Dev.         | 0.17             | 0.35             |
| COV              | 0.13             | 0.22             |
The Wilmington St Bridge (skew offset = 62 degrees) is presented in Figure 6.21 to illustrate the improvements made by the ASP method in predicting field measured deflections. Most significant is the closely matching deflected shapes.
6.7.4 SGL Straight Line Predictions vs. Field Measured Deflections

The SGL straight line (SGLSL) method was implemented to predict the deflections of continuous span bridges with equal exterior-to-interior girder load ratios. Although only Bridge 14 and Bridge 10 (two-span continuous bridges) were included in the parametric study, Bridge 1 (three-span continuous bridge) has been included in this investigation. Corresponding predictions were divided by the field measured deflections at each span location for all three bridges and the results are presented in Table 6.14. It is apparent that under predictions and over predictions are consistent within a given span. The SGLSL method entirely over predicts one span in each of the three continuous span bridges, and under predicts the others.
The ratios in Table 6.14 were combined with the related SGL ratios in Table 6.2 and a statistical analysis was performed (see Table 6.15 and Figure 6.22 for results). Note that the two methods predict identical exterior girder deflections, and, therefore, the exterior and interior girder ratios have been combined. It is apparent from the results that only a slight advantage exists in predicting girder deflections by the SGLSL method. The two methods exhibit very similar precision, but the SGLSL average ratio is essentially 1.0, whereas the SGL ratio is slightly higher at 1.04.

<table>
<thead>
<tr>
<th>Span Location</th>
<th>Girder A</th>
<th>Girder B</th>
<th>Girder C</th>
<th>Girder D</th>
<th>Girder E</th>
<th>Girder F</th>
<th>Girder G</th>
</tr>
</thead>
<tbody>
<tr>
<td>4/10 Span A</td>
<td>1.15</td>
<td>1.27</td>
<td>1.03</td>
<td>1.18</td>
<td>1.96</td>
<td>na</td>
<td>na</td>
</tr>
<tr>
<td>6/10 Span B</td>
<td>0.86</td>
<td>0.92</td>
<td>0.80</td>
<td>0.89</td>
<td>0.81</td>
<td>na</td>
<td>na</td>
</tr>
<tr>
<td>4/10 Span B</td>
<td>1.12</td>
<td>1.15</td>
<td>1.26</td>
<td>1.09</td>
<td>na</td>
<td>na</td>
<td>na</td>
</tr>
<tr>
<td>6/10 Span C</td>
<td>0.64</td>
<td>0.80</td>
<td>0.80</td>
<td>0.80</td>
<td>na</td>
<td>na</td>
<td>na</td>
</tr>
<tr>
<td>35/100 Span C</td>
<td>1.36</td>
<td>1.53</td>
<td>-</td>
<td>1.40</td>
<td>-</td>
<td>1.23</td>
<td>1.01</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bridge Location</th>
<th>Girder A</th>
<th>Girder B</th>
<th>Girder C</th>
<th>Girder D</th>
<th>Girder E</th>
<th>Girder F</th>
<th>Girder G</th>
</tr>
</thead>
<tbody>
<tr>
<td>4/10 Span A</td>
<td>0.73</td>
<td>0.84</td>
<td>-</td>
<td>0.95</td>
<td>-</td>
<td>0.82</td>
<td>0.71</td>
</tr>
<tr>
<td>4/10 Span B</td>
<td>0.71</td>
<td>0.74</td>
<td>-</td>
<td>0.78</td>
<td>-</td>
<td>0.81</td>
<td>0.82</td>
</tr>
<tr>
<td>35/100 Span C</td>
<td>1.36</td>
<td>1.53</td>
<td>-</td>
<td>1.40</td>
<td>-</td>
<td>1.23</td>
<td>1.01</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Average</th>
<th>SGLSL Prediction/Measured</th>
<th>SGL Prediction/Measured</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min</td>
<td>0.64</td>
<td>0.64</td>
</tr>
<tr>
<td>Max</td>
<td>1.96</td>
<td>1.96</td>
</tr>
<tr>
<td>St. Dev.</td>
<td>0.29</td>
<td>0.30</td>
</tr>
<tr>
<td>COV</td>
<td>0.29</td>
<td>0.29</td>
</tr>
</tbody>
</table>
As an example to compare the similar prediction methods, the span B deflections of Bridge 10 (skew offset = 57 degrees) have been plotted in Figure 6.23. The only variation between the two prediction methods is the improved interior girder predictions by the SGLSL method.
6.8 Summary

Comparisons have been made between field measured deflections, ANSYS predicted deflections, SGL predicted deflections, and deflections predicted by three newly developed procedures. Girder deflections for simple span bridges have been predicted by the simplified procedure and the alternative simplified procedure for bridges with equal and unequal exterior-to-interior girder load ratios, respectively. Additionally, deflections of continuous span bridges with equal exterior-to-interior girder load ratios have been predicted by the SGL straight line method. According to multiple statistical analyses, it has been concluded that all three new prediction methods predict dead load deflections in steel plate girder bridges more accurately than traditional SGL analysis.
To verify this conclusion, the SGL method was shown not to accurately predict field measured deflections for either bridge type. Finite element models, created in ANSYS, proved to capture the deflection behavior more accurately than the traditional SGL method. Next, the three new prediction methods were individually compared to the SGL method, as related to ANSYS predicted deflections. Each method demonstrated the ability to predict ANSYS simulated deflections more accurately than the SGL approach. Finally, deflections predicted by the newly developed methods were compared to the field measured deflections.

Following are two tables and ten figures to present the deflection data for all ten measured bridges. Table 6.16 includes various deflection ratios for field measured deflections, SGL predicted deflections, ANSYS predicted deflections, and newly predicted deflections. Similarly, Table 6.17 includes the differences in magnitudes for the aforementioned deflections. Finally, Figures 6.24 – 6.33 present the field measured deflections, SGL predicted deflections, ANSYS predicted deflections, and deflections predicted by the newly developed procedures to compare the girder deflections discussed in this chapter.
Table 6.16: Complete Comparison of Deflection Ratios

<table>
<thead>
<tr>
<th>Bridge</th>
<th>SGL/Measured</th>
<th>SGL/ANSYS</th>
<th>ANSYS/Measured</th>
<th>New Prediction*/Measured</th>
<th>New Prediction*/ANSYS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Exterior</td>
<td>Interior</td>
<td>Exterior</td>
<td>Interior</td>
<td>Exterior</td>
</tr>
<tr>
<td>Eno</td>
<td>1.03</td>
<td>1.24</td>
<td>0.90</td>
<td>1.08</td>
<td>1.15</td>
</tr>
<tr>
<td>Bridge 8</td>
<td>1.26</td>
<td>1.42</td>
<td>0.94</td>
<td>1.10</td>
<td>1.34</td>
</tr>
<tr>
<td>Avondale</td>
<td>1.09</td>
<td>1.25</td>
<td>1.01</td>
<td>1.17</td>
<td>1.08</td>
</tr>
<tr>
<td>US-29</td>
<td>1.08</td>
<td>1.39</td>
<td>1.08</td>
<td>1.29</td>
<td>1.00</td>
</tr>
<tr>
<td>Camden NB</td>
<td>1.01</td>
<td>1.51</td>
<td>0.80</td>
<td>1.41</td>
<td>1.26</td>
</tr>
<tr>
<td>Camden SB</td>
<td>1.07</td>
<td>1.59</td>
<td>0.94</td>
<td>1.62</td>
<td>1.15</td>
</tr>
<tr>
<td>Wilmington St</td>
<td>1.26</td>
<td>1.85</td>
<td>1.58</td>
<td>2.31</td>
<td>0.80</td>
</tr>
<tr>
<td>Bridge 14 - A</td>
<td>1.56</td>
<td>1.20</td>
<td>0.99</td>
<td>1.01</td>
<td>1.57</td>
</tr>
<tr>
<td>Bridge 14 - B</td>
<td>0.83</td>
<td>0.90</td>
<td>0.98</td>
<td>1.02</td>
<td>0.85</td>
</tr>
<tr>
<td>Bridge 10 - B</td>
<td>1.10</td>
<td>1.35</td>
<td>1.19</td>
<td>1.40</td>
<td>0.93</td>
</tr>
<tr>
<td>Bridge 10 - C</td>
<td>0.72</td>
<td>0.90</td>
<td>1.10</td>
<td>1.30</td>
<td>0.65</td>
</tr>
<tr>
<td>Bridge 1 - A</td>
<td>0.72</td>
<td>0.94</td>
<td>0.93</td>
<td>1.02</td>
<td>0.77</td>
</tr>
<tr>
<td>Bridge 1 - B</td>
<td>0.76</td>
<td>0.84</td>
<td>0.85</td>
<td>0.93</td>
<td>0.90</td>
</tr>
<tr>
<td>Bridge 1 - C</td>
<td>1.18</td>
<td>1.50</td>
<td>1.21</td>
<td>1.32</td>
<td>0.97</td>
</tr>
<tr>
<td>Average</td>
<td>1.05</td>
<td>1.28</td>
<td>1.04</td>
<td>1.28</td>
<td>1.03</td>
</tr>
<tr>
<td>Min</td>
<td>0.72</td>
<td>0.84</td>
<td>0.80</td>
<td>0.93</td>
<td>0.65</td>
</tr>
<tr>
<td>Max</td>
<td>1.56</td>
<td>1.85</td>
<td>1.58</td>
<td>2.31</td>
<td>1.57</td>
</tr>
<tr>
<td>St. Dev.</td>
<td>0.23</td>
<td>0.30</td>
<td>0.20</td>
<td>0.35</td>
<td>0.25</td>
</tr>
<tr>
<td>COV</td>
<td>0.22</td>
<td>0.23</td>
<td>0.19</td>
<td>0.28</td>
<td>0.24</td>
</tr>
</tbody>
</table>
Table 6.17: Complete Comparison of Differences in Deflection Magnitudes

<table>
<thead>
<tr>
<th>Bridge</th>
<th>SGL - Measured</th>
<th>SGL - ANSYS</th>
<th>ANSYS - Measured</th>
<th>New Prediction* - Measured</th>
<th>New Prediction* - ANSYS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Exterior</td>
<td>Interior</td>
<td>Exterior</td>
<td>Interior</td>
<td>Exterior</td>
</tr>
<tr>
<td>Eno</td>
<td>0.28</td>
<td>1.77</td>
<td>-0.81</td>
<td>0.72</td>
<td>1.09</td>
</tr>
<tr>
<td>Bridge 8</td>
<td>0.78</td>
<td>1.36</td>
<td>-0.27</td>
<td>0.42</td>
<td>1.04</td>
</tr>
<tr>
<td>Avondale</td>
<td>0.43</td>
<td>1.29</td>
<td>0.03</td>
<td>0.94</td>
<td>0.40</td>
</tr>
<tr>
<td>US-29</td>
<td>0.36</td>
<td>1.62</td>
<td>0.38</td>
<td>1.31</td>
<td>-0.03</td>
</tr>
<tr>
<td>Camden NB</td>
<td>0.02</td>
<td>1.69</td>
<td>-0.79</td>
<td>1.45</td>
<td>0.81</td>
</tr>
<tr>
<td>Camden SB</td>
<td>0.22</td>
<td>1.88</td>
<td>-0.21</td>
<td>1.93</td>
<td>0.43</td>
</tr>
<tr>
<td>Wilmington St</td>
<td>1.18</td>
<td>3.29</td>
<td>2.07</td>
<td>4.07</td>
<td>-0.89</td>
</tr>
<tr>
<td>Bridge 14 - A</td>
<td>0.31</td>
<td>0.17</td>
<td>-0.01</td>
<td>0.01</td>
<td>0.32</td>
</tr>
<tr>
<td>Bridge 14 - B</td>
<td>-0.27</td>
<td>-0.16</td>
<td>-0.03</td>
<td>0.03</td>
<td>-0.24</td>
</tr>
<tr>
<td>Bridge 10 - B</td>
<td>0.21</td>
<td>0.64</td>
<td>0.34</td>
<td>0.70</td>
<td>-0.14</td>
</tr>
<tr>
<td>Bridge 10 - C</td>
<td>-0.54</td>
<td>-0.17</td>
<td>0.11</td>
<td>0.34</td>
<td>-0.65</td>
</tr>
<tr>
<td>Bridge 1 - A</td>
<td>-0.56</td>
<td>-0.11</td>
<td>-0.11</td>
<td>0.03</td>
<td>-0.45</td>
</tr>
<tr>
<td>Bridge 1 - B</td>
<td>-1.04</td>
<td>-0.66</td>
<td>-0.58</td>
<td>-0.27</td>
<td>-0.46</td>
</tr>
<tr>
<td>Bridge 1 - C</td>
<td>0.24</td>
<td>0.62</td>
<td>0.30</td>
<td>0.45</td>
<td>-0.06</td>
</tr>
<tr>
<td><strong>Average</strong></td>
<td><strong>0.11</strong></td>
<td><strong>0.95</strong></td>
<td><strong>0.03</strong></td>
<td><strong>0.87</strong></td>
<td><strong>0.08</strong></td>
</tr>
</tbody>
</table>
* New Prediction* - ANSYS - Measured

**St. Dev.**
Figure 6.24: Field Measured Deflections vs. Predicted Deflections for Bridge 8

Figure 6.25: Field Measured Deflections vs. Predicted Deflections for the Avondale Bridge
Figure 6.26: Field Measured Deflections vs. Predicted Deflections for the US-29 Bridge

Figure 6.27: Field Measured Deflections vs. Predicted Deflections for the Camden NB Bridge
Figure 6.28: Field Measured Deflections vs. Predicted Deflections for the Camden SB Bridge

Figure 6.29: Field Measured Deflections vs. Predicted Deflections for the Eno Bridge
Figure 6.30: Field Measured Deflections vs. Predicted Deflections for the Wilmington St Bridge

Figure 6.31: Field Measured Deflections vs. Predicted Deflections for Bridge 14 (Span B)
Figure 6.32: Field Measured Deflections vs. Predicted Deflections for Bridge 10 (Span B)

Figure 6.33: Field Measured Deflections vs. Predicted Deflections for Bridge 1 (Span B)
Chapter 7

Observations, Conclusions, and Recommendations

7.1 Summary

A simplified procedure has been developed to predict dead load deflections of skewed and non-skewed steel plate girder bridges for use by the North Carolina Department of Transportation (NCDOT). The research was funded to mitigate costly construction delays and maintenance and safety issues in future projects that result from inaccurate deflection predictions via the traditional single girder line (SGL) analysis.

Ten steel plate girder bridges were monitored and field measured deflections were recorded to capture true girder deflection behavior during concrete deck construction. A three-dimensional finite element bridge modeling technique was established and the simulated girder deflections correlated well with field measured deflections. In combination with a preprocessor program developed by the author, the finite element modeling technique was utilized to conduct a parametric study, in which the effects of skew angle, girder spacing, span length, cross frame stiffness, number of girders within the span, and exterior-to-interior girder load ratio on girder deflection behavior were investigated. The results were analyzed and the simplified procedure was developed to predict deflections in steel plate girder bridges. The procedure utilizes empirically derived modifications which are applied to the traditional SGL predictions to account for the effects of skew angle, girder spacing, span length, and exterior-to-interior girder load ratio. Predictions via the simplified procedure were compared to field measured deflections and SGL predictions to validate the procedure.
7.2 Observations

The observations discussed herein relate to field measurements, finite element modeling, automated model generation, the parametric study, the development of the simplified procedure, and comparisons of the deflection results.

- The field measured deflections for the five bridges included in this phase of the research project exhibited five individual deflected shapes, none of which matched SGL predicted deflected shapes.
- Incorporating the SIP metal deck forms into the finite element models resulted in distinctly different simulated deflection behavior, especially for the Wilmington St Bridge.
- SGL predictions over predict field measured deflections for the interior and exterior girders of simple span bridges by approximately 12 and 46 percent, respectively.
- ANSYS finite element models predict field measured deflections more accurately than SGL predictions. The interior and exterior girders of simple span bridges are over predicted by approximately 11 and 7 percent respectively.
- SGL predictions and ANSYS predictions match field measured deflections equally well for interior and exterior girders of continuous span bridges.
- Predictions from the simplified procedure for simple span bridges with equal exterior-to-interior girder load ratios over predict field measured deflections by 8 and 15 percent for the interior and exterior girders, respectively.
- Predictions from the alternative simplified procedure (ASP) for simple span bridges with unequal exterior-to-interior girder load ratios over predict field
measured deflections by 28 and 20 percent for the interior and exterior girders, respectively.

- On average, predictions from the SGL straight line (SGLSL) method match the field measured deflections for the exterior and interior girders of continuous span bridges with equal exterior-to-interior girder load ratios.

7.3 Conclusions

The conclusions discussed herein relate to field measurements, finite element modeling, automated model generation, the parametric study, the development of the simplified procedure, and comparisons of the deflection results.

- The traditional SGL method does not accurately predict dead load deflections of steel plate girder bridges.
- Finite element models created according to the technique presented in this thesis are capable of predicting deflections for skewed and non-skewed steel plate girder bridges.
- Finite element models with SIP forms generate more accurate results, and should be included in the finite element models.
- Skew, the exterior-to-interior girder load ratio, and the girder spacing-to-span ratio affect girder dead load deflections for simple span bridges.
- Cross frame stiffness and the number of girders within the span do not have a significant effect on girder dead load deflections for simple span bridges.
- The simplified procedure (SP), alternative simplified procedure (ASP), and SGL straight line (SGLSL) method can accurately predict girder dead load deflections.
7.4 **Recommended Simplified Procedures**

The recommended simplified procedures to predict the dead load deflections are presented for simple span bridges with equal exterior-to-interior girder load ratios, simple span bridges with unequal exterior-to-interior girder load ratios, and continuous span bridges with equal exterior-to-interior girder load ratios. The three procedures utilize the equations presented in the following sections to predict the exterior girder deflections and the differential deflections between adjacent girders. Additionally, detailed sample calculations are presented in Appendix B.

7.4.1 **Simple Span Bridges with Equal Exterior-to-Interior Girder Load Ratios**

The following simplified procedure was developed in Chapter 5 for simple span bridges with equal exterior-to-interior girder load ratios. Note that the procedure is applied to half of the bridge cross-section and the predictions are then mirrored about an imaginary vertical axis through: the middle girder of a bridge with an odd number of girders or the middle of a bridge with an even number of girders. For instance, the procedure would be utilized to calculate the predicted deflections of girders 1, 2, 3, and 4 in a seven girder bridge. The predictions would then be symmetric about an imaginary vertical axis through girder 4. As a result, the predicted deflection of girder 5 would equal that of girder 3, girder six would equal girder 2, and so on (see Figure 7.1).
Step 1: Calculate the interior girder SGL prediction, $\delta_{SGL\_INT}$, at desired locations along the span (ex. 1/10 points), and at midspan, $\delta_{SGL\_M}$.

Step 2: Calculate the predicted exterior girder deflection at each location along the span using the following:

$$\delta_{EXT} = [\delta_{SGL\_INT} - \Phi(100 - L)][1 - 0.1\tan(1.2\theta)]$$  \hspace{1cm} (eq. 7.1)

where:

- $\delta_{SGL\_INT} = $ interior girder SGL predicted deflection at locations along the span (in)
- $\Phi = 0.03 - a(\theta)$
  - where $a = 0.0002$ if $g \leq 8.2$
  - $a = 0.0002 + 0.000305(8.2 - g)$ if $8.2 < g \leq 11.5$
  - where $g = $ girder spacing (ft)
  - $L = $ exterior-to-interior girder load ratio (in percent, ex: 65 %)
  - $\theta = $ skew offset (degrees) = |skew - 90|

Step 3: Calculate the predicted differential deflection between adjacent girders at each location along the span using the following:


\[ D_{\text{INT}} = x[a(S - 0.04)(1 + z) - 0.1\tan(1.2\theta)] \]  

(eq. 7.2)

where:  
\[ x = \left(\frac{\delta_{\text{SGL INT}}}{\delta_{\text{SGL M}}}\right) \]

where:  
\[ \delta_{\text{SGL M}} = \text{SGL predicted girder deflection at midspan (in)} \]

\[ \alpha = 3.0 - b(\theta) \]

where:  
\[ b = -0.08 \quad \text{if } (S \leq 0.05) \]

\[ b = -0.08 + 8(S - 0.05) \quad \text{if } (0.05 < S \leq 8.2) \]

where:  
\[ S = \text{girder spacing-to-span ratio} \]

\[ z = (10(L - 0.04) + 0.02)(2 - L/50) \]

\[ \theta = \text{skew offset (degrees)} = |\text{skew} - 90| \]

- Step 4: Calculate the predicted interior girder deflections at each location along the span using the following:

\[ \delta_{\text{INT}i} = \delta_{\text{EXT}} + y^*D_{\text{INT}} \]  

(eq. 7.3)

where:  
\[ y = 1 \quad \text{(first interior girder)} \]

\[ y = 2 \quad \text{(other interior girders)} \]

### 7.4.2 Simple Span Bridges with Unequal Exterior-to-Interior Girder Load Ratios

The following recommendation utilizes the alternative simplified procedure (ASP) developed in Chapter 5 for simple span bridges with unequal exterior-to-interior girder load ratios. Note that ‘high ratio’ and ‘low ratio’ refers to the greater and lesser of the two exterior-to-interior girder load ratios respectively. Additionally, the procedure is applicable for a difference in exterior-to-interior girder load ratios of more than 10 percent. For instance, if one exterior girder load is 78 percent of the interior girder load and the other exterior girder load is 90 percent (difference of 12 percent), this method is applicable. If the
second exterior girder load is only 86 percent (difference of 8 percent) the simplified procedure (SP) is applied, as previously discussed.

- Step 1: Calculate the interior girder SGL prediction, $\delta_{SGL_{INT}}$, at desired locations along the span (ex. 1/10 points), and at midspan, $\delta_{SGL_M}$.

- Step 2: Calculate the predicted exterior girder deflections, $\delta_{EXT}$, at each location along the span for both the ‘high ratio’ and ‘low ratio’ using Equation 7.1.

- Step 3: Calculate the predicted differential deflection, $D_{INT}$, between adjacent girders for the ‘low ratio’ according to Equation 7.2.

- Step 4: Calculate the predicted interior girder deflections, $\delta_{INT,i}$, for the ‘low ratio,’ to the middle girder for an odd number of girders and to the center girders for an even number of girders, according to Equation 7.3.
Figure 7.3: Step 4 of the Alternative Simplified Procedure (ASP)

- Step 5: Calculate the ‘slope’ of a line through the predicted exterior girder deflection for the ‘high ratio’ (girder 7 in the Figures) and the predicted center girder deflection for the ‘low ratio’ (girder 4 in the Figures).
- Step 6: Interpolate and extrapolate deflections to predict the entire deflected shape along the straight line referenced in Step 5.
7.4.3 Continuous Span Bridges with Equal Exterior-to-Interior Girder Load Ratios

The following SGL straight line (SGLSL) method was developed in Chapter 5 for continuous span bridges with equal exterior-to-interior girder load ratios.

- Step 1: Calculate the exterior girder SGL predictions, $\delta_{SGL, EXT}$, at desired locations along the span (ex. 1/20 points).
- Step 2: Use the predicted exterior girder SGL deflections as the interior girder deflections, resulting in a straight line prediction (see Figure 7.5).
7.5 Future Considerations

Future research can be directed to improve upon the recommendations concluded in this research. Additional steel plate girder bridges should be monitored in the field to further validate the measured deflections to finite element models. Consequently, increased variance in measured bridge parameters would provide further validation to the simplified procedure and allow for future improvements. Additional bridges should include the possible bridge configurations: simple span bridges with equal and unequal exterior-to-interior girder load ratios and continuous span bridges with equal and unequal exterior-to-interior girder load ratios.
References


*ANSYS 7.1 Documentation* (2003), Swanson Analysis System, Inc.


Currah, R.M. (1993). “Shear Strength and Shear Stiffness of Permanent Steel Bridge Deck Forms,” M.S. Thesis, Department of Civil Engineering, University of Texas, Austin, TX.


Helwig, T., and Yura, J. (2003), “Strength Requirements for Diaphragm Bracing of Beams,” Draft manuscript to be submitted.


Appendices
Appendix A

Simplified Procedure Flow Chart

This appendix contains a flow chart outlining the simplified procedures developed to predict dead load deflections of skewed and non-skewed steel plate girder bridges.

The steps (1-4) for the Simplified Procedure (SP) are described in Section 7.4.1, steps (1-6) for the Alternative Simplified Procedure (ASP) are described in Section 7.4.2, and steps (1-2) for the SGL Straight Line Method (SGLSL) are described in Sections 7.4.3.

The flow chart can be utilized for the following: simple span bridges with equal exterior-to-interior girder load ratios, simple span bridges with unequal exterior-to-interior girder load ratios, and continuous span bridges with equal exterior-to-interior girder load ratios.
START: SGL ANALYSIS AT DESIRED LOCATIONS ALONG THE SPAN

SIMPLE SPAN BRIDGE?

YES

EXTERIOR-TO-INTERIOR GIRDER LOAD RATIO WITHIN 10 PERCENT DIFFERENCE?

YES (SP)

NO (ASP)

STEP 1

STEP 2

STEP 3

STEP 4

STEP 1

STEP 2

STEP 3

STEP 4

STEP 5

STEP 6

NO (SGLSL)
Appendix B

Sample Calculations of the Simplified Procedure

This appendix contains a step-by-step sample calculation of the simplified procedure developed to predict dead load deflections in steel plate girder bridges. In this sample, deflections are predicted for the US-29 Bridge (simple span). Two cases were considered: equal exterior-to-interior girder load ratios and unequal exterior-to-interior girder load ratios.

Single girder line (SGL) analysis is utilized for the base prediction on which the simplified procedure predicts deflections. In this appendix, the girders are assumed to have constant cross-section and the SGL deflections are predicted for a prismatic beam with a uniformly distributed dead load, determined from tributary width assumptions.
Sample Calculations of the Simplified Procedure for the US-29 Bridge

Given
Number of Girders = 7
Skew Angle = 46 degrees
Constant, $E_s = 30,000$ ksi
Girder Spacing, $g = 7.75$ ft
Interior girder load, $w_i = 2$ k/ft

Girder Spacing, $g = 7.75$ ft

Case I: Equal Exterior-to-Interior Girder Load Ratios, $w_i = w_7 = 1.7$ k/ft
Case II: Unequal Exterior-to-Interior Girder Load Ratios, $w_i = 1.7$ k/ft, $w_7 = 1.3$ k/ft

Case I Calculations
Equivalent Skew Offset: $\theta = |90 - \text{skew}| = |90 - 46| = 44$ degrees
Girder Spacing to Span Ratio: $S = g/L = 7.75/123.83 = 0.063$

$\frac{1}{2}$ Span: $\delta_{SGL \_ INT} = \frac{5wl^4}{384EI} = \frac{5(2)(123.83)^4}{384(1.225 \times 10^7)} = 0.50$ ft = 6.00 in

$\delta_{EXT} = |\delta_{SGL \_ INT} - \Phi(100 - L)|(1 - 0.1 \tan(1.2 \theta))$

$= [6.00 - 0.018(100 - 85)](1 - 0.1 \tan(1.2 \times 44)) = 4.93$ in

where: $\Phi = 0.03 - a(\theta) = 0.03 - 0.0002(44) = 0.018$

Note: $a = 0.0002 \quad (g \leq 8.2$ ft$)$

$L = \frac{w_{EXT}}{w_{INT}} = \frac{1.7}{2.0} = 85\%$

$D_{INT} = x[\alpha(S - 0.04)(1 + z) - 0.1 \tan(1.2 \theta)]$

$= 1.0[1.94(0.063 - 0.04)(1 + 0.074) - 0.1 \tan(1.2 \times 44)] = -0.08$ in

where: $x = \frac{\delta_{SGL \_ INT}}{\delta_{SGL \_ M}} = \frac{6.0}{6.0} = 1.0$
Case I (cont.)

where:  \( \alpha = 3.0 - b(\theta) = 3.0 - 0.024(44) = 1.94 \)

Note:  \( b = -0.08 + 8(S - 0.05) = -0.08 + 8(0.063 - 0.05) = 0.021 \)
\((0.05 < S \leq 0.08)\)

\[ z = (10(S - 0.04) + 0.02)(2 - \frac{L}{50}) \]
\[ = (10(0.063 - 0.04) + 0.02)(2 - \frac{85}{50}) = 0.074 \]

\( \frac{1}{4} \) Span:  \( \delta_{SGL\_INT} = \frac{57wl^4}{6144EI} = \frac{57(2)(123.83)^4}{6144(1.225 \times 10^7)} = 0.36 \text{ ft} = 4.27 \text{ in} \)

\( \delta_{EXT} = [4.27 - 0.018(100 - 85)][1 - 0.1 \tan(1.2 \times 44)] = 3.43 \text{ in} \)

\( D_{INT} = 0.71[1.94(0.063 - 0.04)(1 + 0.074) - 0.1 \tan(1.2 \times 44)] = -0.06 \text{ in} \)

where:  \( x = \frac{\delta_{SGL\_INT}}{\delta_{SGL\_M}} = \frac{4.27}{6.0} = 0.71 \)

Results (inches):

<table>
<thead>
<tr>
<th></th>
<th>G1</th>
<th>G2</th>
<th>G3</th>
<th>G4</th>
<th>G5</th>
<th>G6</th>
<th>G7</th>
</tr>
</thead>
<tbody>
<tr>
<td>SGL</td>
<td>3.63</td>
<td>4.27</td>
<td>4.27</td>
<td>4.27</td>
<td>4.27</td>
<td>4.27</td>
<td>4.27</td>
</tr>
<tr>
<td>Simplified Procedure</td>
<td>3.43</td>
<td>3.37</td>
<td>3.32</td>
<td>3.32</td>
<td>3.32</td>
<td>3.37</td>
<td>3.43</td>
</tr>
<tr>
<td></td>
<td>5.10</td>
<td>6.00</td>
<td>6.00</td>
<td>6.00</td>
<td>6.00</td>
<td>6.00</td>
<td>5.10</td>
</tr>
<tr>
<td></td>
<td>4.93</td>
<td>4.85</td>
<td>4.77</td>
<td>4.77</td>
<td>4.77</td>
<td>4.85</td>
<td>4.93</td>
</tr>
</tbody>
</table>
Case II (Midspan Only)

65% Load (‘Light Load’):

\[
\delta_{\text{EXT}} = [6.00 - 0.018(100 - 65)][1 - 0.1 \tan(1.2 \times 44)] = 4.57 \text{ in}
\]

where: \[ L = \frac{w_{\text{EXT}}}{w_{\text{INT}}} = \frac{1.3}{2.0} = 65\% \]

\[ D_{\text{INT}} = 1.0[1.94(0.063 - 0.04)(1 + 0.172) - 0.1 \tan(1.2 \times 44)] = -0.08 \text{ in} \]

where: \[ z = (10(0.063 - 0.04) + 0.02)(2 - \frac{65}{50}) = 0.172 \]

Girder 4 Deflection (middle): \[ \delta_4 = \delta_{\text{EXT}} + 2(D_{\text{INT}}) = 4.57 + 2(-0.08) = 4.41 \text{ in} \]

Recall, Girder 1 Deflection: \[ \delta_1 = \delta_{\text{EXT}} = 4.93 \text{ in} \quad \text{(from Case I)} \]

Predict other girder deflections with straight line passing through \( \delta_1 \) and \( \delta_4 \)

\[ \text{Slope = Differential} = \frac{\delta_4 - \delta_1}{4 - 1} = \frac{4.41 - 4.93}{4 - 1} = -0.173 \]

Results (inches):

<table>
<thead>
<tr>
<th></th>
<th>G1</th>
<th>G2</th>
<th>G3</th>
<th>G4</th>
<th>G5</th>
<th>G6</th>
<th>G7</th>
</tr>
</thead>
<tbody>
<tr>
<td>SGL</td>
<td>5.10</td>
<td>6.00</td>
<td>6.00</td>
<td>6.00</td>
<td>6.00</td>
<td>6.00</td>
<td>3.90</td>
</tr>
<tr>
<td>ASP</td>
<td>4.93</td>
<td>4.76</td>
<td>4.58</td>
<td>4.41</td>
<td>4.24</td>
<td>4.06</td>
<td>3.89</td>
</tr>
</tbody>
</table>

![Graph showing deflection predictions](image)
Appendix C

Deflection Summary for Bridge 8

This appendix contains a detailed description of Bridge 8 including bridge geometry, material data, cross frame type and size, and dead loads calculated from slab geometry. Illustrations detailing the bridge geometry and field measurement locations are included, along with tables and graphs of the field measured non-composite girder deflections.

A summary of the ANSYS finite element model created for Bridge 8 is also included in this appendix. This summary includes a picture of the ANSYS model, details about the elements used in the model generation, the loads applied to the model, and tables and graphs of the deflections predicted by the model.
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER: R-2547 (EB Bridge on US 64 Bypass over Smithfield Rd.)
MEASUREMENT DATE: August 24, 2004

BRIDGE DESCRIPTION

<table>
<thead>
<tr>
<th>Type</th>
<th>One Span Simple</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length</td>
<td>153.04 ft (46.648 m)</td>
</tr>
<tr>
<td>Number of Girders</td>
<td>6</td>
</tr>
<tr>
<td>Girder Spacing</td>
<td>11.29 ft (3.440 m)</td>
</tr>
<tr>
<td>Skew</td>
<td>60 deg</td>
</tr>
<tr>
<td>Overhang</td>
<td>2.85 ft (870 mm) (from web centerline)</td>
</tr>
<tr>
<td>Bearing Type</td>
<td>Elastomeric Pad</td>
</tr>
</tbody>
</table>

MATERIAL DATA

<table>
<thead>
<tr>
<th>Structural Steel</th>
<th>Grade</th>
<th>Yield Strength</th>
</tr>
</thead>
<tbody>
<tr>
<td>Girder:</td>
<td>AASHTO M270</td>
<td>50 ksi (345 MPa)</td>
</tr>
<tr>
<td>Other:</td>
<td>AASHTO M270</td>
<td>50 ksi (345 MPa)</td>
</tr>
</tbody>
</table>

Concrete Unit Weight: 150 pcf (nominal)
SIP Form Weight: 4.69 psf (CSI Catalog)

GIRDER DATA

<table>
<thead>
<tr>
<th>Length</th>
<th>153.04 ft (46.648 m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web Thickness</td>
<td>0.63 in (16 mm)</td>
</tr>
<tr>
<td>Web Depth</td>
<td>68.03 in (1728 mm)</td>
</tr>
<tr>
<td>Top Flange Width</td>
<td>17.99 in (457 mm)</td>
</tr>
<tr>
<td>Bottom Flange Width</td>
<td>17.99 in (457 mm)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Flange Thickness</th>
<th>Begin</th>
<th>End</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.00 in (51 mm)</td>
<td>0.00</td>
<td>153.04 ft (46.648 m)</td>
</tr>
<tr>
<td>Bottom:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.00 in (51 mm)</td>
<td>0.00</td>
<td>40.98 ft (12.490 m)</td>
</tr>
<tr>
<td>3.00 in (76 mm)</td>
<td>40.98 ft (12.490 m)</td>
<td>112.07 ft (34.158 m)</td>
</tr>
<tr>
<td>2.00 in (51 mm)</td>
<td>112.07 ft (34.158 m)</td>
<td>153.04 ft (46.648 m)</td>
</tr>
</tbody>
</table>

CROSS-FRAME DATA

<table>
<thead>
<tr>
<th>End Bent (Type K)</th>
<th>Diagonals</th>
<th>Horizontals</th>
<th>Verticals</th>
</tr>
</thead>
<tbody>
<tr>
<td>WT 4 x 14</td>
<td>C 15 x 33.9 (top)</td>
<td>WT 4 x 14 (bottom)</td>
<td>NA</td>
</tr>
<tr>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>L 3 x 3 x 3/8&quot;</td>
<td>L 3 x 3 x 3/8&quot; (bottom)</td>
<td>NA</td>
<td></td>
</tr>
</tbody>
</table>

174
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER: R-2547 (EB Bridge on US 64 Bypass over Smithfield Rd.)
MEASUREMENT DATE: August 24, 2004

STIFFENERS

Longitudinal: NA
Bearing: PL 0.87" × 7.09" (22 mm × 180 mm)
Intermediate: PL 0.63" × NA (16 mm × NA, connector plate)
No Intermediate Stiffeners
Middle Bearing: NA
End Bent Connector: PL 0.87" × NA (22 mm × NA, connector plate)

SLAB DATA

THICKNESS 9.25 in (235 mm) nominal
BUILD-UP 3.74 in (95 mm) nominal

LONGITUDINAL REBAR SIZE (metric) SPACING (nominal)
Top: #13 17.72 in (450 mm)
Bottom: #16 8.27 in (210 mm)

TRANSVERSE REBAR
Top: #16 5.51 in (140 mm)
Bottom: #16 5.51 in (140 mm)

DECK LOADS

<table>
<thead>
<tr>
<th>Girder</th>
<th>Concrete$^1$</th>
<th>Slab$^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>lb/ft</td>
<td>N/mm</td>
</tr>
<tr>
<td>G1</td>
<td>1186.80</td>
<td>17.32</td>
</tr>
<tr>
<td>G2</td>
<td>1414.97</td>
<td>20.65</td>
</tr>
<tr>
<td>G3</td>
<td>1414.97</td>
<td>20.65</td>
</tr>
<tr>
<td>G4</td>
<td>1414.97</td>
<td>20.65</td>
</tr>
<tr>
<td>G5</td>
<td>1414.97</td>
<td>20.65</td>
</tr>
<tr>
<td>G6</td>
<td>1186.80</td>
<td>17.32</td>
</tr>
</tbody>
</table>

$^1$ Calculated with nominal slab thicknesses

$^2$ Includes slab, buildups, and stay-in-place forms (nominal)
FIELD MEASUREMENT SUMMARY

Project Number: R-2547 (EB Bridge on US 64 Bypass over Smithfield Rd.)
Measurement Date: August 24, 2004

Girder Centerline:
Measurement Location:

(a) Plan View (Not to Scale)

(b) Elevation View (Not to Scale)

Plan and Elevation View of Bridge 8 (Knightdale, NC)
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER: R-2547 (EB Bridge on US 64 Bypass over Smithfield Rd.)
MEASUREMENT DATE: August 24, 2004

BEARING SETTLEMENTS (data in inches)

<table>
<thead>
<tr>
<th>Point</th>
<th>End 1</th>
<th>End 2</th>
<th>Avg.</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>0.09</td>
<td>0.10</td>
<td>0.09</td>
</tr>
<tr>
<td>G2</td>
<td>0.13</td>
<td>0.11</td>
<td>0.12</td>
</tr>
<tr>
<td>G3</td>
<td>---</td>
<td>0.12</td>
<td>0.12</td>
</tr>
<tr>
<td>G4</td>
<td>0.04</td>
<td>0.04</td>
<td>0.04</td>
</tr>
<tr>
<td>G5</td>
<td>0.07</td>
<td>0.09</td>
<td>0.08</td>
</tr>
<tr>
<td>G6</td>
<td>0.08</td>
<td>0.10</td>
<td>0.09</td>
</tr>
</tbody>
</table>

GIRDER DEFLECTIONS (data in inches)

MEASURED

<table>
<thead>
<tr>
<th>Point</th>
<th>1/4 Span Loading</th>
<th>Midspan</th>
<th>3/4 Span Loading</th>
<th>Full Span Loading</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1/4</td>
<td>Midspan</td>
<td>3/4</td>
<td>1/4</td>
</tr>
<tr>
<td>G1</td>
<td>0.50</td>
<td>0.45</td>
<td>0.28</td>
<td>2.15</td>
</tr>
<tr>
<td>G2</td>
<td>0.49</td>
<td>0.64</td>
<td>0.29</td>
<td>2.19</td>
</tr>
<tr>
<td>G3</td>
<td>0.59</td>
<td>0.68</td>
<td>0.26</td>
<td>2.25</td>
</tr>
<tr>
<td>G4</td>
<td>0.57</td>
<td>0.75</td>
<td>0.32</td>
<td>2.24</td>
</tr>
<tr>
<td>G5</td>
<td>0.60</td>
<td>0.82</td>
<td>0.32</td>
<td>2.26</td>
</tr>
<tr>
<td>G6</td>
<td>0.62</td>
<td>0.81</td>
<td>0.34</td>
<td>2.23</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>3/4 Span Loading</th>
<th>Midspan</th>
<th>3/4</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>2.15</td>
<td>2.75</td>
<td>2.24</td>
</tr>
<tr>
<td>G2</td>
<td>2.19</td>
<td>2.94</td>
<td>2.19</td>
</tr>
<tr>
<td>G3</td>
<td>2.25</td>
<td>2.94</td>
<td>2.09</td>
</tr>
<tr>
<td>G4</td>
<td>2.24</td>
<td>3.02</td>
<td>2.22</td>
</tr>
<tr>
<td>G5</td>
<td>2.26</td>
<td>3.05</td>
<td>2.13</td>
</tr>
<tr>
<td>G6</td>
<td>2.23</td>
<td>3.03</td>
<td>2.17</td>
</tr>
</tbody>
</table>

3 Midspan measurement location was 5.02 m offset from actual midspan.

PREDICTIONS4 (Single Girder-Line Model in SAP 2000)

<table>
<thead>
<tr>
<th>Point</th>
<th>1/4</th>
<th>Midspan</th>
<th>3/4</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>3.01</td>
<td>4.03</td>
<td>3.01</td>
</tr>
<tr>
<td>G2</td>
<td>3.59</td>
<td>4.81</td>
<td>3.59</td>
</tr>
<tr>
<td>G3</td>
<td>3.59</td>
<td>4.81</td>
<td>3.59</td>
</tr>
<tr>
<td>G4</td>
<td>3.59</td>
<td>4.81</td>
<td>3.59</td>
</tr>
<tr>
<td>G5</td>
<td>3.59</td>
<td>4.81</td>
<td>3.59</td>
</tr>
<tr>
<td>G6</td>
<td>3.01</td>
<td>4.03</td>
<td>3.01</td>
</tr>
</tbody>
</table>

4 Using nominal slab thicknesses
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER: R-2547 (EB Bridge on US 64 Bypass over Smithfield Rd.)
MEASUREMENT DATE: August 24, 2004

GIRDER DEFLECTIONS
CROSS SECTION VIEW

1/4 Span Loading
"Midspan" Loading
3/4 Span Loading
Full Span Loading

1/4 Span

Deflection (inches)

G1 G2 G3 G4 G5 G6

Midspan

Deflection (inches)

G1 G2 G3 G4 G5 G6

3/4 Span

Deflection (inches)

G1 G2 G3 G4 G5 G6
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER:
R-2547 (EB Bridge on US 64 Bypass over Smithfield Rd.)

MEASUREMENT DATE:
August 24, 2004

GIRDER DEFLECTIONS
CROSS SECTION VIEW

GIRDER DEFLECTIONS
ELEVATION VIEW
ANSYS FINITE ELEMENT MODELING SUMMARY

PROJECT NUMBER: R-2547 (EB Bridge on US 64 Bypass over Smithfield Rd.)

MODEL PICTURE: (Steel Only, Oblique View)
## ANSYS FINITE ELEMENT MODELING SUMMARY

**PROJECT NUMBER:** R-2547 (EB Bridge on US 64 Bypass over Smithfield Rd.)

### MODEL DESCRIPTION

- **COMPONENT** Element Type
  - Girder: SHELL93
  - Connector Plates: SHELL93
  - Stiffener Plates: SHELL93
  - Cross-frame Members: LINK8 (diagonal)
  - LINK8 (horizontal)
  - End Diaphragm: BEAM4 (horizontal)
  - LINK8 (diagonal)
  - Stay-in-place Deck Forms: LINK8
  - Concrete Slab: SHELL63
  - Shear Studs: MPC184

### APPLIED LOADS

<table>
<thead>
<tr>
<th>Girder</th>
<th><em>Load</em></th>
<th>lb/ft</th>
<th>N/mm</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>1186.80</td>
<td>17.32</td>
<td></td>
</tr>
<tr>
<td>G2</td>
<td>1414.97</td>
<td>20.65</td>
<td></td>
</tr>
<tr>
<td>G3</td>
<td>1414.97</td>
<td>20.65</td>
<td></td>
</tr>
<tr>
<td>G4</td>
<td>1414.97</td>
<td>20.65</td>
<td></td>
</tr>
<tr>
<td>G5</td>
<td>1414.97</td>
<td>20.65</td>
<td></td>
</tr>
<tr>
<td>G6</td>
<td>1186.80</td>
<td>17.32</td>
<td></td>
</tr>
</tbody>
</table>

*applied as a uniform pressure to area of top flange

### GIRDER DEFLECTIONS

#### ANSYS vs ANSYS (SIP)

<table>
<thead>
<tr>
<th>Point</th>
<th>G1</th>
<th>G2</th>
<th>G3</th>
<th>G4</th>
<th>G5</th>
<th>G6</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/4</td>
<td>2.94</td>
<td>3.07</td>
<td>3.16</td>
<td>3.17</td>
<td>3.09</td>
<td>2.95</td>
</tr>
<tr>
<td>Midspan</td>
<td>3.98</td>
<td>4.16</td>
<td>4.28</td>
<td>4.29</td>
<td>4.17</td>
<td>3.99</td>
</tr>
<tr>
<td>3/4</td>
<td>2.94</td>
<td>3.09</td>
<td>3.17</td>
<td>3.16</td>
<td>3.07</td>
<td>2.94</td>
</tr>
</tbody>
</table>

#### Measured vs Predicted

<table>
<thead>
<tr>
<th>Point</th>
<th>G1</th>
<th>G2</th>
<th>G3</th>
<th>G4</th>
<th>G5</th>
<th>G6</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/4</td>
<td>2.29</td>
<td>2.35</td>
<td>2.43</td>
<td>2.42</td>
<td>2.43</td>
<td>2.34</td>
</tr>
<tr>
<td>Midspan</td>
<td>2.89</td>
<td>3.14</td>
<td>3.17</td>
<td>3.26</td>
<td>3.30</td>
<td>3.24</td>
</tr>
<tr>
<td>3/4</td>
<td>2.38</td>
<td>2.40</td>
<td>2.34</td>
<td>2.49</td>
<td>2.42</td>
<td>2.45</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Point</th>
<th>1/4</th>
<th>Midspan</th>
<th>3/4</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>3.01</td>
<td>4.03</td>
<td>3.01</td>
</tr>
<tr>
<td>G2</td>
<td>3.59</td>
<td>4.81</td>
<td>3.59</td>
</tr>
<tr>
<td>G3</td>
<td>3.59</td>
<td>4.81</td>
<td>3.59</td>
</tr>
<tr>
<td>G4</td>
<td>3.59</td>
<td>4.81</td>
<td>3.59</td>
</tr>
<tr>
<td>G5</td>
<td>3.59</td>
<td>4.81</td>
<td>3.59</td>
</tr>
<tr>
<td>G6</td>
<td>3.01</td>
<td>4.03</td>
<td>3.01</td>
</tr>
</tbody>
</table>
ANSYS FINITE ELEMENT MODELING SUMMARY

PROJECT NUMBER: R-2547 (EB Bridge on US 64 Bypass over Smithfield Rd.)

GIRDER DEFLECTIONS CROSS SECTION VIEW

Measured
ANSYS
ANSYS (SIP)
Predicted

Deflection (inches)

0.00 1.00 2.00 3.00 4.00 5.00
G1 G2 G3 G4 G5 G6

1/4 Span

Deflection (inches)

0.00 1.00 2.00 3.00 4.00 5.00
G1 G2 G3 G4 G5 G6

Midspan

Deflection (inches)

0.00 1.00 2.00 3.00 4.00 5.00
G1 G2 G3 G4 G5 G6

3/4 Span

182
Appendix D

Deflection Summary for the Wilmington St Bridge

This appendix contains a detailed description of the Wilmington St Bridge including bridge geometry, material data, cross frame type and size, and dead loads calculated from slab geometry. Illustrations detailing the bridge geometry and field measurement locations are included, along with tables and graphs of the field measured non-composite girder deflections.

A summary of the ANSYS finite element model created for the Wilmington St Bridge is also included in this appendix. This summary includes a picture of the ANSYS model, details about the elements used in the model generation, the loads applied to the model, and tables and graphs of the deflections predicted by the model.
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER: B-3257 (South Wilmington Street Bridge)
MEASUREMENT DATE: November 1, 2004

BRIDGE DESCRIPTION

TYPE One Span Simple
LENGTH 149.50 ft (44.85 m)
NUMBER OF GIRDERS 5
GIRDER SPACING 8.25 ft (2.475 m)
SKEW 152 deg
OVERHANG 3.042 ft (Overhang Side)
1 ft (ADJ to Stage I side)
BEARING TYPE Pot Bearing

MATERIAL DATA

STRUCTURAL STEEL
Girder: AASHTO M270 50 ksi (345 MPa)
Other: AASHTO M270 50 ksi (345 MPa)

CONCRETE UNIT WEIGHT 118 pcf (measured)
SIP FORM WEIGHT 3 psf (nominal)

GIRDER DATA

LENGTH 149.50 ft (44.85 m)
WEB THICKNESS 0.5 in (13 mm)
WEB DEPTH 54 in (1371.6 mm)

TOP FLANGE WIDTH 16 in (406.4 mm)
BOTTOM FLANGE WIDTH 20 in (508.0 mm)

<table>
<thead>
<tr>
<th>Flange Thickness</th>
<th>Begin</th>
<th>End</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 in (25.4 mm)</td>
<td>0.00</td>
<td>31.25 ft (9.375 m)</td>
</tr>
<tr>
<td>1.375 in (34.93 mm)</td>
<td>31.25 ft (9.375 m)</td>
<td>118.25 ft (35.475 m)</td>
</tr>
<tr>
<td>1 in (25.4 mm)</td>
<td>118.25 ft (35.475 m)</td>
<td>149.5 ft (44.85 m)</td>
</tr>
<tr>
<td>Bottom:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.125 ft (28.575 mm)</td>
<td>0.00</td>
<td>31.25 ft (9.375 m)</td>
</tr>
<tr>
<td>1.875 in (34.93 mm)</td>
<td>31.25 ft (9.375 m)</td>
<td>118.25 ft (35.475 m)</td>
</tr>
<tr>
<td>1.125 ft (28.575 mm)</td>
<td>118.25 ft (35.475 m)</td>
<td>149.5 ft (44.85 m)</td>
</tr>
</tbody>
</table>

CROSS-FRAME DATA

END BENT (Type K) Diagonals Horizontals Verticals
WT 4×12 C 15×50 (top) NA
MIDDLE BENT NA NA NA
INTERMEDIATE (Type K) L 3×3×5/16 L 3×3×5/16 (bottom) NA
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER: B-3257 (South Wilmington Street Bridge)
MEASUREMENT DATE: November 1, 2004

STIFFENERS

Longitudinal: NA
Bearing: PL 1" × 7" (25.4 mm × 177.8 mm)
Intermediate: PL 0.5 " x NA (12.7 mm x NA, connector Plate)
   No Intermediate Stiffeners
Middle Bearing: NA
End Bent Connector: PL 0.5 " x NA (12.7 mm x NA, connector Plate)

SLAB DATA

THICKNESS 8.5 in (215.9 mm) nominal
BUILD-UP 2.5 in (63.5 mm) nominal

LONGITUDINAL REBAR SIZE (US) SPACING (nominal)
Top: #4 18.0 in (457.2 mm)
Bottom: #5 10.0 in (254.0 mm)

TRANSVERSE REBAR
Top: #5 7.0 in (177.8 mm)
Bottom: #5 7.0 in (177.8 mm)

DECK LOADS

<table>
<thead>
<tr>
<th>Girder</th>
<th>Concrete 1 lb/ft</th>
<th>N/mm</th>
<th>Slab 2 lb/ft</th>
<th>N/mm</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>G6</td>
<td>518.71</td>
<td>7.57</td>
<td>549.54</td>
<td>8.02</td>
<td>0.94</td>
</tr>
<tr>
<td>G7</td>
<td>800.33</td>
<td>11.68</td>
<td>861.32</td>
<td>12.57</td>
<td>0.93</td>
</tr>
<tr>
<td>G8</td>
<td>769.50</td>
<td>11.23</td>
<td>831.17</td>
<td>12.13</td>
<td>0.93</td>
</tr>
<tr>
<td>G9</td>
<td>800.33</td>
<td>11.68</td>
<td>861.32</td>
<td>12.57</td>
<td>0.93</td>
</tr>
<tr>
<td>G10</td>
<td>743.46</td>
<td>10.85</td>
<td>774.30</td>
<td>11.30</td>
<td>0.96</td>
</tr>
</tbody>
</table>

1 Calculated with measured slab thicknesses
2 Includes slab, buildups, and stay-in-place forms (nominal)
FIELD MEASUREMENT SUMMARY

Project Number: B-3257 (South Wilmington Street Bridge)
Measurement Date: November 1, 2004

Girder Centerline: ————
Measurement Location: ●

(a) Plan View (Not to Scale)

Girder: ❀❑❑❑❒
String Pots: ■
Fixed Support: ▲
Expansion Support: ●

(b) Elevation View (Not to Scale)

Plan and Elevation View of the Wilmington St Bridge (Raleigh, NC)
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER: B-3257 (South Wilmington Street Bridge)
MEASUREMENT DATE: November 1, 2004

BEARING SETTLEMENTS (data in inches)

<table>
<thead>
<tr>
<th>Point</th>
<th>End 1</th>
<th>End 2</th>
<th>Avg.</th>
</tr>
</thead>
<tbody>
<tr>
<td>G6</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>G7</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>G8</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>G9</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>G10</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
</tbody>
</table>

GIRDER DEFLECTIONS (data in inches)

MEASURED

<table>
<thead>
<tr>
<th>Point</th>
<th>1/4 Span Loading</th>
<th>Midspan</th>
<th>3/4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1/4 Midspan 3/4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>G6</td>
<td>1.52 2.01 1.30</td>
<td>2.28 3.01 2.13</td>
<td></td>
</tr>
<tr>
<td>G7</td>
<td>1.42 1.75 1.04</td>
<td>2.20 2.81 1.85</td>
<td></td>
</tr>
<tr>
<td>G8</td>
<td>1.36 1.65 0.94</td>
<td>2.15 2.80 1.79</td>
<td></td>
</tr>
<tr>
<td>G9</td>
<td>1.38 1.67 1.00</td>
<td>2.20 2.99 1.93</td>
<td></td>
</tr>
<tr>
<td>G10</td>
<td>1.59 1.93 1.04</td>
<td>2.60 3.46 2.15</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Point</th>
<th>3/4 Span Loading</th>
<th>Full Span Loading</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1/4 Midspan 3/4</td>
<td>1/4 Midspan 3/4</td>
</tr>
<tr>
<td>G6</td>
<td>2.68 3.88 2.93</td>
<td>2.64 3.80 2.83</td>
</tr>
<tr>
<td>G7</td>
<td>2.70 3.76 2.72</td>
<td>2.64 3.70 2.72</td>
</tr>
<tr>
<td>G8</td>
<td>2.72 3.84 2.80</td>
<td>2.72 3.78 2.80</td>
</tr>
<tr>
<td>G9</td>
<td>2.97 4.25 3.17</td>
<td>2.97 4.19 3.17</td>
</tr>
<tr>
<td>G10</td>
<td>3.66 5.10 3.64</td>
<td>3.62 5.04 3.70</td>
</tr>
</tbody>
</table>

3 Midspan measurement location was 14.95 ft offset from actual midspan.

PREDICTIONS 4 (Single Girder-Line Model in SAP 2000)

<table>
<thead>
<tr>
<th>Point</th>
<th>1/4</th>
<th>Midspan</th>
<th>3/4</th>
</tr>
</thead>
<tbody>
<tr>
<td>G6</td>
<td>2.94</td>
<td>3.86</td>
<td>2.94</td>
</tr>
<tr>
<td>G7</td>
<td>4.53</td>
<td>5.96</td>
<td>4.53</td>
</tr>
<tr>
<td>G8</td>
<td>4.36</td>
<td>5.73</td>
<td>4.36</td>
</tr>
<tr>
<td>G9</td>
<td>4.53</td>
<td>5.96</td>
<td>4.53</td>
</tr>
<tr>
<td>G10</td>
<td>4.21</td>
<td>5.54</td>
<td>4.21</td>
</tr>
</tbody>
</table>

4 Using measured slab thicknesses
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER: B-3257 (South Wilmington Street Bridge)
MEASUREMENT DATE: November 1, 2004

GIRDER DEFLECTIONS
CROSS SECTION VIEW

1/4 Span Loading
"Midspan" Loading
3/4 Span Loading
Full Span Loading

Deflection (inches)

1/4 Span
Midspan
3/4 Span

G6 G7 G8 G9 G10
G6 G7 G8 G9 G10
G6 G7 G8 G9 G10

Deflection (inches)

188
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER: B-3257 (South Wilmington Street Bridge)
MEASUREMENT DATE: November 1, 2004

GIRDER DEFLECTIONS
CROSS SECTION VIEW

GIRDER DEFLECTIONS
ELEVATION VIEW

Measured
Predicted

Deflection (inches)

G6 G7 G8 G9 G10
"Midspan"

Girder 1
Girder 2
Girder 3
Girder 4
Girder 5

Deflection (inches)

1/4 Span Midspan 3/4 Span

Location Along Span
ANSYS FINITE ELEMENT MODELING SUMMARY

PROJECT NUMBER: B-3257 (South Wilmington Street Bridge)

MODEL DESCRIPTION: (Steel Only, Isometric View)
ANSYS FINITE ELEMENT MODELING SUMMARY

PROJECT NUMBER: B-3257 (South Wilmington Street Bridge)

MODEL DESCRIPTION

<table>
<thead>
<tr>
<th>COMPONENT</th>
<th>Element Type</th>
<th>APPLIED LOADS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Girder</td>
<td>SHELL93</td>
<td></td>
</tr>
<tr>
<td>Connector Plates</td>
<td>SHELL93</td>
<td></td>
</tr>
<tr>
<td>Stiffener Plates</td>
<td>SHELL93</td>
<td></td>
</tr>
<tr>
<td>Cross-frame Members</td>
<td>LINK8 (diagonal)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>LINK8 (horizontal)</td>
<td></td>
</tr>
<tr>
<td>End Diaphragm</td>
<td>BEAM4 (horizontal)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>LINK8 (diagonal)</td>
<td></td>
</tr>
<tr>
<td>Stay-in-place Deck Forms</td>
<td>LINK8</td>
<td></td>
</tr>
<tr>
<td>Concrete Slab</td>
<td>SHELL63</td>
<td></td>
</tr>
<tr>
<td>Shear Studs</td>
<td>MPC184</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Girder</th>
<th>*Load</th>
<th>lb/ft</th>
<th>N/mm</th>
</tr>
</thead>
<tbody>
<tr>
<td>G6</td>
<td>447.17</td>
<td>6.52</td>
<td></td>
</tr>
<tr>
<td>G7</td>
<td>686.15</td>
<td>10.01</td>
<td></td>
</tr>
<tr>
<td>G8</td>
<td>655.73</td>
<td>9.57</td>
<td></td>
</tr>
<tr>
<td>G9</td>
<td>686.15</td>
<td>10.01</td>
<td></td>
</tr>
<tr>
<td>G10</td>
<td>619.68</td>
<td>9.04</td>
<td></td>
</tr>
</tbody>
</table>

*applied as a uniform pressure to area of top flange

GIRDER DEFLECTIONS

<table>
<thead>
<tr>
<th>Point</th>
<th>ANSYS</th>
<th>ANSYS (SIP)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1/4</td>
<td>Midspan</td>
</tr>
<tr>
<td>G6</td>
<td>2.05</td>
<td>2.74</td>
</tr>
<tr>
<td>G7</td>
<td>2.51</td>
<td>3.41</td>
</tr>
<tr>
<td>G8</td>
<td>2.63</td>
<td>3.60</td>
</tr>
<tr>
<td>G9</td>
<td>2.63</td>
<td>3.56</td>
</tr>
<tr>
<td>G10</td>
<td>2.77</td>
<td>3.72</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Point</th>
<th>Measured</th>
<th>Predicted</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1/4</td>
<td>Midspan</td>
</tr>
<tr>
<td>G6</td>
<td>2.64</td>
<td>3.80</td>
</tr>
<tr>
<td>G7</td>
<td>2.64</td>
<td>3.70</td>
</tr>
<tr>
<td>G8</td>
<td>2.72</td>
<td>3.78</td>
</tr>
<tr>
<td>G9</td>
<td>2.97</td>
<td>4.19</td>
</tr>
<tr>
<td>G10</td>
<td>3.62</td>
<td>5.04</td>
</tr>
</tbody>
</table>
ANSYS FINITE ELEMENT MODELING SUMMARY

PROJECT NUMBER: B-3257 (South Wilmington Street Bridge)

GIRDER DEFLECTIONS
CROSS SECTION VIEW

0.00
1.00
2.00
3.00
4.00
5.00
6.00

G6 G7 G8 G9 G10

1/4 Span

Deflection (inches)

Measured
ANSYS
ANSYS (SIP)
Predicted

0.00
1.00
2.00
3.00
4.00
5.00
6.00

G6 G7 G8 G9 G10

Midspan

Deflection (inches)

0.00
1.00
2.00
3.00
4.00
5.00
6.00

G6 G7 G8 G9 G10

3/4 Span

Deflection (inches)
Appendix E

Deflection Summary for Bridge 14

This appendix contains a detailed description of Bridge 14 including bridge geometry, material data, cross frame type and size, and dead loads calculated from slab geometry. Illustrations detailing the bridge geometry and field measurement locations are included, along with tables and graphs of the field measured non-composite girder deflections.

A summary of the ANSYS finite element model created for Bridge 14 is also included in this appendix. This summary includes a picture of the ANSYS model, details about the elements used in the model generation, the loads applied to the model, and tables and graphs of the deflections predicted by the model.

193
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER: R-2547 (Ramp (RPBDY1) Over US-64 Business)
MEASUREMENT DATE: June 29 & July 2, 2004

BRIDGE DESCRIPTION

<table>
<thead>
<tr>
<th>TYPE</th>
<th>Two Span Continuous</th>
</tr>
</thead>
<tbody>
<tr>
<td>LENGTH</td>
<td>208.26 ft (63.477 m)</td>
</tr>
<tr>
<td>NUMBER OF GIRDERS</td>
<td>5</td>
</tr>
<tr>
<td>GIRDER SPACING</td>
<td>9.97 ft (3.04 m)</td>
</tr>
<tr>
<td>SKEW</td>
<td>65.6 deg</td>
</tr>
<tr>
<td>OVERHANG</td>
<td>3.70 ft (1130 mm)</td>
</tr>
<tr>
<td>BEARING TYPE</td>
<td>Elastomeric Pad</td>
</tr>
</tbody>
</table>

MATERIAL DATA

<table>
<thead>
<tr>
<th>STRUCTURAL STEEL</th>
<th>Grade</th>
<th>Yield Strength</th>
</tr>
</thead>
<tbody>
<tr>
<td>Girder: AASHTO M270</td>
<td>50 ksi (345 MPa)</td>
<td></td>
</tr>
<tr>
<td>Other: AASHTO M270</td>
<td>50 ksi (345 MPa)</td>
<td></td>
</tr>
</tbody>
</table>

| CONCRETE UNIT WEIGHT | 150 pcf (nominal) |
| SIP FORM WEIGHT     | 2.98 psf (CSI Catalog) |

GIRDER DATA

| LENGTH | "Span A" | "Span B"
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>101.92 ft (31.064 m)</td>
<td>106.34 ft (32.413 m)</td>
<td></td>
</tr>
</tbody>
</table>

| WEB THICKNESS | 0.47 in (12 mm) |
| WEB DEPTH    | 62.99 in (1600 mm) |

| TOP FLANGE WIDTH | 14.96 in (380 mm) |
| BOTTOM FLANGE WIDTH | 17.72 in (450 mm) |

<table>
<thead>
<tr>
<th>Flange Thickness</th>
<th>Begin</th>
<th>End</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top: 0.79 in (20 mm)</td>
<td>0.00</td>
<td>92.07 ft (28.064 m)</td>
</tr>
<tr>
<td>1.18 in (30 mm)</td>
<td>92.07 ft (28.064 m)</td>
<td>111.76 ft (34.064 m)</td>
</tr>
<tr>
<td>0.79 in (20 mm)</td>
<td>111.76 ft (34.064 m)</td>
<td>208.26 ft (63.477 m)</td>
</tr>
<tr>
<td>Bottom: 0.79 in (20 mm)</td>
<td>0.00</td>
<td>92.07 ft (28.064 m)</td>
</tr>
<tr>
<td>1.38 in (35 mm)</td>
<td>92.07 ft (28.064 m)</td>
<td>111.76 ft (34.064 m)</td>
</tr>
<tr>
<td>0.79 in (20 mm)</td>
<td>111.76 ft (34.064 m)</td>
<td>208.26 ft (63.477 m)</td>
</tr>
</tbody>
</table>
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER: R-2547 (Ramp (RPBDY1) Over US-64 Business)
MEASUREMENT DATE: June 29 & July 2, 2004

STIFFENERS

Longitudinal: N/A
Bearing: PL 0.98" × 8.27" (25 mm × 210 mm)
Intermediate: PL 0.63" × NA (16 mm × NA, connector plate)
               PL 0.47" × 5.12" (12 mm × 130 mm)
Middle Bearing: PL 0.98" × 8.27" (25 mm × 210 mm)
End Bent Connector: NA (Integral Bent)

CROSS-FRAME DATA

<table>
<thead>
<tr>
<th>END BENT</th>
<th>Diagonals</th>
<th>Horizontals</th>
<th>Verticals</th>
</tr>
</thead>
<tbody>
<tr>
<td>MIDDLE BENT (Type X)</td>
<td>L 4 x 4 x 5/8&quot;</td>
<td>L 4 x 4 x 5/8&quot; (Bottom)</td>
<td>NA</td>
</tr>
<tr>
<td>INTERMEDIATE (Type X)</td>
<td>L 4 x 4 x 5/8&quot;</td>
<td>L 4 x 4 x 5/8&quot; (Bottom)</td>
<td>NA</td>
</tr>
</tbody>
</table>

SLAB DATA

THICKNESS 8.86 in (225 mm) nominal
BUILD-UP 2.95 in (75 mm) nominal

Over Middle Bent:

<table>
<thead>
<tr>
<th>LONGITUDINAL REBAR</th>
<th>SIZE (metric)</th>
<th>SPACING (nominal)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top: #16</td>
<td>4.33 in (110 mm)</td>
<td></td>
</tr>
<tr>
<td>Bottom: #16</td>
<td>8.66 in (220 mm)</td>
<td></td>
</tr>
</tbody>
</table>

TRANSVERSE REBAR

| Top: #16 | 5.91 in (150 mm) |
| Bottom: #16 | 5.91 in (150 mm) |

Otherwise:

<table>
<thead>
<tr>
<th>LONGITUDINAL REBAR</th>
<th>SIZE (metric)</th>
<th>SPACING (nominal)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top: #16</td>
<td>21.65 in (550 mm)</td>
<td></td>
</tr>
<tr>
<td>Bottom: #16</td>
<td>8.66 in (220 mm)</td>
<td></td>
</tr>
</tbody>
</table>

TRANSVERSE REBAR

| Top: #16 | 5.91 in (150 mm) |
| Bottom: #16 | 5.91 in (150 mm) |
FIELD MEASUREMENT SUMMARY

Project Number: R-2547 (Ramp (RBPDY1) over US-64 Business)
Measurement Date: June 29 & July 2, 2004

Girder Centerline:  
Construction Joint:  
Measurement Location:  

Span A = 101.92 ft (31.06 m)
Span B = 106.34 ft (32.41 m)

(a) Plan View (Not to Scale)

(b) Elevation View (Not to Scale)

Plan and Elevation View of Bridge 14 (Knightdale, NC)
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER: R-2547 (Ramp (RPBDY1) Over US-64 Business)
MEASUREMENT DATE: June 29 & July 2, 2004

DECK LOADS

<table>
<thead>
<tr>
<th>Girder</th>
<th>Concrete 1</th>
<th>Slab 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>lb/ft</td>
<td>N/mm</td>
</tr>
<tr>
<td>G1</td>
<td>1160.07</td>
<td>16.93</td>
</tr>
<tr>
<td>G2</td>
<td>1204.61</td>
<td>17.58</td>
</tr>
<tr>
<td>G3</td>
<td>1204.61</td>
<td>17.58</td>
</tr>
<tr>
<td>G4</td>
<td>1204.61</td>
<td>17.58</td>
</tr>
<tr>
<td>G5</td>
<td>1160.07</td>
<td>16.93</td>
</tr>
</tbody>
</table>

1 Calculated with nominal slab thicknesses
2 Includes slab, buildups, and stay-in-place forms (nominal)

BEARING SETTLEMENTS 3 (data in inches, negative is deflection upwards)

<table>
<thead>
<tr>
<th>Point</th>
<th>Pour 1 Settlement</th>
<th>Pour 2 Settlement</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>---</td>
<td>-0.03</td>
</tr>
<tr>
<td>G2</td>
<td>---</td>
<td>-0.03</td>
</tr>
<tr>
<td>G3</td>
<td>---</td>
<td>-0.02</td>
</tr>
<tr>
<td>G4</td>
<td>---</td>
<td>-0.03</td>
</tr>
<tr>
<td>G5</td>
<td>---</td>
<td>-0.05</td>
</tr>
</tbody>
</table>

3 Noticeably, the settlement totaled from the two pours was very close to zero.

GIRDER DEFLECTIONS (data in inches, negative is deflection upwards)

POUR 1 MEASURED

<table>
<thead>
<tr>
<th>Point</th>
<th>7/10 Span B Loading</th>
<th>End of Span B</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>-0.07</td>
<td>0.23</td>
</tr>
<tr>
<td>G2</td>
<td>0.00</td>
<td>0.26</td>
</tr>
<tr>
<td>G3</td>
<td>0.01</td>
<td>0.25</td>
</tr>
<tr>
<td>G4</td>
<td>-0.07</td>
<td>0.20</td>
</tr>
<tr>
<td>G5</td>
<td>-0.09</td>
<td>0.27</td>
</tr>
</tbody>
</table>
## FIELD MEASUREMENT SUMMARY

**PROJECT NUMBER:** R-2547 (Ramp (RPBDY1) Over US-64 Business)  
**MEASUREMENT DATE:** June 29 & July 2, 2004

### GIRDER DEFLECTIONS (data in inches, negative is deflection upwards)

#### POUR 2 MEASURED

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>0.28</td>
<td>0.63</td>
<td>0.52</td>
<td>0.11</td>
<td>0.43</td>
<td>0.38</td>
<td>0.43</td>
<td>0.30</td>
<td>0.29</td>
</tr>
<tr>
<td>G2</td>
<td>-0.19</td>
<td>0.48</td>
<td>0.50</td>
<td>0.14</td>
<td>0.30</td>
<td>0.37</td>
<td>0.38</td>
<td>0.18</td>
<td>0.28</td>
</tr>
<tr>
<td>G3</td>
<td>-0.18</td>
<td>0.49</td>
<td>0.58</td>
<td>0.17</td>
<td>0.31</td>
<td>0.44</td>
<td>0.37</td>
<td>0.19</td>
<td>0.35</td>
</tr>
<tr>
<td>G4</td>
<td>-0.25</td>
<td>0.50</td>
<td>0.52</td>
<td>0.27</td>
<td>0.30</td>
<td>0.35</td>
<td>0.42</td>
<td>0.18</td>
<td>0.28</td>
</tr>
<tr>
<td>G5</td>
<td>-0.10</td>
<td>0.49</td>
<td>0.43</td>
<td>0.29</td>
<td>0.25</td>
<td>0.26</td>
<td>0.37</td>
<td>0.13</td>
<td>0.18</td>
</tr>
</tbody>
</table>

#### TOTAL MEASURED

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>1.19</td>
<td>0.14</td>
<td>0.16</td>
<td>1.36</td>
<td>0.07</td>
<td>0.13</td>
</tr>
<tr>
<td>G2</td>
<td>1.04</td>
<td>0.02</td>
<td>0.14</td>
<td>1.16</td>
<td>-0.05</td>
<td>0.11</td>
</tr>
<tr>
<td>G3</td>
<td>1.09</td>
<td>0.03</td>
<td>0.22</td>
<td>1.21</td>
<td>-0.02</td>
<td>0.20</td>
</tr>
<tr>
<td>G4</td>
<td>1.18</td>
<td>0.03</td>
<td>0.14</td>
<td>1.27</td>
<td>-0.02</td>
<td>0.15</td>
</tr>
<tr>
<td>G5</td>
<td>0.90</td>
<td>-0.02</td>
<td>0.04</td>
<td>0.91</td>
<td>-0.04</td>
<td>0.07</td>
</tr>
</tbody>
</table>

### PREDICTIONS\(^4\) (Single Girder-Line Model in SAP 2000)

#### Pour 1

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>-0.45</td>
<td>0.93</td>
<td>1.39</td>
<td>1.43</td>
<td>-0.17</td>
<td>-0.09</td>
</tr>
<tr>
<td>G2</td>
<td>-0.47</td>
<td>0.97</td>
<td>1.44</td>
<td>1.49</td>
<td>-0.17</td>
<td>-0.17</td>
</tr>
<tr>
<td>G3</td>
<td>-0.47</td>
<td>0.97</td>
<td>1.44</td>
<td>1.49</td>
<td>-0.17</td>
<td>-0.17</td>
</tr>
<tr>
<td>G4</td>
<td>-0.47</td>
<td>0.97</td>
<td>1.44</td>
<td>1.49</td>
<td>-0.17</td>
<td>-0.17</td>
</tr>
<tr>
<td>G5</td>
<td>-0.45</td>
<td>0.93</td>
<td>1.39</td>
<td>1.43</td>
<td>-0.17</td>
<td>-0.09</td>
</tr>
</tbody>
</table>

#### Pour 2

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>0.98</td>
<td>0.76</td>
<td>1.30</td>
<td>1.02</td>
<td>0.80</td>
<td>1.26</td>
</tr>
<tr>
<td>G2</td>
<td>1.02</td>
<td>0.80</td>
<td>1.26</td>
<td>1.02</td>
<td>0.80</td>
<td>1.26</td>
</tr>
<tr>
<td>G3</td>
<td>1.02</td>
<td>0.80</td>
<td>1.26</td>
<td>1.02</td>
<td>0.80</td>
<td>1.26</td>
</tr>
<tr>
<td>G4</td>
<td>1.02</td>
<td>0.80</td>
<td>1.26</td>
<td>1.02</td>
<td>0.80</td>
<td>1.26</td>
</tr>
<tr>
<td>G5</td>
<td>0.98</td>
<td>0.76</td>
<td>1.30</td>
<td>1.02</td>
<td>0.80</td>
<td>1.26</td>
</tr>
</tbody>
</table>

### Super-Imposed Total

<table>
<thead>
<tr>
<th>Point</th>
<th>4/10 A</th>
<th>3/10 B</th>
<th>6/10 B</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>0.98</td>
<td>0.76</td>
<td>1.30</td>
</tr>
<tr>
<td>G2</td>
<td>1.02</td>
<td>0.80</td>
<td>1.26</td>
</tr>
<tr>
<td>G3</td>
<td>1.02</td>
<td>0.80</td>
<td>1.26</td>
</tr>
<tr>
<td>G4</td>
<td>1.02</td>
<td>0.80</td>
<td>1.26</td>
</tr>
<tr>
<td>G5</td>
<td>0.98</td>
<td>0.76</td>
<td>1.30</td>
</tr>
</tbody>
</table>

\(^4\) Using nominal slab thicknesses
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER: R-2547 (Ramp (RPBDY1) Over US-64 Business)
MEASUREMENT DATE: June 29 & July 2, 2004

GIRDER DEFLECTIONS
CROSS SECTION VIEW

-1.00 0.00 1.00 2.00

G1 G2 G3 G4 G5

Deflection (inches)

Pour 1 Measured
Pour 2 Measured
Total Measured

G1 G2 G3 G4 G5

Deflection (inches)

G1 G2 G3 G4 G5

Deflection (inches)

199

199

199

199
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER: R-2547 (Ramp (RPBDY1) Over US-64 Business)
MEASUREMENT DATE: June 29 & July 2, 2004

GIRDER DEFLECTIONS CROSS SECTION VIEW

G1 G2 G3 G4 G5

3/10 Span B

Deflection (inches)

Pour 1 Measured
Pour 1 Predicted
Pour 2 Measured
Pour 2 Predicted
Total Measured
Total Predicted

4/10 Span A

Deflection (inches)

5/10 Span B

Deflection (inches)

6/10 Span B

Deflection (inches)
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER: R-2547 (Ramp (RPBDY1) Over US-64 Business)
MEASUREMENT DATE: June 29 & July 2, 2004

GIRDER DEFLECTIONS
ELEVATION VIEW

- POUR 1

- POUR 2

- TOTAL

201
ANSYS FINITE ELEMENT MODELING SUMMARY

PROJECT NUMBER: R-2547 (Ramp (RPBDY1) Over US-64 Business)

MODEL PICTURE: (Steel Only, Oblique View)
ANSYS FINITE ELEMENT MODELING SUMMARY

PROJECT NUMBER:  R-2547 (Ramp (RPBDY1) Over US-64 Business)

MODEL DESCRIPTION
- Component: Element Type
  - Girder: SHELL93
  - Connector Plates: SHELL93
  - Stiffener Plates: SHELL93
  - Cross-frame Members: LINK8 (diagonal)
  - LINK8 (horizontal)
  - Middle Diaphragm: LINK8 (diagonal)
  - LINK8 (horizontal)
  - Stay-in-place Deck Forms: LINK8
  - Concrete Slab: SHELL63
  - Shear Studs: MPC184

APPLIED LOADS

<table>
<thead>
<tr>
<th>Component</th>
<th>Element Type</th>
<th>Girder</th>
<th>*Load</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>lb/ft</td>
<td>N/mm</td>
</tr>
<tr>
<td>Girder</td>
<td>SHELL93</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Connector Plates</td>
<td>SHELL93</td>
<td>G1</td>
<td>1229.3</td>
</tr>
<tr>
<td>Stiffener Plates</td>
<td>SHELL93</td>
<td>G2</td>
<td>1296.4</td>
</tr>
<tr>
<td>Cross-frame Members</td>
<td>LINK8 (diagonal)</td>
<td>G3</td>
<td>1296.4</td>
</tr>
<tr>
<td></td>
<td>LINK8 (horizontal)</td>
<td>G4</td>
<td>1296.4</td>
</tr>
<tr>
<td></td>
<td>LINK8 (diagonal)</td>
<td>G5</td>
<td>1229.3</td>
</tr>
<tr>
<td>Middle Diaphragm</td>
<td>LINK8 (horizontal)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*applied as a uniform pressure to area of top flange

<table>
<thead>
<tr>
<th>Point</th>
<th>ANSYS Pour 1</th>
<th>ANSYS Pour 1 (SIP)</th>
<th>Pour 1 Measured</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>-0.45</td>
<td>0.96</td>
<td>1.43</td>
</tr>
<tr>
<td>G2</td>
<td>-0.45</td>
<td>0.96</td>
<td>1.43</td>
</tr>
<tr>
<td>G3</td>
<td>-0.45</td>
<td>0.97</td>
<td>1.43</td>
</tr>
<tr>
<td>G4</td>
<td>-0.45</td>
<td>0.96</td>
<td>1.43</td>
</tr>
<tr>
<td>G5</td>
<td>-0.45</td>
<td>0.95</td>
<td>1.41</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Point</th>
<th>ANSYS Pour 2</th>
<th>ANSYS Pour 2 (SIP)</th>
<th>Pour 2 Measured</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>1.45</td>
<td>-0.15</td>
<td>-0.07</td>
</tr>
<tr>
<td>G2</td>
<td>1.48</td>
<td>-0.15</td>
<td>-0.07</td>
</tr>
<tr>
<td>G3</td>
<td>1.50</td>
<td>-0.15</td>
<td>-0.07</td>
</tr>
<tr>
<td>G4</td>
<td>1.49</td>
<td>-0.14</td>
<td>-0.07</td>
</tr>
<tr>
<td>G5</td>
<td>1.45</td>
<td>-0.15</td>
<td>-0.07</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Point</th>
<th>ANSYS Total</th>
<th>ANSYS Total (SIP)</th>
<th>Total Measured</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>1.00</td>
<td>0.81</td>
<td>1.35</td>
</tr>
<tr>
<td>G2</td>
<td>1.03</td>
<td>0.82</td>
<td>1.36</td>
</tr>
<tr>
<td>G3</td>
<td>1.05</td>
<td>0.82</td>
<td>1.36</td>
</tr>
<tr>
<td>G4</td>
<td>1.04</td>
<td>0.82</td>
<td>1.36</td>
</tr>
<tr>
<td>G5</td>
<td>1.00</td>
<td>0.81</td>
<td>1.34</td>
</tr>
</tbody>
</table>

Note: When ANSYS numbers were compared with ANSYS (SIP) numbers, there was 1% difference, therefore, ANSYS with SIP will not be shown on graphs.
ANSYS FINITE ELEMENT MODELING SUMMARY

PROJECT NUMBER: R-2547 (Ramp (RPBDY1) Over US-64 Business)

GIRDER DEFLECTIONS CROSS SECTION VIEW

Measured
ANSYS (no SIP)
SAP Prediction

0.00
1.00
2.00

Deflection (inches)

0.00
1.00
2.00

Deflection (inches)

0.00
1.00
2.00

Deflection (inches)

G1 G2 G3 G4 G5

4/10 Span A

G1 G2 G3 G4 G5

3/10 Span B

G1 G2 G3 G4 G5

6/10 Span B

204
Appendix F

Deflection Summary for Bridge 10

This appendix contains a detailed description of Bridge 10 including bridge geometry, material data, cross frame type and size, and dead loads calculated from slab geometry. Illustrations detailing the bridge geometry and field measurement locations are included, along with tables and graphs of the field measured non-composite girder deflections.

A summary of the ANSYS finite element model created for Bridge 10 is also included in this appendix. This summary includes a picture of the ANSYS model, details about the elements used in the model generation, the loads applied to the model, and tables and graphs of the deflections predicted by the model.
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER: R-2547 (Knightdale-Eagle Rock Rd. Over US-64 Bypass)
MEASUREMENT DATE: March 20 & March 29, 2004

BRIDGE DESCRIPTION

<table>
<thead>
<tr>
<th>TYPE</th>
<th>Two Span Continous, Two Simple Spans (Continuous Spans Measured)</th>
</tr>
</thead>
<tbody>
<tr>
<td>LENGTH</td>
<td>300.19 ft (91.5 m)</td>
</tr>
<tr>
<td>NUMBER OF GIRDERS</td>
<td>4</td>
</tr>
<tr>
<td>GIRDER SPACING</td>
<td>9.51 ft (2.9 m)</td>
</tr>
<tr>
<td>SKEW</td>
<td>147.1 deg</td>
</tr>
<tr>
<td>OVERHANG</td>
<td>3.02 ft (920 mm) (from web centerline)</td>
</tr>
<tr>
<td>BEARING TYPE</td>
<td>Elastomeric Pad</td>
</tr>
</tbody>
</table>

MATERIAL DATA

<table>
<thead>
<tr>
<th>STRUCTURAL STEEL</th>
<th>Grade</th>
<th>Yield Strength</th>
</tr>
</thead>
<tbody>
<tr>
<td>Girder:</td>
<td>AASHTO M270</td>
<td>50 ksi (345 MPa)</td>
</tr>
<tr>
<td>Other:</td>
<td>AASHTO M270</td>
<td>50 ksi (345 MPa)</td>
</tr>
</tbody>
</table>

| CONCRETE UNIT WEIGHT | 150 pcf (nominal) |
| SIP FORM WEIGHT      | 2.57 psf (CSI Catalog) |

GIRDER DATA

<table>
<thead>
<tr>
<th>LENGTH</th>
<th>&quot;Span B&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>155.51 ft (47.4 m)</td>
</tr>
<tr>
<td></td>
<td>144.68 ft (44.1 m)</td>
</tr>
<tr>
<td>WEB THICKNESS</td>
<td>0.55 in (14 mm)</td>
</tr>
<tr>
<td>WEB DEPTH</td>
<td>75.79 in (1925 mm)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Flange Thickness</th>
<th>Begin</th>
<th>End</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top: 1.26 in (32 mm)</td>
<td>0.00</td>
<td>112.86 ft (34.4 m)</td>
</tr>
<tr>
<td>1.26 in (32 mm)</td>
<td></td>
<td>132.55 ft (40.4 m)</td>
</tr>
<tr>
<td>1.97 in (50 mm)</td>
<td></td>
<td>178.48 ft (54.4 m)</td>
</tr>
<tr>
<td>1.26 in (32 mm)</td>
<td></td>
<td>199.80 ft (60.9 m)</td>
</tr>
<tr>
<td>1.26 in (32 mm)</td>
<td></td>
<td>300.19 ft (91.5 m)</td>
</tr>
<tr>
<td>Flange Width</td>
<td>Begin</td>
<td>End</td>
</tr>
<tr>
<td>---------------------</td>
<td>--------</td>
<td>-------------------</td>
</tr>
<tr>
<td>15.75 in (400 mm)</td>
<td>0.00</td>
<td>112.86 ft (34.4 m)</td>
</tr>
<tr>
<td>18.50 in (470 mm)</td>
<td></td>
<td>132.55 ft (40.4 m)</td>
</tr>
<tr>
<td>18.50 in (470 mm)</td>
<td></td>
<td>178.48 ft (54.4 m)</td>
</tr>
<tr>
<td>18.50 in (470 mm)</td>
<td></td>
<td>199.80 ft (60.9 m)</td>
</tr>
<tr>
<td>15.75 in (400 mm)</td>
<td></td>
<td>300.19 ft (91.5 m)</td>
</tr>
</tbody>
</table>

Bottom: Same as Top Flange
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER: R-2547 (Knightdale-Eagle Rock Rd. Over US-64 Bypass)
MEASUREMENT DATE: March 20 & March 29, 2004

STIFFENERS

Longitudinal: N/A
Bearing: PL 0.79" × 7.09" (20 mm × 180 mm)
Intermediate: PL 0.47" × NA (12 mm × NA, connector plate)
               PL 0.55" × 5.91" (14 mm × 150 mm)
Middle Bearing: PL 1.10" × 8.27" (28 mm × 210 mm)
End Bent Connector: PL 0.47" × NA (12 mm × NA)

CROSS-FRAME DATA

<table>
<thead>
<tr>
<th></th>
<th>Diagonals</th>
<th>Horizontals</th>
<th>Verticals</th>
</tr>
</thead>
<tbody>
<tr>
<td>END BENT (Type K)</td>
<td>WT 4×12</td>
<td>MC 18×42.7</td>
<td>WT 4×12</td>
</tr>
<tr>
<td></td>
<td></td>
<td>WT 4×12</td>
<td></td>
</tr>
<tr>
<td>MIDDLE BENT</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>INTERMEDIATE (Type X)</td>
<td>WT 4×12</td>
<td>WT 4×12 (bottom)</td>
<td>NA</td>
</tr>
</tbody>
</table>

SLAB DATA

THICKNESS 8.86 in (225 mm) nominal
BUILD-UP 2.56 in (65 mm) nominal

Over Middle Bent:
LONGITUDINAL REBAR SIZE (metric) SPACING (nominal)
Top: #19 6.69 in (170 mm)
Bottom: #16 9.45 in (240 mm)
TRANSVERSE REBAR
Top: #16 6.30 in (160 mm)
Bottom: #16 6.30 in (160 mm)

Otherwise:
LONGITUDINAL REBAR SIZE (metric) SPACING (nominal)
Top: #16 13.39 in (340 mm)
Bottom: #16 9.45 in (240 mm)
TRANSVERSE REBAR
Top: #16 6.30 in (160 mm)
Bottom: #16 6.30 in (160 mm)
FIELD MEASUREMENT SUMMARY

Project Number: R-2547 (Knightdale-Eagle Rock Rd. over US-64 Bypass)
Measurement Date: March 20 & March 29, 2004

Girder Centerline: Span B = 155.51 ft (47.40 m)
Construction Joint: Span C = 144.69 ft (44.10 m)
Measurement Location:

(a) Plan View (Not to Scale)

(b) Elevation View (Not to Scale)

Plan and Elevation View of Bridge 10 (Knightdale, NC)

208
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER: R-2547 (Knightdale-Eagle Rock Rd. Over US-64 Bypass)
MEASUREMENT DATE: March 20 & March 29, 2004

### DECK LOADS

<table>
<thead>
<tr>
<th>Girder</th>
<th>Concrete(^1)</th>
<th>Slab(^2)</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>lb/ft</td>
<td>N/mm</td>
<td>lb/ft</td>
</tr>
<tr>
<td>G1</td>
<td>1014.81</td>
<td>14.81</td>
<td>1081.27</td>
</tr>
<tr>
<td>G2</td>
<td>1138.83</td>
<td>16.62</td>
<td>1227.91</td>
</tr>
<tr>
<td>G3</td>
<td>1138.83</td>
<td>16.62</td>
<td>1227.91</td>
</tr>
<tr>
<td>G4</td>
<td>1014.81</td>
<td>14.81</td>
<td>1081.27</td>
</tr>
</tbody>
</table>

\(^1\) Calculated with nominal slab thicknesses
\(^2\) Includes slab, buildups, and stay-in-place forms (nominal)

### BEARING SETTLEMENTS\(^3\) (data in inches, negative is deflection upwards)

<table>
<thead>
<tr>
<th>Point</th>
<th>Pour 1 Settlement</th>
<th>Pour 2 Settlement</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>End 1</td>
<td>Middle</td>
</tr>
<tr>
<td>G1</td>
<td>0.05</td>
<td>0.06</td>
</tr>
<tr>
<td>G2</td>
<td>0.03</td>
<td>0.07</td>
</tr>
<tr>
<td>G3</td>
<td>0.03</td>
<td>0.06</td>
</tr>
<tr>
<td>G4</td>
<td>0.01</td>
<td>0.06</td>
</tr>
</tbody>
</table>

\(^3\) Noticeably, the settlement totaled from the two pours was very close to zero.

### GIRDER DEFLECTIONS (data in inches, negative is deflection upwards)

<table>
<thead>
<tr>
<th>Point</th>
<th>7/10 Span C Loading</th>
<th>8/10 Span C Loading</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>4/10 B</td>
<td>7/10 B</td>
</tr>
<tr>
<td>G1</td>
<td>-0.70</td>
<td>-0.60</td>
</tr>
<tr>
<td>G2</td>
<td>-0.60</td>
<td>-0.58</td>
</tr>
<tr>
<td>G3</td>
<td>-0.69</td>
<td>-0.60</td>
</tr>
<tr>
<td>G4</td>
<td>-0.73</td>
<td>-0.55</td>
</tr>
</tbody>
</table>

End of Span C

<table>
<thead>
<tr>
<th>Point</th>
<th>4/10 B</th>
<th>7/10 B</th>
<th>2/10 C</th>
<th>6/10 C</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>-0.87</td>
<td>-0.71</td>
<td>1.42</td>
<td>1.99</td>
</tr>
<tr>
<td>G2</td>
<td>-0.75</td>
<td>-0.73</td>
<td>0.67</td>
<td>1.76</td>
</tr>
<tr>
<td>G3</td>
<td>-0.83</td>
<td>-0.71</td>
<td>0.78</td>
<td>1.66</td>
</tr>
<tr>
<td>G4</td>
<td>-0.89</td>
<td>-0.63</td>
<td>0.71</td>
<td>1.68</td>
</tr>
</tbody>
</table>
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER: R-2547 (Knightdale-Eagle Rock Rd. Over US-64 Bypass)
MEASUREMENT DATE: March 20 & March 29, 2004

GIRDER DEFLECTIONS (data in inches, negative is deflection upwards)

<table>
<thead>
<tr>
<th>Point</th>
<th>4/10 B</th>
<th>7/10 B</th>
<th>2/10 C</th>
<th>6/10 C</th>
<th>4/10 B</th>
<th>7/10 B</th>
<th>2/10 C</th>
<th>6/10 C</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>0.85</td>
<td>0.56</td>
<td>-0.08</td>
<td>0.20</td>
<td>2.08</td>
<td>1.39</td>
<td>-0.38</td>
<td>-0.12</td>
</tr>
<tr>
<td>G2</td>
<td>0.98</td>
<td>0.54</td>
<td>-0.15</td>
<td>-0.04</td>
<td>2.10</td>
<td>1.34</td>
<td>-0.43</td>
<td>-0.36</td>
</tr>
<tr>
<td>G3</td>
<td>1.08</td>
<td>0.70</td>
<td>-0.08</td>
<td>0.10</td>
<td>2.13</td>
<td>1.50</td>
<td>-0.36</td>
<td>-0.19</td>
</tr>
<tr>
<td>G4</td>
<td>1.46</td>
<td>0.79</td>
<td>-0.19</td>
<td>0.07</td>
<td>2.63</td>
<td>1.66</td>
<td>-0.49</td>
<td>-0.24</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Point</th>
<th>4/10 B</th>
<th>7/10 B</th>
<th>2/10 C</th>
<th>6/10 C</th>
<th>4/10 B</th>
<th>7/10 B</th>
<th>2/10 C</th>
<th>6/10 C</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>2.91</td>
<td>2.02</td>
<td>-0.57</td>
<td>-0.32</td>
<td>3.28</td>
<td>2.41</td>
<td>-0.73</td>
<td>-0.50</td>
</tr>
<tr>
<td>G2</td>
<td>2.76</td>
<td>1.90</td>
<td>-0.62</td>
<td>-0.58</td>
<td>3.07</td>
<td>2.23</td>
<td>-0.75</td>
<td>-0.69</td>
</tr>
<tr>
<td>G3</td>
<td>2.77</td>
<td>2.09</td>
<td>-0.58</td>
<td>-0.40</td>
<td>3.01</td>
<td>2.34</td>
<td>-0.68</td>
<td>-0.51</td>
</tr>
<tr>
<td>G4</td>
<td>3.26</td>
<td>2.31</td>
<td>-0.76</td>
<td>-0.51</td>
<td>3.40</td>
<td>2.52</td>
<td>-0.86</td>
<td>-0.60</td>
</tr>
</tbody>
</table>

TOTAL MEASURED

<table>
<thead>
<tr>
<th>Point</th>
<th>4/10 B</th>
<th>7/10 B</th>
<th>2/10 C</th>
<th>6/10 C</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>1.97</td>
<td>1.29</td>
<td>1.18</td>
<td>2.07</td>
</tr>
<tr>
<td>G2</td>
<td>1.91</td>
<td>1.10</td>
<td>0.39</td>
<td>1.64</td>
</tr>
<tr>
<td>G3</td>
<td>1.74</td>
<td>1.21</td>
<td>0.56</td>
<td>1.66</td>
</tr>
<tr>
<td>G4</td>
<td>2.02</td>
<td>1.36</td>
<td>0.38</td>
<td>1.64</td>
</tr>
</tbody>
</table>

PREDICTIONS\textsuperscript{4} (Single Girder-Line Model in SAP 2000)

<table>
<thead>
<tr>
<th>Point</th>
<th>4/10 B</th>
<th>7/10 B</th>
<th>2/10 C</th>
<th>6/10 C</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>-0.71</td>
<td>-0.74</td>
<td>0.82</td>
<td>1.81</td>
</tr>
<tr>
<td>G2</td>
<td>-0.80</td>
<td>-0.83</td>
<td>0.92</td>
<td>2.03</td>
</tr>
<tr>
<td>G3</td>
<td>-0.80</td>
<td>-0.83</td>
<td>0.92</td>
<td>2.03</td>
</tr>
<tr>
<td>G4</td>
<td>-0.71</td>
<td>-0.74</td>
<td>0.82</td>
<td>1.81</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Point</th>
<th>4/10 B</th>
<th>7/10 B</th>
<th>2/10 C</th>
<th>6/10 C</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>2.17</td>
<td>1.29</td>
<td>0.35</td>
<td>1.43</td>
</tr>
<tr>
<td>G2</td>
<td>2.43</td>
<td>1.44</td>
<td>0.39</td>
<td>1.60</td>
</tr>
<tr>
<td>G3</td>
<td>2.43</td>
<td>1.44</td>
<td>0.39</td>
<td>1.60</td>
</tr>
<tr>
<td>G4</td>
<td>2.17</td>
<td>1.29</td>
<td>0.35</td>
<td>1.43</td>
</tr>
</tbody>
</table>

\textsuperscript{4} Using nominal slab thicknesses
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER: R-2547 (Knightdale-Eagle Rock Rd. Over US-64 Bypass)
MEASUREMENT DATE: March 20 & March 29, 2004

GIRDER DEFLECTIONS
CROSS SECTION VIEW

- Pour 1 Measured
- Pour 2 Measured
- Total Measured
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER: R-2547 (Knightdale-Eagle Rock Rd. Over US-64 Bypass)
MEASUREMENT DATE: March 20 & March 29, 2004

GIRDER DEFLECTIONS
CROSS SECTION VIEW

Pour 1 Measured
Pour 2 Measured
Total Measured

Pour 1 Predicted
Pour 2 Predicted
Total Predicted

Pour 1 Measured
Pour 1 Predicted
Pour 2 Measured
Pour 2 Predicted
Total Measured
Total Predicted
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER: R-2547 (Knightdale-Eagle Rock Rd. Over US-64 Bypass)
MEASUREMENT DATE: March 20 & March 29, 2004

GIRDER DEFLECTIONS
ELEVATION VIEW

POUR 1

-2.00
-1.00
0.00
1.00
2.00
3.00
4.00

Deflection (inches)

4/10 7/10 2/10 6/10
Span B Span B Span C Span C

POUR 2

-2.00
-1.00
0.00
1.00
2.00
3.00
4.00

Deflection (inches)

4/10 7/10 2/10 6/10
Span B Span B Span C Span C

TOTAL

-2.00
-1.00
0.00
1.00
2.00
3.00
4.00

Deflection (inches)

4/10 7/10 2/10 6/10
Span B Span B Span C Span C

Girder 1
Girder 2
Girder 3
Girder 4
ANSYS FINITE ELEMENT MODELING SUMMARY

PROJECT NUMBER:  R-2547 (Knightdale-Eagle Rock Rd. Over US-64 Bypass)

MODEL PICTURE:  (Steel Only, Isometric View)
**ANSYS FINITE ELEMENT MODELING SUMMARY**

**PROJECT NUMBER:** R-2547 (Knightdale-Eagle Rock Rd. Over US-64 Bypass)

**MODEL DESCRIPTION**

<table>
<thead>
<tr>
<th>COMPONENT</th>
<th>Element Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Girder</td>
<td>SHELL93</td>
</tr>
<tr>
<td>Connector Plates</td>
<td>SHELL93</td>
</tr>
<tr>
<td>Stiffener Plates</td>
<td>SHELL93</td>
</tr>
<tr>
<td>Cross-frame Members</td>
<td>LINK8 (diagonal)</td>
</tr>
<tr>
<td></td>
<td>LINK8 (horizontal)</td>
</tr>
<tr>
<td>End Diaphragm</td>
<td>LINK8 (diagonal)</td>
</tr>
<tr>
<td></td>
<td>LINK8 (vertical)</td>
</tr>
<tr>
<td></td>
<td>BEAM4 (horizontal)</td>
</tr>
<tr>
<td>Stay-in-place Deck Forms</td>
<td>LINK8</td>
</tr>
<tr>
<td>Concrete Slab</td>
<td>SHELL63</td>
</tr>
<tr>
<td>Shear Studs</td>
<td>MPC184</td>
</tr>
</tbody>
</table>

**APPLIED LOADS**

<table>
<thead>
<tr>
<th>Component</th>
<th>*Load</th>
</tr>
</thead>
<tbody>
<tr>
<td>Girder</td>
<td></td>
</tr>
<tr>
<td>G1</td>
<td>674.3</td>
</tr>
<tr>
<td>G2</td>
<td>1068.1</td>
</tr>
<tr>
<td>G3</td>
<td>1065.4</td>
</tr>
<tr>
<td>G4</td>
<td>1062.8</td>
</tr>
</tbody>
</table>

*applied as a uniform pressure to area of top flange

<table>
<thead>
<tr>
<th>Point</th>
<th>ANSYS Pour 1 Loading</th>
<th>ANSYS Pour 1 Loading (SIP)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>4/10 B</td>
<td>7/10 B</td>
</tr>
<tr>
<td>G1</td>
<td>-0.56</td>
<td>-0.59</td>
</tr>
<tr>
<td>G2</td>
<td>-0.49</td>
<td>-0.53</td>
</tr>
<tr>
<td>G3</td>
<td>-0.48</td>
<td>-0.50</td>
</tr>
<tr>
<td>G4</td>
<td>-0.50</td>
<td>-0.53</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Point</th>
<th>ANSYS Pour 2 Loading</th>
<th>ANSYS Pour 2 Loading (SIP)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>4/10 B</td>
<td>7/10 B</td>
</tr>
<tr>
<td>G1</td>
<td>2.34</td>
<td>1.63</td>
</tr>
<tr>
<td>G2</td>
<td>2.27</td>
<td>1.61</td>
</tr>
<tr>
<td>G3</td>
<td>2.30</td>
<td>1.63</td>
</tr>
<tr>
<td>G4</td>
<td>2.42</td>
<td>1.73</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Point</th>
<th>ANSYS Total</th>
<th>ANSYS Total (SIP)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>4/10 B</td>
<td>7/10 B</td>
</tr>
<tr>
<td>G1</td>
<td>1.79</td>
<td>1.04</td>
</tr>
<tr>
<td>G2</td>
<td>1.78</td>
<td>1.08</td>
</tr>
<tr>
<td>G3</td>
<td>1.82</td>
<td>1.13</td>
</tr>
<tr>
<td>G4</td>
<td>1.92</td>
<td>1.20</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Point</th>
<th>Pour 1 Measured</th>
<th>Pour 2 Measured</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>4/10 B</td>
<td>7/10 B</td>
</tr>
<tr>
<td>G1</td>
<td>-0.87</td>
<td>-0.71</td>
</tr>
<tr>
<td>G2</td>
<td>-0.75</td>
<td>-0.73</td>
</tr>
<tr>
<td>G3</td>
<td>-0.83</td>
<td>-0.71</td>
</tr>
<tr>
<td>G4</td>
<td>-0.89</td>
<td>-0.63</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Point</th>
<th>Total Measured</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>4/10 B</td>
</tr>
<tr>
<td>G1</td>
<td>1.97</td>
</tr>
<tr>
<td>G2</td>
<td>1.91</td>
</tr>
<tr>
<td>G3</td>
<td>1.74</td>
</tr>
<tr>
<td>G4</td>
<td>2.02</td>
</tr>
</tbody>
</table>
PROJECT NUMBER: R-2547 (Knightdale-Eagle Rock Rd. Over US-64 Bypass)

GIRDER DEFLECTIONS
CROSS SECTION VIEW

- Measured
- ANSYS (no SIP)
- ANSYS (SIP)
- SAP Prediction

Deflection (inches)

4/10 Span B

Deflection (inches)

7/10 Span B

Deflection (inches)

2/10 Span C

216
ANSYS FINITE ELEMENT MODELING SUMMARY

PROJECT NUMBER: R-2547 (Knightdale-Eagle Rock Rd. Over US-64 Bypass)

GIRDER DEFLECTIONS
CROSS SECTION VIEW

- Measured
- ANSYS (no SIP)
- ANSYS (SIP)
- SAP Prediction

Deflection (inches) vs. 6/10 Span C
Appendix G

Deflection Summary for Bridge 1

This appendix contains a detailed description of Bridge 1 including bridge geometry, material data, cross frame type and size, and dead loads calculated from slab geometry. Illustrations detailing the bridge geometry and field measurement locations are included, along with tables and graphs of the field measured non-composite girder deflections.

A summary of the ANSYS finite element model created for Bridge 1 is also included in this appendix. This summary includes a picture of the ANSYS model, details about the elements used in the model generation, the loads applied to the model, and tables and graphs of the deflections predicted by the model.
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER: R-2547 (Rogers Ln. Extension over US 64 Bypass)
MEASUREMENT DATE: October 19, October 26, & November 3, 2004

BRIDGE DESCRIPTION

<table>
<thead>
<tr>
<th>TYPE</th>
<th>Three Span Continuous</th>
</tr>
</thead>
<tbody>
<tr>
<td>LENGTH</td>
<td>585.98 ft (178.608 m)</td>
</tr>
<tr>
<td>NUMBER OF GIRDERS</td>
<td>7</td>
</tr>
<tr>
<td>GIRDER SPACING</td>
<td>9.68 ft (2.95 m)</td>
</tr>
<tr>
<td>SKEW</td>
<td>57.6 deg</td>
</tr>
<tr>
<td>OVERHANG</td>
<td>3.28 ft (1000 mm)</td>
</tr>
<tr>
<td>BEARING TYPE</td>
<td>Pot Bearing</td>
</tr>
</tbody>
</table>

BEARING TYPE (from web centerline)

MATERIAL DATA

<table>
<thead>
<tr>
<th>STRUCTURAL STEEL</th>
<th>Grade</th>
<th>Yield Strength</th>
</tr>
</thead>
<tbody>
<tr>
<td>Girder: AASHTO M270</td>
<td>50 ksi (345 MPa)</td>
<td></td>
</tr>
<tr>
<td>Other: AASHTO M270</td>
<td>50 ksi (345 MPa)</td>
<td></td>
</tr>
</tbody>
</table>

CONCRETE UNIT WEIGHT: 150 pcf (nominal)
SIP FORM WEIGHT: 2.57 psf (CSI Catalog)

GIRDER DATA

<table>
<thead>
<tr>
<th>LENGTH</th>
<th>&quot;Span A&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;Span B&quot;</td>
<td>233.61 ft (71.205 m)</td>
</tr>
<tr>
<td>&quot;Span C&quot;</td>
<td>188.28 ft (57.388 m)</td>
</tr>
</tbody>
</table>

| WEB THICKNESS | 0.63 in (16 mm) |
| WEB DEPTH    | 90.55 in (2300 mm) |

| TOP FLANGE WIDTH | 19.69 in (500 mm) |
| BOTTOM FLANGE WIDTH | 22.05 in (560 mm) |

FLANGE THICKNESSES

<table>
<thead>
<tr>
<th>Top</th>
<th>Bottom</th>
<th>Begin</th>
<th>End</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.87 in (22 mm)</td>
<td>0.98 in (25 mm)</td>
<td>0.00</td>
<td>104.92 ft (31.981 m)</td>
</tr>
<tr>
<td>1.38 in (35 mm)</td>
<td>1.38 in (35 mm)</td>
<td>104.92 ft (31.981 m)</td>
<td>147.69 ft (45.016 m)</td>
</tr>
<tr>
<td>2.17 in (55 mm)</td>
<td>2.36 in (60 mm)</td>
<td>147.69 ft (45.016 m)</td>
<td>180.50 ft (55.016 m)</td>
</tr>
<tr>
<td>1.38 in (35 mm)</td>
<td>1.38 in (35 mm)</td>
<td>180.50 ft (55.016 m)</td>
<td>223.03 ft (67.981 m)</td>
</tr>
<tr>
<td>0.87 in (22 mm)</td>
<td>1.18 in (30 mm)</td>
<td>223.03 ft (67.981 m)</td>
<td>343.27 ft (104.629 m)</td>
</tr>
<tr>
<td>1.38 in (35 mm)</td>
<td>1.38 in (35 mm)</td>
<td>343.27 ft (104.629 m)</td>
<td>378.02 ft (115.221 m)</td>
</tr>
<tr>
<td>2.76 in (70 mm)</td>
<td>2.76 in (70 mm)</td>
<td>378.02 ft (115.221 m)</td>
<td>417.39 ft (127.221 m)</td>
</tr>
<tr>
<td>1.38 in (35 mm)</td>
<td>1.38 in (35 mm)</td>
<td>417.39 ft (127.221 m)</td>
<td>464.66 ft (141.629 m)</td>
</tr>
<tr>
<td>0.87 in (22 mm)</td>
<td>1.18 in (30 mm)</td>
<td>464.66 ft (141.629 m)</td>
<td>585.98 ft (178.608 m)</td>
</tr>
</tbody>
</table>
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER: R-2547 (Rogers Ln. Extension over US 64 Bypass)
MEASUREMENT DATE: October 19, October 26, & November 3, 2004

STIFFENERS

Longitudinal: N/A
Bearing: PL 0.98" × 9.06" (25 mm × 230 mm)
Intermediate: PL 0.47" × NA (12 mm × NA, connector plate)
    PL 0.71" × 7.68" (18 mm × 195 mm)
Middle Bearing: PL 1.57" × 9.06" (40 mm × 230 mm)
End Bent Connector: PL 0.79" × NA (20 mm × NA, connector plate)

CROSS-FRAME DATA

<table>
<thead>
<tr>
<th></th>
<th>Diagonals</th>
<th>Horizontals</th>
<th>Verticals</th>
</tr>
</thead>
<tbody>
<tr>
<td>END BENT (D1, Type K)</td>
<td>WT 5 x 15</td>
<td>C 15 x 33.9 (Top)</td>
<td>NA</td>
</tr>
<tr>
<td></td>
<td></td>
<td>WT 5 x 15 (Bottom)</td>
<td></td>
</tr>
<tr>
<td>MIDDLE BENT (D3, Type K)</td>
<td>L 4 x 4 x 1/2&quot;</td>
<td>L 4 x 4 x 1/2&quot; (Top)</td>
<td>NA</td>
</tr>
<tr>
<td></td>
<td></td>
<td>L 4 x 4 x 1/2&quot; (Bottom)</td>
<td></td>
</tr>
<tr>
<td>INTERMEDIATE (D4, Type K)</td>
<td>L 4 x 4 x 1/2&quot;</td>
<td>L 4 x 4 x 1/2&quot; (Top)</td>
<td>NA</td>
</tr>
<tr>
<td></td>
<td></td>
<td>L 4 x 4 x 1/2&quot; (Bottom)</td>
<td></td>
</tr>
<tr>
<td>INTERMEDIATE (D2, Type K)</td>
<td>L 4 x 4 x 1/2&quot;</td>
<td>L 4 x 4 x 1/2&quot; (Bottom)</td>
<td>NA</td>
</tr>
</tbody>
</table>

SLAB DATA

<table>
<thead>
<tr>
<th></th>
<th></th>
<th>nominal</th>
</tr>
</thead>
<tbody>
<tr>
<td>THICKNESS</td>
<td>8.86 in (225 mm)</td>
<td></td>
</tr>
<tr>
<td>BUILD-UP</td>
<td>3.54 in (90 mm)</td>
<td></td>
</tr>
</tbody>
</table>

Over Middle 2 Bents:

LONGITUDINAL REBAR           SIZE (metric) | SPACING (nominal)
Top:                        #16 | 5.12 in (130 mm)
Bottom:                    #16 | 9.45 in (240 mm)

TRANSVERSE REBAR           SIZE (metric) | SPACING (nominal)
Top:                        #16 | 6.30 in (160 mm)
Bottom:                    #16 | 6.30 in (160 mm)

Otherwise:

LONGITUDINAL REBAR           SIZE (metric) | SPACING (nominal)
Top:                        #16 | 20.47 in (520 mm)
Bottom:                    #16 | 9.45 in (240 mm)

TRANSVERSE REBAR           SIZE (metric) | SPACING (nominal)
Top:                        #16 | 6.30 in (160 mm)
Bottom:                    #16 | 6.30 in (160 mm)
FIELD MEASUREMENT SUMMARY

Project Number: R-2547 (Rogers Ln. Extension over US 64 Bypass)
Measurement Date: October 19, October 26, & November 3, 2004

Girder Centerline: Span A = 164.09 ft (50.015 m)
Construction Joint: Span B = 233.61 ft (71.205 m)
Measurement Location: Span C = 188.28 ft (57.388 m)

(a) Plan View (Not to Scale)

Girder: Girder Centerline
String Pots: Construction Joint
Cable from Girder to String Pot

(b) Elevation View (Not to Scale)

Plan and Elevation View of Bridge 1 (Raleigh, NC)
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER: R-2547 (Rogers Ln. Extension over US 64 Bypass)
MEASUREMENT DATE: October 19, October 26, & November 3, 2004

DECK LOADS

<table>
<thead>
<tr>
<th>Girder</th>
<th>Concrete $^1$</th>
<th>Slab $^2$</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>lb/ft</td>
<td>N/mm</td>
<td>lb/ft</td>
</tr>
<tr>
<td>G1</td>
<td>1109.37</td>
<td>16.19</td>
<td>1177.89</td>
</tr>
<tr>
<td>G2</td>
<td>1183.37</td>
<td>17.27</td>
<td>1273.82</td>
</tr>
<tr>
<td>G4</td>
<td>1183.37</td>
<td>17.27</td>
<td>1273.82</td>
</tr>
<tr>
<td>G6</td>
<td>1183.37</td>
<td>17.27</td>
<td>1273.82</td>
</tr>
<tr>
<td>G7</td>
<td>1109.37</td>
<td>16.19</td>
<td>1177.89</td>
</tr>
</tbody>
</table>

$^1$ Calculated with nominal slab thicknesses
$^2$ Includes slab, buildups, and stay-in-place forms (nominal)

GIRDER DEFLECTIONS (data in inches, negative is deflection upwards)

POUR 1 MEASURED

<table>
<thead>
<tr>
<th>Point</th>
<th>4/10 A</th>
<th>4/10 B</th>
<th>35/100 C</th>
<th>4/10 A</th>
<th>4/10 B</th>
<th>35/100 C</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>0.60</td>
<td>-0.21</td>
<td>0.06</td>
<td>2.39</td>
<td>-0.99</td>
<td>0.25</td>
</tr>
<tr>
<td>G2</td>
<td>0.54</td>
<td>-0.20</td>
<td>0.06</td>
<td>2.23</td>
<td>-0.99</td>
<td>0.24</td>
</tr>
<tr>
<td>G4</td>
<td>0.46</td>
<td>-0.14</td>
<td>0.07</td>
<td>2.08</td>
<td>-0.95</td>
<td>0.30</td>
</tr>
<tr>
<td>G6</td>
<td>0.49</td>
<td>-0.19</td>
<td>0.07</td>
<td>2.14</td>
<td>-1.03</td>
<td>0.28</td>
</tr>
<tr>
<td>G7</td>
<td>0.54</td>
<td>-0.19</td>
<td>0.07</td>
<td>2.26</td>
<td>-1.01</td>
<td>0.29</td>
</tr>
</tbody>
</table>

POUR 2 MEASURED

<table>
<thead>
<tr>
<th>Point</th>
<th>4/10 A</th>
<th>4/10 B</th>
<th>35/100 C</th>
<th>4/10 A</th>
<th>4/10 B</th>
<th>35/100 C</th>
<th>4/10 A</th>
<th>4/10 B</th>
<th>35/100 C</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>-0.56</td>
<td>2.87</td>
<td>-1.53</td>
<td>-1.18</td>
<td>5.79</td>
<td>-2.23</td>
<td>-1.47</td>
<td>6.80</td>
<td>-2.61</td>
</tr>
<tr>
<td>G2</td>
<td>-0.55</td>
<td>2.84</td>
<td>-1.52</td>
<td>-1.15</td>
<td>5.58</td>
<td>-2.26</td>
<td>-1.42</td>
<td>6.56</td>
<td>-2.66</td>
</tr>
<tr>
<td>G4</td>
<td>-0.57</td>
<td>2.86</td>
<td>-1.43</td>
<td>-1.10</td>
<td>5.41</td>
<td>-2.16</td>
<td>-1.34</td>
<td>6.32</td>
<td>-2.48</td>
</tr>
<tr>
<td>G6</td>
<td>-0.58</td>
<td>2.98</td>
<td>-1.41</td>
<td>-0.97</td>
<td>5.40</td>
<td>-2.15</td>
<td>-1.10</td>
<td>6.22</td>
<td>-2.42</td>
</tr>
<tr>
<td>G7</td>
<td>-0.57</td>
<td>3.05</td>
<td>-1.40</td>
<td>-0.90</td>
<td>5.44</td>
<td>-2.07</td>
<td>-0.94</td>
<td>6.15</td>
<td>-2.29</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Point</th>
<th>4/10 A</th>
<th>4/10 B</th>
<th>35/100 C</th>
<th>4/10 A</th>
<th>4/10 B</th>
<th>35/100 C</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>-0.95</td>
<td>6.49</td>
<td>-2.62</td>
<td>-0.69</td>
<td>6.33</td>
<td>-2.54</td>
</tr>
<tr>
<td>G2</td>
<td>-0.96</td>
<td>6.28</td>
<td>-2.63</td>
<td>-0.73</td>
<td>6.14</td>
<td>-2.58</td>
</tr>
<tr>
<td>G4</td>
<td>-0.95</td>
<td>6.05</td>
<td>-2.47</td>
<td>-0.74</td>
<td>5.96</td>
<td>-2.45</td>
</tr>
<tr>
<td>G6</td>
<td>-0.72</td>
<td>5.91</td>
<td>-2.38</td>
<td>-0.60</td>
<td>5.87</td>
<td>-2.38</td>
</tr>
<tr>
<td>G7</td>
<td>-0.71</td>
<td>5.82</td>
<td>-2.30</td>
<td>-0.58</td>
<td>5.80</td>
<td>-2.30</td>
</tr>
</tbody>
</table>
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER: R-2547 (Rogers Ln. Extension over US 64 Bypass)
MEASUREMENT DATE: October 19, October 26, & November 3, 2004

GIRDER DEFLECTIONS (data in inches, negative is deflection upwards)

<table>
<thead>
<tr>
<th>Point</th>
<th>8/10 Span C Loading</th>
<th>4/10 Span C Loading</th>
<th>2/10 Span C Loading</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>4/10 A</td>
<td>4/10 B</td>
<td>35/100 C</td>
</tr>
<tr>
<td>G1</td>
<td>0.02</td>
<td>-0.23</td>
<td>0.82</td>
</tr>
<tr>
<td>G2</td>
<td>0.03</td>
<td>-0.21</td>
<td>0.82</td>
</tr>
<tr>
<td>G4</td>
<td>0.02</td>
<td>-0.21</td>
<td>0.82</td>
</tr>
<tr>
<td>G6</td>
<td>0.02</td>
<td>-0.22</td>
<td>0.95</td>
</tr>
<tr>
<td>G7</td>
<td>0.00</td>
<td>-0.22</td>
<td>1.05</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Point</th>
<th>Middle Bent 2</th>
<th>Complete Loading</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>4/10 A</td>
<td>4/10 B</td>
</tr>
<tr>
<td>G1</td>
<td>0.36</td>
<td>-1.11</td>
</tr>
<tr>
<td>G2</td>
<td>0.30</td>
<td>-1.14</td>
</tr>
<tr>
<td>G4</td>
<td>0.28</td>
<td>-1.20</td>
</tr>
<tr>
<td>G6</td>
<td>0.32</td>
<td>-1.23</td>
</tr>
<tr>
<td>G7</td>
<td>0.34</td>
<td>-1.20</td>
</tr>
</tbody>
</table>

TOTAL MEASURED

<table>
<thead>
<tr>
<th>Point</th>
<th>Super-Imposed Total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>4/10 A</td>
</tr>
<tr>
<td>G1</td>
<td>1.99</td>
</tr>
<tr>
<td>G2</td>
<td>1.73</td>
</tr>
<tr>
<td>G4</td>
<td>1.53</td>
</tr>
<tr>
<td>G6</td>
<td>1.77</td>
</tr>
<tr>
<td>G7</td>
<td>2.03</td>
</tr>
</tbody>
</table>

PREDICTIONS^4 (Single Girder-Line Model in SAP 2000)

<table>
<thead>
<tr>
<th>Point</th>
<th>Pour 1</th>
<th>Pour 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>4/10 A</td>
<td>4/10 B</td>
</tr>
<tr>
<td>G1</td>
<td>2.36</td>
<td>-1.29</td>
</tr>
<tr>
<td>G2</td>
<td>2.52</td>
<td>-1.38</td>
</tr>
<tr>
<td>G3</td>
<td>2.52</td>
<td>-1.38</td>
</tr>
<tr>
<td>G4</td>
<td>2.52</td>
<td>-1.38</td>
</tr>
<tr>
<td>G5</td>
<td>2.36</td>
<td>-1.29</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Point</th>
<th>Pour 3</th>
<th>Super-Imposed Total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>4/10 A</td>
<td>4/10 B</td>
</tr>
<tr>
<td>G1</td>
<td>0.26</td>
<td>-1.06</td>
</tr>
<tr>
<td>G2</td>
<td>0.28</td>
<td>-1.11</td>
</tr>
<tr>
<td>G3</td>
<td>0.28</td>
<td>-1.11</td>
</tr>
<tr>
<td>G4</td>
<td>0.28</td>
<td>-1.11</td>
</tr>
<tr>
<td>G5</td>
<td>0.26</td>
<td>-1.06</td>
</tr>
</tbody>
</table>

^4 Using nominal slab thicknesses
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER: R-2547 (Rogers Ln. Extension over US 64 Bypass)
MEASUREMENT DATE: October 19, October 26, & November 3, 2004

GIRDER DEFLECTIONS
CROSS SECTION VIEW

<table>
<thead>
<tr>
<th>Deflection (inches)</th>
<th>G1</th>
<th>G2</th>
<th>G4</th>
<th>G6</th>
<th>G7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pour 1Measured</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pour 2Measured</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pour 3Measured</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total Measured</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Pour 1 Measured
Pour 2 Measured
Pour 3 Measured
Total Measured

Pour 1 Measured
Pour 2 Measured
Pour 3 Measured
Total Measured

Pour 1 Measured
Pour 2 Measured
Pour 3 Measured
Total Measured
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER: R-2547 (Rogers Ln. Extension over US 64 Bypass)
MEASUREMENT DATE: October 19, October 26, & November 3, 2004

GIRDER DEFLECTIONS CROSS SECTION VIEW

- Total Measured
- Total Predicted

G1 G2 G4 G6 G7

4/10 Span A

G1 G2 G4 G6 G7

4/10 Span B

G1 G2 G4 G6 G7

35/100 Span C

Total Measured: 225
Total Predicted: 225
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER: R-2547 (Rogers Ln. Extension over US 64 Bypass)
MEASUREMENT DATE: October 19, October 26, & November 3, 2004

GIRDER DEFLECTIONS
ELEVATION VIEW

POUR 1

POUR 2

POUR 3
FIELD MEASUREMENT SUMMARY

PROJECT NUMBER: R-2547 (Rogers Ln. Extension over US 64 Bypass)
MEASUREMENT DATE: October 19, October 26, & November 3, 2004

GIRDER DEFLECTIONS
ELEVATION VIEW

<table>
<thead>
<tr>
<th>Girder</th>
<th>Deflection (inches)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-3.00</td>
</tr>
<tr>
<td>2</td>
<td>1.00</td>
</tr>
<tr>
<td>4</td>
<td>3.00</td>
</tr>
<tr>
<td>6</td>
<td>7.00</td>
</tr>
<tr>
<td>7</td>
<td>9.00</td>
</tr>
</tbody>
</table>

TOTAL 227

[Graph showing deflections of Girder 1, 2, 4, 6, and 7]
ANSYS FINITE ELEMENT MODELING SUMMARY

PROJECT NUMBER: R-2547 (Ramp (RPBDY1) Over US-64 Business)

MODEL PICTURE: (Steel Only, Oblique View)
**ANSYS FINITE ELEMENT MODELING SUMMARY**

**PROJECT NUMBER:**  R-2547 (Ramp (RPBDY1) Over US-64 Business)

**MODEL DESCRIPTION**

- **Component**
  - Girder: SHELL93
  - Connector Plates: SHELL93
  - Stiffener Plates: SHELL93
  - Cross-frame Members: LINK8 (diagonal)
  - BEAM4 (horizontal)
  - Middle Diaphragm: LINK8 (diagonal)
  - BEAM4 (horizontal)
  - Stay-in-place Deck Forms: LINK8
  - Concrete Slab: SHELL63
  - Shear Studs: MPC184

**APPLIED LOADS**

<table>
<thead>
<tr>
<th>Component</th>
<th>Element Type</th>
<th>Load (lb/ft N/mm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Girder</td>
<td>SHELL93</td>
<td></td>
</tr>
<tr>
<td>Connector Plates</td>
<td>SHELL93</td>
<td></td>
</tr>
<tr>
<td>Stiffener Plates</td>
<td>SHELL93</td>
<td></td>
</tr>
<tr>
<td>Cross-frame Members</td>
<td>LINK8 (diagonal)</td>
<td></td>
</tr>
<tr>
<td>BEAM4 (horizontal)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Middle Diaphragm</td>
<td>LINK8 (diagonal)</td>
<td></td>
</tr>
<tr>
<td>Stay-in-place Deck Forms</td>
<td>LINK8</td>
<td></td>
</tr>
<tr>
<td>Concrete Slab</td>
<td>SHELL63</td>
<td></td>
</tr>
<tr>
<td>Shear Studs</td>
<td>MPC184</td>
<td></td>
</tr>
</tbody>
</table>

*Note: When ANSYS numbers were compared with ANSYS (SIP) numbers, there was 1% difference, therefore, ANSYS with SIP will not be shown on graphs.*
PROJECT NUMBER: R-2547 (Ramp (RPBDY1) Over US-64 Business)

GIRDER DEFLECTIONS CROSS SECTION VIEW

Measured
ANSYS (no SIP)
SAP Prediction
Appendix H

MATLAB Files

This appendix contains a detailed description of the specific input variables required to define a given bridge structure in the MATLAB preprocessor program. Additionally, the individual input files for all ten studied bridges and the thirty-eight source code files are included. Note that the source code files are presented in alphabetical order, and follow the bridge input files and ‘main.m’, which is the only source code file called in the MATLAB execution window.
This is a description of the preprocessor file for a given bridge.

The input file will have the name of the appropriate bridge to be evaluated in ANSYS. Users can change the following variables to represent the desired bridge.

Main Data

- **skew:** Given skew angle of the bridge (NO NEGATIVE VALUES)
- **spantype:** Type of Bridge: 1=simple span
- 2=two span continuous
- 3=three span continuous
- **spanlending:** Coordinate at end of first span
- **span2ending:** Coordinate at end of second span (if needed)
- **span3ending:** Coordinate at end of third span (if needed)
- **girders:** Number of girders
- **spacing:** Spacing dimension of the girders
- **overhang1:** Overhang width dimension outside of first girder (ext)
- **overhang2:** Overhang width dimension outside of last girder (ext)
- **steelmod:** Steel's modulus of elasticity
- **steelpois:** Poisson's ratio for steel
- **concmod:** Concrete's modulus of elasticity
- **concpois:** Poisson's ratio for concrete

Girder Section Data - There are normally flange changes along the length of the spans. This program will allow for up to 9 changes along one girder length. The user must start at the zero coordinate end of the girder and count changes until it reaches the other end - even if there is a symmetrical change, it must be re-entered.

It is assumed that the web height and thickness will not change along the length of the girder (has never been the case during study).

- **numsections:** Number of sections along a girder
- **webheight:** Height dimension of the web (bottom of top flange to top of bottom flange)
- **webthickness:** Thickness dimension of the web
- **flangewidth:** The widest section's flange width (default)
- **thickesttopflange:** Thickest top flange of all the sections (for buildup)
- **thickestbottomflange:** Thickest bottom flange of all the sections
- **sectionarray:** Array for all data of the girder section
  - The first column is the ending coordinate of that section
  - The second column is the top flange width dimension
  - The third column is the top flange thickness dimension
  - The fourth column is the bottom flange width dimension
  - The fifth column is the bottom flange thickness dimension

Note: The row number in the sectionarray is the number of that section, therefore, if user has six different sections - the first number will go 1-6.

Concrete Data

- **buildup:** Build-up concrete thickness
- **slab:** Slab thickness dimension
integralbents: Variable defining presence of integral pour
numpours: Number of pour sequences
pourarray: Bridge construction joint coordinate array

Bearing Stiffeners - 3 and 4 are only used if needed (for 2 or 3 span continuous bridges). Bearing stiffeners are always present on both sides of the girders.

numbearing: Number of bearing coordinates (i.e. 2 for simple span)
beararray: Bearing stiffener coordinate array (w/ tag IDs)
beararray(1,1): Z-coordinate of first end bent bearing stiffeners
beararray(2,1): Z-coordinate of other end bent bearing stiffeners
beararray(3,1): Z-coordinate of first middle bent bearing stiffeners
beararray(4,1): Z-coordinate of other middle bent bearing stiffeners
beararray(1,2): Thickness of first end bent bearing stiffener
beararray(2,2): Thickness of other end bent bearing stiffener
beararray(3,2): Thickness of middle bent bearing stiffeners
beararray(4,2): Thickness of other middle bent stiffeners

Intermediate Stiffeners - The left and right come from looking at a cross-section view (towards direction of stationing) of the bridge from plans. The right side stiffeners are in the positive x-direction and the left side stiffeners are in the negative x-direction.

Note: User must enter right and left side stiffeners as looking at an interior girder. The rest of the program will handle placing the stiffeners on the correct side for an exterior girder.

stiffthick: Thickness of all intermediate stiffeners
numstiffright: # of intermediate stiffeners on right side of girder
numstiffleft: # of intermediate stiffeners on left side of girder
rightstiffarray: Right side intermediate stiffener array
leftstiffarray: Left side intermediate stiffener array

Connector Plates - All that is required is the first coordinate of a right side connector plate and the spacing (per span).

Note: If the cross frame layout is = 2, like in Bridge 10, then the variable: connright2 (first coordinate) will equal the coordinate where the second span ended (represented by a variable above) so that the cross frames start at the middle bent.

connthick: Connector Plate Thickness
connright: Coord of the first right side conn plate on first span
rightspacing: Spacing of conn plates on right side of first span
connright2: Coord of the first right side conn plate on second span
rightspacing2: Spacing of conn plates on right side of second span
connright3: Coord of the first right side conn plate on third span
rightspacing3: Spacing of conn plates on right side of third span
%Pot Location Data / Measurement Locations - There are up to eight allowed
% locations to analyze the bridge behavior. User must enter the number
% of desired locations and the z-coordinates of those locations.
%Note: If the bridge is 2 or 3 span continuous, the coordinates for pot
% locations must be global - from one end to the other.
%
%numpots: Desired number of measurement locations along girder span
%potarray: Array for the z-coordinates of the pot locations
%potarray(1): First measurement/pot location
%potarray(2): Second measurement/pot location
%potarray(3): Third measurement/pot location
%etc...

%Intermediate Cross Frame (ICF) Data - For the truss members of the
%ICFs, the user must enter the area. For the horizontal/beam members
%of the ICFs, the user must enter the area, moment of inertia about the
%y-y axis and moment of inertia about the z-z axis.
%Note: The user must leave zeros or some other number in for the beam
%area, iyy and izz instead of commenting (%) the variable.
%
crossframelayout: Variable to tell how the cross frames are layed out in
%continuous bridges:
%1 is like for Bridges 14 & 1
%2 is like for Bridge 10
%icftype: 1 - "X" type, 2 - "K" type, 3 - "K" type w/ top chord
%(type 3 is for Bridge 1)
icfrussarea: Area of the truss members
%icfbeamarea: Area of the horizontal beam members for a "K" frame
%icfbeamiiy: Moment of Inertia about y-axis for beams in "K" frame
%icfbeamizz: Moment of Inertia about z-axis for beams in "K" frame

%End Bent Diaphragm (EBD) Data - For the truss members of the EBD, the
%user must enter the area. The truss members are all of the
%non-horizontal members of the EBD. For the horizontal/beam members of
%the EBD, the user must enter the area, moment of inertia about the
%y-y axis and the moment of inertia about the z-z axis. This is done
%separately for the top and bottom beam (chord) members of the EBD.
%Note: Looking up inertias in the AISC Steel Manual, you find Ixx and Iyy
%For equivalencies, the Ixx in AISC = Iyy in ANSYS
% & the Iyy in AISC = Izz in ANSYS
%ebdtype: 1 - No Vertical Member & 2 - W/ a Vertical Member
%ebdtrussarea: Area of truss members
%topbeamarea: Area for the top beam/chord member
%topbeamiyy: Moment of Inertia about y-axis for the top chord
%topbeamizz: Moment of Inertia about z-axis for the top chord
%bottombeamarea: Area for the bottom beam/chord member
%bottombeamiyy: Moment of Inertia about y-axis for the bottom chord
%bottombeamizz: Moment of Inertia about z-axis for the bottom chord

%Middle Bent Diaphragm (MBD) Data - For the truss members of the MBD, the
%user must enter the area. The truss members are all of the
%non-horizontal members of the diaphragm. For the horizontal/beam
members of the diaphragm, the user must enter the area, moment of
inertia about the y-y axis and the moment of inertia about the z-z
axis.

If mbdtype = 3, then the bottom beam member must have the same properties
as the top beam member, as is the case in Bridge 1 - the program will
only handle it this way.

Note: Same inertia equivalencies as for EBDs

mbdtype:           1 - "X" type & 2 - "K" type & 3 - "K" type w/ top beam
mbdtrussarea:      Area of truss members
mbdbeamarea:       Area for the bottom & top beam/chord member
mbdbeamiyy:        Moment of Inertia about y-axis for the horizontal beam
mbdbeamizz:        Moment of Inertia about z-axis for the horizontal beam

%SIP Form Data - The pan spacing variable is found in shop drawings (when
provided). Usually, this number is around 610 mm (2 ft). The areas
are from a SAP 2000 analysis of a target deflection. The tolerance
specifies where the pan nodes search for existing nodes to be coupled
to (76 mm (3in) usually works).

panspacing=610;     %SIP form spacing along girders
diagonalarea=32.39;  %Area of diagonal truss members
rodarea=25.81;       %Area of horizontal (rod) truss members
couples=76;          %Specified tolerance for node coupling
%Filename: avondale_input.m

%Main Data
skew=53;                %degrees -> if no skew, put 90 (not 0)
span=1;
span1ending=-43575;     %mm
%span2ending=
%span3ending=
girders=7;              %girders
spacing=3410;           %mm
overhang1=1040;         %mm
overhang2=1040;         %mm
steelmod=200000;        %N/mm^2
steelpois=0.3;          %Poisson's Ratio
concmod=21526;          %N/mm^2
concpois=0.2;           %Poisson's Ratio

%Flange Sections
numsections=5;              %Input number of different sections
webheight=1650;             %mm
webthickness=16;            %mm
thickesttopflange=32;       %mm
thickestbottomflange=45;    %mm

%The following variable is not changed by the user
sectionarray=zeros(numsections,5);   %initialize array of thicknesses

%Remove "%" on the following variables to make active.
%NOTE: WHICHEVER SECTION IS THE LAST SECTION, THE ENDING Z-COORDINATE
% WILL BE THE END OF THE GIRDER

%First Section
sectionarray(1,1)=-5500;        %mm - ending zcoordinate
sectionarray(1,2)=380;          %mm - top flange width
sectionarray(1,3)=28;           %mm - top flange thickness
sectionarray(1,4)=510;          %mm - bottom flange width
sectionarray(1,5)=28;           %mm - bottom flange thickness

%Second Section
sectionarray(2,1)=-12500;       %mm
sectionarray(2,2)=380;          %mm
sectionarray(2,3)=28;           %mm
sectionarray(2,4)=510;          %mm
sectionarray(2,5)=35;           %mm

%Third Section
sectionarray(3,1)=-31075;       %mm
sectionarray(3,2)=380;          %mm
sectionarray(3,3)=32;           %mm
sectionarray(3,4)=510;          %mm
sectionarray(3,5)=45; %mm

%Fourth Section
sectionarray(4,1)=-38075; %mm
sectionarray(4,2)=380; %mm
sectionarray(4,3)=28; %mm
sectionarray(4,4)=510; %mm
sectionarray(4,5)=35; %mm

%Fifth Section
sectionarray(5,1)=-43575; %mm
sectionarray(5,2)=380; %mm
sectionarray(5,3)=28; %mm
sectionarray(5,4)=510; %mm
sectionarray(5,5)=28; %mm

%Concrete Data
buildup=65; %mm
slab=230; %mm
integralbents=0; % 0 = no integral bents & 1 = integral bents
numpours=1; %input number of pour sequences on bridge

%The following variable is not changed by user
pourarray=zeros(numpours-1,2);

%Remove "%" in front of variable to make active
pourarray(1,1)=; %mm
pourarray(2,1)=; %mm

%Small loop not to be changed by user to fill in tag ID #
for ii=1:numpours-1
   pourarray(ii,2)=20;
end

%Bearing Stiffeners
numbearing=2; %input the number of bearing locations on bridge

%The following variable is not changed by user
beararray=zeros(numbearing,2); %initialize bearing stiffener array

%Remove "%" in front of variables to make active
beararray(1,1)=-250; %mm - coordinate
beararray(1,2)=28; %mm - thickness
beararray(2,1)=-43325; %mm - coordinate
beararray(2,2)=28; %mm - thickness

%Intermediate Stiffeners
stiffthick=12; %mm
numstiffright=0; %input the number on the right side of interior girder
numstiffleft=0; %input the number on the left side of interior

%The following two variables are not changed by user.
rightstiffarray=zeros(numstiffright,1); %initializes array
leftstiffarray=zeros(numstiffleft,1); %initializes array

%Note: For the following input, the arrays for left and right must be filled in correctly. For instance, if there were 3 right side stiffeners and 2 left side stiffeners, the user will have up to rightstiffarray(3,1) and leftstiffarray(2,1).

%fill in z-coordinates for right side intermediate stiffeners
%rightstiffarray(1)=;

%fill in z-coordinates for left side intermediate stiffeners
%leftstiffarray(1)=;

%-----------------------------------------------

%Connector Plates

connthick=16; %mm
connright=-7870; %mm
rightspacing=6009.9; %mm
%connright2= %mm
%rightspacing2= %mm
%connright3= %mm
%rightspacing3= %mm

%-----------------------------------------------

%Measurement Locations / Pot Locations

numpots=3; %input the number of measurement locations along girder

%The following variable is not changed by user
potarray=zeros(numpots,1);

%Remove "%" in front of variables to make active
potarray(1)=-11018.75; %mm
potarray(2)=-21787.5; %mm
potarray(3)=-32556.25; %mm
%potarray(4)=

%-----------------------------------------------

%Intermediate Cross Frame Data

crossframelayout=1; %selected type
icftype=1; %selected type
icfrussarea=1612.9; %mm^2
icfbeamarea=0; %mm^2
icfbamiyy=0; %mm^4
icfbeamizz=0; %mm^4

%-----------------------------------------------

%End Bent Diaphragm Data

ebdtype=1; %selected type
%Middle Bent Diaphragm Data

mbdtype=1; %selected type
mbdtrussarea=0; %mm^2
mbdbeamarea=0; %mm^2
mbdbeamiiyy=0; %mm^4
mbdbeamizz=0; %mm^4

%S.I.P. Form Data

panspacing=305; %mm
diagonalarea=38.71; %mm
rodarea=38.71; %mm
couples=75; %mm
%Filename: bridgel_input.m

%User Input

%Main Data
skew=57.6;              %degrees -> if no skew, put 90 (not 0)
spantype=3;
span1ending=-50015;     %mm
span2ending=-121220;    %mm
span3ending=-178608;    %mm
girders=7;              %girders
spacing=2950;           %mm
overhang1=1000;         %mm
overhang2=1000;         %mm
steelmod=200000;        %N/mm^2
steelpois=0.3;          %Poisson's Ratio
concmod=21526;          %N/mm^2
concpois=0.2;           %Poisson's Ratio

%-------------------------------------------------------------------------
%Flange Sections

numsections=9;              %Input number of different sections
webheight=2300;             %mm
webthickness=16;            %mm
thickesttopflange=70;       %mm
thickestbottomflange=70;    %mm

%The following variable is not changed by the user
sectionarray=zeros(numsections,5);   %initialize array of thicknesses

%Remove "%" on the following variables to make active.
%NOTE: WHICHEVER SECTION IS THE LAST SECTION, THE ENDING Z-COORDINATE %
% WILL BE THE END OF THE GIRDER

%First Section
sectionarray(1,1)=-31981;       %mm - ending z-coordinate
sectionarray(1,2)=500;          %mm - top flange width
sectionarray(1,3)=22;           %mm - top flange thickness
sectionarray(1,4)=560;          %mm - bottom flange width
sectionarray(1,5)=25;           %mm - bottom flange thickness

%Second Section
sectionarray(2,1)=-45016;       %mm
sectionarray(2,2)=500;          %mm
sectionarray(2,3)=35;           %mm
sectionarray(2,4)=560;          %mm
sectionarray(2,5)=35;           %mm

%Third Section
sectionarray(3,1)=-55016;       %mm
sectionarray(3,2)=500;          %mm
sectionarray(3,3)=55;           %mm
sectionarray(3,4)=560;          %mm
%Fourth Section
sectionarray(4,1)=-67981;  %mm
sectionarray(4,2)=500;     %mm
sectionarray(4,3)=35;      %mm
sectionarray(4,4)=560;     %mm
sectionarray(4,5)=35;      %mm

%Fifth Section
sectionarray(5,1)=-104629; %mm
sectionarray(5,2)=500;     %mm
sectionarray(5,3)=22;      %mm
sectionarray(5,4)=560;     %mm
sectionarray(5,5)=30;      %mm

%Sixth Section
sectionarray(6,1)=-115221; %mm
sectionarray(6,2)=500;     %mm
sectionarray(6,3)=35;      %mm
sectionarray(6,4)=560;     %mm
sectionarray(6,5)=35;      %mm

%Seventh Section
sectionarray(7,1)=-127220; %mm
sectionarray(7,2)=500;     %mm
sectionarray(7,3)=70;      %mm
sectionarray(7,4)=560;     %mm
sectionarray(7,5)=70;      %mm

%Eighth Section
sectionarray(8,1)=-141629; %mm
sectionarray(8,2)=500;     %mm
sectionarray(8,3)=35;      %mm
sectionarray(8,4)=560;     %mm
sectionarray(8,5)=35;      %mm

%Ninth Section
sectionarray(9,1)=-178608; %mm
sectionarray(9,2)=500;     %mm
sectionarray(9,3)=22;      %mm
sectionarray(9,4)=560;     %mm
sectionarray(9,5)=30;      %mm

%Concrete Data
buildup=90;               %mm
slab=225;                 %mm
integralbents=0;           % 0 = no integral bents & 1 = integral bents
numpours=3;               %input number of pour sequences on bridge

%The following variable is not changed by user
pourarray=zeros(numpours-1,2);

%Remove "%" in front of variable to make active
pourarray(1,1)=-30104;    %mm
pourarray(2,1)=-107004;    %mm

%Small loop not to be changed by user to fill in tag ID #
for ii=1:numpours-1
  pourarray(ii,2)=20;
end

%-------------------------------------------------------------------------
%Bearing Stiffeners
numbearing=4;    %input the number of bearing locations on bridge

%The following variable is not changed by user
beararray=zeros(numbearing,2);    %initialize bearing stiffener array
%Remove "%" in front of variables to make active
beararray(1,1)=-240;            %mm - coordinate
beararray(1,2)=25;              %mm - thickness
beararray(2,1)=-50015;          %mm - coordinate
beararray(2,2)=40;              %mm - thickness
beararray(3,1)=-121220;         %mm - coordinate
beararray(3,2)=40;              %mm - thickness
beararray(4,1)=-178368;         %mm - coordinate
beararray(4,2)=25;              %mm - thickness

%-------------------------------------------------------------------------
%Intermediate Stiffeners
stiffthick=18;      %mm

numstiffright=9;    %input the number on the right side of interior girder
numstiffleft=9;     %input the number on the left side of interior

%The following two variables are not changed by user.
rightstiffarray=zeros(numstiffright,1);    %initializes array
leftstiffarray=zeros(numstiffleft,1);       %initializes array
%Note:  For the following input, the arrays for left and right must be
%filled in correctly.  For instance, if there were 3 right side
%stiffeners and 2 left side stiffeners, the user will have up to
%rightstiffarray(3,1) and leftstiffarray(2,1).

%fill in z-coordinates for right side intermediate stiffeners
rightstiffarray(1)=-37265;
rightstiffarray(2)=-45765;
rightstiffarray(3)=-57515;
rightstiffarray(4)=-65015;
rightstiffarray(5)=-106220;
rightstiffarray(6)=-113720;
rightstiffarray(7)=-124220;
rightstiffarray(8)=-130220;
rightstiffarray(9)=-136220;
%fill in z-coordinates for left side intermediate stiffeners
leftstiffarray(1)=-41515;
leftstiffarray(2)=-53765;
leftstiffarray(3)=-61265;
leftstiffarray(4)=-102470;
leftstiffarray(5)=-109970;
leftstiffarray(6)=-117470;
leftstiffarray(7)=-127220;
leftstiffarray(8)=-133220;
leftstiffarray(9)=-139220;

%-------------------------------------------------------------------------

%Connector Plates

connthick=12;           %mm
connright=-5863.5;      %mm
rightspacing=6733.33;   %mm
connright2=-56154;      %mm
rightspacing2=7600;     %mm
connright3=-126730;     %mm
rightspacing3=6000;     %mm

%-------------------------------------------------------------------------

%Measurement Locations / Pot Locations

numpots=3; %input the number of measurement locations along girder

%The following variable is not changed by user
potarray=zeros(numpots,1);

%Remove "%" in front of variables to make active
potarray(1)=20000;     %mm
potarray(2)=78030;     %mm
potarray(3)=141083;    %mm
potarray(4)=

%-------------------------------------------------------------------------

%Intermediate Cross Frame Data

crossframelayout=1; %selected type
icftype=3; %selected type
icftrussarea=2419.4; %mm^2
icfbeamarea=2419.4; %mm^2
icfbeamiyy=2297597; %mm^4
icfbeamizz=2297597; %mm^4

%-------------------------------------------------------------------------

%End Bent Diaphragm Data

ebdtype=1; %selected type
ebdtrussarea=2851.6; %mm^2
topbeamarea=6419.3; %mm^2
topbeamiyy=131112899; %mm^4
topbeamizz=3358987; %mm^4
bottombeamarea=2851.6; %mm^2
bottombeamiyy=3862627; %mm^4
bottombeamizz=3475532; %mm^4

%-------------------------------------------------------------------------

%Middle Bent Diaphragm Data
mbdtype=3;  %selected type
mbdtrussarea=2419.4;  %mm^2
mbddeamarea=2419.4;  %mm^2
mbdbeamixy=2297597;  %mm^4
mbdbeamizz=2297597;  %mm^4

%-------------------------------------------------------------------------
%S.I.P. Form Data

panspacing=610;  %mm
diagonalarea=32.39;  %mm
rodarea=25.81;  %mm
couples=150;  %mm

%-------------------------------------------------------------------------
User Input

skew=60;                %degrees -> if no skew, put 90 (not 0)
spantype=1;
span1ending=-46648;     %mm
%span2ending=
%span3ending=
girders=6;              %girders
spacing=3440;           %mm
overhang1=870;          %mm
overhang2=870;          %mm
steelmod=200000;        %N/mm^2
steelpois=0.3;          %Poisson's Ratio
concmmod=21526;         %N/mm^2
concpois=0.2;           %Poisson's Ratio

Flange Sections

numsections=3;            %Input number of different sections
webheight=1728;              %mm
webthickness=16;            %mm
thickesttopflange=51;       %mm
thickestbottomflange=76;    %mm

The following variable is not changed by the user
sectionarray=zeros(numsections,5);   %initialize array of thicknesses

Remove "%" on the following variables to make active.
%NOTE: WHICHEVER SECTION IS THE LAST SECTION, THE ENDING Z-COORDINATE
% WILL BE THE END OF THE GIRDER

First Section
sectionarray(1,1)=-12490;       %mm - ending zcoordinate
sectionarray(1,2)=457;          %mm - top flange width
sectionarray(1,3)=51;           %mm - top flange thickness
sectionarray(1,4)=457;          %mm - bottom flange width
sectionarray(1,5)=51;           %mm - bottom flange thickness

Second Section
sectionarray(2,1)=-34158;       %mm
sectionarray(2,2)=457;          %mm
sectionarray(2,3)=51;           %mm
sectionarray(2,4)=457;          %mm
sectionarray(2,5)=76;           %mm

Third Section
sectionarray(3,1)=-46648;       %mm
sectionarray(3,2)=457;          %mm
sectionarray(3,3)=51;           %mm
sectionarray(3,4)=457;          %mm
sectionarray(3,5)=51; %mm

%-------------------------------------------------------------------------
%Concrete Data
buildup=95; %mm
slab=235; %mm
integralbents=0; % 0 = no integral bents & 1 = integral bents
numpours=1; %input number of pour sequences on bridge

%The following variable is not changed by user
pourarray=zeros(numpours-1,2);

%Remove "%" in front of variable to make active
pourarray(1,1); %mm
pourarray(2,1); %mm

%Small loop not to be changed by user to fill in tag ID #
for ii=1:numpours-1
pourarray(ii,2)=20;
end

%-------------------------------------------------------------------------
%Bearing Stiffeners
numbearing=2; %input the number of bearing locations on bridge

%The following variable is not changed by user
beararray=zeros(numbearing,2); %initialize bearing stiffener array

%Remove "%" in front of variables to make active
beararray(1,1)=-240; %mm - coordinate
beararray(1,2)=22; %mm - thickness
beararray(2,1)=-46408; %mm - coordinate
beararray(2,2)=22; %mm - thickness

%-------------------------------------------------------------------------
%Intermediate Stiffeners
stiffthick=14; %mm
numstiffright=0; %input the number on the right side of interior girder
numstiffleft=0; %input the number on the left side of interior

%The following two variables are not changed by user.
rightstiffarray=zeros(numstiffright,1); %initializes array
leftstiffarray=zeros(numstiffleft,1); %initializes array

%Note: For the following input, the arrays for left and right must be
%filled in correctly. For instance, if there were 3 right side
%stiffeners and 2 left side stiffeners, the user will have up to
%rightstiffarray(3,1) and leftstiffarray(2,1).

%fill in z-coordinates for right side intermediate stiffeners
%rightstiffarray(1)=;
%fill in z-coordinates for left side intermediate stiffeners
%leftstiffarray(1)=;

%Connector Plates

connthick=14;            %mm
connright=-4728;         %mm
rightspacing=6575.5;     %mm
%connright2=
%rightspacing2=          %mm
%connright3=
%rightspacing3=          %mm

%Measurement Locations / Pot Locations

numpots=2;  %input the number of measurement locations along girder

%The following variable is not changed by user
potarray=zeros(numpots,1);

%Remove "%" in front of variables to make active
potarray(1)=-18300;     %mm
potarray(2)=-35348;     %mm
%potarray(3)=           %mm

%Intermediate Cross Frame Data

crossframelayout=1;         %selected type
icftype=1;                  %selected type
icftrussarea=1361.29;       %mm^2
icfbeamarea=0;              %mm^2
icfbeamiyy=0;               %mm^4
icfbeamizz=0;               %mm^4

%End Bent Diaphragm Data

ebdtype=1;                  %selected type
ebdtrussarea=2658.06;       %mm^2
topbeamiyy=6419.34;         %mm^2
topbeamiyy=131113000;       %mm^4
topbeamizz=3358990;         %mm^4
bottombeamiyy=2658.06;      %mm^2
bottombeamiyy=1760660;      %mm^4
bottombeamizz=4495300;      %mm^4

%Middle Bent Diaphragm Data

mbdtype=1;                  %selected type
mbdtrussarea=0;             %mm^2
mbdbeamiyy=0;               %mm^4
mbdbeamizz=0; \text{ \text{ mm}^4} \\

%------------------------------------------------------------------------
%  S.I.P. Form Data

panspacing=610; \text{ \text{ mm}} \\
\text{ diagonalarea}=45.68; \text{ \text{ mm}} \\
\text{ rodarea}=36.77; \text{ \text{ mm}} \\
\text{ couples}=75; \text{ \text{ mm}} \\

%------------------------------------------------------------------------
%Filename: bridge10_input.m

%User Input

%Main Data
skew=147.1; %degrees -> if no skew, put 90 (not 0)
spantype=2;
span1ending=-47260; %mm
span2ending=-91220; %mm
span3ending=

%degrees

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input

%Main Data

%User Input
%Fourth Section
sectionarray(4,1)=-60760; %mm
sectionarray(4,2)=470; %mm
sectionarray(4,3)=32; %mm
sectionarray(4,4)=470; %mm
sectionarray(4,5)=38; %mm

%Fifth Section
sectionarray(5,1)=-91220; %mm
sectionarray(5,2)=400; %mm
sectionarray(5,3)=32; %mm
sectionarray(5,4)=400; %mm
sectionarray(5,5)=38; %mm

%Concrete Data
buildup=65; %mm
slab=225; %mm
integralbents=0; % 0 = no integral bents & 1 = integral bents
numpours=2; %input number of pour sequences on bridge

%The following variable is not changed by user
pourarray=zeros(numpours-1,2);

%Remove "%" in front of variable to make active
pourarray(1,1)=-65760; %mm
pourarray(2,1); %mm

%Small loop not to be changed by user to fill in tag ID #
for ii=1:numpours-1
    pourarray(ii,2)=20;
end

%Bearing Stiffeners
numbearing=3; %input the number of bearing locations on bridge

%The following variable is not changed by user
beararray=zeros(numbearing,2); %initialize bearing stiffener array

%Remove "%" in front of variables to make active
beararray(1,1)=-230; %mm - coordinate
beararray(1,2)=20; %mm - thickness
beararray(2,1)=-47260; %mm - coordinate
beararray(2,2)=28; %mm - thickness
beararray(3,1)=-90990; %mm - coordinate
beararray(3,2)=20; %mm - thickness

%Intermediate Stiffeners
stiffthick=14; %mm
numstiffright=5; %input the number on the right side of interior girder
numstiffleft=5; %input the number on the left side of interior

%The following two variables are not changed by user.
rightstiffarray=zeros(numstiffright,1); %initializes array
leftstiffarray=zeros(numstiffleft,1); %initializes array

%Note: For the following input, the arrays for left and right must be
% filled in correctly. For instance, if there were 3 right side
% stiffeners and 2 left side stiffeners, the user will have up to
% rightstiffarray(3,1) and leftstiffarray(2,1).

%fill in z-coordinates for right side intermediate stiffeners
rightstiffarray(1)=-35760;
rightstiffarray(2)=-41960;
rightstiffarray(3)=-52560;
rightstiffarray(4)=-58760;
rightstiffarray(5)=-89190;
%fill in z-coordinates for left side intermediate stiffeners
leftstiffarray(1)=-2030;
leftstiffarray(2)=-38860;
leftstiffarray(3)=-45060;
leftstiffarray(4)=-49460;
leftstiffarray(5)=-55660;

%Connector Plates
connthick=12; %mm
connright=-3100; %mm
rightspacing=7360; %mm
connright2=-47260; %mm
rightspacing2=7290; %mm
%connright3=
%rightspacing3=

%Measurement Locations / Pot Locations
numpots=4; %input the number of measurement locations along girder

%The following variable is not changed by user
potarray=zeros(numpots,1);

%Remove "%" in front of variables to make active
potarray(1)=-18904; %mm
potarray(2)=-33082; %mm
potarray(3)=-56052; %mm
potarray(4)=-73613; %mm
potarray(5)=

%Intermediate Cross Frame Data
crossframelayou=2; %selected type
icftype=1; %selected type
icftrussarea=2283.9; %mm^2
icfbeamarea=0; \hspace{1cm} \text{%mm}^2
icfbeamiyy=0; \hspace{1cm} \text{%mm}^4
icfbeamizz=0; \hspace{1cm} \text{%mm}^4

%-------------------------------------------------------------------------
%End Bent Diaphragm Data

ebdtype=2; \hspace{1cm} \text{selected type}
ebdtrussarea=2283.9; \hspace{1cm} \text{%mm}^2
topbeamarea=8129; \hspace{1cm} \text{%mm}^2
topbeamiyy=230592000; \hspace{1cm} \text{%mm}^4
topbeamizz=5952110; \hspace{1cm} \text{%mm}^4
bottombeamarea=2283.9; \hspace{1cm} \text{%mm}^2
bottombeamiyy=1469300; \hspace{1cm} \text{%mm}^4
bottombeamizz=3804360; \hspace{1cm} \text{%mm}^4

%-------------------------------------------------------------------------
%Middle Bent Diaphragm Data

mbdtype=1; \hspace{1cm} \text{selected type}
mbdtrussarea=0; \hspace{1cm} \text{%mm}^2
mbdbeamarea=0; \hspace{1cm} \text{%mm}^2
mbdbeamiyy=0; \hspace{1cm} \text{%mm}^4
mbdbeamizz=0; \hspace{1cm} \text{%mm}^4

%-------------------------------------------------------------------------
%S.I.P. Form Data

panspacing=610; \hspace{1cm} \text{%mm}
diagonalarea=33.94; \hspace{1cm} \text{%mm}
rodarea=26.45; \hspace{1cm} \text{%mm}
couples=75; \hspace{1cm} \text{%mm}
%Filename: bridge14_input.m

%User Input

%Main Data
skew=65.6;               %degrees -> if no skew, put 90 (not 0)
spantype=2;
span1ending=-31064;      %mm
span2ending=-63477;      %mm
%span3ending=
girders=5;              %girders
spacing=3040;           %mm
overhang1=1130;         %mm
overhang2=1130;         %mm
steelmod=200000;        %N/mm^2
steelpois=0.3;          %Poisson's Ratio
concmod=21526;          %N/mm^2
concpois=0.2;           %Poisson's Ratio

%-------------------------------------------------------------------------
%Flange Sections

numsections=3;              %Input number of different sections
webheight=1600;             %mm
webthickness=12;            %mm
thickesttopflange=30;       %mm
thickestbottomflange=35;    %mm

%The following variable is not changed by the user
sectionarray=zeros(numsections,5);   %initialize array of thicknesses

%Remove "%" on the following variables to make active.
%NOTE: WHICHEVER SECTION IS THE LAST SECTION, THE ENDING Z-COORDINATE
%       WILL BE THE END OF THE GIRDER

%First Section
sectionarray(1,1)=-28064;       %mm - ending zcoordinate
sectionarray(1,2)=380;          %mm - top flange width
sectionarray(1,3)=20;           %mm - top flange thickness
sectionarray(1,4)=450;          %mm - bottom flange width
sectionarray(1,5)=20;           %mm - bottom flange thickness

%Second Section
sectionarray(2,1)=-34064;       %mm
sectionarray(2,2)=380;          %mm
sectionarray(2,3)=30;           %mm
sectionarray(2,4)=450;          %mm
sectionarray(2,5)=35;           %mm

%Third Section
sectionarray(3,1)=-63477;       %mm
sectionarray(3,2)=380;          %mm
sectionarray(3,3)=20;           %mm
sectionarray(3,4)=450;          %mm

253
sectionarray(3,5)=20; %mm

%---------------------------------------------------------------
%Concrete Data

buildup=75; %mm
slab=225; %mm
integralbents=1; % 0 = no integral bents & 1 = integral bents
numpours=2; %input number of pour sequences on bridge

%The following variable is not changed by user
pourarray=zeros(numpours-1,2);

%Remove "%" in front of variables to make active
pourarray(1,1)=-45177; %mm
pourarray(2,1); %mm

%Small loop not to be changed by user to fill in tag ID #
for ii=1:numpours-1
    pourarray(ii,2)=20;
end

%---------------------------------------------------------------
%Bearing Stiffeners

numbearing=3; %input the number of bearing locations on bridge

%The following variable is not changed by user
beararray=zeros(numbearing,2); %initialize bearing stiffener array

%Remove "%" in front of variables to make active
beararray(1,1)=-153; %mm - coordinate
beararray(1,2)=25; %mm - thickness
beararray(2,1)=-31064; %mm - coordinate
beararray(2,2)=25; %mm - thickness
beararray(3,1)=-63324; %mm - coordinate
beararray(3,2)=25; %mm - thickness

%---------------------------------------------------------------
%Intermediate Stiffeners

stiffthick=12; %mm

numstiffright=3; %input the number on the right side of interior girder
numstiffleft=3; %input the number on the left side of interior

%The following two variables are not changed by user.
rightstiffarray=zeros(numstiffright,1); %initializes array
leftstiffarray=zeros(numstiffleft,1); %initializes array

%Note: For the following input, the arrays for left and right must be
%filled in correctly. For instance, if there were 3 right side
%stiffeners and 2 left side stiffeners, the user will have up to
%rightstiffarray(3,1) and leftstiffarray(2,1).

%fill in z-coordinates for right side intermediate stiffeners

254
rightstiffarray(1)=-22564;
rightstiffarray(2)=-35314;
rightstiffarray(3)=-60924;
%fill in z-coordinates for left side intermediate stiffeners
leftstiffarray(1)=-2553;
leftstiffarray(2)=-26814;
leftstiffarray(3)=-39564;

%Connector Plates

connthick=16;            %mm 
connright=-3544;         %mm 
rightspacing=7200;       %mm 
connright2=-37114;       %mm 
rightspacing2=6050;      %mm 
%connright3=             %mm 
%rightspacing3=          %mm 

%Measurement Locations / Pot Locations

numpots=3; %input the number of measurement locations along girder

%The following variable is not changed by user
potarray=zeros(numpots,1);

%Remove "%" in front of variables to make active
potarray(1)=-12426;     %mm
potarray(2)=-40788;     %mm
potarray(3)=-50512;     %mm
%potarray(4)=

%Intermediate Cross Frame Data

crossframelayout=1;         %selected type 
icftype=1;                  %selected type 
icftrussarea=2974.2;        %mm^2 
icfbeamarea=0;              %mm^2 
icfbeamiyy=0;               %mm^4 
icfbeamizz=0;               %mm^4 

%End Bent Diaphragm Data

ebdtype=1;                  %selected type 
ebdrussarea=0;             %mm^2 
topbeamarea=0;             %mm^2 
topbeamiyy=0;              %mm^4 
topbeamiz=0;               %mm^4 
bottombeamarea=0;           %mm^2 
bottombeamiyy=0;            %mm^4 
bottombeamiz=0;             %mm^4 

%Middle Bent Diaphragm Data

mbdtype=1;                  %selected type
mbdtrussarea=2974.2;        %mm^2
mbdbeamarea=0;              %mm^2
mbdbeamiyy=0;               %mm^4
mbdbeamizz=0;               %mm^4

%-------------------------------------------------------------------
%S.I.P. Form Data

panspacing=610;             %mm
diagonalarea=37.42;         %mm
rodarea=29.03;              %mm
couples=75;                 %mm

%-------------------------------------------------------------------
%Filename: camden_NB_input.m

%User Input

%Main Data
skew=150.3;             %degrees -> if no skew, put 90 (not 0)
spantype=1;
span1ending=-43966;     %mm
%span2ending=
%span3ending=
girders=6;              %girders
spacing=2650;           %mm
overhang1=300;          %mm
overhang2=300;          %mm
steelmod=200000;        %N/mm^2
steelpois=0.3;          %Poisson's Ratio
concmmod=21526;         %N/mm^2
concpois=0.2;           %Poisson's Ratio

%-------------------------------------------------------------------------

%Flange Sections
numsections=3;              %Input number of different sections
webheight=1680;             %mm
webthickness=16;            %mm
thickesttopflange=30;       %mm
thickestbottomflange=45;    %mm

%The following variable is not changed by the user
sectionarray=zeros(numsections,5);   %initialize array of thicknesses

%Remove "%" on the following variables to make active.
%NOTE: WHICHEVER SECTION IS THE LAST SECTION, THE ENDING Z-COORDINATE %
%      WILL BE THE END OF THE GIRDER

%First Section
sectionarray(1,1)=-10983;       %mm - ending zcoordinate
sectionarray(1,2)=410;          %mm - top flange width
sectionarray(1,3)=25;           %mm - top flange thickness
sectionarray(1,4)=480;          %mm - bottom flange width
sectionarray(1,5)=28;           %mm - bottom flange thickness

%Second Section
sectionarray(2,1)=-32983;       %mm
sectionarray(2,2)=410;          %mm
sectionarray(2,3)=30;           %mm
sectionarray(2,4)=480;          %mm
sectionarray(2,5)=45;           %mm

%Third Section
sectionarray(3,1)=-43966;       %mm
sectionarray(3,2)=410;          %mm
sectionarray(3,3)=25;           %mm
sectionarray(3,4)=480;          %mm
sectionarray(3,5)=28; %mm

% Concrete Data
buildup=65; %mm
slab=225; %mm
integralbents=1; % 0 = no integral bents & 1 = integral bents
numpours=1; %input number of pour sequences on bridge

% The following variable is not changed by user
pourarray=zeros(numpours-1,2);

% Remove "%" in front of variable to make active
pourarray(1,1); %mm
pourarray(2,1); %mm

% Small loop not to be changed by user to fill in tag ID #
for ii=1:numpours-1
  pourarray(ii,2)=20;
end

% Bearing Stiffeners
numbearing=2; %input the number of bearing locations on bridge

% The following variable is not changed by user
beararray=zeros(numbearing,2); %initialize bearing stiffener array

% Remove "%" in front of variables to make active
beararray(1,1)=-228; %mm - coordinate
beararray(1,2)=18; %mm - thickness
beararray(2,1)=-43738; %mm - coordinate
beararray(2,2)=18; %mm - thickness

% Intermediate Stiffeners
stiffthick=14; %mm

numstiffright=0; %input the number on the right side of interior girder
numstiffleft=0; %input the number on the left side of interior

% The following two variables are not changed by user.
rightstiffarray=zeros(numstiffright,1); %initializes array
leftstiffarray=zeros(numstiffleft,1); %initializes array

% Note: For the following input, the arrays for left and right must be
% filled in correctly. For instance, if there were 3 right side
% stiffeners and 2 left side stiffeners, the user will have up to
% rightstiffarray(3,1) and leftstiffarray(2,1).

% fill in z-coordinates for right side intermediate stiffeners
%rightstiffarray(1)=;
%fill in z-coordinates for left side intermediate stiffeners
%leftstiffarray(1)=;

%Connector Plates
connthick=10;  %mm
conright=-1983; %mm
rightspacing=7000; %mm
conright2= %mm
rightspacing2= %mm
conright3= %mm
rightspacing3= %mm

%Measurement Locations / Pot Locations
numpots=3;  %input the number of measurement locations along girder
%The following variable is not changed by user
potarray=zeros(numpots,1);
%Remove "%" in front of variables to make active
potarray(1)=-10991.5;  %mm
potarray(2)=-21983; %mm
potarray(3)=-32974.5; %mm
%potarray(4)=

%Intermediate Cross Frame Data

crossframelayout=1;  %selected type
icftype=1;  %selected type
icftrussarea=1612.9; %mm^2
icfbeamarea=0; %mm^2
icfbeamiyy=0; %mm^4
icfbeamizz=0; %mm^4

%End Bend Diaphragm Data

ebdtype=1;  %selected type
ebdtrussarea=3225.8; %mm^2
topbeamarea=8129; %mm^2
topbeamiyy=230592210; %mm^4
topbeamizz=5952109; %mm^4
bottombeamarea=3225.8; %mm^2
bottombeamiyy=8699237; %mm^4
bottombeamizz=4828285; %mm^4

%Middle Bent Diaphragm Data

mbdtype=1;  %selected type
mbdtrussarea=0; %mm^2
mbdbeamarea=0; %mm^2
mbdbeamiyy=0; %mm^4
mbdbeamzz=0; %mm^4

%-------------------------------------------------------------------------
%S.I.P. Form Data

panspacing=610; %mm
diagonalarea=103.2; %mm
rodarea=25.81; %mm
couples=75; %mm

%-------------------------------------------------------------------------
% User Input

% Main Data
skew=150.3; % degrees -> if no skew, put 90 (not 0)
spantype=1;
span1ending=-43966; % mm
% span2ending=
% span3ending=

girders=7;
spacing=2650; % mm
overhang1=300; % mm
overhang2=300; % mm

steelmod=200000; % N/mm^2
steelpois=0.3; % Poisson's Ratio
concmod=21526; % N/mm^2
concpois=0.2; % Poisson's Ratio

% Flange Sections
numsections=3; % Input number of different sections
webheight=1680; % mm
webthickness=16; % mm

thickesttopflange=30; % mm
thickestbottomflange=45; % mm

% The following variable is not changed by the user
sectionarray=zeros(numsections,5); % initialize array of thicknesses

% Remove "%%" on the following variables to make active.
% NOTE: WHICHEVER SECTION IS THE LAST SECTION, THE ENDING Z-COORDINATE % WILL BE THE END OF THE GIRDER

% First Section
sectionarray(1,1)=-10983; % mm - ending z-coordinate
sectionarray(1,2)=410; % mm - top flange width
sectionarray(1,3)=25; % mm - top flange thickness
sectionarray(1,4)=480; % mm - bottom flange width
sectionarray(1,5)=28; % mm - bottom flange thickness

% Second Section
sectionarray(2,1)=-32983; % mm
sectionarray(2,2)=410; % mm
sectionarray(2,3)=30; % mm
sectionarray(2,4)=480; % mm
sectionarray(2,5)=45; % mm

% Third Section
sectionarray(3,1)=-43966; % mm
sectionarray(3,2)=410; % mm
sectionarray(3,3)=25; % mm
sectionarray(3,4)=480; % mm
sectionarray(3,5)=28; %mm

%Concrete Data
buildup=65; %mm
slab=225; %mm
integralbents=1; % 0 = no integral bents & 1 = integral bents
numpours=1; %input number of pour sequences on bridge

%The following variable is not changed by user
pourarray=zeros(numpours-1,2);

%Remove "%" in front of variable to make active
pourarray(1,1); %mm
pourarray(2,1); %mm

%Small loop not to be changed by user to fill in tag ID #
for ii=1:numpours-1
    pourarray(ii,2)=20;
end

%Bearing Stiffeners
numbearing=2; %input the number of bearing locations on bridge

%The following variable is not changed by user
beararray=zeros(numbearing,2); %initialize bearing stiffener array

%Remove "%" in front of variables to make active
beararray(1,1)=-228; %mm - coordinate
beararray(1,2)=18; %mm - thickness
beararray(2,1)=-43738; %mm - coordinate
beararray(2,2)=18; %mm - thickness

%Intermediate Stiffeners

stiffthick=14; %mm
numstiffright=0; %input the number on the right side of interior girder
numstiffleft=0; %input the number on the left side of interior

%The following two variables are not changed by user.
rightstiffarray=zeros(numstiffright,1); %initializes array
leftstiffarray=zeros(numstiffleft,1); %initializes array

%Note: For the following input, the arrays for left and right must be filled in correctly. For instance, if there were 3 right side stiffeners and 2 left side stiffeners, the user will have up to rightstiffarray(3,1) and leftstiffarray(2,1).

%fill in z-coordinates for right side intermediate stiffeners
%rightstiffarray(1)=;
fill in z-coordinates for left side intermediate stiffeners
leftstiffarray(1)=;

Connector Plates

connthick=10; %mm
connright=-1983; %mm
rightspacing=7000; %mm
connright2= %mm
rightspacing2= %mm
connright3= %mm
rightspacing3= %mm

Measurement Locations / Pot Locations

numpots=3; %input the number of measurement locations along girder

The following variable is not changed by user
potarray=zeros(numpots,1);

Remove "%" in front of variables to make active
potarray(1)=-10991.5; %mm
potarray(2)=-21983; %mm
potarray(3)=-32974.5; %mm

Intermediate Cross Frame Data

crossframelayout=1; %selected type
icftype=1; %selected type
icftrussarea=1612.9; %mm^2
icfbeamarea=0; %mm^2
icfbeamiiyy=0; %mm^4
icfbeamizz=0; %mm^4

End Bend Diaphragm Data

ebdtype=1; %selected type
ebdtrussarea=3225.8; %mm^2
topbeamarea=8129; %mm^2
topbeamiiyy=230592210; %mm^4
topbeamizz=5952109; %mm^4
bottombeamarea=3225.8; %mm^2
bottombeamiiyy=8699237; %mm^4
bottombeamizz=4828285; %mm^4

Middle Bent Diaphragm Data

mdbtype=1; %selected type
mdbtrussarea=0; %mm^2
mdbbeamarea=0; %mm^2
mdbbeamiiyy=0; %mm^4
mbdbeamizz=0; %mm^4

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% S.I.P. Form Data

panspacing=610; %mm
diagonalarea=103.2; %mm
rodarea=25.81; %mm
couples=75; %mm

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Filename: eno_input.m

%User Input

%Main Data
skew=90;                %degrees -> if no skew, put 90 (not 0)
spantype=1;
span1ending=-71940;     %mm
%span2ending=
%span3ending=

girders=5;              %girders
spacing=2940;           %mm
overhang1=1040;         %mm
overhang2=470;          %mm

steelmod=200000;        %N/mm^2
steelpois=0.3;          %Poisson's Ratio
concmod=21526;          %N/mm^2
concpois=0.2;           %Poisson's Ratio

-------------------------------------------------------------------------
%Flange Sections
numsections=3;              %Input number of different sections
webheight=2580;             %mm
webthickness=14;            %mm

thickesttopflange=28;       %mm
thickestbottomflange=50;    %mm

%The following variable is not changed by the user
sectionarray=zeros(numsections,5);   %initialize array of thicknesses

%Remove "%" on the following variables to make active.
%NOTE: WHICHEVER SECTION IS THE LAST SECTION, THE ENDING Z-COORDINATE
% WILL BE THE END OF THE GIRDER
%First Section
sectionarray(1,1)=-17910;       %mm - ending zcoordinate
sectionarray(1,2)=510;          %mm - top flange width
sectionarray(1,3)=28;           %mm - top flange thickness
sectionarray(1,4)=580;          %mm - bottom flange width
sectionarray(1,5)=30;           %mm - bottom flange thickness

%Second Section
sectionarray(2,1)=-54030;       %mm
sectionarray(2,2)=510;          %mm
sectionarray(2,3)=28;           %mm
sectionarray(2,4)=580;          %mm
sectionarray(2,5)=50;           %mm

%Third Section
sectionarray(3,1)=-71940;       %mm
sectionarray(3,2)=510;          %mm
sectionarray(3,3)=28;           %mm
sectionarray(3,4)=580;          %mm
sectionarray(3,5)=30;           %mm
%Concrete Data
buildup=75;           %mm
slab=230;              %mm
integralbents=0;        % 0 = no integral bents & 1 = integral bents
numpours=1;             %input number of pour sequences on bridge

%The following variable is not changed by user
pourarray=zeros(numpours-1,2);

%Remove "%" in front of variable to make active
%pourarray(1,1)=;       %mm
%pourarray(2,1)=;       %mm

%Small loop not to be changed by user to fill in tag ID 
for ii=1:numpours-1
    pourarray(ii,2)=20;
end

%Bearing Stiffeners
numbearing=2;           %input the number of bearing locations on bridge

%The following variable is not changed by user
beararray=zeros(numbearing,2);      %initialize bearing stiffener array

%Remove "%" in front of variables to make active
beararray(1,1)=-290;            %mm - coordinate
beararray(1,2)=28;              %mm - thickness
beararray(2,1)=-71650;          %mm - coordinate
beararray(2,2)=28;              %mm - thickness

%Intermediate Stiffeners
stiffthick=12;      %mm
numstiffright=0;    %input the number on the right side of interior girder
numstiffleft=15;    %input the number on the left side of interior

%The following two variables are not changed by user.
rightstiffarray=zeros(numstiffright,1);  %initializes array
leftstiffarray=zeros(numstiffleft,1);   %initializes array

%Note: For the following input, the arrays for left and right must be
%filled in correctly. For instance, if there were 3 right side
%stiffeners and 2 left side stiffeners, the user will have up to
%rightstiffarray(3,1) and leftstiffarray(2,1).

%fill in z-coordinates for right side intermediate stiffeners
%rightstiffarray(1)=;
%fill in z-coordinates for left side intermediate stiffeners
leftstiffarray(1)=-1640;
leftstiffarray(2)=-2990;
leftstiffarray(3)=-8390;
leftstiffarray(4)=-13790;
leftstiffarray(5)=-19470;
leftstiffarray(6)=-24970;
leftstiffarray(7)=-30470;
leftstiffarray(8)=-35970;
leftstiffarray(9)=-41470;
leftstiffarray(10)=-46970;
leftstiffarray(11)=-52470;
leftstiffarray(12)=-58150;
leftstiffarray(13)=-63550;
leftstiffarray(14)=-68950;
leftstiffarray(15)=-70300;
%-------------------------------------------------------------------------
%Connector Plates

connthick=20;           %mm
connright=-5690;        %mm
rightspacing=5505.45;   %mm
%connright2=            %mm
%rightspacing2=         %mm
%connright3=            %mm
%rightspacing3=         %mm
%-------------------------------------------------------------------------
%Measurement Locations / Pot Locations

numpots=3;  %input the number of measurement locations along girder

%The following variable is not changed by user
potarray=zeros(numpots,1);

%Remove "%" in front of variables to make active
potarray(1)=-17985;         %mm
potarray(2)=-35970;         %mm
potarray(3)=-53955;         %mm
%potarray(4)=

%-------------------------------------------------------------------------
%Intermediate Cross Frame Data

crossframelayout=1;         %selected type
icftype=1;                  %selected type
icfrussarea=1980.64;        %mm^2
icfbeamarea=0;              %mm^2
icfbeamiyy=0;               %mm^4
icfbeamizz=0;               %mm^4
%-------------------------------------------------------------------------
%End Bend Diaphragm Data

ebdtype=1;                  %selected type
ebdrussarea=1980.6;         %mm^2
topbeamarea=5883.9;         %mm^2
topbeamiyy=84078700;        %mm^4
topbeamizz=4703420; %\text{mm}^4
bottombeamarea=1980.6; %\text{mm}^2
bottombeamiyy=3096760; %\text{mm}^4
bottombeamizz=3096760; %\text{mm}^4

%-------------------------------------------------------------------------
%Middle Bent Diaphragm Data

mbdtype=1; %selected type
mbdtrussarea=0; %\text{mm}^2
mbdbeamarea=0; %\text{mm}^2
mbdbeamiyy=0; %\text{mm}^4
mbdbeamizz=0; %\text{mm}^4

%-------------------------------------------------------------------------
%S.I.P. Form Data

panspacing=574; %\text{mm}
diagonalarea=103.23; %\text{mm}
rodarea=25.81; %\text{mm}
couples=75; %\text{mm}

%-------------------------------------------------------------------------

268
%Filename:  us29_input.m

%User Input
%Main Data
skew=46;                %degrees  ->  if no skew, put 90 (not 0)
spantype=1;
span1ending=-37744.4;   %mm
%span2ending=           %mm
%span3ending=           %mm
girders=7;             %girders
spacing=2362.2;        %mm
overhang1=698.5;       %mm
overhang2=698.5;       %mm
steelmod=200000;       %N/mm^2
steelpois=0.3;         %Poisson's Ratio
concmod=21526;         %N/mm^2
concpois=0.2;          %Poisson's Ratio
%-------------------------------------------------------------------------
%Flange Sections
numsections=3;                  %Input number of different sections
webheight=1320.8;               %mm
webthickness=12.7;              %mm
thickesttopflange=19.1;         %mm
thickestbottomflange=34.925;    %mm
%The following variable is not changed by the user
sectionarray=zeros(numsections,5);   %initialize array of thicknesses
%Remove "#" on the following variables to make active.
%NOTE:  WHICHEVER SECTION IS THE LAST SECTION, THE ENDING Z-COORDINATE
%      WILL BE THE END OF THE GIRDER
%First Section
sectionarray(1,1)=-8204.2;      %mm - ending zcoordinate
sectionarray(1,2)=381;          %mm - top flange width
sectionarray(1,3)=19.1;         %mm - top flange thickness
sectionarray(1,4)=381;          %mm - bottom flange width
sectionarray(1,5)=19.1;         %mm - bottom flange thickness
%Second Section
sectionarray(2,1)=-29540.2;     %mm
sectionarray(2,2)=381;          %mm
sectionarray(2,3)=19.1;         %mm
sectionarray(2,4)=381;          %mm
sectionarray(2,5)=34.925;       %mm
%Third Section
sectionarray(3,1)=-37744.4;     %mm
sectionarray(3,2)=381;          %mm
sectionarray(3,3)=19.1;         %mm
sectionarray(3,4)=381;          %mm
sectionarray(3,5)=19.1;  %mm

%Concrete Data
buildup=63.5;  %mm
slab=209.55;  %mm
integralbents=0;  % 0 = no integral bents & 1 = integral bents
numpours=1;  %input number of pour sequences on bridge

%The following variable is not changed by user
pourarray=zeros(numpours-1,2);

%Remove "%" in front of variable to make active
%pourarray(1,1)=;  %mm
%pourarray(2,1)=;  %mm

%Small loop not to be changed by user to fill in tag ID #
for ii=1:numpours-1
    pourarray(ii,2)=20;
end

%Bearing Stiffeners
numbearing=2;  %input the number of bearing locations on bridge

%The following variable is not changed by user
beararray=zeros(numbearing,2);  %initialize bearing stiffener array

%Remove "%" in front of variables to make active
beararray(1,1)=-165;  %mm - coordinate
beararray(1,2)=19.1;  %mm - thickness
beararray(2,1)=-37579.3;  %mm - coordinate
beararray(2,2)=19.1;  %mm - thickness

%Intermediate Stiffeners
stiffthick=9.525;  %mm
numstiffright=2;  %input the number on the right side of interior girder
numstiffleft=2;  %input the number on the left side of interior

%The following two variables are not changed by user.
rightstiffarray=zeros(numstiffright,1);  %initializes array
leftstiffarray=zeros(numstiffleft,1);  %initializes array

%Note:  For the following input, the arrays for left and right must be
% filled in correctly.  For instance, if there were 3 right side
% stiffeners and 2 left side stiffeners, the user will have up to
% rightstiffarray(3,1) and leftstiffarray(2,1).

%fill in z-coordinates for right side intermediate stiffeners
rightstiffarray(1)=-832.36;
rightstiffarray(2)=-37270.18;
%fill in z-coordinates for left side intermediate stiffeners
leftstiffarray(1)=-474.22;
leftstiffarray(2)=-36912.04;

%Connector Plates

connthick=9.525;        %mm
connright=-6261.1;      %mm
rightspacing=6305.55;   %mm
%connright2=            %mm
%rightspacing2=         %mm
%connright3=            %mm
%rightspacing3=         %mm

%Measurement Locations / Pot Locations

numpots=3;  %input the number of measurement locations along girder

%The following variable is not changed by user
potarray=zeros(numpots,1);

%Remove "%" in front of variables to make active
potarray(1)=-9518.7;        %mm
potarray(2)=-18872.2;       %mm
potarray(3)=-28225.8;       %mm
%potarray(4)=

%Intermediate Cross Frame Data

crossframelayout=1;         %selected type
icftype=2;                  %selected type
icftrussarea=1148.4;        %mm^2
icfbeamarea=1148.4;         %mm^2
icfbeamiyy=624347;          %mm^4
icfbeamizz=624347;          %mm^4

%End Bent Diaphragm Data

ebdtype=2;                  %selected type
ebdtrussarea=1696.8;        %mm^2
topbeamarea=6419.3;         %mm^2
topbeamiyy=131113000;       %mm^4
topbeamizz=3358990;         %mm^4
bottombeamarea=1696.8;      %mm^2
bottombeamiyy=1419350;      %mm^4
bottombeamizz=1656600;      %mm^4

%Middle Bent Diaphragm Data

mbdtype=1;                  %selected type
mbdtrussarea=0;             %mm^2
mbdbeamarea=0; \quad \text{mm}^2
mbdbeamiyy=0; \quad \text{mm}^4
mbdbeamizz=0; \quad \text{mm}^4

%------------------------------------------------------------------------
%S.I.P. Form Data

panspacing=864; \quad \text{mm}
diagonalarea=25.81; \quad \text{mm}
rodarea=25.81; \quad \text{mm}
couples=75; \quad \text{mm}

%------------------------------------------------------------------------
%Filename: wilmington_input.m

%User Input

%Main Data
skew=152; %degrees -> if no skew, put 90 (not 0)
spantype=1;
span1ending=-45567.6; %mm
%span2ending=
%span3ending=
girders=5; %girders
spacing=2514.6; %mm
overhang1=304.8; %mm
overhang2=915.6; %mm
steelmod=200000; %N/mm^2
steelpois=0.3; %Poisson's Ratio
concmd=21526; %N/mm^2
concpois=0.2; %Poisson's Ratio

%-------------------------------------------------------------------------
%Flange Sections

numsections=3; %Input number of different sections
webheight=1371.6; %mm
webthickness=12.7; %mm
thickesttopflange=34.925; %mm
thickestbottomflange=47.625; %mm

%The following variable is not changed by the user
sectionarray=zeros(numsections,5); %initialize array of thicknesses

%Remove "%" on the following variables to make active.
%NOTE: WHICHEVER SECTION IS THE LAST SECTION, THE ENDING Z-COORDINATE % WILL BE THE END OF THE GIRDER

%First Section
sectionarray(1,1)=-9525; %mm - ending z-coordinate
sectionarray(1,2)=406.4; %mm - top flange width
sectionarray(1,3)=25.4; %mm - top flange thickness
sectionarray(1,4)=508; %mm - bottom flange width
sectionarray(1,5)=28.575; %mm - bottom flange thickness

%Second Section
sectionarray(2,1)=-36042.6; %mm
sectionarray(2,2)=406.4; %mm
sectionarray(2,3)=34.925; %mm
sectionarray(2,4)=508; %mm
sectionarray(2,5)=47.625; %mm

%Third Section
sectionarray(3,1)=-45567.6; %mm
sectionarray(3,2)=406.4; %mm
sectionarray(3,3)=25.4; %mm
sectionarray(3,4)=508; %mm
sectionarray(3,5)=28.575; %mm

%---------------------------------------------------------------
%Concrete Data
buildup=63.5; %mm
slab=215.9; %mm
integralbents=0; % 0 = no integral bents & 1 = integral bents

numpours=1; %input number of pour sequences on bridge

%The following variable is not changed by user
pourarray=zeros(numpours-1,2);

%Remove "%" in front of variable to make active
pourarray(1,1)=; %mm
pourarray(2,1)=; %mm

%Small loop not to be changed by user to fill in tag ID #
for ii=1:numpours-1
    pourarray(ii,2)=20;
end

%---------------------------------------------------------------
%Bearing Stiffeners
numbearing=2; %input the number of bearing locations on bridge

%The following variable is not changed by user
beararray=zeros(numbearing,2); %initialize bearing stiffener array

%Remove "%" in front of variables to make active
beararray(1,1)=-203.2; %mm - coordinate
beararray(1,2)=25.4; %mm - thickness
beararray(2,1)=-45364.4; %mm - coordinate
beararray(2,2)=25.4; %mm - thickness

%---------------------------------------------------------------
%Intermediate Stiffeners
stiffthick=12; %mm

numstiffright=0; %input the number on the right side of interior girder
numstiffleft=0; %input the number on the left side of interior

%The following two variables are not changed by user.
rightstiffarray=zeros(numstiffright,1); %initializes array
leftstiffarray=zeros(numstiffleft,1); %initializes array

%Note: For the following input, the arrays for left and right must be
%filled in correctly. For instance, if there were 3 right side
%stiffeners and 2 left side stiffeners, the user will have up to
%rightstiffarray(3,1) and leftstiffarray(2,1).

%fill in z-coordinates for right side intermediate stiffeners
%rightstiffarray(1)=;
%fill in z-coordinates for left side intermediate stiffeners
%leftstiffarray(1)=;

%-------------------------------------------------------------------------
%Connector Plates

connthick=12.7; %mm
connright=-2127.25; %mm
rightspacing=6096; %mm
%connright2=
%rightspacing2=
%connright3=
%rightspacing3=

%-------------------------------------------------------------------------
%Measurement Locations / Pot Locations

numpots=3; %input the number of measurement locations along girder

%The following variable is not changed by user
potarray=zeros(numpots,1);

%Remove "%" in front of variables to make active
potarray(1)=-11493.5; %mm
potarray(2)=-27297.9; %mm
potarray(3)=-34074.1; %mm
potarray(4)=

%-------------------------------------------------------------------------
%Intermediate Cross Frame Data

crossframelayout=1; %selected type
icftype=2; %selected type
icftrussarea=1148.4; %mm^2
icfbeamarea=1148.4; %mm^2
icfbeamixy=624347; %mm^4
icfbeamizz=624347; %mm^4

%-------------------------------------------------------------------------
%End Bent Diaphragm Data

ebdtype=2; %selected type
ebdtrussarea=2283.9; %mm^2
topbeamaarea=9483.9; %mm^2
topbeamixy=168157496; %mm^4
topbeamizz=4578546; %mm^4
bottombeamaarea=2283.9; %mm^2
bottombeamixy=1469297; %mm^4
bottombeamizz=3804355; %mm^4

%-------------------------------------------------------------------------
%Middle Bent Diaphragm Data

mbdtype=1; %selected type
mbdtrussarea=0; %mm^2
mbdbeamaarea=0; %mm^2

275
mbdbeamiyy=0; \text{ %mm}^4
mbdbeamizz=0; \text{ %mm}^4

%-------------------------------------------------------------------------
%S.I.P. Form Data

panspacing=813; \text{ %mm}
diagonalarea=21.68; \text{ %mm}
rodarea=25.81; \text{ %mm}
couples=75; \text{ %mm}

%-------------------------------------------------------------------------
%Filename: main.m

%This is the file to run in the command window in MATLAB. The file calls
% the necessary MATLAB files to run the program and to output commands.
%_________________________________________________________________________
%Variables:
% fid: This is a file identifier representing the output file
%_________________________________________________________________________

clear               %clears the memory
clc                 %clears the screen

%Open the output file to print ANSYS commands to:

%***** User changes file name as desired *****
fid=fopen('bridge10_output.m','w');

%Need to print out this for the coordinate system shifts
fprintf(fid,'csys,4 
');

%Call the file with the user's data of the given bridge:
% ***** Type correct input file here *****
bridge10_input

%Call code files to write output to file
elements
preliminary
sections
bearings
stiffeners
connplates
pots
zcoords
dummycoords
girderkeypoints
girderareas
drawconnplates
drawstiffeners
drawbearings
slabkeypoints
slabareas
areamesh
rigidlinks
ebdarrays

%Only create middle bent diaphragms if needed
if spantype > 1
    if crossframelayou=1
        mbdarrays
    end %end IF
end %end IF
icfarrays

%Print command to select all entities

277
fprintf(fid,'allsel \n');

%Close the output command file after writing to file
close(fid);

%Open a new file for the SIP commands

%***** User changes file name as desired *****
fid=fopen('bridge10_pans.m','w');

pans

%Close the SIP output command file after writing to file
fclose(fid);

%END OF FILE & END OF PROGRAM
%-------------------------------------------------------------------------
%_______________________________________________________________________

278
% This file selects areas and applies the correct attributes to them. It 
% does all areas from the girders to the slab. By "applying" the 
% attributes, the file just prints the correctly formatted ANSYS 
% commands into the output file. 
% Note: The concrete deck is not meshed, but can be meshed manually later 
% in ANSYS if needed for composite action analysis. 

% Variables: 
% marker: rcplatearray marker variable 
% counter1: Keeps up with the first area selected 
% counter2: Keeps up with the last area selected 
% mesharray: Applies attributes, common for printing commands to 
% file 
% pointer: Random pointer, multiple applications 
% starting: Keeps up with the given starting area 
% ending: Keeps up with the given ending area 
% bearmesharray: Applies bearing stiffener attributes 

% Calculate the number of areas on each girder (all the same) 
pergirder=girderareacounter/girders; 
% Apply attributes to bottom flanges 
marker=6; % sixth entry of rcplatearray (first bottom flange) 
counter1=1; % initialize 

% Loop to build and print array for each bottom flange section 
for ii=1:numdummyfirst 
   if dummyfca(ii,2)==10|dummyfca(ii,2)==11|dummyfca(ii,2)==0|... 
      dummyfca(ii,2)==21 
      mesharray=zeros(girders*2,3); % initialize array 
      counter2=dummyfca(ii,3); % initialize variable 
   end %end IF 
   mesharray(1,1)=counter1; 
   mesharray(2,1)=counter1+1; 
   mesharray(1:2,2)=(counter2-1)*5; 
   pointer=3; 

   % Fill in other rows 
   for jj=3:girders*2 
      mesharray(pointer,1:2)=mesharray(pointer-2,1:2)+pergirder; 
      pointer=pointer+1; 
   end %end FOR 
   mesharray(pointer,3)=5; 
end %end FOR 

% Print statement to select the first line of areas 
fprintf(fid,'ASEL,s,area,, %5.0f , %5.0f , %1.0f \n',...
mesharray(1,:));

%Loop to print statements selecting the other areas
for jj=2:girders*2
    fprintf(fid,'ASEL,a,area,, %5.0f , %5.0f , %1.0f \n',...
            mesharray(jj,:));
end %end FOR

%Print the attribute statement and size the areas
fprintf(fid,'AATT,1, %2.0f ,1 \n',rcplatearray(marker,1));
fprintf(fid,'AESIZE,all,300 \n');

counter1=(counter2-1)*5+1;      %renew the counter1 variable
marker=marker+2;                %increment
end %end IF
end %end big FOR

%-------------------------------------------------------------------------
%Apply attributes to the top flanges
marker=5;       %fifth entry of rcplatearray (first top flange)
counter1=1;     %initialize

%Loop to build and print array for each top flange section
for ii=1:numdummyfirst
    if dummyfca(ii,2)==10|dummyfca(ii,2)==11|dummyfca(ii,2)==0|...
        dummyfca(ii,2)==21
        mesharray=zeros(girders*2,3);   %initialize array
        counter2=dummyfca(ii,3);        %initialize variable
    end %end IF
    mesharray(1,1)=counter1+3;
    mesharray(2,1)=counter1+4;
    mesharray(1:2,2)=(counter2-1)*5;
    pointer=3;

    %Fill in the other rows
    for jj=3:girders*2
        mesharray(pointer,1:2)=mesharray(pointer-2,1:2)+pergirder;
        pointer=pointer+1;
    end %end FOR

    %Fill in the third column with a 5
    for jj=1:girders*2
        mesharray(jj,3)=5;
    end %end FOR

    %Print statement to select the first line of areas
    fprintf(fid,'ASEL,s,area,, %5.0f , %5.0f , %1.0f \n',...
            mesharray(1,:));

    %Loop to print statements selecting the other areas
    for jj=2:girders*2
        fprintf(fid,'ASEL,a,area,, %5.0f , %5.0f , %1.0f \n',...
                mesharray(jj,:));
    end %end FOR

%Print the attribute statement and size the areas
fprintf(fid,'AATT,1, %2.0f ,1 \n',rcplatearray(marker,1));
fprintf(fid,'AESIZE,all,300 \n');

counter1=(counter2-1)*5+1;  %renew the counter1 variable
marker=marker+2;              %increment
end  %end IF
end  %end FOR
%-------------------------------------------------------------------------
%Apply the attributes to the girder web

%Print statements to file
fprintf(fid,'ASEL,s,area,,3, %5.0f ,5 \n',girderareacounter);
fprintf(fid,'AATT,1,8,1 \n');
fprintf(fid,'AESIZE,all,300 \n');

%Note:  The web thickness will always be the eigth real constant set.
%-------------------------------------------------------------------------
%Apply the attributes to connector plates
starting=girderareacounter+1;
ending=girderareacounter+connplateareacounter;

%Print statements to file
fprintf(fid,'ASEL,s,area,, %5.0f , %5.0f \n',starting,ending);
fprintf(fid,'AATT,1,6,1 \n');
fprintf(fid,'AESIZE,all,300 \n');

%Note:  The connector plate thickness will always be the sixth real constant set.
%-------------------------------------------------------------------------
%Apply the attributes to the intermediate stiffeners

%Only if intermediate stiffeners are present
if numstiff > 0
    starting=ending+1;
    ending=ending+stiffareacounter;
    fprintf(fid,'ASEL,s,area,, %5.0f , %5.0f \n',starting,ending);
    fprintf(fid,'AATT,1,7,1 \n');
    fprintf(fid,'AESIZE,all,300 \n');
end  %end IF

%Note:  The intermediate stiffener thickness will always be the seventh real constant set.
%-------------------------------------------------------------------------
%Apply the attributes to the bearing stiffeners

%Calculate location of first bearing plate thickness in rcplatearray
marker=4+numsections*2+1;
%Initialize
starting=ending+1;
ending=ending+bearingareacounter;
pointer=starting;
%Loop to select and print attributes to bearing stiffeners
for ii=1:numbearing
    bearmesharray=zeros(2,3);       %initialize array
    %Fill in first column
    bearmesharray(1,1)=pointer;
    bearmesharray(2,1)=pointer+1;
    %Fill in second and third columns
    bearmesharray(:,2)=ending;
    bearmesharray(:,3)=numbearing*2;
    %Print statements
    fprintf(fid,'ASEL,s,area,, %5.0f , %5.0f , %1.0f \n',...        
        bearmesharray(1,:));
    fprintf(fid,'ASEL,a,area,, %5.0f , %5.0f , %1.0f \n',...        
        bearmesharray(2,:));
    fprintf(fid,'AATT,1, %2.0f ,1 \n',rcplatearray(marker,1));
    fprintf(fid,'AESIZE,all,300 \n');
    %increment variables
    pointer=pointer+2;
    marker=marker+1;
end %end FOR
%-------------------------------------------------------------------------
%Mesh all of the areas, except for the deck
fprintf(fid,'ASEL,s,area,,1, %5.0f \n',ending);
fprintf(fid,'AMESH,all \n');
%-------------------------------------------------------------------------
%Apply attributes to the concrete deck areas
starting=ending+1;
ending=ending+numslabarears;
fprintf(fid,'ASEL,s,area,, %5.0f , %5.0f \n',starting,ending);
fprintf(fid,'AATT,2,5,2 \n');
fprintf(fid,'AESIZE,all,1500 \n');
%Note:  The concrete slab thickness will always be the fifth
%      real constant set.
%-------------------------------------------------------------------------
%Filename: bearings.m

%This file takes the bearing plate coordinate array entered by the user and correctly sizes the array and fills in the correct tag IDs. Note that the program fills in the array differently for different bridge types. The bearing plate coordinate array (beararray) will be used later in the program.

%Variables:
% bearholder: Temporary duplicate array to hold the coordinate values
% Tags Used: 5: First end bent bearing stiffeners
% 6: Other end bent bearing stiffeners
% 7: First middle bent bearing stiffeners
% 8: Other middle bent bearing stiffeners

bearholder=beararray; %set the temporary array

%Re-initialize bearing array to correct size
beararray=zeros(numbearing,2);

%Loop to correctly fill in the bearing plate array of z-coordinates
%First, for a simple span
if spantype==1
    for ii=1:numbearing
        beararray(ii,1)=bearholder(ii,1);
        if ii==1
            beararray(ii,2)=5; %for first end bent bearing plates
        elseif ii==2
            beararray(ii,2)=6; %for other end bent bearing plates
        end %end IF
    end %end FOR
%For two-span continuous
elseif spantype==2
    for ii=1:numbearing
        beararray(ii,1)=bearholder(ii,1);
        if ii==1
            beararray(ii,2)=5; %for first end bent bearing plates
        elseif ii==2
            beararray(ii,2)=7; %for other end bent bearing plates
        elseif ii==3
            beararray(ii,2)=6; %for first middle bent bearing plates
        end %end IF
    end %end FOR
%For three-span continuous
elseif spantype==3
    for ii=1:numbearing
        beararray(ii,1)=bearholder(ii,1);
        if ii==1
            beararray(ii,2)=5; %for first end bent bearing plates
        elseif ii==2
            beararray(ii,2)=7; %for other end bent bearing plates
        elseif ii==3
            beararray(ii,2)=6; %for first middle bent bearing plates
        elseif ii==4
            beararray(ii,2)=8; %for other middle bent bearing plates
        end %end IF
    end %end FOR
beararray(ii,2)=8; %for first middle bent bearing plates
elseif ii==4
    beararray(ii,2)=6; %for other middle bent bearing plates
end %end IF
end %end FOR
end %end IF/ELSE
%---------------------------------------------------------------
%Filename: connplates.m

%This program takes the users input (in 'main.m') for the first connector
% plate location and the spacing of the cross frames (hence the
% connector plates) and creates an array to hold the z-coordinates of
% the connector plates - one for the right side and one for the left
% side.

%_________________________________________________________________________
%Variables:
%_________________________________________________________________________
%rightconnarray: %array for connector plates on right side of girders
%leftconnarray: %array for connector plates on left side of girders
%rightholder: %duplicate array of right side connector plates for
%           % manipulation
%leftholder: %see rightholder above (left side now)
%npointer: %pointer variable to find the first zero in array and
%          % then moved up one to the last item in array
%numconns: %variable to remember the number of connector plates
%_________________________________________________________________________
%Tags Used:     1:  Right Side Connector Plates
%               2:  Left Side Connector Plates
%_________________________________________________________________________

%initialize connector plate arrays
rightconnarray=zeros(100,1);
leftconnarray=zeros(100,1);
ii=connright;       %counter for coordinate along span
jj=1;               %counter for entry into the connector plate arrays

%The arrays must be treated differently for different skews
if skew>90
    while ii >= 0.98*(span1ending+zshift)
        rightconnarray(jj,1)=ii;
        leftconnarray(jj,1)=ii-zshift;
        ii=ii-rightspacing;
        jj=jj+1;
    end %end WHILE
else
    while ii >= 0.98*span1ending
        rightconnarray(jj,1)=ii;
        leftconnarray(jj,1)=ii-zshift;
        ii=ii-rightspacing;
        jj=jj+1;
    end %end WHILE
end %end IF
%Do the next steps only for continuous span bridges
if spantype==2|spantype==3
    ii=connright2;
    if skew>90
        while ii >= 0.98*(span2ending+zshift)
            rightconnarray(jj,1)=ii;
            leftconnarray(jj,1)=ii-zshift;
            ii=ii-rightspacing2;
            jj=jj+1;
        end %end WHILE
    else
        %end WHILE
    end %end IF
end %end IF
while ii >= 0.98*span2ending
    rightconnarray(jj,1)=ii;
    leftconnarray(jj,1)=ii-zshift;
    ii=ii-rightspacing2;
    jj=jj+1;
end %end WHILE
end %end IF/ELSE
%Do the next step only if there is a third span
if spantype==3
    ii=connright3;
    if skew>90
        while ii >= 0.98*(span3ending+zshift)
            rightconnarray(jj,1)=ii;
            leftconnarray(jj,1)=ii-zshift;
            ii=ii-rightspacing3;
            jj=jj+1;
        end %end WHILE
    else
        while ii >= 0.98*span3ending
            rightconnarray(jj,1)=ii;
            leftconnarray(jj,1)=ii-zshift;
            ii=ii-rightspacing3;
            jj=jj+1;
        end %end WHILE
    end %end IF/ELSE
end %end IF spantype == 3
end %end IF spantype == 2 or 3

%Recreate arrays to correct size
rightholder=rightconnarray;     %temp array for the right conn plates
leftholder=leftconnarray;       %temp array for the left conn plates

%Only need one pointer b/c # of right conn plates should = # of left ones
%point to where the first zero is in the array
pointer=find(rightconnarray==0);
pointer=pointer(1,1);          %grab the first reference to a zero term
pointer=pointer-1;             %subtract one for reference

%Re-initialize arrays to correct size, but now with an extra column for
%the tag identification to be used later.
rightconnarray=zeros(pointer,2);
leftconnarray=zeros(pointer,2);

%Loops to fill back in the connector plate arrays
for ii=1:pointer
    rightconnarray(ii,1)=rightholder(ii,1);
    rightconnarray(ii,2)=1;
    leftconnarray(ii,1)=leftholder(ii,1);
    leftconnarray(ii,2)=2;
end %end FOR

numconns=pointer;       %hold this number for later
%-------------------------------------------------------------------------------------
%Filename: couplefixes.m (the following is an example)

%This file takes the bad couples from ANSYS (input by the user) and
% outputs commands to delete the bad couples and create new couples
% where needed.

%_________________________________________________________________________

%Variables:
%
%numdeletes:    Number of coupled node sets to be deleted
%deletearray:   Deleted coupled sets array
%numredos:      Number of coupled node sets to be re-coupled
%                   (Each will be re-coupled in all three directions)
%redoarray:     Array holding node numbers to be coupled together
%
%_________________________________________________________________________

%Clear screen and memory
clear
cic

numdeletes=18;  %initialize
%Create/Fill-in array that holds coupled sets to be deleted
deletearray=[
    1
    48
    95
    142
    188
    234
    280
    327
    374
    421
    467
    513
    559
    606
    653
    700
    746
    792
];

%******** USER enters NODE NUMBERS here (IN ORDER and IN PAIRS) ********
numredos=3;     %initialize
%Create/Fill-in array that holds coupled sets to be re-coupled
redoarray=[
    3623   100047
    7164   100139
    10705  100231
];

%Print commands to delete the coupled sets
for ii=1:numdeletes
    fprintf('cpdele, %6.0f ,,,any \n',deletearray(ii))
end %end FOR

%Print commands to create a new coupled set
for ii=1:numredos
    fprintf('cp,next,ux, %8.0f , %8.0f \n',redoarray(ii,:))
    fprintf('cp,next,uy, %8.0f , %8.0f \n',redoarray(ii,:))
    fprintf('cp,next,uz, %8.0f , %8.0f \n',redoarray(ii,:))
end %end FOR
%-------------------------------------------------------------------------
%Filename: drawbearings.m

%This file searches the girder coordinate arrays and finds where the
% bearing stiffeners (left and right) should be drawn. This file also
% draws the actual bearing stiffeners and increments the area & bearing
% stiffener area counters.

%_________________________________________________________________________
%Variables:
%
%holder: Tag ID match coordinate
:pointer: Points the the holder coordinate in the dummy array
%one: Represents the first keypoint of the stiffener area
%two: Represents the second keypoint of the stiffener area
%three: Represents the third keypoint of the stiffener area
%four: Represents the fourth keypoint of the stiffener area
%five: Represents the fifth keypoint of the stiffener area
%six: Represents the sixth keypoint of the stiffener area
%bearingareacounter: Bearing stiffener area counter
%temp: Variable to make if statements easier to read

%_________________________________________________________________________

%Initialize counter variable
bearingareacounter=0;

%Loop to "draw" bearing stiffeners on correct sides of girders
for ii=1:girders
    %First Girder (bearing stiffeners on the left and right)
    if ii==1
        %Loop to search through coordinate array
        for jj=1:numfirstcoords
            temp=firstcoordarray(jj,2); %for simpler IF expression
            if temp==5 | temp==6 | temp==7 | temp==8 %search for tag ID
                holder=firstcoordarray(jj,1);
                pointer=find(dummyfca(:,1)==holder);
                one=pointer*6-5;
                two=one+2;
                three=one+5;
                four=one+3;
                five=one+1;
                six=one+4;
                fprintf(fid,'A, %8.0f , %8.0f , %8.0f , %8.0f 
',
                        one,two,three,four);
                fprintf(fid,'A, %8.0f , %8.0f , %8.0f , %8.0f 
',
                        one,five,six,four);
                areacounter=areacounter+2;
                bearingareacounter=bearingareacounter+2;
            end %end IF
        end %end FOR
    elseif ii==girders
        %Loop to search through coordinate array
        for jj=1:numlastcoords
            temp=lastcoordarray(jj,2);
            if temp==5 | temp==6 | temp==7 | temp==8 %search for tag ID
                holder=lastcoordarray(jj,1);
                fprintf(fid,'A, %8.0f , %8.0f , %8.0f , %8.0f 
',
                        one,two,three,four);
                fprintf(fid,'A, %8.0f , %8.0f , %8.0f , %8.0f 
',
                        one,five,six,four);
                areacounter=areacounter+2;
                bearingareacounter=bearingareacounter+2;
            end %end IF
        end %end FOR
    end %end FOR

%Last Girder (bearing stiffeners on the left and right)
elseii=girders
    %Loop to search through coordinate array
    for jj=1:numlastcoords
        temp=lastcoordarray(jj,2);
        if temp==5 | temp==6 | temp==7 | temp==8 %search for tag ID
            holder=lastcoordarray(jj,1);
        end %end FOR

    end %end FOR

end %end FOR
pointer = find(dummylca(:,1) == holder);
one = pointer*6-5+(girders-1)*1000;
two = one+2;
three = one+5;
four = one+3;
five = one+1;
six = one+4;
fprintf(fid,'A, %8.0f , %8.0f , %8.0f , %8.0f \n','
one, two, three, four);
fprintf(fid,'A, %8.0f , %8.0f , %8.0f , %8.0f \n','
one, five, six, four);
%increment area counters
areacounter = areacounter + 2;
bearingareacounter = bearingareacounter + 2;
end
end %end IF
end %end FOR

%Interior Girders (bearing stiffeners on the left and right)
else ii==2
%Must draw bearing stiffeners on ALL interior girders
for kk = 1:numintgirders
  %Loop to search through coordinate array
  for jj = 1:numinteriorcoords
    temp = interiorcoordarray(jj,2); %for simpler IF expression
    %search for tag ID
    if temp == 5 | temp == 6 | temp == 7 | temp == 8
      holder = interiorcoordarray(jj,1);
      pointer = find(dummyica(:,1) == holder);
one = pointer*6-5+1000*kk;
two = one+2;
three = one+5;
four = one+3;
five = one+1;
six = one+4;
fprintf(fid,'A, %8.0f , %8.0f , %8.0f , %8.0f \n','
one, two, three, four);
fprintf(fid,'A, %8.0f , %8.0f , %8.0f , %8.0f \n','
one, five, six, four);
  %increment area counters
  areacounter = areacounter + 2;
bearingareacounter = bearingareacounter + 2;
end
end %end FOR
end %end IF
end %end ELSE
end %end big FOR
%-------------------------------------------------------------
% This file searches the girder coordinate arrays and finds where a connector plate (left or right) should be drawn. This file also draws the actual plate area and increments the area & connector plate area counters.

% Variables:

% holder:                Holds the tag ID match coordinate
% pointer:               Points to the holder coordinate in dummy array
% one:                   Represents the first keypoint of the plate area
% two:                   Represents the second keypoint of the plate area
% three:                 Represents the third keypoint of the plate area
% four:                  Represents the fourth keypoint of the plate area
% connplateareacounter:  Connector plate area counter
% temparray1:            Temporary array to hold first coordinate array
% temparray2:            Temporary array to hold interior coordinate array
% temparray3:            Temporary array to hold last coordinate array

% Initialize counter variable
connplateareacounter=0;
% Initialize temporary arrays
temparray1=firstcoordarray;
temparray2=interiorcoordarray;
temparray3=lastcoordarray;
%----------------------------------------------------------------------------------

% Loop to change the tag ID's of coordinates where there may be a bearing and a connector plate
if spantype==2|spantype==3
  for ii=1:numfirstcoords
    % right side connector plates
    if firstcoordarray(ii,2)==1
      if firstcoordarray(ii,1)==span1ending
        firstcoordarray(ii,2)=25;
        interiorcoordarray(ii,2)=25;
        lastcoordarray(ii,2)=25;
      end
    end
    % left side connector plates
    if firstcoordarray(ii,2)==2
      if firstcoordarray(ii,1)==span1ending
        firstcoordarray(ii,2)=25;
        interiorcoordarray(ii,2)=25;
        lastcoordarray(ii,2)=25;
      end
    end
  end
end
if spantype==3
  % right side connector plates
  if firstcoordarray(ii,2)==1
    if firstcoordarray(ii,1)==span2ending
      firstcoordarray(ii,2)=25;
      interiorcoordarray(ii,2)=25;
      lastcoordarray(ii,2)=25;
    end
  end
end
if spantype==3
  % left side connector plates
  if firstcoordarray(ii,2)==2
    if firstcoordarray(ii,1)==span2ending
      firstcoordarray(ii,2)=25;
      interiorcoordarray(ii,2)=25;
      lastcoordarray(ii,2)=25;
    end
  end
end
if spantype==3
  % right side connector plates
  if firstcoordarray(ii,2)==1
    if firstcoordarray(ii,1)==span1ending
      firstcoordarray(ii,2)=25;
      interiorcoordarray(ii,2)=25;
      lastcoordarray(ii,2)=25;
    end
  end
end
end %end IF  
%left side connector plates  
if firstcoordarray(ii,2)==2  
   if firstcoordarray(ii,1)==span2ending  
      firstcoordarray(ii,2)=25;  
      interiorcoordarray(ii,2)=25;  
      lastcoordarray(ii,2)=25;  
   end %end IF  
end %end IF  
end %end IF spantype==3  
end %end FOR  
end %end IF  

%-------------------------------------------------------------------------  
%Loop to "draw" connector plates on correct sides of girders  
for ii=1:girders  
  %First Girder (connector plates on the right only)  
  if ii==1  
     %Loop to search through coordinate array  
     for jj=1:numfirstcoords  
        if firstcoordarray(jj,2)==1     %search for tag ID  
           holder=firstcoordarray(jj,1);  
           pointer=find(dummyfca(:,1)==holder);  
           one=pointer*6-5;  
           two=one+2;  
           three=one+5;  
           four=one+3;  
           fprintf(fid,'A, %8.0f , %8.0f , %8.0f , %8.0f 
',...
            one,two,three,four);  
           %increment area counters  
           areacounter=areacounter+1; 
           connplateareacounter=connplateareacounter+1;  
      end %end IF  
   end %end FOR  
  %Last Girder (connector plates on the left only)  
  elseif ii==girders  
     %Loop to search through coordinate array  
     for jj=1:numlastcoords  
        if lastcoordarray(jj,2)==2      %search for tag ID  
           holder=lastcoordarray(jj,1);  
           pointer=find(dummylca(:,1)==holder);  
           one=pointer*6-5+(girders-1)*1000;  
           two=one+1;  
           three=one+4;  
           four=one+3;  
           fprintf(fid,'A, %8.0f , %8.0f , %8.0f , %8.0f 
',...
           one,two,three,four);  
           %increment area counters  
           areacounter=areacounter+1;  
           connplateareacounter=connplateareacounter+1;  
      end %end IF  
   end %end FOR  
  %Interior Girders (connector plates on left and right)  
  elseif ii==2  
     %Loop to search through coordinate array  
     for jj=1:numinteriorcoords  
        if interiorcoordarray(jj,2)==1      %search for tag ID  
           holder=interiorcoordarray(jj,1);  
           pointer=find(dummyintca(:,1)==holder);  
           one=pointer*6-5+(girders-1)*1000;  
           two=one+1;  
           three=one+4;  
           four=one+3;  
           fprintf(fid,'A, %8.0f , %8.0f , %8.0f , %8.0f 
',...
           one,two,three,four);  
           %increment area counters  
           areacounter=areacounter+1;  
           connplateareacounter=connplateareacounter+1;  
      end %end IF  
   end %end FOR  
end %end IF
holder=interiorcoordarray(jj,1);
pointer=find(dummyica(:,1)==holder);
one=pointer*6-5+1000;
for kk=1:numintgirders
    two=one+2;
    three=one+5;
    four=one+3;
    fprintf(fid,'A, %8.0f , %8.0f , %8.0f , %8.0f \n',
        one,two,three,four);
    %increment area counters
    areacounter=areacounter+1;
    connplateareacounter=connplateareacounter+1;
    one=one+1000;
end %end FOR
elseif interiorcoordarray(jj,2)==2
    holder=interiorcoordarray(jj,1);
    pointer=find(dummyica(:,1)==holder);
one=pointer*6-5+1000;
for kk=1:numintgirders
    two=one+1;
    three=one+4;
    four=one+3;
    fprintf(fid,'A, %8.0f , %8.0f , %8.0f , %8.0f \n',
        one,two,three,four);
    %increment area counters
    areacounter=areacounter+1;
    connplateareacounter=connplateareacounter+1;
    one=one+1000;
end %end FOR
end %end IF/ELSE
end %end FOR
end %end IF/ELSE
end %end big FOR

%-------------------------------------------------------------------------
%Replace coordinate arrays:
firstcoordarray=temparray1;
interiorcoordarray=temparray2;
lastcoordarray=temparray3;
%-------------------------------------------------------------------------
%Filename:  drawebdtype1.m

%This file prints to file the ANSYS commands to draw the first type of end bent diaphragms. The file also grabs the correct lines to apply attributes.

%Variables:

%starting:      Represents the starting line number to be grabbed
%ending:        Represents the ending line number to be grabbed

%For the bottom chord members in the end bent diaphragms
starting=ending+1;
ending=ending+numbottomchords;

%Loop to print ANSYS commands to draw lines for bottom chords
for ii=1:numbottomchords
    fprintf(fid,'l, %8.0f , %8.0f \n',bottomchordarray(ii,:));
end %end FOR

%Commands to apply the attributes
fprintf(fid,'LSEL,s,line,, %8.0f , %8.0f \n',starting,ending);
fprintf(fid,'LATT,1,4,3 \n');

%Note:  The bottom beam members for the end bent diaphragms always have a real constant set = 4 and are BEAM4 Elements.

%For the top chord members in the end bent diaphragms
starting=ending+1;
ending=ending+numtopchords;

%Loop to print ANSYS commands to draw lines for top chords
for ii=1:numtopchords
    fprintf(fid,'l, %8.0f , %8.0f \n',topchordarray(ii,:));
end %end FOR

%Commands to apply the attributes
fprintf(fid,'LSEL,s,line,, %8.0f , %8.0f \n',starting,ending);
fprintf(fid,'LATT,1,3,3 \n');

%Note:  The top beam members for the end bent diaphragms always have a real constant set = 3 and are BEAM4 Elements.

%For the diagonal truss members in the end bent diaphragms
starting=ending+1;
ending=ending+numebddiagonals;

%Loop to print ANSYS commands to draw lines for truss members
for ii=1:numebddiagonals
    fprintf(fid,'l, %8.0f , %8.0f \n',ebddiagonalarray(ii,:));
end %end FOR

%Commands to apply the attributes
fprintf(fid,'LSEL,s,line,, %8.0f , %8.0f \n',starting,ending);
fprintf(fid,'LATT,1,2,4 \n');

%Note: The truss members for the end bent diaphragms always have a % real constant set = 2 and are LINK8 Elements.
%---------------------------------------------------------------
%Filename: drawebdtype2.m

%This file prints to file the ANSYS commands to draw the first type of
% end bent diaphragms. The file also grabs the correct lines to apply
% attributes.

%_________________________________________________________________________
%Variables:
%
%starting: Represents the starting line number to be grabbed
%ending: Represents the ending line number to be grabbed
%_________________________________________________________________________

%For the bottom chord members in the end bent diaphragms
starting=ending+1;
ending=ending+numbottomchords;

%Loop to print ANSYS commands to draw lines for bottom chords
for ii=1:numbottomchords
  fprintf(fid,'l, %8.0f , %8.0f \n',bottomchordarray(ii,:));
end %end FOR

%Commands to apply the attributes
fprintf(fid,'LSEL,s,line,, %8.0f , %8.0f \n',starting,ending);
fprintf(fid,'LATT,1,4,3 \n');

%Note: The bottom beam members for the end bent diaphragms always have a
% real constant set = 4 and are BEAM4 Elements.
%-------------------------------------------------------------------------

%For the top chord members in the end bent diaphragms
starting=ending+1;
ending=ending+numtopchords;

%Loop to print ANSYS commands to draw lines for top chords
for ii=1:numtopchords
  fprintf(fid,'l, %8.0f , %8.0f \n',topchordarray(ii,:));
end %end FOR

%Commands to apply the attributes
fprintf(fid,'LSEL,s,line,, %8.0f , %8.0f \n',starting,ending);
fprintf(fid,'LATT,1,3,3 \n');

%Note: The top beam members for the end bent diaphragms always have a
% real constant set = 3 and are BEAM4 Elements.
%-------------------------------------------------------------------------

%For all truss members in the end bent diaphragms
starting=ending+1;
ending=ending+numebddiagonals+numebdvertical;

%Loop to print ANSYS commands to draw lines for truss members
for ii=1:numebddiagonals
  fprintf(fid,'l, %8.0f , %8.0f \n',ebddiagonalarray(ii,:));
end %end FOR

%Loop to print ANSYS commands to draw lines for vertical members
for ii=1:numebdvertical
    fprintf(fid,'1, %8.0f , %8.0f \n',ebdverticalarray(ii,:));
end %end FOR

%Commands to apply the attributes
fprintf(fid,'LSEL,s,line,, %8.0f , %8.0f \n',starting,ending);
fprintf(fid,'LATT,1,2,4 \n');

%Note: The truss members for the end bent diaphragms always have a
%real constant set = 2 and are LINK8 Elements.
%---------------------------------------------------------------
%Filename:  drawicftypel.m

%This file prints to file the ANSYS commands to draw the "X" type
% intermediate cross frames.  The file also applies the correct
% attributes to the correct lines.  Since these are the last lines to be
% drawn by the program, the file also grabs all of the drawn lines and
% sizes them to 1 and meshes them.
%_________________________________________________________________________
%Variables:
% %starting:      Starting line number to be grabbed
% %ending:        Ending line number to be grabbed
%_________________________________________________________________________

%For the truss members in the intermediate cross frames  
starting=ending+1;  
ending=ending+numcrossframes*3;

%Print the ANSYS commands to draw lines for the bottom truss members  
for ii=1:numcrossframes  
   fprintf(fid,'l, %8.0f , %8.0f \n',bottomicfkparray(ii,:));  
end %end FOR

%Loop to print the ANSYS commands to draw the diagonals in the icf  
for ii=1:numicfdiagonals  
   fprintf(fid,'l, %8.0f , %8.0f \n',typexdiagonalarray(ii,:));  
end %end FOR

%Commands to apply the attributes  
fprintf(fid,'LSEL,s,line,, %8.0f , %8.0f \n',starting,ending);  
fprintf(fid,'LATT,1,1,4 \n');

%Note:  The members for the intermediate cross frames always have a  
% real constant set = 1 and are LINK8 Elements.
%-------------------------------------------------------------------------

%Size and mesh the line elements  
starting=linesthusfar+1;

%ANSYS commands  
fprintf(fid,'LSEL,s,line,, %8.0f , %8.0f \n',starting,ending);  
fprintf(fid,'LESIZE,all,,,1 \n');  
fprintf(fid,'LMESH,all \n');  
%-------------------------------------------------------------------------
%Filename: drawicftype2.m

%This file prints to file the ANSYS commands to draw the "K" type
% intermediate cross frames. The file also applies the correct
% attributes to the correct lines. Since these are the last lines to be
% drawn by the program, the file also grabs all of the drawn lines and
% sizes them to 1 and meshes them.

%_________________________________________________________________________
%Variables:
% %starting:      Starting line number to be grabbed
% %ending:        Ending line number to be grabbed
%_________________________________________________________________________

%For the bottom truss members in the intermediate cross frames
starting=ending+1;
ending=ending+numicfbottomchords;

%Print the ANSYS commands to draw lines for the bottom beam members
for ii=1:numicfbottomchords
    fprintf(fid,'l, %8.0f , %8.0f \n',icfchordarray(ii,:));
end %end FOR

%Commands to apply the attributes
fprintf(fid,'LSEL,s,line,, %8.0f , %8.0f \n',starting,ending);
fprintf(fid,'LATT,1,9,3 \n');

%Note:  The beam members for the intermediate cross frames always have a
%       real constant set = 9 and are BEAM4 Elements.
%-------------------------------------------------------------------------

%For the diagonal truss members in the intermediate cross frames
starting=ending+1;
ending=ending+numicfdiagonals;

%Print the ANSYS commands to draw the diagonals in the icf
for ii=1:numicfdiagonals
    fprintf(fid,'l, %8.0f , %8.0f \n',typekdiagonalarray(ii,:));
end %end FOR

%Commands to apply the attributes
fprintf(fid,'LSEL,s,line,, %8.0f , %8.0f \n',starting,ending);
fprintf(fid,'LATT,1,1,4 \n');

%Note:  The truss members for the intermediate cross frames always have a
%       real constant set = 1 and are LINK8 Elements.
%-------------------------------------------------------------------------

%Size and mesh the line elements
starting=linesthusfar+1;

%ANSYS commands
fprintf(fid,'LSEL,s,line,, %8.0f , %8.0f \n',starting,ending);
fprintf(fid,'LESIZE,all,,,1 \n');
fprintf(fid,'LMESH,all \n');
%Filename: drawicftype3.m

%This file prints to file the ANSYS commands to draw the "K" type (with
% the top beam member) intermediate cross frames. The file also applies
% the correct attributes to the correct lines. Since these are the last
% lines to be drawn by the program, the file also grabs all of the drawn
% lines, sizes them to 1 and meshes them.

%_________________________________________________________________________
%Variables:

%starting: Starting line number to be grabbed
%ending: Ending line number to be grabbed
%_________________________________________________________________________

%For the bottom beam members in the intermediate cross frames
starting=ending+1;
ending=ending+numicfbottomchords;

%Print the ANSYS commands to draw lines for the bottom beam members
for ii=1:numicfbottomchords
    fprintf(fid,'l, %8.0f , %8.0f \n',icfchordarray(ii,:));
end %end FOR

%Commands to apply the attributes
fprintf(fid,'LSEL,s,line,, %8.0f , %8.0f \n',starting,ending);
fprintf(fid,'LATT,1,9,3 \n');

%Note: The beam members for the intermediate cross frames always have a
% real constant set = 9 and are BEAM4 Elements.
%---------------------------------------------------------------

%For the diagonal truss members in the intermediate cross frames
starting=ending+1;
ending=ending+numicfdiagonals;

%Print the ANSYS commands to draw the diagonals in the icf
for ii=1:numicfdiagonals
    fprintf(fid,'l, %8.0f , %8.0f \n',typekdiagonalarray(ii,:));
end %end FOR

%Commands to apply the attributes
fprintf(fid,'LSEL,s,line,, %8.0f , %8.0f \n',starting,ending);
fprintf(fid,'LATT,1,1,4 \n');

%Note: The truss members for the intermediate cross frames always have a
% real constant set = 1 and are LINK8 Elements.
%---------------------------------------------------------------

%For the top beam members in the intermediate cross frames (type 3)
starting=ending+1;
ending=ending+numcrossframes;

%Print the ANSYS commands to draw lines for the top beam members
for ii=1:numcrossframes
    fprintf(fid,'l, %8.0f , %8.0f \n',topicfkparray(ii,:));
end %end FOR
% Commands to apply the attributes
fprintf(fid,'LSEL,s,line,, %8.0f , %8.0f \n',starting,ending);
fprintf(fid,'LATT,1,9,3 \n');

% Note: The members for the intermediate cross frames always have a
% real constant set = 9 and are BEAM Elements.
%-------------------------------------------------------------------------
% Size and mesh the line elements
starting=linesthusfar+1;

% ANSYS commands
fprintf(fid,'LSEL,s,line,, %8.0f , %8.0f \n',starting,ending);
fprintf(fid,'LESIZE,all,,1 \n');
fprintf(fid,'LMESH,all \n');
%-------------------------------------------------------------------------
%Filename: drawmbdtype1.m

%This file prints to file the ANSYS commands to draw the "X" type
%middle bent diaphragms. The file also applies the correct attributes
%to the correct lines.

%Variables:
%
%starting: Starting line number to be "grabbed"
%ending: Ending line number to be "grabbed"

%For the truss members in the middle bent diaphragms
starting=ending+1;
ending=ending+((girders-1)*nummiddles)*3;

%Print the ANSYS commands to draw lines for the bottom truss members
for ii=1:(girders-1)*nummiddles
    fprintf(fid,'l, %8.0f , %8.0f 
',bottommbdkparray(ii,:));
end %end FOR

%Print the ANSYS commands to draw the diagonals in the mbd
for ii=1:nummbddiagonals
    fprintf(fid,'l, %8.0f , %8.0f 
',typexdiagmbdarray(ii,:));
end %end FOR

%Commands to apply the attributes
fprintf(fid,'LSEL,s,line,, %8.0f , %8.0f \n',starting,ending);
fprintf(fid,'LATT,1,32,4 
');

%Note: The members for the intermediate cross frames always have a
%real constant set = 32 and are LINK8 Elements.
%Filename: drawmbdtype2.m

%This file prints to file the ANSYS commands to draw the "K" type
% middle bent diaphragms. The file also applies the correct attributes
% to the correct lines.

%Variables:
%
%starting: Starting line number to be grabbed
%ending: Ending line number to be grabbed
%
%For the bottom truss members in the middle bent diaphragms
starting=ending+1;
ending=ending+nummbdchords;

%Print the ANSYS commands to draw lines for the bottom beam members
for ii=1:nummbdchords
    fprintf(fid,'l, %8.0f , %8.0f \n',mbdchordarray(ii,:));
end %end FOR

%Commands to apply the attributes
fprintf(fid,'LSEL,s,line,, %8.0f , %8.0f \n',starting,ending);
fprintf(fid,'LATT,1,33,3 \n');

%Note: The beam members for the middle bent diaphragms always have a
% real constant set = 33 and are BEAM4 Elements.

%For the diagonal truss members in the middle bent diaphragms
starting=ending+1;
ending=ending+nummbddiagonals;

%Print the ANSYS commands to draw the diagonals in the mbd
for ii=1:nummbddiagonals
    fprintf(fid,'l, %8.0f , %8.0f \n',typekdiagmbdarray(ii,:));
end %end FOR

%Commands to apply the attributes
fprintf(fid,'LSEL,s,line,, %8.0f , %8.0f \n',starting,ending);
fprintf(fid,'LATT,1,32,4 \n');

%Note: The truss members for the middle bent diaphragms always have a
% real constant set = 32 and are LINK8 Elements.

---
%Filename:  drawmbdtype3.m

%This file prints to file the ANSYS commands to draw the "K" type
% middle bent diaphragms w/ a top horizontal chord member. The file
% also applies the correct attributes to the correct lines.
%_________________________________________________________________________
%Variables:
% %starting:      Starting line number to be grabbed
% %ending:        Ending line number to be grabbed
%_________________________________________________________________________

%For the bottom truss members in the middle bent diaphragms
starting=ending+1;
ending=ending+nummbdchords;

%Print the ANSYS commands to draw lines for the bottom beam members
for ii=1:nummbdchords
    fprintf(fid,'l, %8.0f , %8.0f \n',mbdchordarray(ii,:));
end %end FOR
%Commands to apply the attributes
fprintf(fid,'LSEL,s,line,, %8.0f , %8.0f \n',starting,ending);
fprintf(fid,'LATT,1,33,3 \n');
%Note:  The beam members for the middle bent diaphragms always have a
%       real constant set = 33 and are BEAM4 Elements.
%-------------------------------------------------------------------------

%For the diagonal truss members in the middle bent diaphragms
starting=ending+1;
ending=ending+nummbddiagonals;

%Print the ANSYS commands to draw the diagonals in the mbd
for ii=1:nummbddiagonals
    fprintf(fid,'l, %8.0f , %8.0f \n',typekdiagmbdarray(ii,:));
end %end FOR
%Commands to apply the attributes
fprintf(fid,'LSEL,s,line,, %8.0f , %8.0f \n',starting,ending);
fprintf(fid,'LATT,1,32,4 \n');
%Note:  The truss members for the middle bent diaphragms always have a
%       real constant set = 32 and are LINK8 Elements.
%-------------------------------------------------------------------------

%For the top beam members in the middle bent diaphrams
starting=ending+1;
ending=ending+nummbdtopchords;

%Print the ANSYS commands to draw the top beam members
for ii=1:nummbdtopchords
    fprintf(fid,'l, %8.0f , %8.0f \n',topmbdkparray(ii,:));
end %end FOR
%Commands to apply the attributes
fprintf(fid,'LSEL,s,line,, %8.0f , %8.0f \n',starting,ending);
fprintf(fid,'LATT,1,33,3 \n');

%Note: The top beam members for the middle bent diaphragms always have a
% real constant set = 33 and are BEAM4 Elements.
%-------------------------------------------------------------
%Filename: drawstiffeners.m

%This file searches the girder coordinate arrays and finds where an
% intermediate stiffener (left or right) should be "drawn". The file
% also draws the actual stiffener area and increments the area and
% stiffener area counters.

%_________________________________________________________________________
%Variables:
%
%holder:              Tag ID match coordinate
:pointer:             Points to the holder coordinate in dummy array
%one:                 Represents the first keypoint of the stiffener area
%two:                 Represents the second keypoint of the stiffener area
%three:               Represents the third keypoint of the stiffener area
%four:                Represents the fourth keypoint of the stiffener area
%stiffareacounter:    Intermediate stiffener area counter

%_________________________________________________________________________

%Initialize counter variable
stiffareacounter=0;

%Loop to draw stiffeners on correct sides of girders
for ii=1:girders
    %First Girder (intermediate stiffeners on the right only)
    if ii==1
        %Loop to search through coordinate array
        for jj=1:numfirstcoords
            if firstcoordarray(jj,2)==3 %search for tag ID
                holder=firstcoordarray(jj,1);
                pointer=find(dummyfca(:,1)==holder);
                one=pointer*6-5;
                two=one+2;
                three=one+5;
                four=one+3;
                fprintf(fid,'A, %8.0f , %8.0f , %8.0f , %8.0f \n',... 
                    one,two,three,four);
                %increment area counters
                areacounter=areacounter+1;
                stiffareacounter=stiffareacounter+1;
            end %end IF
        end %end FOR
    end %end IF
    %Last Girder (intermediate stiffeners on the left only)
    elseif ii==girders
        %Loop to search through coordinate array
        for jj=1:numlastcoords
            if lastcoordarray(jj,2)==4 %search for tag ID
                holder=lastcoordarray(jj,1);
                pointer=find(dummylca(:,1)==holder);
                one=pointer*6-5+(girders-1)*1000;
                two=one+1;
                three=one+4;
                four=one+3;
                fprintf(fid,'A, %8.0f , %8.0f , %8.0f , %8.0f \n',... 
                    one,two,three,four);
                %increment area counters
                areacounter=areacounter+1;
                stiffareacounter=stiffareacounter+1;
            end %end IF
        end %end FOR
    elseif
        %Middle Girders (intermediate stiffeners on both sides)
        %Code for middle girders
    else
        %Last Girder (intermediate stiffeners on the left only)
        %Code for last girders
    end %end IF
end %end FOR
elseif ii==2
%Loop to search through coordinate array
for jj=1:numinteriorcoords
    if interiorcoordarray(jj,2)==3  %search for tag ID
        holder=interiorcoordarray(jj,1);
        pointer=find(dummyica(:,1)==holder);
        one=pointer*6-5+1000;
        for kk=1:numintgirders
            two=one+2;
            three=one+5;
            four=one+3;
            fprintf(fid,'A, %8.0f , %8.0f , %8.0f , %8.0f \n',...
                one,two,three,four);
            %increment area counters
            areacounter=areacounter+1;
            stiffareacounter=stiffareacounter+1;
            one=one+1000;
        end
    end
elseif interiorcoordarray(jj,2)==4  %search for tag ID
    holder=interiorcoordarray(jj,1);
    pointer=find(dummyica(:,1)==holder);
    one=pointer*6-5+1000;
    for kk=1:numintgirders
        two=one+1;
        three=one+4;
        four=one+3;
        fprintf(fid,'A, %8.0f , %8.0f , %8.0f , %8.0f \n',...
            one,two,three,four);
        %increment area counters
        areacounter=areacounter+1;
        stiffareacounter=stiffareacounter+1;
        one=one+1000;
    end
end
%end IF/ELSE
end %end FOR
endif
% Filename: dummycoords

% This file takes the coordinate arrays from zcoords.m and creates dummy
% arrays (for first, last and interior girders) to get rid of the
% duplicate coordinates for drawing keypoints and areas.

% Variables:
% temparray: Array used to hold the place of the coordinate arrays
% dummyfca: Coordinate array for first girder w/out duplicates
% dummylca: Coordinate array for last girder w/out duplicates
% dummyica: Coordinate array for interior girders w/out duplicates
% pointer: Pointer Variable used to shorten coordinate array
% numdummyfirst: Holds the number of non-duplicate coordinates
% numdummylast: Holds the number of non-duplicate coordinates
% numdummyinterior: Holds the number of non-duplicate coordinates
% numdummycoords: Number of coordinates in any girder's dummy array
% dummycoordarray: Array of the non-repetative coordinates only

%For the first girder
temparray=zeros(500,3); %array of pointers to coordinates to be used
jj=1; %initialize local temparray counter
%Loop to find pointers at non-duplicate coordinates
for ii=1:numfirstcoords
    if ii==numfirstcoords
        temparray(jj,1:3)=firstcoordarray(ii,1:3);
    elseif ii==1
        temparray(jj,1:3)=firstcoordarray(ii,1:3);
        jj=jj+1; %increment
    else
        if firstcoordarray(ii,1)==firstcoordarray(ii-1,1)
        else
            temparray(jj,1:3)=firstcoordarray(ii,1:3);
            jj=jj+1; %increment
        end %end IF
    end %end IF
end %end FOR
pointer=find(temparray==0); %find end of array
%Select pointer(2) because first zero is beginning of girder
pointer=pointer(2)-1;
dummyfca=zeros(pointer,3); %initialize the dummy coordinate array
numdummyfirst=pointer; %keep track of number of dummy coordinates
%Loop to fill in the dummy array
for ii=1:pointer
    dummyfca(ii,1:2)=temparray(ii,1:2);
    dummyfca(ii,3)=ii;
end %end FOR

%Warn user for close coordinates (< 100 mm apart)
% - it's the same for all girders
for ii=1:numdummyfirst-1
    if dummyfca(ii,1)-dummyfca(ii+1,1)<=99
fprintf(...
  'Warning, coordinates are too close at coordinate %8.2f \n'
, dummyfca(ii,1))
fprintf('They are %5.2f mm apart \n',...
  dummyfca(ii,1)-dummyfca(ii+1,1))
end %end IF
end %end FOR
%-------------------------------------------------------------------------
%For the last girder

jj=1;   %initialize local temparray counter
%Loop to find pointers at non-duplicate coordinates
for ii=1:numlastcoords
  if ii==numlastcoords
    temparray(jj,1:3)=lastcoordarray(ii,1:3);
  elseif ii==1
    temparray(jj,1:3)=lastcoordarray(ii,1:3);
    jj=jj+1;    %increment
  else
    if lastcoordarray(ii,1)==lastcoordarray(ii-1,1)
    else
      temparray(jj,1:3)=lastcoordarray(ii,1:3);
      jj=jj+1;    %increment
    end %end IF
  end %end IF
end %end FOR

pointer=find(temparray==0); %find end of array
%Select pointer(2) because first zero is beginning of girder
pointer=pointer(2)-1;

dummylca=zeros(pointer,3);      %initialize the dummy coordinate array
numdummylast=pointer;           %keep track of number of dummy coordinates

%Loop to fill in the dummy array
for ii=1:pointer
  dummylca(ii,1:2)=temparray(ii,1:2);
  dummylca(ii,3)=ii;
end %end FOR
%-------------------------------------------------------------------------
%For the interior girders

jj=1;   %initialize local temparray counter
%Loop to find pointers at non-duplicate coordinates
for ii=1:numinteriorcoords
  if ii==numinteriorcoords
    temparray(jj,1:3)=interiorcoordarray(ii,1:3);
  elseif ii==1
    temparray(jj,1:3)=interiorcoordarray(ii,1:3);
    jj=jj+1;    %increment
  else
    if interiorcoordarray(ii,1)==interiorcoordarray(ii-1,1)
    else
    end %end IF
  end %end IF
end %end FOR

%-------------------------------------------------------------------------
temparray(jj,1:3)=interiorcoordarray(ii,1:3);
jj=jj+1;    %increment
end %end IF
end %end IF
end %end FOR

pointer=find(temparray==0); %find end of array
%Select pointer(2) because first zero is beginning of girder
pointer=pointer(2)-1;

dummyica=zeros(pointer,3);     %initialize the dummy coordinate array
numdummyinterior=pointer;      %keep track of number of dummy coordinates

%Loop to fill in the dummy array
for ii=1:pointer
    dummyica(ii,1:2)=temparray(ii,1:2);
    dummyica(ii,3)=ii;
end
%-------------------------------------------------------------------------
%Need generic array of just the dummy coordinates
numdummycoords=pointer;       %the number is the same for all girders

%Initialize array
dummycoordarray=zeros(numdummycoords,1);
%Fill in coordinates (could use any dummy coordinate array - they match)
dummycoordarray(1)=dummyica(1);
%-------------------------------------------------------------------------
% This file creates many different arrays for help with the end bent diaphragms. It does this for both the top and bottom beam members, the truss members (diagonal and vertical). The vertical members will only be created if needed.

% Variables:
% %bottomgirderkp: Number of girder keypoints at bottom of girders
% %topgirderkp: Number of girder keypoints at the top of girders
% %bottomarray: Bottom of girder keypoint array
% %toparray: Top of girder keypoint array
% %numbottomchords: Number of required beam members at bottom
% %bottomchordarray: Array for bottom beam (chord) members
% %numtopchords: Number of required beam members at top
% %topchordarray: Array for top beam members
% %numebddiagonals: Number of diagonal members for ebd's
% %ebddiagonalarray: Diagonal members' keypoint array
% %numebdvertical: Number of vertical members for ebd's
% %ebdverticalarray: Vertical members' keypoint array
% %pointer: Variable used repeatedly to point along arrays
% %marker: Variable used repeatedly to point along arrays
% %match: Variable that holds a match b/t arrays

% Call the ebdkeypoints.m file
bdkeypoints

% Next, grab the keypoints needed for the end bent diaphragms

% Initialize variables and arrays
bottomgirderkp=girders*2;
topgirderkp=girders*2;
bottomarray=zeros(bottomgirderkp,1);
toparray=zeros(topgirderkp,1);
pointer=1;
for ii=1:numfirstcoords
    match=firstcoordarray(ii,2);
    if match==5 | match==6
        marker=find(dummyfca(:,1)==firstcoordarray(ii,1));
        bottomarray(pointer,1)=marker(1)*6-5;
        toparray(pointer,1)=marker(1)*6-2;
        pointer=pointer+1; %increment
    end %end IF
end %end FOR

% Loop to fill in the rest of the arrays
for ii=pointer:bottomgirderkp
    bottomarray(pointer,1)=bottomarray(pointer-2,1)+1000;
    toparray(pointer,1)=toparray(pointer-2,1)+1000;
    pointer=pointer+1; %increment
end %end for

% Top and bottom beam members

% Initialize arrays and variables:
numbottomchords=2*2*(girders-1);
bottomchordarray=zeros(numbottomchords,2);
numtopchords=numbottomchords;
topchordarray=zeros(numtopchords,2);

% Re-initialize variables
pointer=1;
marker=1;

% Two Loops to fill in the first column of the chord arrays
for ii=1:(girders-1)
    bottomchordarray(pointer,1)=bottomarray(marker,1);
    bottomchordarray(pointer+1,1)=bottomarray(marker+2,1);
    topchordarray(pointer,1)=toparray(marker,1);
    topchordarray(pointer+1,1)=toparray(marker+2,1);
    % increment variables
    pointer=pointer+4;
    marker=marker+2;
end % end FOR

% Re-initialize variables
pointer=3;
marker=2;

for ii=1:(girders-1)
    bottomchordarray(pointer,1)=bottomarray(marker,1);
    bottomchordarray(pointer+1,1)=bottomarray(marker+2,1);
    topchordarray(pointer,1)=toparray(marker,1);
    topchordarray(pointer+1,1)=toparray(marker+2,1);
    % increment variables
    pointer=pointer+4;
    marker=marker+2;
end % end FOR

% Re-initialize variables
pointer=1;
marker=1;

% Loop to fill in the second column of the chord arrays
for ii=1:(girders-1)
    bottomchordarray(pointer:pointer+1,2)=ebdkparray(marker,1);
    bottomchordarray(pointer+2:pointer+3,2)=ebdkparray(marker+2,1);
    topchordarray(pointer:pointer+1,2)=ebdkparray(marker+1,1);
    topchordarray(pointer+2:pointer+3,2)=ebdkparray(marker+3,1);
    % increment variables
    pointer=pointer+4;
    marker=marker+4;
end % end FOR

% Diagonal truss members

numebddiagonals=2*(girders-1)*2;
ebddiagonalarray=zeros(numebddiagonals,2);

312
% Re-initialize variables
pointer=1;
marker=1;

% Loop to fill in the first column of the diagonal member array
for ii=1:(girders-1)*2
    ebddiagonalarray(pointer,1)=bottomarray(marker,1);
    ebddiagonalarray(pointer+1,1)=bottomarray(marker+2,1);
    % increment variables
    pointer=pointer+2;
    marker=marker+1;
end % end FOR

% Re-initialize variables
pointer=1;
marker=2;

% Loop to fill in the second column of the diagonal member array
for ii=1:(girders-1)*2
    ebddiagonalarray(pointer:pointer+1,2)=ebdkparray(marker,1);
    % increment variables
    pointer=pointer+2;
    marker=marker+2;
end % end FOR

%-------------------------------------------------------------------------
% Vertical truss members
numebdvertical=(girders-1)*2;
ebdverticalarray=zeros(numebdvertical,2);

% Re-initialize variables
pointer=1;
marker=1;

% Loop to fill in the columns of the vertical member array
for ii=1:numebdvertical
    ebdverticalarray(pointer,1)=ebdkparray(marker,1);
    ebdverticalarray(pointer,2)=ebdkparray(marker+1,1);
    % increment variables
    pointer=pointer+1;
    marker=marker+2;
end % end FOR

%-------------------------------------------------------------------------
% Finally, call the correct file to draw in the ebd members, but only if
% integral bents are not present
if integralbents==0
    if ebdtype==1
        drawebdtype1
    elseif ebdtype==2
        drawebdtype2
    end % end IF
end % end IF

%-------------------------------------------------------------------------
%Filename: ebdkeypoints.m

%This file creates arrays to print commands to ANSYS for the keypoints needed for the end bent diagrapms. The file will then be called by two other files that will use the keypoints to draw the end bent diagrams in two different ways.

%_________________________________________________________________________
%Variables:
%   %currentkp:     Starting keypoint number for the end bent diagrapms
%   %counter:       Number of keypoints created in this file
%   %ebdkparray:    Array with new keypoints used for ANSYS command output
%   %pointer:       End bent diagrapm array pointer
%   %marker:        Determines when shifts are necessary
%   %temp:          Checks for necessary shifts
%   %shifts:        Number of shifts
%_________________________________________________________________________

%Initialize variables and arrays
currentkp=40001;
counter=0;
ebdkparray=zeros(2*(girders-1)*2,4);
pointer=1;

%Fill in the end bent diagrapm array

%Loop through first and interior girders
for jj=1:1+numintgirders
    %Loop through the first girder's z-coordinates
    for ii=1:numfirstcoords
        match=firstcoordarray(ii,2);
        %find bearing by tag ID (5 or 6)
        if match==5 | match==6
            %fill in the array
            ebdkparray(pointer,1)=currentkp;
            ebdkparray(pointer+1,1)=currentkp+1;
            ebdkparray(pointer:pointer+1,2)=0.5*xshift;
            ebdkparray(pointer,3)=0;
            ebdkparray(pointer+1,3)=webheight;
            ebdkparray(pointer:pointer+1,4)=firstcoordarray(ii,1)...
                +0.5*zshift;
            %increment variables
            pointer=pointer+2;
            counter=counter+2;
            currentkp=currentkp+2;
        end %end IF
    end %end FOR
end %end FOR

%Print out the keypoints

%Initialize
marker=counter/jj;
temp=1;
shifts=0;
%Loop to print commands to file for ANSYS
for ii=1:counter
    %Print keypoints
    fprintf(fid,'k, %8.0f , %8.2f , %8.2f , %8.2f \n', ebdkparray(ii,:));
    %Print shift statement if needed
    if ii==marker*temp
        fprintf(fid,'wpoffs, %8.2f , %8.2f , %8.2f \n',...
            xshift,yshift,zshift);
        temp=temp+1;
        shifts=shifts+1;
    end %end IF
end %end FOR

%Loop to move axis back to original location
while shifts>0
    fprintf(fid,'wpoffs, %8.2f , %8.2f , %8.2f \n',...
        -1*xshift,-1*yshift,-1*zshift);
    shifts=shifts-1;  %decrement
end %end WHILE

%-------------------------------------------------------------------------
% This file reads in all the element types, real constant sets and material properties for the bridge, creates arrays, and prints them to the output file.

% Variables:
% rctrussarray: Array to hold the truss variables of Real Constants
% rcbeamarray: Array to hold the beam variables of Real Constants
% rcplatearray: Array to hold the thickness type of Real Constants
% numplates: Number of RC sets needed for all the thicknesses
% section: Variable keeping up with the section along the girder
% pointer: Pointer incrementing down the RC plate array
% bearing: Variable keeping up with the bearing along the girder
% rcpointer: Variable for the RC set numbers of the bearing plates

% Print out material property data:
% First for Steel
fprintf(fid,'MPTEMP,,,,,,,, \n'); % two lines needed in ANSYS
fprintf(fid,'MPTEMP,1,0 \n');
fprintf(fid,'MPDATA,EX,1,, %6.2f \n',steelmod); % elastic modulus
fprintf(fid,'MPDATA,PRXY,1,, %2.2f \n',steelpois); % Poisson's Ratio
% For Concrete
fprintf(fid,'MPTEMP,,,,,,,, \n'); % two lines needed in ANSYS
fprintf(fid,'MPTEMP,1,0 \n');
fprintf(fid,'MPDATA,EX,2,, %6.2f \n',concmod); % elastic modulus
fprintf(fid,'MPDATA,PRXY,2,, %2.2f \n',concpois); % Poisson's Ratio

% Print out the element types needed for bridge:
fprintf(fid,'ET,1,SHELL93 \n'); % element type for steel shells
fprintf(fid,'ET,2,SHELL63 \n'); % element type for concrete deck
fprintf(fid,'ET,3,BEAM4 \n'); % element type for beams of diaphragms
fprintf(fid,'ET,4,LINK8 \n'); % element type for trusses
fprintf(fid,'ET,5,MPC184 \n'); % element type for the rigid links
fprintf(fid,'KEYOPT,5,1,1 \n'); % This option is for the rigid links

% Gather the information about the real constant sets for bridge:
% Only areas are required for the truss elements (intermediate cross frames, end bent diaphragms, middle bent diaphragms and SIP Forms).
rcctrussarray=zeros(5,2); % initialize array
rcctrussarray(1,1)=1;
rcctrussarray(2,1)=2;
rcctrussarray(3,1)=32;
rcctrussarray(4,1)=51;
rcctrussarray(5,1)=52;
rcctrussarray(1,2)=icftrussarea;
rcctrussarray(2,2)=ebdtrussarea;
rctrussarray(3,2)=mbdtrussarea;
rctrussarray(4,2)=diagonalarea;
rctrussarray(5,2)=rodarea;

rctimearray=zeros(4,4); %initialize array
rcbeamarray(1,1)=3;
rbeamarray(2,1)=4;
rbeamarray(3,1)=9;
rbeamarray(4,1)=33;
rbeamarray(1,2)=topbeamarea;
rbeamarray(2,2)=bottombeamarea;
rbeamarray(3,2)=icfbeamarea;
rbeamarray(4,2)=mbdbeamarea;
rbeamarray(1,3)=topbeamiyy;
rbeamarray(2,3)=bottombeamiyy;
rbeamarray(3,3)=icfbeamiyy;
rbeamarray(4,3)=mbdbeamiyy;
rbeamarray(1,4)=topbeaimizz;
rbeamarray(2,4)=bottombeaimizz;
rbeamarray(3,4)=icfbeaimizz;
rbeamarray(4,4)=mbdbeaimizz;

numplates=4+numsections*2+numbearing;

rcplatearray=zeros(numplates,5); %initialize array

for ii=1:4
    if odd or even - odd for top flange and even for bottom
    end %end IF
    end %end FOR

%The following is to fill in the RC sets for the girder sections
section=1; %initialize

for ii=5:4+numsections*2
    if odd or even - odd for top flange and even for bottom
    end %end IF
end %end FOR
pointer=ii+1;  %increment

bearing=1;      %initialize
rcpointer=28;   %initialize known starting set number

for ii=pointer:pointer+numbearing-1
    thisthickness=beararray(bearing,2);
    %fill in array appropriately
    rcplatearray(ii,1)=rcpointer;
    rcplatearray(ii,2:5)=thisthickness;
    %increment counters
    bearing=bearing+1;
    rcpointer=rcpointer+1;
end %end FOR

%-------------------------------------------------------------------------
%Print out the real constant sets:
%Loop to print truss set
for ii=1:5
    fprintf(fid,'R, %2.0f , %8.2f 
', rctrussarray(ii,:));
end %end FOR
%Loop to print beam set
for ii=1:4
    fprintf(fid,'R, %2.0f , %8.2f , %8.2f , %8.2f ,,,90 
', rcbeamarray(ii,:));
end %end FOR
%Loop to print plate set
for ii=1:numplates
    fprintf(fid,'R, %2.0f , %8.2f , %8.2f , %8.2f , %8.2f \n', ...
            rcplatearray(ii,:));
end %end FOR
%-------------------------------------------------------------------------
%Filename: girderareas.m

%This file creates arrays for the areas of the girders in a format
% friendly to commands in ANSYS by using the keypoint arrays created in
% girderkeypoints.m. These created arrays are then printed out in the
% correct ANSYS format.

%_________________________________________________________________________
%Variables:
%
%numfirstareas:         Number of areas for first girder
%numlastareas:          Number of areas for last girder
%numinteriorareas:      Number of areas for all of the interior girders
%oneinteriorarea:       Number of areas for only one interior girder
%firstareaarray:        Array to hold keypoints for first girder
%lastareaarray:         Array to hold keypoints for last girder
%interiorareaarray:     Array to hold keypoints for interior girder
%marker:                First keypoint in array
%keeper:                Temporary variable for interior girders
%pointer:               Array pointer for interior areas
%areacounter:           Global variable to keep up w/ the number of areas
%girderareacounter:     Number of areas for girders only
%
%_________________________________________________________________________

%Find the number of areas needed for girders
numfirstareas=(numdummyfirst-1)*5;
umlastareas=(numdummylast-1)*5;
oneinteriorarea=(numdummyinterior-1)*5;
uminteriorareas=(numdummyinterior*numintgirders-numintgirders)*5;

%Intialize Arrays
firstareaarray=zeros(numfirstareas,4);
lastareaarray=zeros(numlastareas,4);
interiorareaarray=zeros(numinteriorareas,4);

%-------------------------------
%First Girder

%Initialize marker
marker=firstkparray(1,1);
%The first five areas are manipulated manually
firstareaarray(1,1)=marker+1;
firstareaarray(1,2)=marker;
firstareaarray(1,3)=marker+6;
firstareaarray(1,4)=marker+7;
firstareaarray(2,1)=marker;
firstareaarray(2,2)=marker+2;
firstareaarray(2,3)=marker+8;
firstareaarray(2,4)=marker+6;
firstareaarray(3,1)=marker;
firstareaarray(3,2)=marker+6;
firstareaarray(3,3)=marker+9;
firstareaarray(3,4)=marker+3;
firstareaarray(4,1)=marker+4;
firstareaarray(4,2)=marker+3;
firstareaarray(4,3)=marker+9;
firstareaarray(4,4)=marker+10;
%Loop to fill in the rest of the Area Array
for ii=6:numfirstareas
    firstareaarray(ii,:)=firstareaarray(ii-5,:)+6;
end %end FOR

%Loop to print commands for areas to file
for ii=1:numfirstareas
    fprintf(fid,'A, %8.0f , %8.0f , %8.0f , %8.0f \n',...
        firstareaarray(ii,:));
end %end FOR

%-------------------------------------------------------------------------
%Interior Girders
%Initialize marker to first keypoint number in interior keypoint array
marker=interiorkparray(1,1);
%Initialize other variables
keeper=0;
pointer=1;
%Loop for all interior girders
for ii=1:numintgirders
    %The first five areas are manipulated manually
    interiorareaarray(1+keeper*oneinteriorarea,1)=marker+1;
    interiorareaarray(1+keeper*oneinteriorarea,2)=marker;
    interiorareaarray(1+keeper*oneinteriorarea,3)=marker+6;
    interiorareaarray(1+keeper*oneinteriorarea,4)=marker+7;
    interiorareaarray(2+keeper*oneinteriorarea,1)=marker;
    interiorareaarray(2+keeper*oneinteriorarea,2)=marker+2;
    interiorareaarray(2+keeper*oneinteriorarea,3)=marker+8;
    interiorareaarray(2+keeper*oneinteriorarea,4)=marker+6;
    interiorareaarray(3+keeper*oneinteriorarea,1)=marker;
    interiorareaarray(3+keeper*oneinteriorarea,2)=marker+6;
    interiorareaarray(3+keeper*oneinteriorarea,3)=marker+9;
    interiorareaarray(3+keeper*oneinteriorarea,4)=marker+3;
    interiorareaarray(4+keeper*oneinteriorarea,1)=marker+4;
    interiorareaarray(4+keeper*oneinteriorarea,2)=marker+3;
    interiorareaarray(4+keeper*oneinteriorarea,3)=marker+9;
    interiorareaarray(4+keeper*oneinteriorarea,4)=marker+10;
    interiorareaarray(5+keeper*oneinteriorarea,1)=marker+3;
    interiorareaarray(5+keeper*oneinteriorarea,2)=marker+5;
    interiorareaarray(5+keeper*oneinteriorarea,3)=marker+11;
    interiorareaarray(5+keeper*oneinteriorarea,4)=marker+9;
    pointer=pointer+5;

    %Loop to fill in remaining array
    for jj=pointer:oneinteriorarea+keeper*oneinteriorarea
        interiorareaarray(jj,:)=interiorareaarray(jj-5,:)+6;
        pointer=pointer+1;
    end %end FOR

    %Increment variables
    marker=marker+1000;
    keeper=keeper+1;
%Loop to print commands for areas to file
for ii=1:numinteriorareas
    fprintf(fid,'A, %8.0f, %8.0f, %8.0f, %8.0f \n',...
        interiorareaarray(ii,:));
end %end FOR
%-------------------------------------------------------------
%Last Girder

%initialize marker:
marker=lastkparray(1,1);
%The first five areas are manipulated manually
lastareaarray(1,1)=marker+1;
lastareaarray(1,2)=marker;
lastareaarray(1,3)=marker+6;
lastareaarray(1,4)=marker+7;
lastareaarray(2,1)=marker;
lastareaarray(2,2)=marker+2;
lastareaarray(2,3)=marker+8;
lastareaarray(2,4)=marker+6;
lastareaarray(3,1)=marker;
lastareaarray(3,2)=marker+6;
lastareaarray(3,3)=marker+9;
lastareaarray(3,4)=marker+3;
lastareaarray(4,1)=marker+4;
lastareaarray(4,2)=marker+3;
lastareaarray(4,3)=marker+9;
lastareaarray(4,4)=marker+10;
lastareaarray(5,1)=marker+3;
lastareaarray(5,2)=marker+5;
lastareaarray(5,3)=marker+11;
lastareaarray(5,4)=marker+9;
%Loop to fill in the rest of the Area Array
for ii=6:numlastareas
    lastareaarray(ii,:)=lastareaarray(ii-5,:)+6;
end %end FOR
%Loop to print commands for areas to file
for ii=1:numlastareas
    fprintf(fid,'A, %8.0f, %8.0f, %8.0f, %8.0f \n',...
        lastareaarray(ii,:));
end %end FOR
%-----------------------------------------------
%Initialize area counter variable
areacounter=numfirstareas+numlastareas+numinteriorareas;
%Initialize girder area counter variable
girderareatcounter=areacounter;
%--------------------------------------------------
%Filename: girderkeypoints.m

%This file takes the coordinates from the dummy coordinate arrays and
% creates array for the first, last and interior girders that hold
% keypoint data - the keypoint number along with the corresponding x, y
% and z coordinates.
%The arrays created will be used to print out the keypoints for ANSYS.

%Variables:
%
%firstkparray:          Array for keypoints for first girder
%firstnumkp:            Number of keypoints for first girder
%interiorkparray:       Array for keypoints for all interior girders
%interiornumkp:         Number of keypoints for an interior girder
%lastkparray:           Array for keypoints for last girder
%lastnumkp:             Number of keypoints for last girder
%kpcounter:             Total number of keypoints
%section:               Current section along girder
%topwidth:              Section's top flange width
%bottomwidth:           Section's bottom flange width
:pointer:               Temporary variable
%temparray:             Temporary array variable
%tempholder:            Temporary counter variable
%
%Initialize keypoint arrays & multiply by six because there are six
% keypoints at each coordinate location.
firstkparray=zeros(1000,4);
firstnumkp=6*numdummyfirst;
interiorkparray=zeros(8000,4);
interiornumkp=6*numdummyinterior;
lastkparray=zeros(1000,4);
lastnumkp=6*numdummylast;

%Initialize the global keypoint counter
kpcounter=firstnumkp+interiornumkp*numintgirders+lastnumkp;

%Fill in KP arrays for all girders
for zz=1:girders
  %For the first girder (starting at kp 1)
  if zz==1
    section=1;       %start with first girder section
    for ii=1:numdummyfirst
      %flange widths
      topwidth=sectionarray(section,2);
      bottomwidth=sectionarray(section,4);
      %x-coordinates
      firstkparray(ii*6-5,2)=0;
      firstkparray(ii*6-4,2)=-bottomwidth*0.5;
      firstkparray(ii*6-3,2)=bottomwidth*0.5;
      firstkparray(ii*6-2,2)=0;
      firstkparray(ii*6-1,2)=-topwidth*0.5;
      firstkparray(ii*6,2)=topwidth*0.5;
      %y-coordinates
  
322
firstkparray(ii*6-5:ii*6-3,3)=0;
firstkparray(ii*6-2:ii*6,3)=webheight;
%z-coordinates
firstkparray(ii*6-5:ii*6,4)=dummyfca(ii,1);
%keypoint numbers
for jj=ii*6-5:ii*6
    firstkparray(jj,1)=jj;
end %end FOR
%Change section if correct tag ID is identified
if dummyfca(ii,2)==10|dummyfca(ii,2)==13|dummyfca(ii,2)==21
    section=section+1;
end %end IF
end %end FOR
%For the last girder (starting at next 1000 kp)
elseif zz==girders
    section=1;     %start with first girder section
for ii=1:numdummylast
    %flange widths
    topwidth=sectionarray(section,2);
    bottomwidth=sectionarray(section,4);
    %x-coordinates
    lastkparray(ii*6-5,2)=0;
    lastkparray(ii*6-4,2)=-bottomwidth*0.5;
    lastkparray(ii*6-3,2)=bottomwidth*0.5;
    lastkparray(ii*6-2,2)=0;
    lastkparray(ii*6-1,2)=-topwidth*0.5;
    lastkparray(ii*6,2)=topwidth*0.5;
    %y-coordinates
    lastkparray(ii*6-5:ii*6-3,3)=0;
    lastkparray(ii*6-2:ii*6,3)=webheight;
    %z-coordinates
    lastkparray(ii*6-5:ii*6,4)=dummylca(ii,1);
    %keypoint numbers
    for jj=ii*6-5:ii*6
        lastkparray(jj,1)=jj+((girders-1)*1000);
    end %end FOR
    %Change section if correct tag ID is identified
    if dummylca(ii,2)==10|dummylca(ii,2)==13|dummylca(ii,2)==21
        section=section+1;
    end %end IF
end %end FOR
%For the first interior girder (starting at kp 1001)
elseif zz==2
    section=1;     %start with first girder section
for ii=1:numdummyinterior
    %flange widths
    topwidth=sectionarray(section,2);
    bottomwidth=sectionarray(section,4);
    %x-coordinates
    interiorkparray(ii*6-5,2)=0;
    interiorkparray(ii*6-4,2)=-bottomwidth*0.5;
    interiorkparray(ii*6-3,2)=bottomwidth*0.5;
    interiorkparray(ii*6-2,2)=0;
    interiorkparray(ii*6-1,2)=-topwidth*0.5;
    interiorkparray(ii*6,2)=topwidth*0.5;
    %y-coordinates
    interiorkparray(ii*6-5:ii*6-3,3)=0;
interiorkparray(ii*6-2:ii*6,3)=webheight;
%z-coordinates
interiorkparray(ii*6-5:ii*6,4)=dummyica(ii,1);
%keypoint numbers
for jj=ii*6-5:ii*6
    interiorkparray(jj,1)=jj+1000;
end
%Change section if correct tag ID is identified
if dummyica(ii,2)==10|dummyica(ii,2)==13|dummyica(ii,2)==21
    section=section+1;
end %end IF
end %end FOR loop
end %end IF/ELSEIF statement
end %end big FOR loop
%-------------------------------------------------------------------------
%For the interior girders, I have to account for all of them
pointer=find(interiorkparray==0);
pointer=pointer(1);
%Initialize a temporary array and fill in
temparray=zeros(interiornumkp,4);
temparray(1:pointer-1,:)=interiorkparray(1:pointer-1,:);
%Loop to fill in the interior kp array for all interior girders
%Start at 2 since the first interior girder's keypoints are already there
for ii=2:numintgirders
    tempholder=1;   %initialize counter
    %Loop to fill in the next girder's keypoints
    for jj=pointer:pointer+interiornumkp-1
        interiorkparray(jj,1)=temparray(tempholder,1)+1000;
        interiorkparray(jj,2:4)=temparray(tempholder,2:4);
        tempholder=tempholder+1;    %increment
    end
    pointer=find(interiorkparray==0);
    pointer=pointer(1); %pointer moves down array to new zero space
    %the new temparray now has larger kp numbers (others don't change):
temparray(:,1)=temparray(:,1)+1000;
end %end big FOR loop
%-------------------------------------------------------------------------
%Correctly size the 3 girder kp arrays for output
%First Girder:
pointer=find(firstkparray==0);
temparray=firstkparray;
firstkparray=zeros(pointer(1)-1,4);
%Loop to fill in new first kp array
for ii=1:pointer(1)-1
    firstkparray(ii,:)=temparray(ii,:);
end %end FOR
%Interior Girders:
pointer=find(interiorkparray==0);
temparray=interiorkparray;
teriorkparray=zeros(pointer(1)-1,4);
%Loop to fill in new interior kp array
for ii=1:pointer(1)-1
    interiorkparray(ii,:)=temparray(ii,:);
end %end FOR

%Last Girder:
pointer=find(lastkparray==0);
temparray=lastkparray;
lastkparray=zeros(pointer(1)-1,4);
%Loop to fill in new first kp array
for ii=1:pointer(1)-1
    lastkparray(ii,:)=temparray(ii,:);
end %end FOR
%
%Finally, print out the keypoints to file by calling printgirderkp.m
printgirderkp
%---------------------------------------------------------------
% This file creates many different arrays for help with the intermediate cross frames. It does this for the bottom beam members (if needed for the "K" type) and the diagonal truss members (the horizontal members are trusses in the "X" type icf).

% Variables:

% bottomicfkparray: Array for keypoints along top of girders
% topicfkparray: Array for keypoints along bottom of girders
% pointer1: First column array pointer
% pointer2: Second column array pointer
% match1: Variable representing an item in an array
% match2: Variable representing an item in an array
% holder: Variable that holds a match between arrays
% numicfbottomchords: Number of beams needed for type "K" icf's
% icfchordarray: Array to hold correct keypoint numbers for beams
% numicfdiagonals: Number of diagonal trusses for icf's
% typekdiagonalarray: Array to hold correct keypoint numbers for diagonals in type "K" icf
% typexdiagonalarray: Array to hold correct keypoint numbers for diagonals in type "X" icf
% pointer: Variable used repeatedly to point along arrays
% marker: Variable used repeatedly to point along arrays

% First, call the icfkeypoints.m file
icfkeypoints
%------------------------------------------------------------------------------

% Next, grab the keypoints needed for the intermediate cross frames

% Initialize variables and arrays
bottomicfkparray=zeros(numcrossframes,2);
topicfkparray=zeros(numcrossframes,2);
pointer1=1;
pointer2=1;

% Loop to fill in part of the keypoints needed in the girders
for ii=1:numfirstcoords
    % initialize match variables
    match1=firstcoordarray(ii,2);
    match2=firstcoordarray(ii,2);
    if match1==1
        holder=find(dummyfca(:,1)==firstcoordarray(ii,1));
        bottomicfkparray(pointer1,1)=holder(1)*6-5;
        topicfkparray(pointer1,1)=holder(1)*6-2;
        pointer1=pointer1+1;  %increment
    end  %end IF
    if match2==2
        holder=find(dummyfca(:,1)==firstcoordarray(ii,1));
        bottomicfkparray(pointer2,2)=holder(1)*6-5+1000;
        topicfkparray(pointer2,2)=holder(1)*6-2+1000;
        pointer2=pointer2+1;  %increment
    end  %end IF
end  %end FOR
%Second Loop to fill in the rest of the arrays
for ii=pointer1:numcrossframes
    bottomicfkparray(ii,1:2)=bottomicfkparray(ii-icfpergirder,1:2)+1000;
    topicfkparray(ii,1:2)=topicfkparray(ii-icfpergirder,1:2)+1000;
end %end FOR

%Create arrays for different types (possibilities) of cross frames

%First, as if the bottom chord is a beam member w/ needed keypoint:

%Initialize
numicfbottomchords=2*numcrossframes;
icfchordarray=zeros(numicfbottomchords,2);
pointer=1;
marker=1;

%Loop to fill in the columns of the icf chord array
for ii=1:numcrossframes
    icfchordarray(pointer,1)=bottomicfkparray(marker,1);
    icfchordarray(pointer+1,1)=bottomicfkparray(marker,2);
    icfchordarray(pointer:pointer+1,2)=icfkparray(marker,1);
    %increment
    pointer=pointer+2;
    marker=marker+1;
end %end FOR

%Second, as if the bottom chord is one long truss (no kp needed):
%To do this, the bottomicfkparray is already set up to draw what would be % needed.

%Third, as if the diagonals need the middle keypoint ("K" type icf):

%Initialize
numicfdiagonals=2*numcrossframes;
typekdiagonalarray=zeros(numicfdiagonals,2);
pointer=1;
marker=1;

%Loop to fill in the columns of the type "K" truss array
for ii=1:numcrossframes
    typekdiagonalarray(pointer,1)=topicfkparray(marker,1);
    typekdiagonalarray(pointer+1,1)=topicfkparray(marker,2);
    typekdiagonalarray(pointer:pointer+1,2)=icfkparray(marker,1);
    %increment
    pointer=pointer+2;
    marker=marker+1;
end %end FOR

%Fourth, as if the diagonals don't need the middle keypoint ("X" type icf)

%Initialize
typexdiagonalarray=zeros(numicfdiagonals,2);
pointer=1;
marker=1;

%Loop to fill in the columns of the type "X" truss array
for ii=1:numcrossframes
    typexdiagonalarray(pointer,1)=topicfkparray(marker,1);
    typexdiagonalarray(pointer+1,1)=bottomicfkparray(marker,1);
    typexdiagonalarray(pointer,2)=bottomicfkparray(marker,2);
    typexdiagonalarray(pointer+1,2)=topicfkparray(marker,2);
    %increment
    pointer=pointer+2;
    marker=marker+1;
end %end FOR

%Fifth, as if the top chord is one long truss (no kp needed):
%To do this, the topicfkparray is already set up to draw what would be
% needed.

%Finally, call the correct file to "draw" in the icf members
if icftype==1
    drawicftype1
elseif icftype==2
    drawicftype2
elseif icftype==3
    drawicftype3
end %end IF

%---------------------------------------------------------------
328
%Filename: icfkeypoints.m

%This file creates the extra keypoints needed for the "K" type intermediate cross frames. The file then prints commands to file needed to "draw" the actual keypoints created.

%Variables:
%currentkp: Starting keypoint number of intermediate cross frames
%counter: Number of keypoints created in this file
%icfkparray: Array with new keypoints used for ANSYS command output
%pointer: Intermediate cross frame array pointer
%temparray: Temporary array to hold the values of the icfkparray
%marker: Determines when shifts are necessary
%temp: Checks for necessary shifts
%shifts: Number of shifts
%numcrossframes: Number of actual cross frames on entire bridge
%icfpergirder: Number of cross frames between each "bay"

%Initialize variables and arrays
currentkp=30001; %these keypoints are set to start at # 30,001
counter=0;
icfkparray=zeros(100,4);
pointer=1;
%-------------------------------------------------------------------------
%Need to fill in the intermediate cross frame array

%Loop through first and interior girders
for jj=1:1+numintgirders
    %Loop through the first girder's z-coordinates
    for ii=1:numfirstcoords
        %find cross frame by tag ID (1)
        if firstcoordarray(ii,2)==1
            %fill in the array
            icfkparray(pointer,1)=currentkp;
icfkparray(pointer,2)=0.5*xshift;
icfkparray(pointer,3)=0;
icfkparray(pointer,4)=firstcoordarray(ii,1);
            %increment variables
            pointer=pointer+1;
counter=counter+1;
currentkp=currentkp+1;
        end %end IF
    end %end FOR
end %end FOR

temparray=icfkparray; %hold temporary array
icfkparray=zeros(counter,4); %re-initialize array
icfkparray(:, :)=temparray(1:counter,:); %fill back in
%-------------------------------------------------------------------------

%Print out the keypoints

329
%Initialize
marker=counter/jj;
temp=1;
shifts=0;

%Loop to print statements for ANSYS
for ii=1:counter
  %Print keypoints
  fprintf(fid,'k, %8.0f , %8.2f , %8.2f , %8.2f 
', icfkparray(ii,:));
  %Print shift statement if needed
  if ii==marker*temp
    fprintf(fid,'wpoffs, %8.2f , %8.2f , %8.2f 
', xshift,yshift,zshift);
    %increment variables
    temp=temp+1;
    shifts=shifts+1;
  end %end IF
end %end FOR

%Loop to move axis back to original location
while shifts>0
  fprintf(fid,'wpoffs, %8.2f , %8.2f , %8.2f 
', -1*xshift,-1*yshift,-1*zshift);
  shifts=shifts-1; %decrement
end %end WHILE

%Initialize some variables for later use
numcrossframes=counter;
icfpergirder=marker;
%-------------------------------------------------------------
% This file creates many different arrays for help with the middle bent diaphragms. It does this for the bottom beam members (if needed for the "K" type), the top beam members for the mbd's in Bridge 1, and the diagonal truss members (the horizontal members are trusses in the "X" type mbd.

% Variables:
%
% bottommbdkp:           Number of keypoints along bottom of girders
% topmbdkp:              Number of keypoints along top of girders
% bottommbdkparray:      Array holding keypoints along top of girders
% topmbdkparray:         Array holding keypoints along bottom of girders
% match:                 Variable representing an item in an array
% holder:                Variable that holds a match between arrays
% nummbdbottomchords:    Number of beams needed for "K" type mbd's
% mbdchordarray:         Array to hold correct keypoint numbers for beams
% nummbddiagonals:       Number of diagonal trusses for mbd's
% typekdiagmbdarray:     Array to hold correct keypoint numbers for diagonals in type "K" mbd
% typexdiagmbdarray:     Array to hold correct keypoint numbers for diagonals in type "X" mbd
% nummbdtopchords:       Number of top beams for type "K" mbd's (Bridge 1)
% pointer:               Common array pointer
% marker:                Common array pointer
% counter:               Loop counter

% Call the mbdkeypoints.m file
mbdkeypoints

% Grab keypoints needed for the end bent diaphragms

% Initialize variables and arrays
bottommbdkp=(girders-1)*nummiddles;
topmbdkp=(girders-1)*nummiddles;
bottommbdkparray=zeros(bottommbdkp,2);
topmbdkparray=zeros(topmbdkp,2);
pointer=1;
counter=0;

for ii=1:numfirstcoords
    match=firstcoordarray(ii,2);
    if match==7 | match==8
        marker=find(dummyfca(:,1)==firstcoordarray(ii,1));
        bottommbdkparray(pointer,1)=marker(1)*6-5;
        topmbdkparray(pointer,1)=marker(1)*6-2;
        % increment variables
        pointer=pointer+1;
        counter=counter+1;
    end % end IF
end % end FOR

% Loop to fill in the rest of the first column
for ii=pointer:bottommbdkp
    bottommbdkparray(pointer,1)=bottommbdkparray(pointer-counter,1)+1000;
    topmbdkparray(pointer,1)=topmbdkparray(pointer-counter,1)+1000;
    pointer=pointer+1; %increment
end %end FOR

%Loop to fill in the second column of the arrays
for ii=1:bottommbdkp
    bottommbdkparray(ii,2)=bottommbdkparray(ii,1)+1000;
    topmbdkparray(ii,2)=topmbdkparray(ii,1)+1000;
end %end FOR
%-------------------------------------------------------------------------
%Create arrays for different possibilities of middle bent diaphragms
%First, bottom chord is a beam member
nummbdchords=2*nummiddles*(girders-1);
mbdchordarray=zeros(nummbdchords,2);
pointer=1;
marker=1;

%Loop to fill in the columns of the mbd chord array
for ii=1:(girders-1)*nummiddles
    mbdchordarray(pointer,1)=bottommbdkparray(marker,1);
    mbdchordarray(pointer+1,1)=bottommbdkparray(marker,2);
    mbdchordarray(pointer:pointer+1,2)=mbdkparray(marker,1);
    %increment variables
    pointer=pointer+2;
    marker=marker+1;
end %close FOR
%-------------------------------------------------------------------------
%Second, bottom chord is a truss member:
% To do this, the bottommbdkparray is already set up to "draw" what
% would be needed.
%-------------------------------------------------------------------------
%Third, if the diagonals need the middle keypoint ("K" type mbd)
% or
%Fourth, same but with a horizontal top chord member in "K" type.
nummbddiagonals=2*(girders-1)*nummiddles;
typekdiagmbdarray=zeros(nummbddiagonals,2);

%Re-initialize variables
pointer=1;
marker=1;

%Loop to fill in the columns of the type "K" truss array
for ii=1:(girders-1)*nummiddles
    typekdiagmbdarray(pointer,1)=topmbdkparray(marker,1);
    typekdiagmbdarray(pointer+1,1)=topmbdkparray(marker,2);
    typekdiagmbdarray(pointer:pointer+1,2)=mbdkparray(marker,1);
    %increment variables
    pointer=pointer+2;
marker=marker+1;
end %end FOR

%Fifth, if the diagonals don't need the middle keypoint ("X" type mbd)
typexdiagmbdarray=zeros(nummbddiagonals,2);

%Re-initialize variables
pointer=1;
marker=1;

%Loop to fill in the columns of the type "X" truss array
for ii=1:(girders-1)*nummiddles
    typexdiagmbdarray(pointer,1)=topmbdkparray(marker,1);
    typexdiagmbdarray(pointer+1,1)=bottommbdkparray(marker,1);
    typexdiagmbdarray(pointer,2)=bottommbdkparray(marker,2);
    typexdiagmbdarray(pointer+1,2)=topmbdkparray(marker,2);
    %increment variables
    pointer=pointer+2;
    marker=marker+1;
end %end FOR

%Sixth, the horizontal top chord
nummbdtopchords=(girders-1)*nummiddles;

%Finally, call the correct file to draw in the mbd members
if mbdtype==1
drawmbdtype1
elseif mbdtype==2
drawmbdtype2
elseif mbdtype==3
drawmbdtype3
end

%Filename: mbdkeypoints.m

%This file creates arrays to print commands to ANSYS for the keypoints
% needed for the middle bent diaphragms. The file will then be called by
% two other files that will use the keypoints to draw the middle bent
% diaphragms in two different ways.

% _______________________________________________________________
% Variables:
% %
% %currentkp: Starting keypoint number for the middle bent diaphragms
% %counter: Number of keypoints created in this file
% %mbdkparray: Array with new keypoints used for ANSYS command output
% %pointer: End bent diaphragm array pointer
% %marker: Determines when shifts are necessary
% %temp: Checks for necessary shifts
% %shifts: Number of shifts
% _______________________________________________________________

%Initialize variables and arrays
currentkp=50001;
counter=0;

%Determine the number of middle bents
if spantype==2
    nummiddles=1;
elseif spantype==3
    nummiddles=2;
end

mbdkparray=zeros((girders-1)*nummiddles,4);
pointer=1;

%-------------------------------------------------------------------------
%Need to fill in the middle bent diaphragm array
%-------------------------------------------------------------------------

%Loop through first and interior girders
for jj=1:1+numintgirders
    %Loop through the first girder's z-coordinates
    for ii=1:numfirstcoords
        match=firstcoordarray(ii,2);
        %find middle bent diaphragm by tag ID (7 or 8)
        if match==7 | match==8
            %fill in array
            mbdkparray(pointer,1)=currentkp;
            mbdkparray(pointer,2)=0.5*xshift;
            mbdkparray(pointer,3)=0;
            mbdkparray(pointer,4)=firstcoordarray(ii,1)+0.5*zshift;
            %increment variables
            pointer=pointer+1;
counter=counter+1;
currentkp=currentkp+1;
        end %end IF
    end %end FOR
end %end FOR

%-------------------------------------------------------------------------

334
%Print out the keypoints
%Initialize
marker=counter/jj;
temp=1;
shifts=0;

%Loop to print commands to file for ANSYS
for ii=1:counter
    %Print keypoints
    fprintf(fid,'k, %8.0f , %8.2f , %8.2f , %8.2f \n', mbdkparray(ii,:));
    %Print shift statement if needed
    if ii==marker*temp
        fprintf(fid,'wpoffs, %8.2f , %8.2f , %8.2f \n',' xshift,yshift,zshift);
        %increment variables
        temp=temp+1;
        shifts=shifts+1;
    end %end IF
end %end FOR

%Loop to move axis back to original location
while shifts>0
    fprintf(fid,'wpoffs, %8.2f , %8.2f , %8.2f \n',' -1*xshift,-1*yshift,-1*zshift);
    shifts=shifts-1; %decrement
end %end WHILE
%-------------------------------------------------------------------------
%Filename: panelements.m

%This file creates the arrays needed for the elements of the SIP forms.

%Variables:

%halfdiags:         Half the number of diagonal elements b/t each girder
%numhorizelems:     Number of horizontal elements b/t each girder
%numdiagelems:      Number of diagonal elements b/t each girder
%totalhorizelems:   Total number of horizontal elements created for pans
%totaldiagelems:    Total number of diagonal elements created for pans
%horizemarray:     Array holding information for the horizontal elements
%diagelemarray:     Array holding information for the diagonal elements
%node:              Temporary variable
%holder:            Temporary variable
%pointer:           Temporary variable

%******************************************************************************
%******************************************************************************

%*******************  ELEMENTS  ***********************
%Determine number of elements
halfdiags=keeper;
numhorizelems=onesetnodes;
umdiagelems=keeper*2;
totalhorizelems=numhorizelems*(girders-1);
totaldiagelems=numdiagelems*(girders-1);

%Initialize pan element arrays
horizemarray=zeros(totalhorizem,2);
diagelemarray=zeros(totaldiagelem,2);

%Initialize
node=100001;          %always starts at node = 100,001
holder=node+onesetnodes;
pointer=1;

%Loops to fill in the pan horizontal element array
for ii=1:(girders-1)
    for jj=pointer:pointer+onesetnodes-1
        horizemarray(pointer,1)=node;
        horizemarray(pointer,2)=holder;
        %increment
        pointer=pointer+1;
        node=node+1;
        holder=holder+1;
    end %end FOR
%re-initialize
node=holder;
holder=holder+onesetnodes;
end %end FOR

%Initialize variables for first diagonal element in array
node=100001;
holder=node+halfdiags+2;
pointer=1;

%Loops to fill in the pan diagonal element array (first diagonal)
for ii=1:(girders-1)
    for jj=pointer:pointer+halfdiags-1
        diagelemarray(pointer,1)=node;
        diagelemarray(pointer,2)=holder;
        %increment
        pointer=pointer+1;
        node=node+1;
        holder=holder+1;
    end %end FOR
%re-initialize
node=holder;
holder=holder+halfdiags+2;
end %end FOR

%Re-initialize variables for second diagonal element in array
node=100002;
holder=node+keeper;

%Loops to fill in the pan diagonal element array (second diagonal)
for ii=1:(girders-1)
    for jj=pointer:pointer+halfdiags-1
        diagelemarray(pointer,1)=node;
        diagelemarray(pointer,2)=holder;
        %increment
        pointer=pointer+1;
        node=node+1;
        holder=holder+1;
    end %end FOR
%re-initialize
node=holder+2;
holder=holder+halfdiags+2;
end %end FOR
%-------------------------------------------------------------------------
%Filename:  pans.m

%This file does general calculations and creates arrays of information
%   for the nodes needed to model the nodes of the SIP forms. It then
%   calls a file to do the same for the elements of the SIP forms and a
%   file to print the commands to file.
%
%Variables:
%
%flangewidth:       Top flange width dimension
%closexnode:        X-coord of node closest to coordinate axis
%farnode:          X-coord of node farthest from coordinate axis
%ynode:             Y-coord of nodes
%span:              Last coordinate of span
%firstpan:          Coordinate at first SIP node
%lastpan:           Coordinate at last SIP node
%pancoverage:       Length of pan coverage
%holder:            Temporary variable
%keeper:            Temporary variable
%remainder:         Decimal remainder variable
%product:           Rounded number for leftover girder length
%split:             Number to split the above variable along the length
%onesetnodes:       Number of nodes per girder, per flange side
%nodespershift:     Number of nodes created prior to each axis shift
%pannodes:          Total number of new nodes created for SIP forms
%panarray:          Array holding all SIP node information
:pointer:           Temporary array pointer
%zpointer:          Temporary array pointer
%SUNSHINE
%

%******************************************************************************
%******************************* NODES ***************************************
%******************************************************************************

%Set some parameters
flangewidth=sectionarray(1,2);
closexnode=flangewidth/2;
farnode=spacing-closexnode;
ynode=webheight;

%Find the last coordinate of span
if spantype==1
    span=span1ending;
elseif spantype==2
    span=span2ending;
elseif spantype==3
    span=span3ending;
end %end IF

%Use correct bridge orientation
if skew<=90
    firstpan=zshift;
    lastpan=span;
elseif skew>90
    firstpan=0;
    lastpan=span+zshift;
end %end IF
%General calculations
pancoverage = (lastpan - firstpan) * -1;
holder = pancoverage / panspacing;
keeper = round(holder);

% Make sure 'keeper' was rounded down, not up
if keeper > holder
    keeper = keeper - 1;
end % end IF

remainder = holder - keeper;
product = round(remainder * panspacing);
split = product / 2;

% Relocate first and last pan nodes
firstpan = firstpan - split;
lastpan = lastpan + split;

% Count Nodes
onesetnodes = keeper + 1;
nodespershift = onesetnodes * 2;
pannodes = nodespershift * (girders - 1);

% Initialize array
panarray = zeros(pannodes, 4);

% Loop for first & third columns (node numbers start at 100,001)
for ii = 1:pannodes
    panarray(ii, 1) = ii + 100000;  % node number column
    panarray(ii, 3) = ynode;        % y-coordinate column
end % end FOR

% Loop for second & fourth columns
pointer = 1;                % initialize
for jj = 1:girders - 1
    zpointer = firstpan;        % re-initialize
    for ii = pointer: pointer + onesetnodes - 1
        panarray(ii, 2) = closexnode;
        panarray(ii, 4) = zpointer;
        % increment
        pointer = pointer + 1;
        zpointer = zpointer - panspacing;
    end % end FOR
    zpointer = firstpan;        % re-initialize
    for ii = pointer: pointer + onesetnodes - 1
        panarray(pointer, 2) = farxnode;
        panarray(ii, 4) = zpointer;
        % increment
        pointer = pointer + 1;
        zpointer = zpointer - panspacing;
    end % end FOR
end % end big FOR

% Call file to create arrays of elements
panelements

% Call file to print commands to file
printpans

%---------------------------------------------------------------
%Filename:  pots.m

%This file takes the pot location coordinate array entered by the user and
% correctly sizes the array and fills in the correct tag ID.
%The measurement location coordinate array (potarray) will be used later
% in the program.

%_________________________________________________________________________
%Variables:

% potholder:     Temporary duplicate array to hold the coordinate values

%Tags Used:     9:  Measurement location along girder span
%_________________________________________________________________________

%Initialize temporary array
potholder=potarray;
potarray=zeros(numpots,2);

%Loop to correctly fill in the measurement locations array
for ii=1:numpots
    potarray(ii,1)=potholder(ii,1);
    potarray(ii,2)=9;
end %end FOR

%-------------------------------------------------------------------------
This file takes the data entered by the user and completes some preliminary calculations to be used later by the program.

% Variables:
% skew: Angle of the skew
% newangle: Angle of the skew to be manipulated
% xshift: Distance coordinate system will be shifted in x-direction
% yshift: Distance coordinate system will be shifted in y-direction
% zshift: Distance coordinate system will be shifted in z-direction
% place: A placeholder for the newangle in radians
% temp: Temporary value for the tangent of the angle
% numintgirders: Number of interior girders there are
% webheight: Height dimension of the web including half the thicknesses of the top and bottom flange for the girder keypoints

% First, the skew angle needs to be transposed into shift coordinates for ANSYS.

% If statement to determine if skew is > 90 degrees
if skew>90
    newangle=180-skew; % newangle is the angle to be manipulated
else
    newangle=skew;
end

xshift=spacing; % girder spacing for the wpoffs function in ANSYS
yshift=0; % yshift for wpoffs in ANSYS is always 0
place=newangle*pi/180; % conversion to radians for MATLAB
temp=tan(place); % tangent of angle in radians
zshift=spacing/temp; % calculate the z coordinate shift

% If the skew is < 90, the zshift should be negative and vice versa
if skew<90
    zshift=-1*round(zshift); % round the number to be used later
elseif skew==90
    zshift=0;
else
    zshift=round(zshift); % round the number to be used later
end

% Calculate number of interior girders
numintgirders=girders-2;

% Dimension to be used for girder keypoints (centerline values)
webheight=webheight+0.5*thickesttopflange+0.5*thickestbottomflange;
%Filename:  printgirdedrkp.m

%This file takes the arrays from the girderkeypoints.m file and prints them correctly to file for ANSYS. The file also prints out the shift commands while keeping track of the number of shifts so that the axis can be moved back to its original location.

%_________________________________________________________________________
%Variables:
%  %
%  %totalinteriorkp:   Total number of keypoints for all interior girders
%  %shifts:            Number of shifts in ANSYS
%_________________________________________________________________________
%Initialize variables
totalinteriorkp=interiornumkp*numintgirders;
shifts=0;   %initialize variable

%Print first girder keypoints and shift
for ii=1:girders
    %for first girder
    if ii==1
        for jj=1:firstnumkp
            fprintf(fid,'k, %8.0f , %8.2f , %8.2f , %8.2f 
',
                firstkparray(jj,:));
        end %end FOR
        fprintf(fid,'wpoffs, %8.2f , %8.2f , %8.2f 
',
            xshift,yshift,zshift);
        shifts=shifts+1;    %increment
    %for last girder
    elseif ii==girders
        for jj=1:lastnumkp
            fprintf(fid,'k, %8.0f , %8.2f , %8.2f , %8.2f 
',
                lastkparray(jj,:));
        end %end FOR
    %for interior girders
    elseif ii==2;
        for jj=1:totalinteriorkp
            if jj>1
                if interiorkparray(jj,1)-interiorkparray(jj-1,1)>1
                    fprintf(fid,'wpoffs, %8.2f , %8.2f , %8.2f 
',
                        xshift,yshift,zshift);
                    shifts=shifts+1;    %increment
                end %end IF
            end %end IF
            fprintf(fid,'k, %8.0f , %8.2f , %8.2f , %8.2f 
',
                interiorkparray(jj,:));
        end %end FOR
    end %end IF/ELSE
end %end big FOR

%Loop to move axis back to original location in ANSYS
while shifts>0
    fprintf(fid,'wpoffs, %8.2f , %8.2f , %8.2f 
',
        -1*xshift,-1*yshift,-1*zshift);
end %end big FOR

343
shifts=shifts-1; %decrement
end %end WHILE

%---------------------------------------------------------------
%Filename: printpans.m

%This file prints the ANSYS commands to file specified by user for
% implementation into the ANSYS interface.

%_________________________________________________________________________
%Variables:
%_________________________________________________________________________
%shifts:        %Shift counter variable
%pointer:       %Panarray pointer variable during command creation

%Initialize
shifts=0;
pointer=1;

%Print node creation commands
for ii=1:(girders-1)
    for jj=pointer:pointer+(onesetnodes*2)-1
        fprintf(fid,'n, %6.0f , %8.2f , %8.2f , %8.2f \n',panarray(jj,:));
        pointer=pointer+1;  %increment
    end %end FOR
    fprintf(fid,'wpoffs, %8.2f , %8.2f , %8.2f \n',xshift,yshift,zshift);
    shifts=shifts+1;    %increment
end %end FOR

%Loop to move axis back to original location
while shifts>0
    fprintf(fid,'wpoffs, %8.2f , %8.2f , %8.2f \n',-1*xshift,-1*yshift,-1*zshift);
    shifts=shifts-1;    %decrement
end %end WHILE

%Print element creation commands
%Print the horizontal elements, real constant set = 52
fprintf(fid,'Type,4 \n');           %Link 8 (truss) is element type 4
fprintf(fid,'Mat,1 \n');            %Steel is Material 1
fprintf(fid,'Real,52 \n');          %Real Constant set 52
fprintf(fid,'Esys,0 \n');           %Needed for ANSYS
fprintf(fid,'Secnum, \n');          %Needed for ANSYS
fprintf(fid,'Tshap,Line \n');       %Needed for ANSYS

%Print the element commands
for ii=1:totalhorizelems
    fprintf(fid,'e, %8.0f , %8.0f \n',horizelemarray(ii,:));
end %end FOR

%Print the diagonal elements, real constant set = 51
fprintf(fid,'Type,4 \n');           %Link 8 (truss) is element type 4
fprintf(fid,'Mat,1 \n');            %Steel is Material 1
fprintf(fid,'Real,51 \n');          %Real Constant set 51
fprintf(fid,'Esys,0 \n');           %Needed for ANSYS
fprintf(fid,'Secnum, \n');          %Needed for ANSYS
fprintf(fid,'Tshap,Line \n');       %Needed for ANSYS

345
%Print the element commands
for ii=1:totaldiagelems
    fprintf(fid,'e, %8.0f , %8.0f \n',diagelemarray(ii,:));
end %end FOR

%Print commands to couple the nodes
shifts=0;  %initialize shift counter

%Loop to print commands
for ii=1:(girders-1)
    fprintf(fid,'nsel,s,loc,y, %8.2f \n',ynode);
    fprintf(fid,'nsel,r,loc,x, %8.2f \n',closexnode);
    fprintf(fid,'cpintf,ux, %5.1f \n',couples);
    fprintf(fid,'cpintf,uy, %5.1f \n',couples);
    fprintf(fid,'cpintf,uz, %5.1f \n',couples);
    fprintf(fid,'nsel,s,loc,y, %8.2f \n',ynode);
    fprintf(fid,'nsel,r,loc,x, %8.2f \n',farxnode);
    fprintf(fid,'cpintf,ux, %5.1f \n',couples);
    fprintf(fid,'cpintf,uy, %5.1f \n',couples);
    fprintf(fid,'cpintf,uz, %5.1f \n',couples);
    fprintf(fid,'wpoffs, %8.2f , %8.2f , %8.2f \n',xshift,yshift,zshift);
    shifts=shifts+1;    %increment
end %end FOR

%Loop to move axis back to original location
while shifts>0
    fprintf(fid,'wpoffs, %8.2f , %8.2f , %8.2f \n',...  
            -1*xshift,-1*yshift,-1*zshift);
    shifts=shifts-1;    %decrement
end %end WHILE

%Select all the nodes for later analysis:
fprintf(fid,'nsel,all \n');

---------------------------------------------------------------
%Filename: rigidlinks.m

%This file draws creates an array to hold the keypoints needed to "draw" 
% in the rigid links. The first column in the array holds the keypoints 
% along the top of the girder and the second column holds the keypoints 
% of the slab. The file also prints to file the commands for ANSYS to 
% both draw the links and apply the correct attributes.

%_________________________________________________________________________
%Variables:
%
%numrigidlinks:       Required number of rigid links
%linesthusfar:        Total number of lines used for bridge before links
%starting:            The first line number representing a rigid link
%ending:              The last line number representing a rigid link
%newlinecounter:      Global counter of lines after all areas are meshed
%rigidlinkarray:      Keypoint number array for rigid links
%marker:              Pointer in rigid link array
%pointer:             Pointer to first keypoint on slab above first girder
%_________________________________________________________________________

%Initialize variables and arrays
numrigidlinks=girders*numdummycoords;
linesthusfar=areacounter+kpcounter-girders-1;
starting=linesthusfar+1;
ending=linesthusfar+numrigidlinks;
newlinecounter=numrigidlinks;
rigidlinkarray=zeros(numrigidlinks,2);
%-------------------------------------------------------------------------
%Start by filling in the array for the rigid links

marker=1;               %initialize
%The following three loops are to fill in the first column of the array
for ii=4:6:firstnumkp
    rigidlinkarray(marker,1)=firstkparray(ii,1);
    marker=marker+1;    %increment
end %end FOR
for ii=4:6:totalinteriorkp
    rigidlinkarray(marker,1)=interiorkparray(ii,1);
    marker=marker+1;    %increment
end %end FOR
for ii=4:6:lastnumkp
    rigidlinkarray(marker,1)=lastkparray(ii,1);
    marker=marker+1;    %increment
end %end FOR

pointer=20001+numdummycoords;       %initialize

%Loop to fill in second column of array
for ii=1:numrigidlinks
    rigidlinkarray(ii,2)=pointer;
    pointer=pointer+1;  %increment
end %end FOR
%-------------------------------------------------------------------------

%Print out statements to draw & mesh rigid links
%Loop for the line commands
for ii=1:numrigidlinks
    fprintf(fid,'l, %8.0f , %8.0f \n',rigidlinkarray(ii,:));
end %end FOR

%Commands to apply the attributes
fprintf(fid,'LSEL,s,line,, %8.0f , %8.0f \n',starting,ending);
fprintf(fid,'LATT,1,,5 \n');
%----------------------------------
% This file takes variables about the girder sections entered by the user and creates: extra coordinates needed for possible girder tapers
% an array for just the coordinates and tag IDs

% Variables:
% sectionholder: Temporary array for coordinates of section changes
% pointer: Pointer at a position in sectionholder array
% numsectioncoords: Counter to keep track of total coordinates
% sectioncoordarray: Z-coordinates and tag ID array for section changes

% Tags Used: 10: Girder Section Change - Flange Widens
% 11: Girder Section Change - Flange Narrows
% 12: Extra Coordinate for Flange Widening
% 13: Extra Coordinate for Flange More Narrow
% 21: Girder Section Change, same flange width

sectionholder=zeros(20,2); %initialize array

%Loop to fill in the temporary holder array with the known sections
for ii=1:numsections
    sectionholder(ii,1)=sectionarray(ii,1);
    ii=ii+1;
end %end FOR

%get pointer to first zero in array
pointer=find(sectionholder==0);
pointer=pointer(1);
numsectioncoords=numsections; %keep track of the number of coordinates

%Loop to find out how many extra z-coordinates are needed and where
for ii=1:numsections-1
    if sectionarray(ii,2)>sectionarray(ii+1,2)|sectionarray(ii,4)>sectionarray(ii+1,4)
        sectionholder(ii,2)=10; %tag ID
        sectionholder(pointer,1)=sectionarray(ii,1)+100;
        sectionholder(pointer,2)=12; %tag ID
        pointer=pointer+1;
        numsectioncoords=numsectioncoords+1;
    elseif sectionarray(ii,2)<sectionarray(ii+1,2)|sectionarray(ii,4)<sectionarray(ii+1,4)
        sectionholder(ii,2)=11; %tag ID
        sectionholder(pointer,1)=sectionarray(ii,1)-100;
        sectionholder(pointer,2)=13; %tag ID
        pointer=pointer+1;
        numsectioncoords=numsectioncoords+1;
    else
        sectionholder(ii,2)=21;
    end %end if-else
end %end FOR

349
end %end IF
end %end FOR

sectioncoordarray=zeros(numsectioncoords,2);   %initialize array

%Loop to fill in the array to be used for total z-coordinates
for ii=1:numsectioncoords
    sectioncoordarray(ii,:)=sectionholder(ii,:);
end %end FOR
%-------------------------------------------------------------------------
%Filename:  slabareas.m

%This file creates an array for the areas of the concrete slab using the
% keypoints created in slabkeypoints.m. It then prints the array in the
% correct ANSYS format to the output file.

% ________________________________________________________________________
% Variables:
%
% numslabareas:       Number of slab areas
% slabareaarray:     Array to write ANSYS commands to file
% pointer:           Array pointer in slabareaarray
% marker:            Keypoint pointer in slabareaarray

% ________________________________________________________________________

%Calculate the number of areas needed for the slab
numslabareas=(numdummycoords-1)*(girders+1);

%Initialize variables
slabareaarray=zeros(numslabareas,4);
pointer=1;
marker=slabkparray(pointer,1);

%Create the area array
for ii=1:girders+1
    for jj=1:numdummycoords-1
        slabareaarray(pointer,1)=marker;
        slabareaarray(pointer,2)=marker+1;
        slabareaarray(pointer,3)=marker+numdummycoords+1;
        slabareaarray(pointer,4)=marker+numdummycoords;
        %increment variables
        pointer=pointer+1;
        marker=marker+1;
    end %end FOR
    marker=marker+1;    %increment
end %end FOR

%Print area commands to file
for ii=1:numslabareas
    fprintf(fid,'A, %8.0f , %8.0f , %8.0f , %8.0f \n',...
            slabareaarray(ii,:));
end %end FOR

%Increment the global area counter
areacounter=areacounter+numslabareas;
% This file creates and fills in an array for the keypoints of the slab starting with keypoint #20001. The file also prints the keypoints in the correct ANSYS format to the output file.

% Variables:

% slabkpheight: Height dimension for location of slab keypoints
% numslabkp: Number of keypoints required for the slab
% slabkparray: Array to hold the keypoints' number and coordinates
% starter: Keeps up with the keypoint number
% pointer: Variable to keep up with locations in the slabkparray
% marker: Variable to keep up with girders for shifts

% Calculate the keypoint height for the slab
slabkpheight=webheight-.5*thickesttopflange+buildup+.5*slab;

% Calculate the required number of keypoints
numslabkp=numdummycoords*(girders+2);

% Increment the global keypoint counter
kpcounter=kpcounter+numslabkp;

% Initialize variables
slabkparray=zeros(numslabkp,4);
starter=20001; % first slab kp = 20,001
pointer=1;
marker=0;

% Three loops to fill in the slab keypoint array
for ii=1:numdummycoords
    slabkparray(pointer,1)=starter;
    slabkparray(pointer,2)=-overhang1;
    slabkparray(pointer,3)=slabkpheight;
    slabkparray(pointer,4)=-overhang1*zshift/xshift+dummycoordarray(ii);
    % increment variables
    starter=starter+1;
    pointer=pointer+1;
end % end FOR

for ii=1:girders
    for jj=1:numdummycoords
        slabkparray(pointer,1)=starter;
        slabkparray(pointer,2)=marker*xshift;
        slabkparray(pointer,3)=slabkpheight;
        slabkparray(pointer,4)=marker*zshift+dummycoordarray(jj);
        % increment variables
        starter=starter+1;
        pointer=pointer+1;
    end % end FOR
    marker=marker+1; % increment
end % end FOR

for ii=1:numdummycoords
    slabkparray(pointer,1)=starter;
end % end FOR
slabkparray(pointer,2)=overhang2+(marker-1)*xshift;
slabkparray(pointer,3)=slabkheight;
slabkparray(pointer,4)=overhang2*zshift/xshift+(marker-1)*...
    zshift+dummycoordarray(ii);
%increment variables
starter=starter+1;
pointer=pointer+1;
end %end FOR
%---------------------------------------------------------------------------------

fprintf(fid,'k, %8.0f , %8.2f , %8.2f , %8.2f \
', slabkparray(ii,:));
end %end FOR
%---------------------------------------------------------------------------------
% This file takes the user's input (from 'main.m') for the intermediate 
% stiffeners and places them into arrays for the different girders: 
% first, interior and last. 
% The reason for this is that the first girder will always only have 
% intermediate stiffeners on the right side (the side towards the 
% interior of the bridge), the last girder will always only have 
% intermediate stiffeners on the left side and the interior girders will 
% have intermediate stiffeners on both sides. 
% This file creates arrays of z-coordinates (with tag IDs) to be used 
% later. 
%
% Variables: 
%
% numstiff:      Total number of stiffeners on each girder 
% fgisa:         The first girder's intermediate stiffener array 
% igisa:         An interior girder's intermediate stiffener array 
% lgisa:         The last girder's intermediate stiffener array 
%
% Tags Used:     3:  Right Side Intermediate Stiffeners 
%                4:  Left Side Intermediate Stiffeners 
%
% Calculate total number of intermediate stiffeners 
numstiff=numstiffright+numstiffleft;

% First Girder Intermediate Stiffener Array (FGISA) 
fgisa=zeros(numstiff,2); 
jj=1;   % array counter 

% Loops for int. stiffeners on right side of the first girder 
% Right Side 
for ii=1:numstiffright 
    fgisa(jj,1)=rightstiffarray(ii,1); 
    fgisa(jj,2)=3; 
    jj=jj+1; 
end % end FOR 
% Left Side 
for ii=1:numstiffleft 
    fgisa(jj,1)=leftstiffarray(ii,1); 
    fgisa(jj,2)=3; 
    jj=jj+1; 
end % end FOR 

% Interior Girder Intermediate Stiffener Array (IGISA) 
igisa=zeros(numstiff,2); 
jj=1;   % array counter 

% Loop for int. stiffeners on both sides of interior girders 
% Right Side 
for ii=1:numstiffright 
    igisa(jj,1)=rightstiffarray(ii,1); 
    igisa(jj,2)=3; 
    jj=jj+1; 
end % end FOR 
% Left Side
for ii=1:numstiffleft
    igisa(jj,1)=leftstiffarray(ii,1);
    igisa(jj,2)=4;
    jj=jj+1;
end %end FOR

%Last Girder Intermediate Stiffener Array (LGISA)
lgisa=zeros(numstiff,2);
jj=1; %array counter

%Loops for int. stiffeners on left side of the last girder
%Right Side
for ii=1:numstiffright
    lgisa(jj,1)=rightstiffarray(ii,1);
    lgisa(jj,2)=4;
    jj=jj+1;
end %end FOR
%Left Side
for ii=1:numstiffleft
    lgisa(jj,1)=leftstiffarray(ii,1);
    lgisa(jj,2)=4;
    jj=jj+1;
end %end FOR
%-------------------------------------------------------------------------
%Filename: zcoords.m

%This file takes the data from all of the files that created separate
% arrays for the different items along the girder span and joins them
% together. The result is three arrays representing the z-coordinates,
% tag IDs and a counter variable for the first girder, an interior
% girder and the last girder.
%Note: The first, last and interior girders will hold the same
% coordinates even though the first and last girders will not
% necessarily use all of them. For instance, the first girder will
% not have left side connector plates, but there will be a coordinate
% there for them.
%
%Variables:
%
% firstcoordarray: Array to hold the entire tally of z-coordinates
% for the first girder.
% interiorcoordarray: Array to hold the entire tally of z-coordinates
% for the interior girders.
% lastcoordarray: Array to hold the entire tally of z-coordinates
% for the first girder.
% pointer: Pointer variable to find the first zero in arrays
% temparray: Temporary array to hold values of other arrays
% tempvalue: Temporary value for iteration in for loop
% numfirstcoords: Holder of number of coordinates for girder
% numlastcoords: Holder of number of coordinates for girder
% numinteriorcoords: Holder of number of coordinates for girder
%
%Tags Used: 14: Start of the Girder
%
%Initialize arrays to random large size
firstcoordarray=zeros(500,3);
interiorcoordarray=zeros(500,3);
lastcoordarray=zeros(500,3);

%Loop through all girders
for ii=1:girders
    %For the first girder
    if ii==1
        jj=1; %initialize global counter
        %Loop for right connector plates
        for kk=1:numconns
            firstcoordarray(jj,1:2)=rightconnarray(kk,1:2);
            jj=jj+1;
        end %end FOR
        %Loop for left connector plates (not used on first girder)
        for kk=1:numconns
            firstcoordarray(jj,1:2)=leftconnarray(kk,1:2);
            jj=jj+1;
        end %end FOR
        %Loop for intermediate stiffeners
        for kk=1:numstiff
            firstcoordarray(jj,1:2)=fgisa(kk,1:2);
            jj=jj+1;
        end %end FOR
        %Loop for bearing plates
    end %end IF
end %end FOR
for kk=1:numbearing
    firstcoordarray(jj,1:2)=beararray(kk,1:2);
    jj=jj+1;
end %end FOR
%Loop for pot measurement locations
for kk=1:numpots
    firstcoordarray(jj,1:2)=potarray(kk,1:2);
    jj=jj+1;
end %end FOR
%Loop for girder sections
for kk=1:numsectioncoords
    firstcoordarray(jj,1:2)=sectioncoordarray(kk,1:2);
    jj=jj+1;
end %end FOR
%Loop for pour sequence
for kk=1:numpours-1
    firstcoordarray(jj,1:2)=pourarray(kk,1:2);
    jj=jj+1;
end %end FOR
%For the last girder
elseif ii==girders
    jj=1; %Re-initialize global counter
%Loop for left connector plates
for kk=1:numconns
    lastcoordarray(jj,1:2)=leftconnarray(kk,1:2);
    jj=jj+1;
end %end FOR
%Loop for right connector plates (not used for last girder)
for kk=1:numconns
    lastcoordarray(jj,1:2)=rightconnarray(kk,1:2);
    jj=jj+1;
end %end FOR
%Loop for intermediate stiffeners
for kk=1:numstiff
    lastcoordarray(jj,1:2)=lgisa(kk,1:2);
    jj=jj+1;
end %end FOR
%Loop for bearing plates
for kk=1:numbearing
    lastcoordarray(jj,1:2)=beararray(kk,1:2);
    jj=jj+1;
end %end FOR
%Loop for pot measurement locations
for kk=1:numpots
    lastcoordarray(jj,1:2)=potarray(kk,1:2);
    jj=jj+1;
end %end FOR
%Loop for girder sections
for kk=1:numsectioncoords
    lastcoordarray(jj,1:2)=sectioncoordarray(kk,1:2);
    jj=jj+1;
end %end FOR
%Loop for pour sequence
for kk=1:numpours-1
    lastcoordarray(jj,1:2)=pourarray(kk,1:2);
    jj=jj+1;
end %end FOR
%For the interior girders
else ii==2
    jj=1;  %Re-initialize global counter
    %Loop for right connector plates
    for kk=1:numconn
        interiorcoordarray(jj,1:2)=rightconnarray(kk,1:2);
        jj=jj+1;
    end %end FOR
    %Loop for left connector plates
    for kk=1:numconn
        interiorcoordarray(jj,1:2)=leftconnarray(kk,1:2);
        jj=jj+1;
    end %end FOR
    %Loop for intermediate stiffeners
    for kk=1:numstiff
        interiorcoordarray(jj,1:2)=igisa(kk,1:2);
        jj=jj+1;
    end %end FOR
    %Loop for bearing plates
    for kk=1: numbearing
        interiorcoordarray(jj,1:2)=beararray(kk,1:2);
        jj=jj+1;
    end %end FOR
    %Loop for pot measurement locations
    for kk=1:numpots
        interiorcoordarray(jj,1:2)=potarray(kk,1:2);
        jj=jj+1;
    end %end FOR
    %Loop for girder sections
    for kk=1: numsectioncoords
        interiorcoordarray(jj,1:2)=sectioncoordarray(kk,1:2);
        jj=jj+1;
    end %end FOR
%end IF/ELSE
end %end FOR
%-------------------------------------------------------------------------
%For the first girder array of z-coordinates
pointer=find(firstcoordarray==0);       %point to the zeros in the array
pointer=pointer(1,1);                   %point to only the first zero
%Hold and sort coordinate array
temparray=sortrows(firstcoordarray,1);
%Re-initialize arrays to correct size
firstcoordarray=zeros(pointer,3);

%Loop to fill back in the values of the array
tempvalue=pointer;  %value to iterate for counter column of array
for ii=1:pointer
    firstcoordarray(ii,1:2)=temparray(ii,1:2);
%fill the third column in w/ counter reference
firstcoordarray(ii,3)=tempvalue;
tempvalue=tempvalue-1;      %decrement counter
end %end FOR

%sort coordinate arrays by the counter tag
firstcoordarray=sortrows(firstcoordarray,3);
%Set tag ID for the beginning of the girders
firstcoordarray(1,2)=14;
%Initialize variable for number of coordinates
numfirstcoords=pointer;
%-------------------------------------------------------------------------

%For the interior girder array of z-coordinates
pointer=find(interiorcoordarray==0);    %point to the zeros in the array
pointer=pointer(1,1);                   %point to only the first zero
%Hold and sort coordinate array
temparray=sortrows(interiorcoordarray,1);
%Re-initialize arrays to correct size
interiorcoordarray=zeros(pointer,3);
%Loop to fill back in the values of the array
tempvalue=pointer;  %value to iterate for counter column of array
for ii=1:pointer
    interiorcoordarray(ii,1:2)=temparray(ii,1:2);
    %Fill the third column in w/ counter reference
    interiorcoordarray(ii,3)=tempvalue;
    tempvalue=tempvalue-1;      %decrement counter
end %end FOR

%Sort coordinate arrays by the counter tag
interiorcoordarray=sortrows(interiorcoordarray,3);
%Set tag ID for the beginning of the girders
interiorcoordarray(1,2)=14;
%Initialize variable for number of coordinates
numinteriorcoords=pointer;
%-------------------------------------------------------------------------

%For the last girder array of z-coordinates
pointer=find(lastcoordarray==0);        %point to the zeros in the array
pointer=pointer(1,1);                   %point to only the first zero
%Hold and sort coordinate array
temparray=sortrows(lastcoordarray,1);
%Re-initialize arrays to correct size
lastcoordarray=zeros(pointer,3);
%Loop to fill back in the values of the array
tempvalue=pointer;  %value to iterate for counter column of array
for ii=1:pointer
    lastcoordarray(ii,1:2)=temparray(ii,1:2);
    %Fill the third column in w/ counter reference
    lastcoordarray(ii,3)=tempvalue;
    tempvalue=tempvalue-1;      %decrement counter
end %end FOR

359
% Sort coordinate arrays by the counter tag
lastcoordarray=sortrows(lastcoordarray,3);
% Set tag ID for the beginning of the girders
lastcoordarray(1,2)=14;
% Initialize variable for number of coordinates
numlastcoords=pointer;
%---------------------------------------------------------------