STRUT-AND-TIE EVALUATION PROGRAM (STEP) FOR THE DESIGN OF BRIDGE COMPONENTS

by

Andi Vicksman

A Thesis

Submitted to the Faculty of Purdue University

In Partial Fulfillment of the Requirements for the degree of

Master of Science in Civil Engineering

Lyles School of Civil Engineering

West Lafayette, Indiana

August 2019

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF COMMITTEE APPROVAL

Dr. Christopher S. Williams, Chair
Department of Civil Engineering
Dr. Robert J. Frosch
Department of Civil Engineering
Dr. Ghadir Haikal
Department of Civil Engineering

Approved by:
Dr. Dulcy Abraham
Head of the Graduate Program

To my parents, Debbie and Sandy Vicksman

Mom and Dad,

Thank you for your unwavering and unconditional support. You have instilled in me perseverance, a love of learning, and compassion. I am forever grateful for all you do for me.

Love,

Andi

ACKNOWLEDGMENTS

I would first like to thank my advisor, Dr. Christopher Williams, for his constant support. I would not have been successful in completing this project without his knowledge and his willingness to share that knowledge. His attention to detail has helped me improve as an engineer. I am also thankful for the help I received from Dr. Robert Frosch and Dr. Ghadir Haikal. Their advice and feedback were indispensable. I am grateful to you all for giving me the opportunity to work on this project and for all that I learned from the experience.

Without INDOT and the JTRP office, this project would have never happened. The Study Advisory Committee was vital during the project and I cannot thank them enough for their expertise and willingness to help when asked. Prince Baah, Jeremy Hunter, Mahmoud Hailat, Donald Shaw, Tim Wells, Pete White, and Tyler Wolf, thank you for your advice, feedback, and support throughout this process.

Merrick Howarth, I have been so fortunate to work with you the past nine months. Your dedication and support on this project were unparalleled. Anything I asked you to help with, you enthusiastically completed. STEP would not have the graphical capabilities it does without your assistance. I cannot wait to see all that you will continue to accomplish in the structural engineering field.

I have met wonderful people at Purdue University through the Structural Engineering Department and Bowen Laboratory. I am grateful to the friends I have made for the everyday conversations and memories created throughout my time here. Thank you for making my graduate school experience more enjoyable. I am fortunate to have an incredibly supportive family and network of friends beyond those at Purdue University. Thank you to all of you as well for always encouraging and supporting me.

Morgan Broberg, Eric Fleet, and Will Rich, your friendship means the world to me. Without your motivation and kind words this project would not be complete. I was extremely fortunate to find such amazing friends who are so selfless, kind, and encouraging. Thank you for always being willing to help and for all the adventures. I cannot wait for many more!

Kyle Godfroy, you are the kindest and most selfless person I have ever met. You remained my rock throughout this project and kept me motivated. Thank you for being understanding and supporting me. I am forever grateful for all that you continuously do for me.

Mom and Dad, thank you for being my voice of reason, my biggest fans, and the best role models. You both are the very definition of compassion and strength. I would not have reached any of my accomplishments without your support. Thank you most of all for loving me no matter what. Adam, you’re the best brother out there. Thank you for all the laughs over the years.

TABLE OF CONTENTS

LIST OF TABLES 10

LIST OF FIGURES 11

ABSTRACT 15

CHAPTER 1. INTRODUCTION 16

1.1 Background 16

1.2 Project Objective and Scope 17

1.3 Organization 18

CHAPTER 2. INTRODUCTION TO THE STRUT-AND-TIE METHOD 20

2.1 Overview 20

2.2 Past Research on the Strut-and-Tie Method 20

2.3 Disturbed Regions 21

2.4 Fundamentals of the Strut-and-Tie Method 22

2.5 Design Procedure 23

2.6 Define Load Cases 24

2.7 Size Structural Component and Determine Initial Reinforcement Layout 25

2.8 Analyze Structural Component 25

2.9 Develop Strut-and-Tie Model 25

2.10 Proportion Ties 29

2.11 Check Nodal Strengths 31

2.11.1 Types of Nodes 32

2.11.2 Proportioning CCC Nodes 32

2.11.3 Proportioning CCT Nodes 35

2.11.4 Proportioning CTT Nodes 36

2.11.5 Nodal Strength Checks 37

2.12 Proportion Crack Control Reinforcement 41

2.13 Provide Anchorage for Ties 42

2.14 Summary 43

CHAPTER 3. OVERVIEW OF COMPUTER PROGRAM 44

3.1 Overview 44

3.2 Coding and Layout 44

3.2.1 Excel VBA Overview 44

3.2.2 Functionality of the Computer Program 46

3.2.3 Computer Program Layout and Formatting 47

3.3 User Inputs 48

3.3.1 Geometric Properties 51

3.3.2 Load Factor and Concrete Material Properties 51

3.3.3 Stirrups and Horizontal Crack Control Reinforcement 52

3.3.4 Longitudinal Reinforcement 54

3.3.5 Factored Applied Loads and Reactions 55

3.4 Program Outputs 57

3.4.1 Strut-and-Tie Model 58

3.4.2 Nodes and Members 58

3.4.3 Node Figures 59

3.4.4 Nodal Checks 59

3.4.5 Reinforcement Design 59

3.4.6 Structural Analysis Outputs 61

3.5 Summary 61

CHAPTER 4. COMPONENT 1 - MULTI-COLUMN BENT CAP 62

4.1 Overview 62

4.2 Computer Program Procedure 62

4.2.1 Instructions 63

4.2.2 Subroutines 65

4.2.3 User Inputs 65

4.2.4 Analyze Structural Component 68

4.2.4.1 Self-Weight Distribution 68

4.2.4.2 Structural Analysis 69

4.2.4.3 Develop Shear and Moment Diagrams 71

4.2.5 Develop Strut-and-Tie Model 72

4.2.5.1 Optimize Top Chord Location 72

4.2.5.2 Node Locations 73

4.2.5.3 Create Members 77
4.2.5.4 Method of Joints 79
4.2.5.5 Plot Model 80
4.2.6 Perform Design Checks 81
4.2.6.1 Design Reinforcement 81
4.2.6.2 Combine Struts and Subdivide Nodes 84
4.2.6.3 Plot Node Figures 89
4.2.6.4 Nodal Strength Checks 89
4.2.6.5 Anchorage Checks 96
4.2.7 Formatting 99
4.2.8 Resetting the Code 99
4.2.9 Rerunning the Code 99
4.3 Additional Limitations and Assumptions 101
4.4 Interpretation of Program Outputs 102
4.5 Design Example: Five-Column Bent Cap 103
4.5.1 Bent Cap Geometry, Material Properties, and Loading 103
4.5.2 User Inputs 110
4.5.3 Computer Program Outputs 113
4.5.3.1 Strut-and-Tie Model 113
4.5.3.2 Nodes and Members 116
4.5.3.3 Node Figures 118
4.5.3.4 Nodal Checks 118
4.5.3.5 Reinforcement 120
4.5.3.6 Continuous Beam Analysis 123
4.5.3.7 Shear and Moment Diagrams 126
4.5.3.8 Final Design 129
4.6 Summary 132
CHAPTER 5. COMPONENT 2 – INTEGRAL AND SEMI-INTEGRAL END BENT CAPS 133
5.1 Overview 133
5.2 Computer Program Procedure 133

5.2.1 Instructions 134
5.2.2 Subroutines 134
5.2.3 User Inputs 134
5.2.4 Structural Analysis 136
5.2.5 Nodal Checks 137
5.2.6 Anchorage Checks 138
5.3 Additional Limitations and Assumptions 138
5.4 Design Example: Integral End Bent 138
5.4.1 End Bent Cap Geometry, Material Properties, and Loading 138
5.4.2 User Inputs 143
5.4.3 Computer Program Outputs 145
5.4.3.1 Strut-and-Tie Model 146
5.4.3.2 Nodes and Members 148
5.4.3.3 Node Figures 148
5.4.3.4 Nodal Checks 149
5.4.3.5 Reinforcement 151
5.4.3.6 Continuous Beam Analysis 154
5.4.3.7 Shear and Moment Diagrams 155
5.4.3.8 Final Design 157
5.5 Summary 160
CHAPTER 6. SUMMARY AND CONCLUDING REMARKS 161
6.1 Summary 161
6.2 Concluding Remarks 162
APPENDIX A. PIER CAP STM DESIGN PROCEDURE 164
APPENDIX B. END BENT CAP STM DESIGN PROCEDURE 171
APPENDIX C. CTR REPORT PERMISSION 180
REFERENCES 183

LIST OF TABLES

Table 2.1: Concrete Efficiency Factors, ν (from AASHTO LRFD, 2017) 39
Table 4.1: Solving Submatrix for Node W 79
Table 4.2: Concrete Efficiency Factors, ν (from AASHTO LRFD (2017)) 93

LIST OF FIGURES

Figure 1.1: Strut-and-Tie Model for a Beam (adapted from Williams et al., 2012) 16

Figure 2.1: Division of Member into B-regions and D-regions Based on St. Venant's Principle (adapted from Williams et al., 2012) 22

Figure 2.2: Example Strut-and-Tie Model for a Simple Beam (from Williams et al., 2012)....... 22

Figure 2.3: Optimization of Moment Arm to Determine Location of Top Chord (adapted from Williams et al., 2012)... 27

Figure 2.4: Using the Least Number of Truss Panels Possible (from Williams et al., 2012)....... 28

Figure 2.5: Principles Used to Determine Members Comprising a Strut-and-Tie Model............ 29

Figure 2.6: Width of Tie for Determining Required Stirrup Spacing... 30

Figure 2.7: Hydrostatic vs. Non-Hydrostatic Nodes (from Williams et al., 2012)....................... 31

Figure 2.8: Types of Nodes in a Strut-and-Tie Model (adapted from Williams et al., 2012) 32

Figure 2.9: CCC Node: Process of Combining Struts and Subdividing Nodes into Two Parts (adapted from Williams et al., 2012) 33

Figure 2.10: CCC Node: Process of Combining Struts and Subdividing Nodes into Three Parts (adapted from Williams et al., 2012) 34

Figure 2.11: Geometry of a CCC Node (adapted from Williams et al., 2012)............................ 35

Figure 2.12: Geometry of a CCT Node (adapted from Williams et al,. 2012) 36

Figure 2.13: Geometry of a CTT Node (adapted from Williams et al., 2012) 37

Figure 2.14: Bearing Area Definitions for Confinement Modification Factor (adapted from Williams et al., 2012, and AASHTO LRFD, 2017).. 38

Figure 2.15: Concrete Efficiency Factors for Different Node Types (from Williams et al., 2012) 39

Figure 2.16: Crack Control Reinforcement (from Williams et al., 2012).................................... 42

Figure 2.17: Anchorage of Ties (from Williams et al., 2012; adapted from Birrcher et al., 2009) 43

Figure 3.1: Computer Program Procedure Overview 47

Figure 3.2: Pier Cap STM Design Workbook Inputs Sheet... 49

Figure 3.3: End Bent Cap STM Design Workbook Inputs Sheet 50

Figure 3.4: "Geometric Properties" Input Table (a) for Pier Caps STM Design Workbook; (b) for End Bent Caps STM Design Workbook 51
Figure 3.5: "Load Factor” Input Table... 52
Figure 3.6: "Concrete Material Properties" Input Table... 52
Figure 3.7: "Stirrups" Input Table 53
Figure 3.8: "Horizontal Crack Control Reinforcement" Input Table 53
Figure 3.9: Horizontal Crack Control Reinforcement for a Thin and Thick Member (adapted from AASHTO LRFD, 2017)... 54
Figure 3.10: "Longitudinal Reinforcement" Input Table.. 55
Figure 3.11: “Factored Applied Loads" Input Table 56
Figure 3.12: "Reactions" Input Table 57
Figure 4.1: Multi-Column Bent Cap.. 62
Figure 4.2: Instructions Sheet within Computer Program 64
Figure 4.3: Inputs Sheet for Multi-Column Bent Cap 67
Figure 4.4: Fixed-End-Moment Calculations - (a) Cantilever Beam Condition; (b) Fixed-Fixed Beam Condition 70
Figure 4.5: Strut-and-Tie Model Near a Column (adapted from Williams et al., 2012) 74
Figure 4.6: Additional Nodes to Satisfy Angle Limit- 2 Panels.................................... 75
Figure 4.7: Satisfying the Angle Limit Between Vertical Ties and Diagonal Struts.................... 76
Figure 4.8: Diagonal Strut Orientation According to Shear Diagram 78
Figure 4.9: Members Intersecting at Node W... 80
Figure 4.10: Nodes Where Struts are Combined 86
Figure 4.11: Subdividing a Node into Two Parts (adapted from Williams et al., 2012).............. 86
Figure 4.12: Subdividing a Node into Three Parts (adapted from Williams et al., 2012)............ 87
Figure 4.13: Slight Change in Angle for Vertical Strut when Node is Subdivided into Three Parts 88
Figure 4.14: Proportions of Three Node Types 91
Figure 4.15: Bearing Area Definitions for Confinement Modification Factor (adapted from Williams et al., 2012, and AASHTO LRFD, 2017).. 92
Figure 4.16: Node Subdivided into Two Parts – (a) Left Portion of Node; (b) Right Portion of Node (from Williams et al., 2012).. 95

Figure 4.17: Node Subdivided into Three Parts (from Williams et al., 2012) 96
Figure 4.18: Nodal Regions where Anchorage Checks are Performed 97
Figure 4.19: Available Length Calculation for Pier Caps Performed by the STM Program (adapted from Williams et al., 2012) 98
Figure 4.20: Example of Limitation within the Program – Changing Strut Orientation due to Subdivision of a Node 101
Figure 4.21: Plan and Elevation Views of Bent Cap – Left (from Williams et al., 2012) 104
Figure 4.22: Plan and Elevation Views of Bent Cap – Right (from Williams et al., 2012) 105
Figure 4.23: Superstructure Cross-Sections (from Williams et al., 2012) 107
Figure 4.24: Assumed Bearing Area for Girder Loads 108
Figure 4.25: Factored Loads and Locations of Reactions (adapted from Williams et al., 2012) 109
Figure 4.26: Inputs Sheet for Design Example (Bent Cap) 112
Figure 4.27: Initial Strut-and-Tie Model from Program Output (Bent Cap) 114
Figure 4.28: Strut-and-Tie Model from Program Output after Removing Unnecessary Node (Bent Cap) 115
Figure 4.29: Nodes and Members Sheet from Program Output (Bent Cap) 117
Figure 4.30: Examples of Node Figures from Program Output (Bent Cap) 118
Figure 4.31: Summary of Nodal Strength Checks from Program Output (Bent Cap) 119
Figure 4.32: Summary of Longitudinal Reinforcement Strength Checks from Program Output (Bent Cap) 120
Figure 4.33: Crack Control Reinforcement from Program Output (Bent Cap) 121
Figure 4.34: Stirrup Spacing Requirements from Program Output with 2 Stirrup Legs Input (Bent Cap) 122
Figure 4.35: Stirrup Spacing Requirements from Program Output with 4 Stirrup Legs Input (Bent Cap) 122
Figure 4.36: Anchorage Check from Program Output (Bent Cap) 123
Figure 4.37: Continuous Beam Analysis from Program Output (Bent Cap) 125
Figure 4.38: Shear Diagram from Program Output (Bent Cap) 126
Figure 4.39: Moment Diagram from Program Output (Bent Cap) 127
Figure 4.40: Shear and Moment Diagram Calculations from Program Output (Bent Cap) 128

Figure 4.41: Elevation with Reinforcement Details based on Program Output- Bent Cap (from Williams et al., 2012) 130
Figure 4.42: Section with Reinforcement Design based on Program Outputs – Bent Cap 131
Figure 5.1: Effective Depth Measurement for End Bent Cap – (a) entire member depth is effective; (b) portion of member depth is effective 135
Figure 5.2: "Pile Reactions" Input Table for End Bent Caps 136
Figure 5.3: Plan View of Integral End Bent Cap 140
Figure 5.4: Elevation View of Integral End Bent 141
Figure 5.5: Factored Loads and Locations of Reactions 142
Figure 5.6: Inputs Sheet for Design Example (End Bent Cap) 144
Figure 5.7: Strut-and-Tie Model from Program Output (End Bent Cap) 147
Figure 5.8: Nodes and Members Sheet from Program Output (End Bent Cap) 148
Figure 5.9: Examples of Node Figures from Program Output (End Bent Cap) 149
Figure 5.10 Summary of Nodal Check Program Outputs (End Bent Cap) 150
Figure 5.11: Summary of Longitudinal Reinforcement Strength Checks from Program Output (End Bent Cap) 151
Figure 5.12: Crack Control Reinforcement Requirements from Program Output (End Bent Cap) 152
Figure 5.13: Stirrup Spacing Requirements from Program Output (End Bent Cap) 153
Figure 5.14: Anchorage Check from Program Output (End Bent Cap) 153
Figure 5.15: Continuous Beam Analysis from Program Output (End Bent Cap) 154
Figure 5.16: Shear Diagram from Program Output (End Bent Cap) 155
Figure 5.17: Moment Diagram from Program Output (End Bent Cap) 156
Figure 5.18: Shear and Moment Diagram Calculations from Program Output (End Bent Cap) 157
Figure 5.19: Elevation with Reinforcement Details based on Program Output (End Bent Cap) 159
Figure 5.20: Section with Reinforcement Design based on Program Outputs (End Bent Cap) . 160

ABSTRACT

Author: Vicksman, Andi, S. MSCE
Institution: Purdue University
Degree Received: August 2019
Title: Strut-and-Tie Evaluation Program (STEP) for the Design of Bridge Components
Committee Chair: Christopher S. Williams

The strut-and-tie method (STM) is a powerful tool used for the design of D-regions (disturbed regions) of reinforced concrete structures. Many typical bridge substructure components consist of D-regions and require the use of the STM for design. Implementation of the STM is more complex than typical design methods, and engineers are often unfamiliar with the design process. As a result, designing using the STM is more time consuming than traditional design methods. The Indiana Department of Transportation (INDOT) identified a need for a tool that assists with the design of typical bridge substructure components using the STM. STEP (Strut-and-Tie Evaluation Program) is a computer program created to fulfill this role. To use the computer program, engineers input geometric conditions, material properties, and reinforcement information for a structural component. STEP uses this information to develop a strut-and-tie model and perform STM design procedures. A graphical representation of the model and a summary of the design results are provided as program outputs for the user.

STEP, created using Excel VBA, is intended to aid in the design of multi-column bent caps and integral and semi-integral end bent caps. Within this thesis, an overview of the STM is provided, including the basic procedures for designing using the STM. An introduction to Excel VBA is also presented. The document describes the layout and formatting of the computer program, required user inputs, and program outputs. Furthermore, limitations and assumptions within the computer program for the substructure components are also included. Finally, design examples focused on the use of STEP for the design of a five-column bent cap and an integral end bent cap are presented. This document can be used as a resource for engineers when designing bridge substructure components using STEP.

CHAPTER 1. INTRODUCTION

1.1 Background

The strut-and-tie method (STM) is a powerful tool used for the design of D-regions (disturbed regions) of reinforced concrete structures. D-regions occur near loads or geometric discontinuities (e.g., corbels, dapped ends, and frame corners). Within D-regions, the distribution of strains is non-linear. As a result, the assumption that plane sections remain plane is no longer valid. Therefore, typical sectional analysis methods that are based on the assumption that strains vary linearly through the depth of a member are not appropriate for design within D-regions. When using the STM, the flow of forces through a structural component are represented through the use of a truss model, known as a strut-and-tie model. The compression members in this model are called struts, and the tension members are called ties. These members intersect at points called nodes. A strut-and-tie model for a beam is shown in Figure 1.1. The entire beam is comprised of D-regions due to the close spacing of the loads and reactions applied to the beam. After creating a strut-and-tie model for a component, the designer must ensure that the member has sufficient capacity to resist the compressive and tensile forces within the model. This is accomplished through proportioning ties, checking nodal strengths, and ensuring longitudinal reinforcement is properly anchored. Distributed reinforcement must also be provided to control cracking.

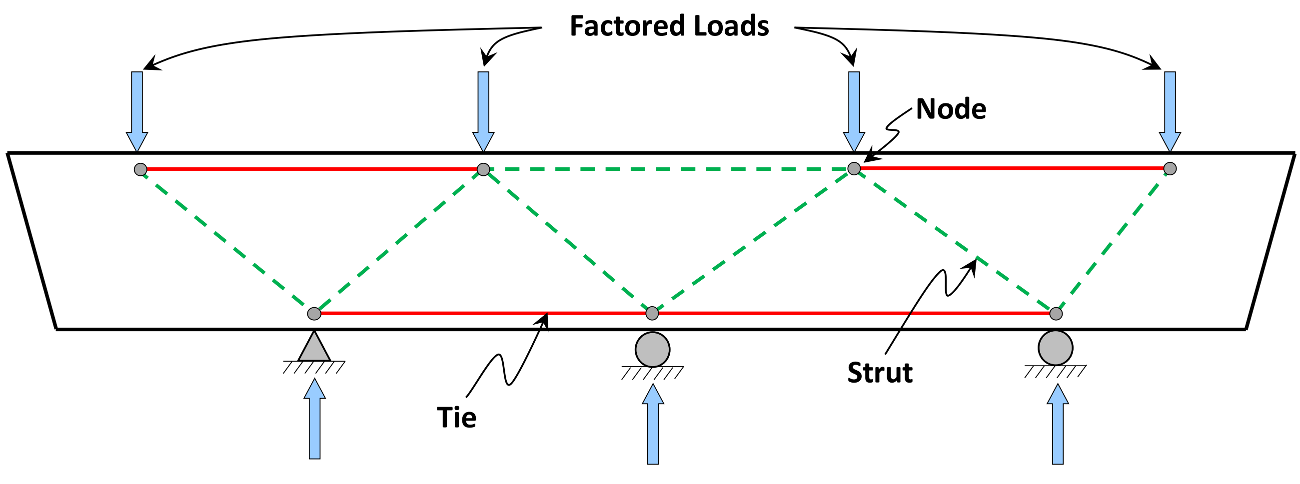


Figure 1.1: Strut-and-Tie Model for a Beam (adapted from Williams et al., 2012)

The STM was first introduced into code provisions in the United States in 1994 when it was included in the *AASHTO LRFD Bridge Design Specifications*. The STM did not appear in the ACI *Building Code Requirements for Structural Concrete* (ACI 318) until 2002 when it was added as an appendix to the code. In ACI 318-14, the STM provisions were moved to Chapter 23 in the main body of the code. Significant updates to the STM provisions in AASHTO LRFD were included in the 2016 Interim Revisions to the 7^{th} Edition of the specifications. The updated provisions were meant to simplify the STM design procedure and improve accuracy (Birrcher et al., 2009; Williams et al., 2012). Despite the 25-year history of the STM within U.S. design specifications, engineers are often reluctant to apply the STM in the design office. The STM is more complex than typical design methods, and engineers are often unfamiliar with the design process. As a result, the Indiana Department of Transportation (INDOT) has identified the need for a tool in the form of a computer program that assists with the STM design of typical bridge substructure components.

1.2 Project Objective and Scope

Performing the STM design of one structural component may potentially take hours, or even days, for an engineer to complete. Designing with the STM is often significantly more time consuming than traditional design methods. Examining Articles 5.7.2.1 and C5.5.1.2.3 of AASHTO LRFD (2017), the use of the STM is the preferred method for the design of D-regions of bridge components. Although engineers routinely design bridge substructure components that warrant the use of the STM, guidance on the implementation of the STM is still needed. Many documents have been created to aid engineers in the use of the STM. However, each state department of transportation has unique needs that are best addressed by tools tailored specifically toward those needs. To develop a tool to help meet Indiana's need for STM design assistance, INDOT initiated a project with the primary objective of creating a computer program to aid engineers with the design of bridge components using STM.

A Study Advisory Committee created by INDOT assisted in the development of the computer program and expressed specific needs within the bridge design community in Indiana. The committee consisted of INDOT bridge engineers and research managers as well as a bridge design consultant from a private firm in Indianapolis, IN. The committee decided that the focus of the

STM computer program should be on the design of three typical bridge substructure components: multi-column bent caps, straddle bent caps, and integral and semi-integral end bent caps. Because these components are generally dominated by D-regions, the STM is typically the appropriate method for the design of these elements. The STM computer program is expected to significantly benefit bridge engineers in Indiana and allow engineers to complete STM designs quickly and accurately.

The STM computer program, titled STEP (Strut-and-Tie Evaluation Program), was created using Excel VBA. Due to the difference in the design procedures for pier caps and end bent caps, two different Excel workbooks have been created for the computer program. The first workbook, titled *Pier Cap STM Design*, is to be used for the design of multi-column bent caps and straddle bent caps. The second workbook, titled *End Bent Cap STM Design*, is to be used for the design of integral and semi-integral end bent caps. Engineers input geometric properties, material properties, reinforcement information, and a factored load case into STEP. This information is used to perform STM design steps, the results of which are output by the program. The engineer must interpret the outputs to ensure an adequate design is achieved.

The version of STEP described in this thesis is a beta version. Revisions to the program are expected based on input from engineers who will be given the opportunity to review the program. The details of the program described in this thesis, however, are expected to remain largely unchanged.

1.3 Organization

This document is organized with background information on the STM and the computer program presented first. Details of the computer program specific to specific bridge components follow. Engineers are encouraged to review the background information before using the computer program. The engineer should also be familiar with the details in the chapter specific to the bridge component being designed. A brief overview of each chapter is presented here for reference:

- Chapter 2: Introduction to the Strut-and-Tie Method

  This chapter serves as an introduction to the strut-and-tie method. The fundamental concepts of the STM are provided. Then, the general STM design procedure incorporated into the computer program is presented.

- Chapter 3: Overview of STM Computer Program

  An introduction to Excel VBA is presented in this chapter as well as a detailed overview of STEP. The layout and formatting of the computer program is included. All required user inputs are described, and the information output by the program is also explained.

- Chapter 4: Component 1 – Multi-Column Bent Caps

  The implementation of STEP to assist with the STM design of multi-column bent caps is presented in this chapter. The procedures used in the computer program are detailed, including a description of the VBA code that was written to perform STM design steps. Limitations of the computer program as well as assumptions included within the program are presented. The chapter ends with an example of the implementation of the computer program for the design of a specific multi-column bent cap.

- Chapter 5: Component 2 – Integral and Semi-Integral End Bent Caps

  The implementation of the program to perform the STM design procedure for integral and semi-integral end bent caps is presented. The techniques used by STEP to perform STM design steps are detailed, specifically focusing on procedures that differ from the procedures used for the design of pier caps. Limitations and assumptions related to the computer program are also included. The chapter ends with an example of the implementation of the computer program for the design of a specific integral end bent cap.

- Chapter 6: Summary

  A summary of the critical elements of the STM are presented along with closing comments regarding the computer program.

CHAPTER 2. INTRODUCTION TO THE STRUT-AND-TIE METHOD

2.1 Overview

This chapter serves as an introduction to the strut-and-tie method (STM) to ensure that designers are familiar with the underlying principles before using the computer program described in the chapters that follow. The theory behind the STM as well as past research and current code provisions are presented. The procedure used by the computer program for the STM analysis of bridge components is also described.

2.2 Past Research on the Strut-and-Tie Method

Multiple research projects have focused on the STM over the past two decades and have resulted in the current STM provisions in AASHTO LRFD (2017).

Texas Department of Transportation (TxDOT) Project 0-5253 titled *D-Region Strength and Serviceability Design* included 37 tests on deep beam specimens. These 37 specimens along with 142 deep beam tests from the literature were analyzed to modify AASHTO LRFD provisions for the STM. The research project focused on eight tasks related to strength and serviceability including distribution of stirrups, triaxial confinement for CCC and CCT nodes (defined in Section 2.11.1), member depth, and limitation of diagonal cracking under service loads. The research program is detailed in Birrcher et al. (2009).

TxDOT Project 5-5253 titled *Strut-and-Tie Model Design Examples for Bridges* included the development of five examples to serve as a reference for designers using the STM for bridge components. The design examples are provided in Williams et al. (2012) along with a detailed description of the STM. The examples include the design of the following components:

- Five-column Bent Cap
- Cantilever Bent Cap
- Inverted-T Straddle Bent Cap: Moment Frame Condition
- Inverted-T Straddle Bent Cap: Simply Supported Condition
- Drilled-Shaft Footing

Williams et al. (2012) is referenced extensively throughout this document. The five-column bent cap example was used as a main reference for the development of STEP. In addition, Chapter 4 uses the same bent cap example to demonstrate how to develop a strut-and-tie model using the computer program.

Furthermore, TxDOT Project 0-6416 titled *Shear Cracking in Inverted-T Straddle Bents* included an experimental program on deep inverted-T specimens. The research results are detailed in Larson et al. (2013) and also influenced the current STM provisions in AASHTO LRFD (2017).

2.3 Disturbed Regions

D-regions, also referred to as disturbed regions or discontinuity regions, occur near applied loads or geometric discontinuities. B-regions, also called Bernoulli or beam regions, occur outside D-regions along a member. Within B-regions, the distribution of strains is linear and typical sectional analysis can be used to design a member. However, within D-regions, the distribution of strains is nonlinear and the assumption that plane sections remain plane is no longer valid. St. Venant's Principle states that linear strains can be assumed at approximately a member depth away from a load or geometric discontinuity. Therefore, a D-region can be assumed to exist within a member depth (or an effective depth, d) away from each applied load or geometric discontinuity, and the transition to a B-region occurs a member depth away from each applied load or geometric discontinuity. The division of a member into B-regions and D-regions is illustrated in Figure 2.1. Shear spans that contain B-regions can be designed using sectional methods. Shear spans that contain only D-regions are dominated by D-region behavior, and the STM analysis is appropriate in these regions. The STM is a powerful design tool that appropriately captures D-region behavior and can therefore be used to design D-regions of reinforced concrete members.

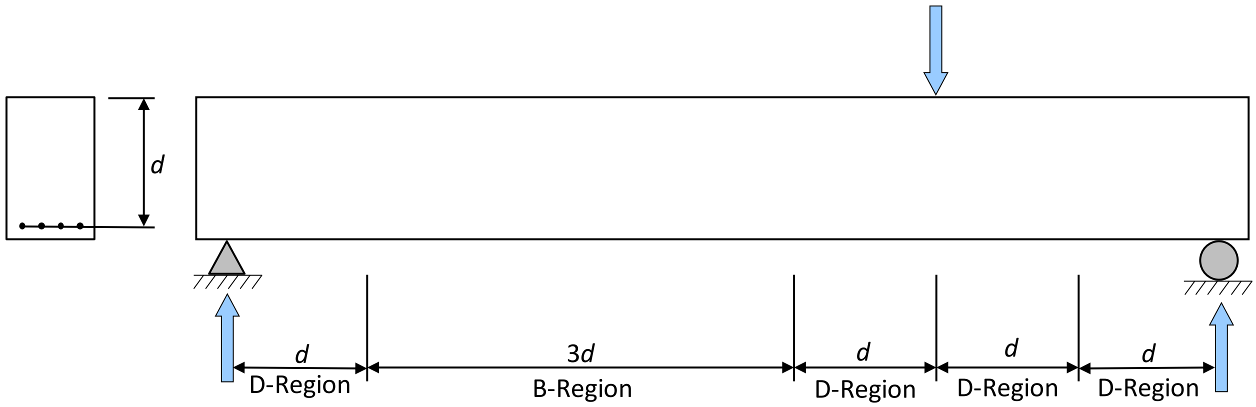


Figure 2.1: Division of Member into B-regions and D-regions Based on St. Venant's Principle (adapted from Williams et al., 2012)

2.4 Fundamentals of the Strut-and-Tie Method

The STM uses strut-and-tie models to perform strength checks of a structural component. A strut-and-tie model is an idealized truss model that represents the flow of forces within a member. The model is comprised of axial tension and compression members. All strut-and-tie models contain three components: struts, ties, and nodes as shown in Figure 2.2. Struts and ties are analogous to the compression and tension members in a truss, respectively. Within a strut-and-tie model, struts are axial compression members that represent compressive stress fields in the concrete. Ties are axial tension members that represent tensile stresses carried by steel reinforcing bars. Nodes are at the intersections of struts and ties and are typically the regions of highest stress in a component. The struts and ties model the flow of forces in a member. Struts are represented by dashed green lines and ties are represented by solid red lines throughout this document.

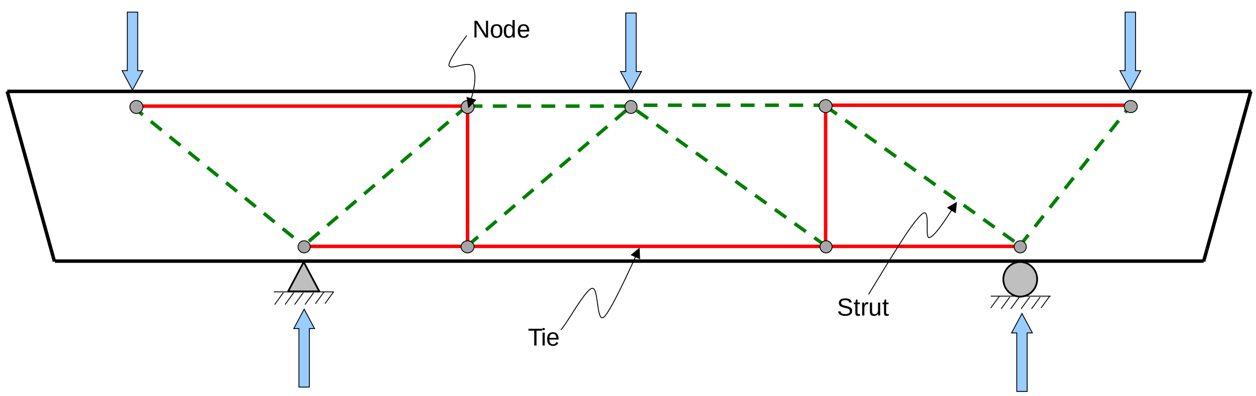


Figure 2.2: Example Strut-and-Tie Model for a Simple Beam (from Williams et al., 2012)

The STM is a lower-bound design approach. Therefore, if the following three conditions are met, a model developed using the STM will be conservative (Birrcher et al., 2009):

1) The model is in equilibrium with external forces.
2) The concrete member has adequate deformation capacity for internal stresses to redistribute into the members of the assumed strut-and-tie model.
3) The strength conditions are met.

As a result, there can be multiple models that are conservative and accurate for a given structure. This can lead to apprehension from engineers when using the STM in design. The computer program uses a series of rules that satisfy structural principles to develop a model that meets the conditions required for a conservative design.

2.5 Design Procedure

STEP, the computer program described in this document, is intended to be a tool to aid in the structural design of bridge components using the STM. For a general procedure not specific to the components and steps used in the computer program, refer to Section 2.3.3 of Williams et al. (2012). The design steps listed below are described in detail in the sections that follow. The first two steps are performed by the engineer prior to using the program. Information from these steps is required as user inputs within the program. The remaining steps are used within the computer program to perform STM design procedures for the structural component. The steps are as follows:

1. *Define load cases* - calculate factored loads acting on the component; all applied loads must be point loads
2. *Size structural component and determine initial reinforcement layout* - determine an initial geometry; perform initial reinforcement design
3. *Analyze structural component* - use continuous beam analysis or other analysis procedure (e.g., pile group analysis) to solve for reactions and develop the shear and moment diagram for the member
4. *Develop Strut-and-Tie Model* - determine locations of nodes and members; then, analyze model to determine member forces
5. *Proportion Ties* - ensure the longitudinal reinforcement provided is sufficient to carry the forces in the longitudinal ties and determine spacing of stirrups based on vertical ties

6. *Check Nodal Strengths* - determine geometries of nodes (except smeared nodes) and perform strength checks for each nodal face
7. *Proportion Crack Control Reinforcement* - specify spacing of reinforcement in both the longitudinal and transverse directions based on bar sizes input by the user
8. *Check Anchorage for Ties* - ensure reinforcement is anchored at the nodal regions located near the ends of the member

2.6 Define Load Cases

The designer is required to input loading conditions into the program for it to perform any STM design procedures. Before using the computer program, the designer should first determine the critical load cases for the member. Each critical load case must be assessed separately in the computer program because a strut-and-tie model with unique member forces results from each load case. Similar to a truss, applied loads acting on a strut-and-tie model can only act at nodes. Therefore, all applied forces and moments acting on the structural component must be transformed into point loads. For example, a uniformly distributed load must be treated as an equivalent series of point loads. Moreover, an applied moment can be replaced by a force couple. Other modifications to loads may be required, such as combining point loads that act very close together (see Section 4.2.2 of Williams et al. (2012)). The decision whether to combine point loads acting close together is left up to the discretion of the designer. These modifications must be performed before entering information into the program.

The user also inputs the load factor to be used for self-weight for the given load case. The program will then automatically distribute this load as point loads. At each point load acting on the component, the corresponding tributary self-weight is factored using the self-weight factor input by the user. This factored self-weight force is then added to the applied point load value at that location. Refer to Sections 3.3.2 and 4.2.4.1 for more information on how the computer program handles self-weight. The option to input self-weight values manually into the program is also described in these sections.

2.7 Size Structural Component and Determine Initial Reinforcement Layout

The designer must input the geometry of the component being designed into the computer program. Selecting an initial member size based on the shear serviceability check developed by Birrcher et al. (2009) and described in Section 2.7 of Williams et al. (2012) is a recommended procedure. Using this check serves to minimize the risk of diagonal crack formation in the member under service loads. Details of the longitudinal reinforcement layout is also input by the user. The initial longitudinal reinforcement layout can be decided based on maximum positive and negative moments along the member. Although sectional design methods should not be used for a final design of a D-region, conventional flexural design procedures can be used to determine the initial reinforcement layout prior to the STM design of the structural component.

2.8 Analyze Structural Component

The remaining steps in the procedure are performed by the computer program based on user inputs. When the user runs the computer program, the program begins by analyzing the structural component to solve for the support reactions. If the engineer is designing an integral or semi-integral end bent, he or she has the option to input pile reactions, and the computer will not perform the continuous beam analysis. Reactions can be calculated using a pile group analysis in this case. The program uses the moment distribution method to solve for reaction values if they are not provided. More details on this methodology are included in Section 5.2.4.

2.9 Develop Strut-and-Tie Model

After calculating the support reactions, the program develops a strut-and-tie model for the component. The development of a strut-and-tie model typically occurs in two steps. The first step is to determine the geometry of the strut-and-tie model based on the locations of applied loads and support reactions. Next, the model is analyzed to determine the forces in the struts and ties.

General principles should be followed when determining the geometry of the strut-and-tie model. Placement of struts and ties should be representative of the elastic flow of forces in the component. Struts are located along compressive stress fields in the member, and ties are located along tensile stress fields. The designer should note that the centroid of the reinforcing bars carrying the forces in the ties of a strut-and-tie model must coincide with the exact location of the ties in the model. When developing a strut-and-tie model outside of the computer program, an engineer can use a finite element analysis to determine where struts and ties should be located. The engineer can also use known cracking patterns to determine locations of struts and ties in a member. The most common method for developing a strut-and-tie model is to follow a logical load path. The program utilizes a set of rules to determine the logical load path and the locations of nodes, struts, and ties in the model.

The location of the top and bottom chords is based on the location of the reinforcement in the component. If there is no negative moment region along the length of the component, the top chord location can be determined based on the depth of the rectangular compression stress block, a, or based on optimizing the height of the strut-and-tie model. The computer program optimizes the strut-and-tie model to determine the location of the top chord when only positive moment exists along the member length. Figure 2.3 depicts how optimizing the moment arm, jd (i.e., the height of the strut-and-tie model), can lead to a more efficient model. If the moment arm is too small, the forces in the top and bottom chord will be larger and will result in more steel being needed to satisfy design checks. If the moment arm is too large, the back face of the nodes along the top chord will be too small and will not have adequate strength. When the moment arm is optimized, the depth of the component is used efficiently (Williams et al., 2012).

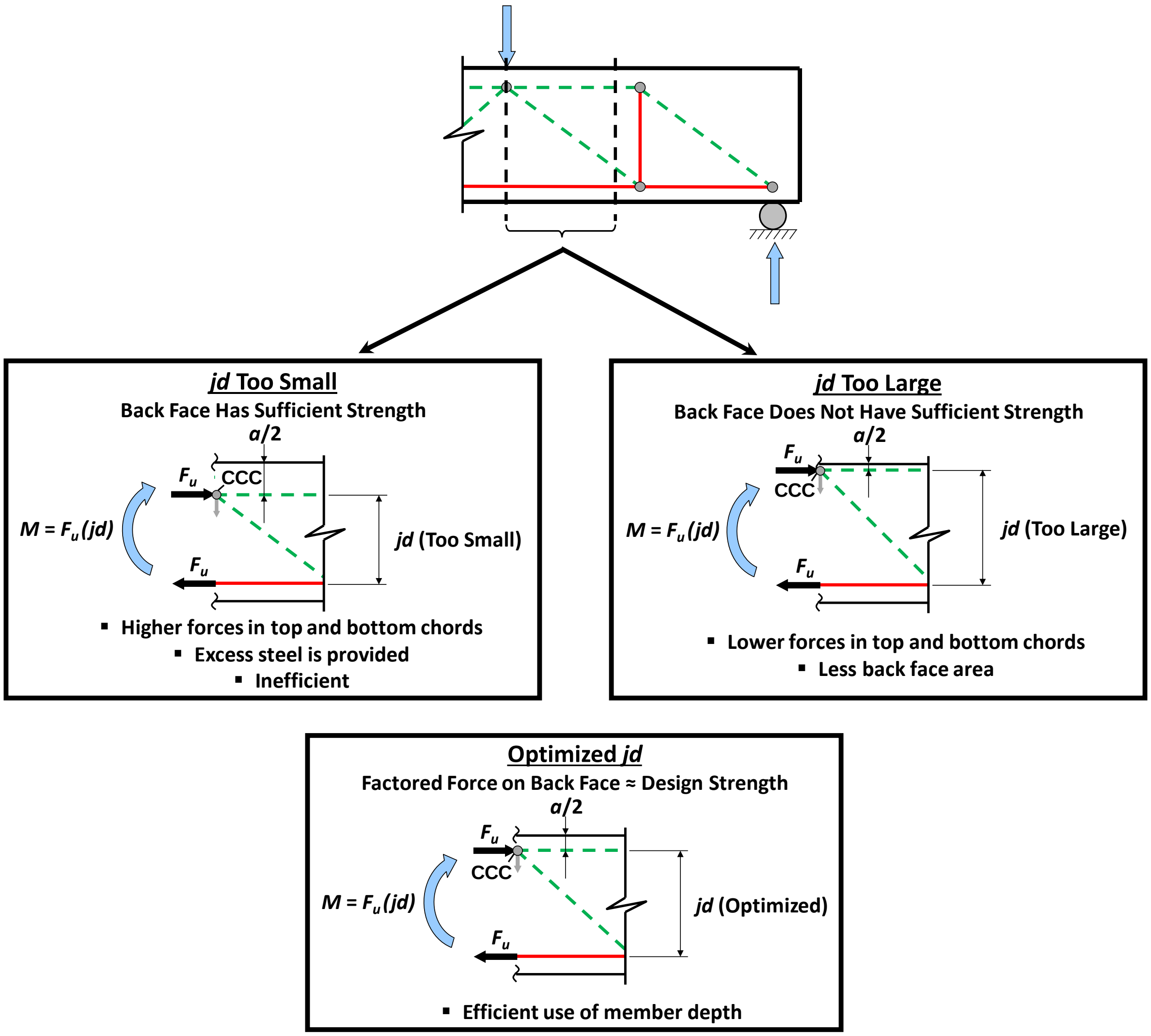


Figure 2.3: Optimization of Moment Arm to Determine Location of Top Chord (adapted from Williams et al., 2012)

Any angle between a strut and a tie entering the same node should not be less than 25° when developing a strut-and-tie model. This angle limit is important for limiting crack openings and preventing excessive strain in the steel (AASHTO LRFD, 2017). When a diagonal strut between an applied load and a reaction will create an angle with a horizontal tie that is less than 25°, additional nodes and vertical ties are required. Using the least number of truss panels is recommended to achieve the most efficient strut-and-tie model, as shown in Figure 2.4. Less reinforcement is required when the minimum number of vertical ties is used, and the resulting design is still safe.

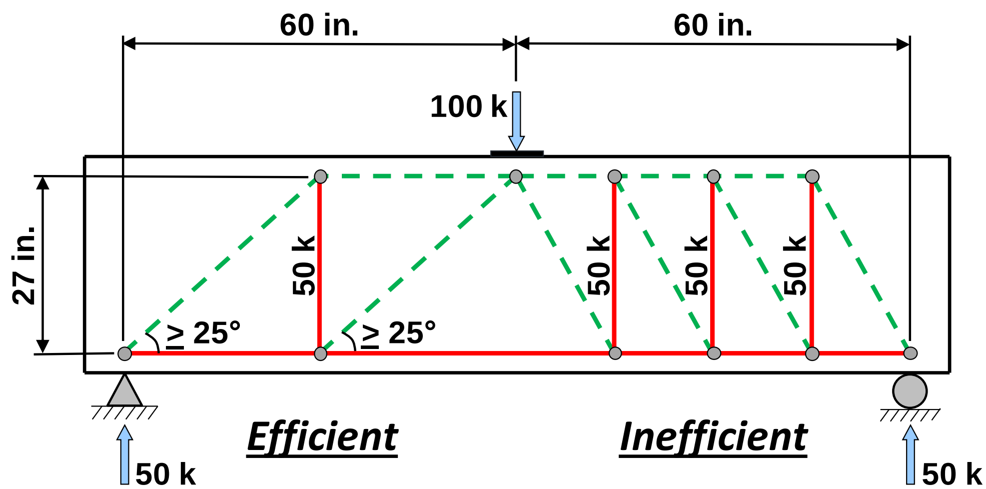


Figure 2.4: Using the Least Number of Truss Panels Possible (from Williams et al., 2012)

When developing a strut-and-tie model for a beam supported on pin supports, simple principles can be considered that will aid a designer with creating a valid model. A beam supported on pin supports is the main focus of this computer program. Thus, simple principles were used by the computer program to develop a strut-and tie model. These principles, shown in Figure 2.5, include the following:

(1) The top and bottom chord are connected by horizontal struts and ties.

(2) Vertical struts exist where a load is directly above a reaction.

(3) Vertical ties exist at loads or reactions where the shear does not change signs.

(4) Diagonal struts will exist between each load point and the bottom chord.

(5) When the shear changes signs at a load or reaction, diagonal struts originating at the node will extend in both directions.

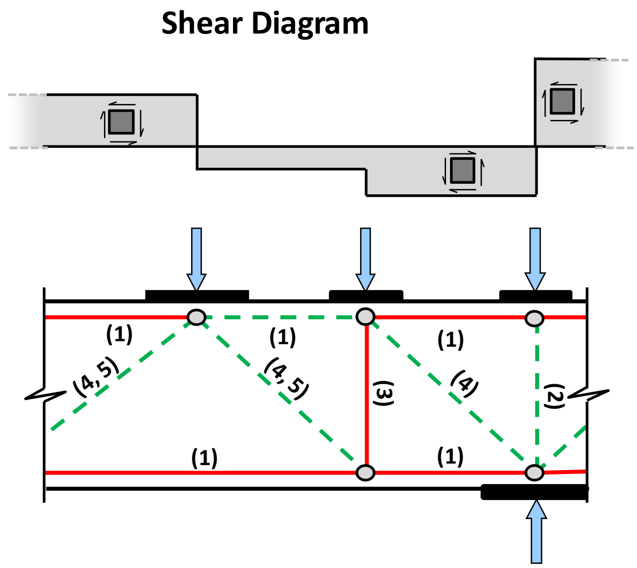


Figure 2.5: Principles Used to Determine Members Comprising a Strut-and-Tie Model

After the strut-and-tie model is developed, the truss is analyzed using statics to determine forces in the members. The computer program uses the method of joints to solve for forces in each member. All members are assumed to be in tension when solving. After solving, positive forces represent ties and negative forces represent struts.

2.10 Proportion Ties

After developing a strut-and-tie model for the component and determining the member forces, the next step is to proportion ties. This step serves to ensure that the area of reinforcement in the component is adequate for resisting the tensile forces in the ties. To calculate the required area of reinforcement for a tie, Equation (2.1) is used:

$$A_{st} = \frac{P_u}{\phi f_y} \tag{2.1}$$

where:

A_{st} = required area of steel (in.^{2})

P_u = tensile force in the tie (kip)

ϕ = resistance factor (=0.9 according to Article 5.5.4.2 of AASHTO LRFD (2017))

f_y = yield strength of the reinforcement (ksi)

The computer program checks to ensure that the area of longitudinal reinforcement specified by the engineer is adequate to carry the forces in the corresponding ties. For longitudinal ties, the program determines if Equation (2.2) is satisfied using the longitudinal reinforcement information entered by the designer:

$$P_u \leq \phi A_{st} f_y \tag{2.2}$$

The stirrups in the member are used to carry the tensile forces represented by the vertical ties in the strut-and-tie model. The user is asked to input the stirrup bar size and the number of stirrup legs provided through the width of the member. Using this information, the computer program determines the stirrup spacing that is required to ensure the stirrups are able to carry the vertical tie forces. To determine this maximum spacing for each vertical tie, the width of the tie, w_t, is assumed to be equal to the smaller length of the two adjacent truss panels, illustrated for Tie AB in Figure 2.6. For Tie AB, w_1 is smaller than w_2 and therefore $w_t = w_1$. This width is centered on the tie because the stirrups considered to carry the force in the tie must be centered on the vertical tie in the model.

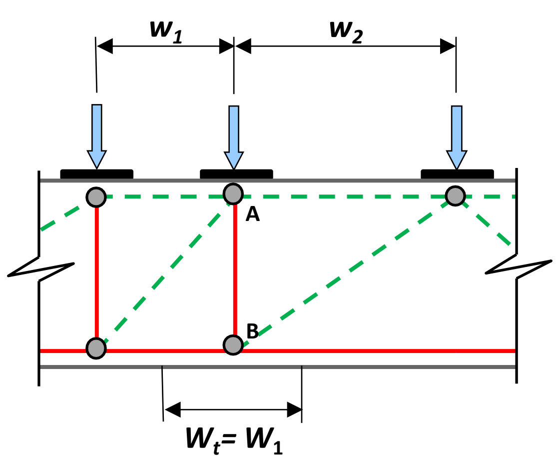


Figure 2.6: Width of Tie for Determining Required Stirrup Spacing

The area of steel required by Equation (2.1) must be provided within w_t. The area of required steel is divided by the area of transverse reinforcement specified by the designer (i.e., the area of each stirrup leg multiplied by the number of legs through the member width) to determine the number of stirrups needed within the width of the tie. The required stirrup spacing is calculated by dividing the width of the tie by the number of stirrups that are needed. Equation (2.3) is used in the program to calculate this spacing:

$$spacing \leq \frac{\phi A_v * f_y * w_t}{P_u} \tag{2.3}$$

where A_v is the transverse area of reinforcement specified by the designer.

2.11 Check Nodal Strengths

After proportioning ties, the strength of nodes is checked to ensure that the loads acting at the nodes do not exceed code-defined factored resistances. Nodes are the regions in a component where, in general, the highest concentration of stresses occur because struts and ties meet at nodes, creating a bottleneck of stresses (Schlaich et al., 1987).

Nodes can be proportioned as either hydrostatic nodes or non-hydrostatic nodes. Hydrostatic nodes are proportioned so that the stress on each face of the node is equivalent. Non-hydrostatic nodes are proportioned according to the origin of the applied stress (AASHTO LRFD, 2017). The difference in proportioning the two types of nodes is illustrated in Figure 2.7. The STM provisions of AASHTO LRFD (2017) assume that nodes are non-hydrostatic and define nodal geometries to correspond to the actual stress distribution at the nodal region. The commentary to Article 5.8.2.1 states that "hydrostatic nodes can sometimes result in unrealistic nodal geometries and impractical reinforcement layouts…Thus, nonhydrostatic nodes are preferred in design" (AASHTO LRFD, 2017). The computer program aligns with AASHTO LRFD provisions and considers all nodes to be non-hydrostatic when defining geometries.

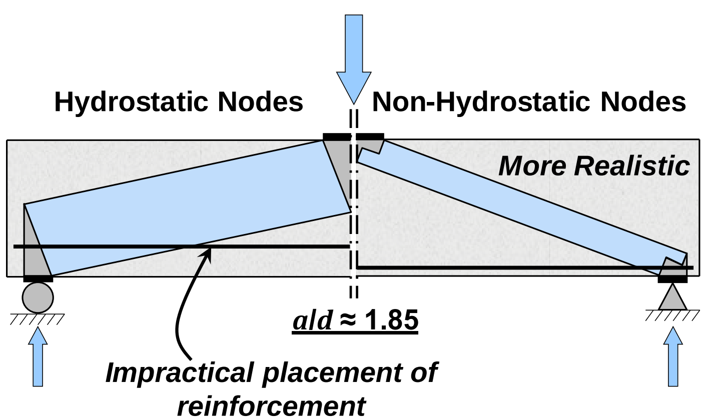


Figure 2.7: Hydrostatic vs. Non-Hydrostatic Nodes (from Williams et al., 2012)

2.11.1 Types of Nodes

There are three types of nodes that are found in strut-and-tie models: CCC, CCT, and CTT, where "C" represents compression members (struts) and "T" represents tension members (ties). The description of each type of node as defined in Article 5.8.2.2 of AASHTO LRFD (2017) is provided below:

CCC: nodes where only struts intersect

CCT: nodes where ties intersect in only one direction

CTT: nodes where ties intersect in two different directions

Each type of node is indicated within a strut-and-tie model in Figure 2.8.

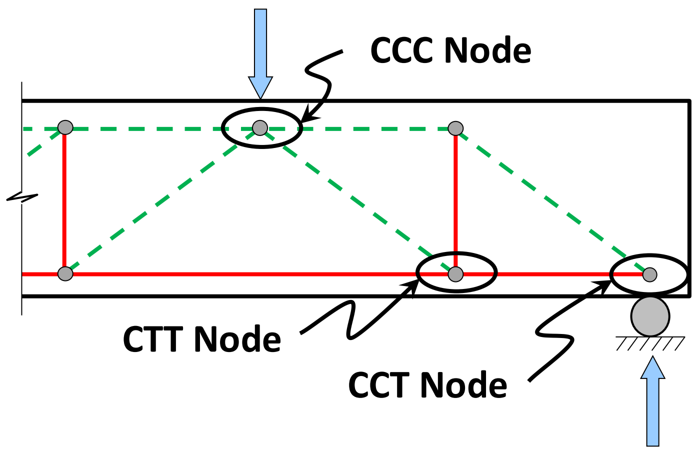


Figure 2.8: Types of Nodes in a Strut-and-Tie Model (adapted from Williams et al., 2012)

2.11.2 Proportioning CCC Nodes

In order to determine the geometry of a CCC node, steps are first taken to simplify the node. This process is depicted in Figure 2.9 for the CCC node in Figure 2.8. Struts intersecting a node from the same direction are typically combined together to simplify nodal geometries and nodal strength checks. The CCC node in Figure 2.8 has two struts on each side of the node. The two struts on the left are combined into one strut as are the two struts on the right.

CCC nodes should be subdivided when struts enter from both sides of the node. The CCC node in Figure 2.8 has struts entering from both sides of the node, and is subdivided into two parts. The applied load is also subdivided. The right portion of the applied load, αP, is equal to $F_R \sin(\theta_2)$. When the node is subdivided into two parts, the angle at which the combined diagonal strut enters the node changes. After combining struts and subdividing the node, the geometry of each subdivided node can be determined. Article C5.8.2.2 of AASHTO LRFD (2017) depicts the process of combining struts and subdividing nodes.

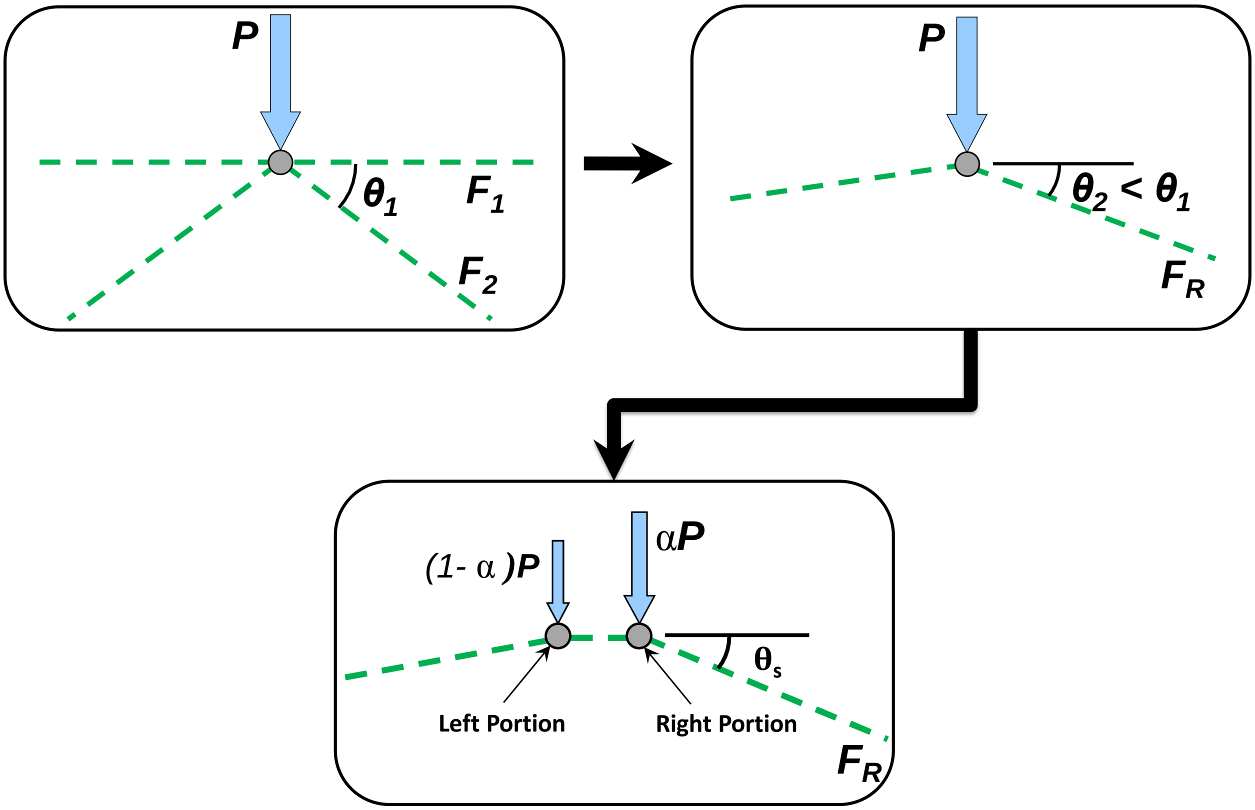


Figure 2.9: CCC Node: Process of Combining Struts and Subdividing Nodes into Two Parts (adapted from Williams et al., 2012)

If a vertical strut also intersects the node in addition to the diagonal struts entering from both sides of the node, the node is subdivided into three parts, as shown in Figure 2.10. In this figure, the struts entering from the left side of the node are first combined into one strut, as are the struts entering the node from the right. Then, the node is subdivided into three parts. The vertical strut will also have a slight change in angle. However, this change in angle is neglected in the figure. The angles at which the diagonal struts enter the node are updated based on the locations of the subdivided nodes.

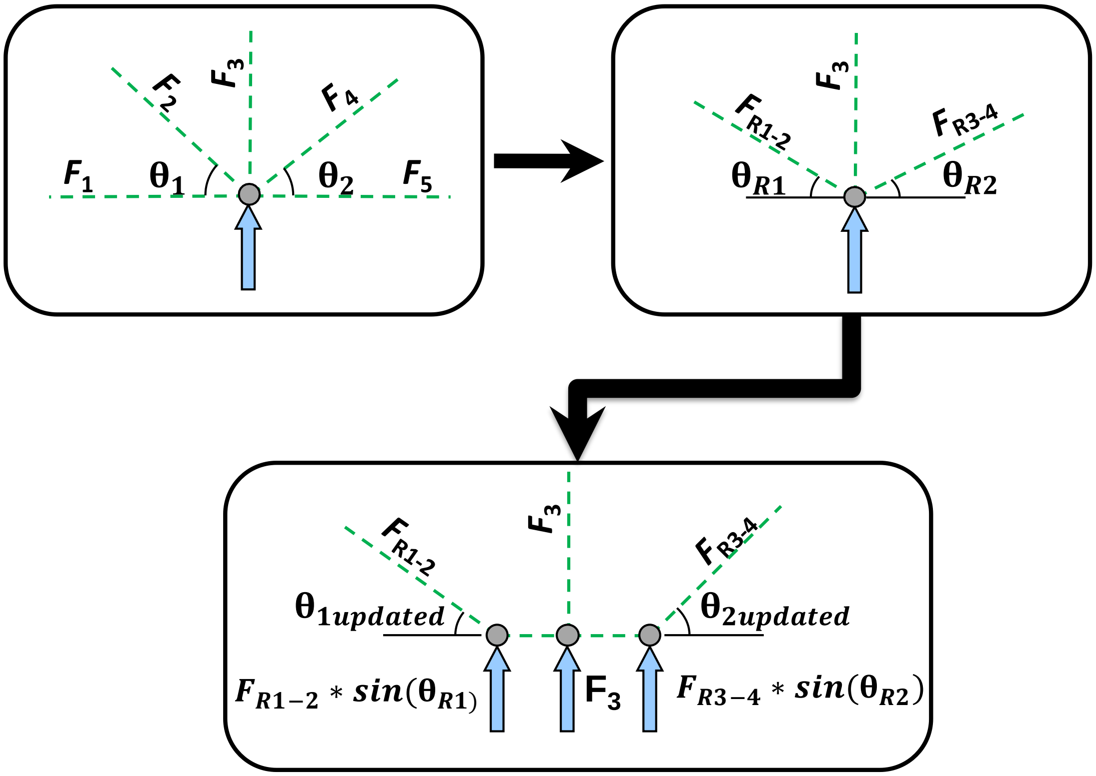


Figure 2.10: CCC Node: Process of Combining Struts and Subdividing Nodes into Three Parts (adapted from Williams et al., 2012)

For each node, three faces are considered for strength checks. These are the bearing face, back face, and strut-to-node interface. A detailed depiction of the geometry of the CCC node labeled in Figure 2.8 is provided in Figure 2.11.

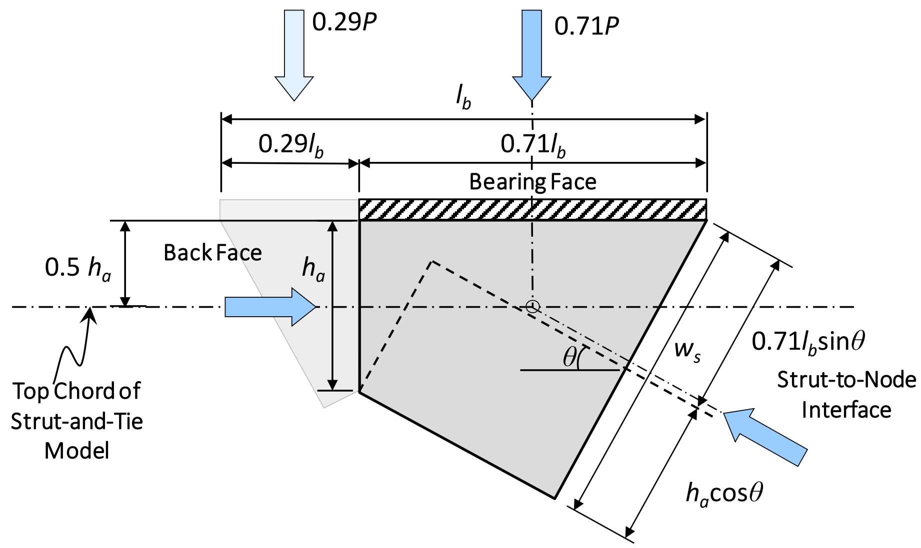


Figure 2.11: Geometry of a CCC Node (adapted from Williams et al., 2012)

The length of the back face of a CCC node, h_a, is twice the distance from the surface of the member to the longitudinal chord of the strut-and-tie model, as seen in Figure 2.11. The length of the strut-to-node interface, w_s, is found using Equation (2.4):

$$w_s = l_b sin\theta + h_a cos\theta \qquad (2.4)$$

where:

l_b = length of the bearing face (in.)

θ = angle between the diagonal strut and the longitudinal axis of the member

h_a = length of the back face (in.)

For subdivided nodes, the length of the bearing face, l_b, in the equation is taken as the bearing face of the subdivided node. For the right side of the node in Figure 2.11, the length of the bearing face is αl_b, where α is equal to 0.71. Correspondingly, the length of the bearing face of the left side is $(1-\alpha)l_b$. These values should be used as l_b in Equation (2.4) to find the length of the strut-to-node interface for each subdivided node.

2.11.3 Proportioning CCT Nodes

The detailed geometry of the CCT node in Figure 2.8 is illustrated in Figure 2.12. CCT nodes are proportioned in a similar method as CCC nodes. As with CCC nodes, if multiple struts intersect a

CCT node from the same direction, these struts should be combined together before proportioning the node. For the CCT node in Figure 2.12, l_b is taken as the width of the bearing plate. The back face of the node, h_a, is calculated in the same manner as for CCC nodes. Because the node depicted is along the bottom chord, the back face length, h_a, is two times the distance from the bottom of the component to the bottom chord of the strut-and-tie model. The bottom chord of the truss model is placed at the centroid of the longitudinal reinforcement in the component ($0.5h_a$ from the bottom surface of the member). The width of the strut-to-node interface is calculated using Equation (2.4).

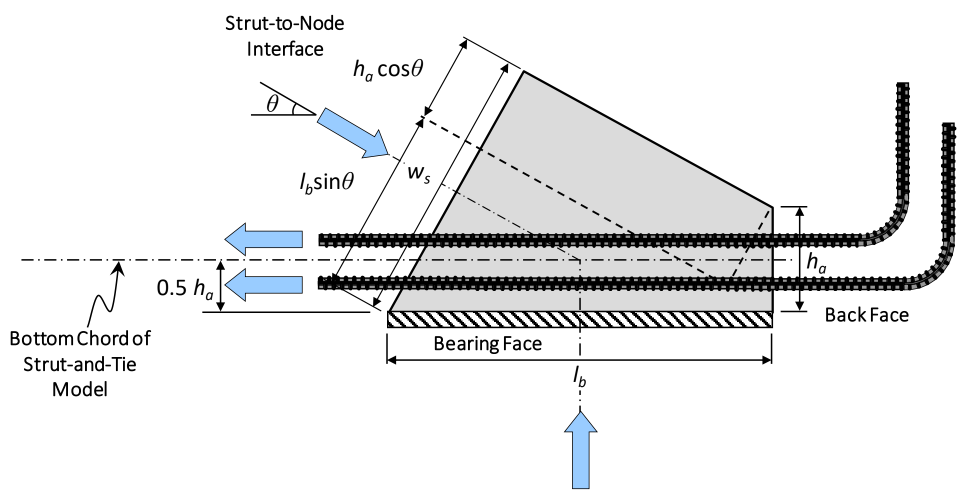


Figure 2.12: Geometry of a CCT Node (adapted from Williams et al,. 2012)

2.11.4 Proportioning CTT Nodes

Many CTT nodes, including the CTT node labeled in Figure 2.8, are smeared nodes. The geometry of a smeared node cannot be easily defined. Because the smeared node in Figure 2.8 does not abut a load or bearing plate, the diagonal strut entering the node is not restricted to the area of the strut-to-node interface that is dependent on a length l_b, resulting in the force from the strut spreading over a large area of concrete. According to Article C5.8.2.2 of AASHTO LRFD (2017), these nodes are not considered critical and strength checks are unnecessary.

In some cases, however, CTT nodes are not smeared nodes. For this scenario, the node is proportioned following the same procedures described for a CCT node. Figure 2.13 shows a proportioned CTT node.

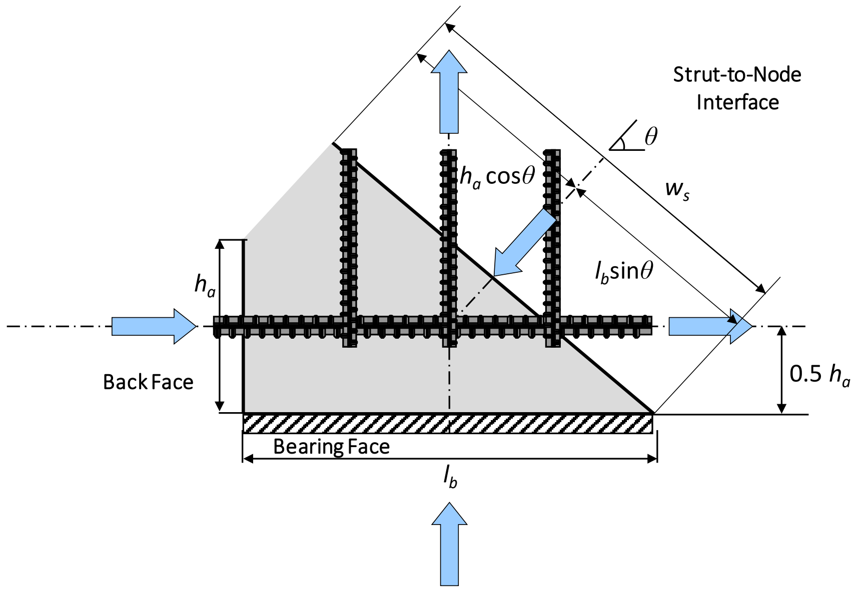


Figure 2.13: Geometry of a CTT Node (adapted from Williams et al., 2012)

2.11.5 Nodal Strength Checks

After the geometry of a node is defined, the strength of the node is checked. The factored resistance of the node is calculated and then compared to the demand. Each face is checked separately. If the node does not have adequate strength, the designer must make modifications to the structural component. Possible modifications include increasing the concrete compressive strength, increasing bearing areas, and/or changing the member geometry. If the geometry of the member changes, the strut-and-tie model is no longer valid, and a new model must be developed.

The strength of a node is calculated by the following three steps: 1) calculate the confinement modification factor, 2) determine the efficiency factor, and 3) calculate the factored resistance of each nodal face (Williams et al., 2012). Each of these steps is explained in detail below.

If the width of the bearing area at a node is smaller than the width of the member, the confinement modification factor, *m*, is calculated. The confinement modification factor is defined in Article 5.6.5 of AASHTO LRFD (2017). The equation for calculating *m* is as follows:

$$m = \sqrt{\frac{A_2}{A_1}} \leq 2.0 \tag{2.5}$$

The areas A_1 and A_2 are defined in Figure 2.14, where A_1 is the loaded area and A_2 is measured on the plane below the loaded area defined by the location at which a line with a slope of 1 vertical to 2 horizontal meets the edge of the member.

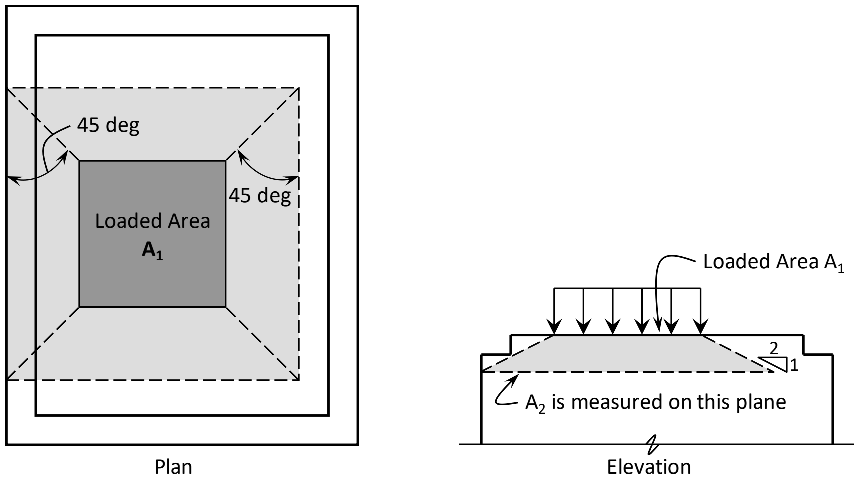


Figure 2.14: Bearing Area Definitions for Confinement Modification Factor (adapted from Williams et al., 2012, and AASHTO LRFD, 2017)

The second step for calculating nodal strengths is to determine the concrete efficiency factor, ν, for each face. The concrete efficiency factor depends on the type of node as well as the face being considered. Efficiency factors are given in Table 5.8.2.5.3a-1 of AASHTO LRFD (2017) and are also listed in Table 2.1 for reference. This table only applies to bridge components that satisfy the crack control reinforcement requirements of Article 5.8.2.6. If these provisions are not satisfied, the efficiency factor is taken as 0.45. The efficiency factors associated with each face of the three different types of nodes when adequate crack control reinforcement is provided are illustrated in Figure 2.15.

Table 2.1: Concrete Efficiency Factors, ν (from AASHTO LRFD, 2017)

	Node Type		
Face	**CCC**	**CCT**	**CTT**
Bearing Face	0.85	0.7	$0.85 - \frac{f_c'}{20\ ksi}$ $0.45 \leq \nu \leq 0.65$
Back Face			
Strut-to-Node Interface	$0.85 - \frac{f_c'}{20\ ksi}$ $0.45 \leq \nu \leq 0.65$	$0.85 - \frac{f_c'}{20\ ksi}$ $0.45 \leq \nu \leq 0.65$	

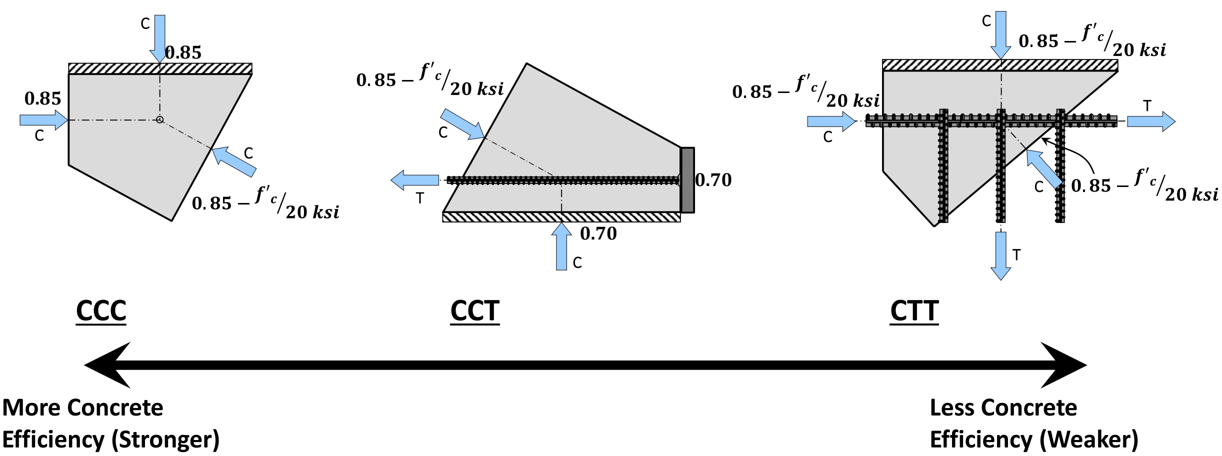


Figure 2.15: Concrete Efficiency Factors for Different Node Types (from Williams et al., 2012)

The last step in determining nodal strengths is to calculate the factored resistance of the node at each face using Equation (2.6)*(2.11)*:

$$\phi P_n = \phi * f_{cu} * A_{cn} \tag{2.6}$$

where:

ϕ = resistance factor (= 0.70 according to Article 5.5.4.2 of AASHTO LRFD (2017))

f_{cu} = limiting compressive stress at the face of the node (ksi)

A_{cn} = effective cross-sectional area of the nodal face (in.^{2})

The effective cross-sectional area of the nodal face, A_{cn}, is calculated as the length of the nodal face determined in Section 2.11.2 to 2.11.4 multiplied by the perpendicular width of the node. If the node abuts a load or bearing plate, the width of the node is equal to the width of the plate. The limiting compressive stress, f_{cu}, is calculated using Equation (2.7):

$$f_{cu} = m * \nu * f_c' \qquad (2.7)$$

where:

$m =$ effective confinement modification factor

$\nu =$ concrete efficiency factor

$f_c' =$ specified compressive strength of the concrete (ksi)

Article 5.8.2.5.1 of AASHTO LRFD (2017) states that “[W]here the back face of a CCC node contains nonprestressed compressive reinforcement the resistance…may be augmented with the yield resistance of the nonprestressed reinforcement.” If the designer wishes to include the effect of compressive reinforcement, Equation (2.8) can be used to find the design strength of the nodal face:

$$\phi P_n = \phi(f_{cu} * A_{cn} + f_y * A_{sn}) \qquad (2.8)$$

where f_y is the yield strength of the reinforcement and A_{sn} is the area of reinforcement entering the back face of the node.

If there is no direct compressive force on the back face of a node, the strength of the back face does not need to be checked (see Article 5.8.2.5.1 of AASHTO LRFD (2017)). Otherwise, a strength check must be performed on all faces of each node that is not a smeared node. In order for the design to be safe, the condition in Equation (2.9) must be met.

$$\phi P_n \geq P_u \qquad (2.9)$$

where P_u is the force acting at the face of the node.

2.12 Proportion Crack Control Reinforcement

When using the STM, crack control reinforcement is required for all components to use the efficiency factors in Table 2.1. The reinforcement controls the width of cracks in the concrete and ensures ductility of the member (see Article C5.8.2.6 of AASHTO LRFD (2017)). Crack control reinforcement assists in redistributing stresses internally within D-regions. Article C5.8.2.6 explains that during previous experimental research, “the width of the first diagonal crack forming in a deep beam was unacceptably large (i.e., greater than 0.016”) if crack control reinforcement provided in that specimen was less than 0.003 times the effective area of the strut” (AASHTO LRFD, 2017).

According to Article 5.8.2.6 of AASHTO LRFD (2017), the reinforcement in the vertical and horizontal directions must satisfy the conditions in Equations (2.10) and (2.11) respectively:

$$\frac{A_v}{b_w s_v} \geq 0.003 \tag{2.10}$$

$$\frac{A_h}{b_w s_h} \geq 0.003 \tag{2.11}$$

where:

A_v = area of vertical crack control reinforcement within spacing s_v (in.^{2})

A_h = area of horizontal crack control reinforcement within spacing s_v (in.^{2})

b_w = width of component’s web (in.)

s_v, s_h = spacing of vertical and horizontal crack control reinforcement respectively (in.)

The spacing of the bars must not be greater than d/4 or 12.0 in. The variables in Equations (2.10) and (2.11) are illustrated in Figure 2.16. The computer program specifies a maximum spacing for horizontal and vertical crack control reinforcement that satisfies Equations (2.10) and (2.11) based on the reinforcement information input by the user.

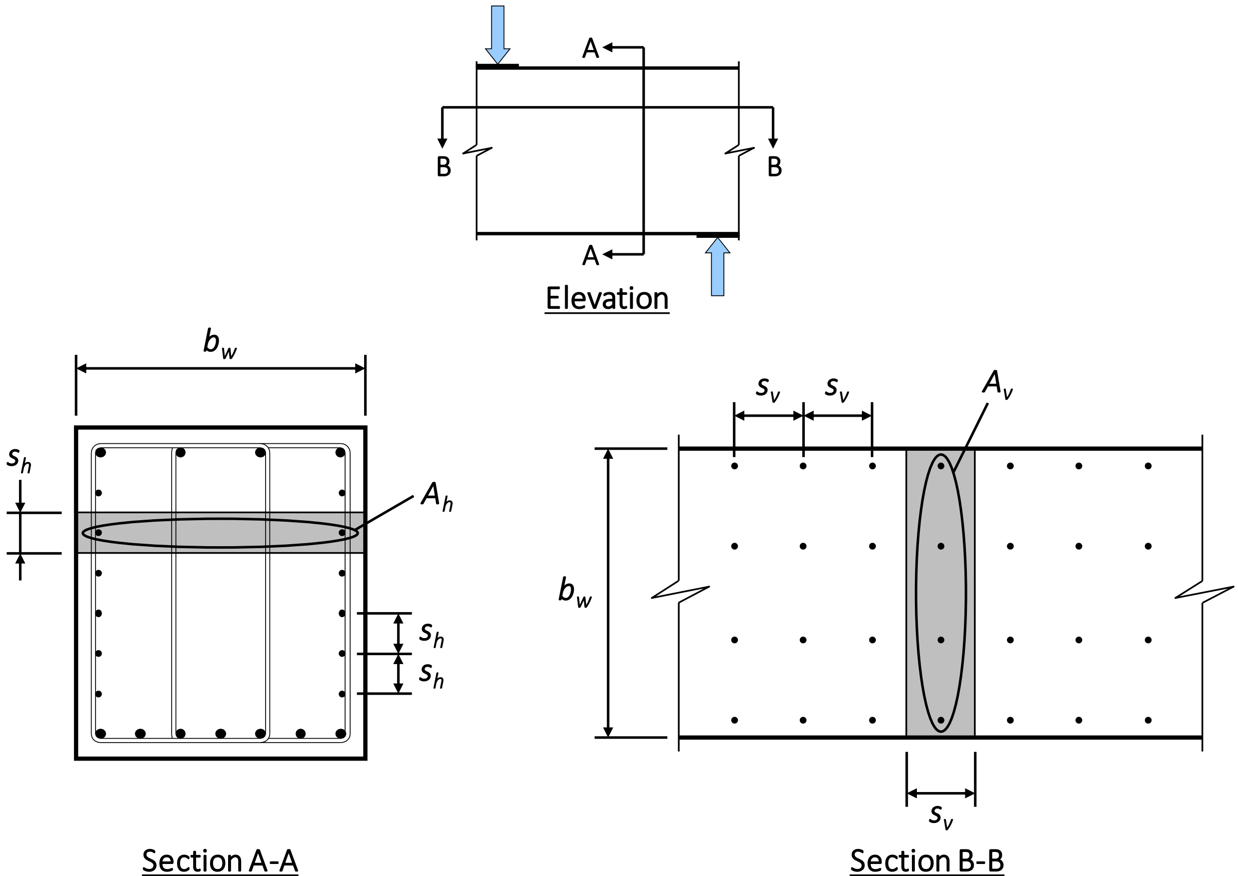


Figure 2.16: Crack Control Reinforcement (from Williams et al., 2012)

2.13 Provide Anchorage for Ties

Ties must be properly anchored to ensure that they can be fully developed to resist the forces in the ties of the strut-and-tie model. In order for a tie to be properly anchored at a node, the reinforcement must be fully developed at the point that the centroid of the reinforcement exits the extended nodal zone, as shown in Figure 2.17. Within the computer program, the designer inputs the required development length of straight and/or hooked bars used as longitudinal reinforcement. The program determines the length that is available for development of the reinforcement and compares it to the required development length(s) input by the user.

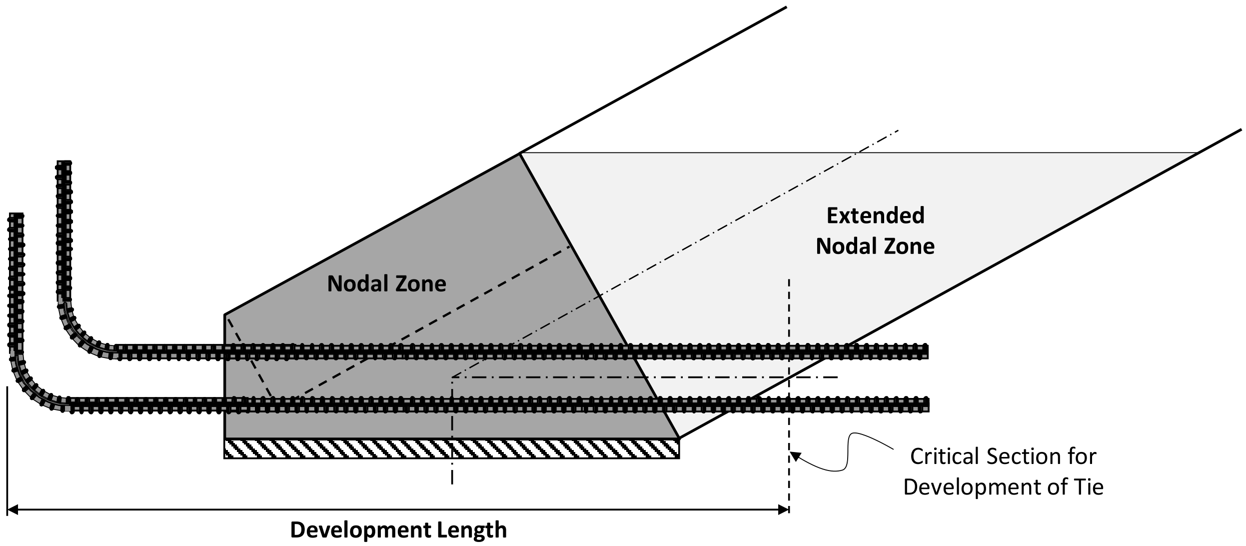


Figure 2.17: Anchorage of Ties (from Williams et al., 2012; adapted from Birrcher et al., 2009)

2.14 Summary

The STM is a powerful design tool used for the design of D-regions in reinforced concrete members. D-regions occur within a member depth away from load points or geometric discontinuities. In these regions, linear strains cannot be assumed, and traditional sectional analysis does not apply. Each strut-and-tie model contains three components: nodes, struts, and ties.

The first step in designing a component using the STM is to define a governing load case. Next, the structural component is sized, which can be accomplished by using a shear serviceability check, and an initial design of the reinforcement is performed. After these steps are completed, the strut-and-tie model is developed and analyzed. Once the strut-and-tie model has been established, ties are proportioned, and the strength of nodes are checked. Finally, crack control reinforcement is detailed, and the designer ensures that longitudinal reinforcement is properly anchored.

An overview of the STM, past research focused on the STM, and the typical design procedure for implementing the STM were presented in this chapter. The following chapter presents an overview of the STM computer program.

CHAPTER 3. OVERVIEW OF COMPUTER PROGRAM

3.1 Overview

This chapter serves as an overview of the computer program to aid engineers with the design of specific bridge substructure components using the strut-and-tie method (STM). An overview of Excel VBA is presented first. Next, formatting and layout information about the computer program is included followed by information on the required user inputs for the computer program. The program outputs are presented last. Details of the computer program specific to each of the types of bridge components for which the program was developed along with specific examples demonstrating the use of the program are presented in Chapters 4 and 5.

3.2 Coding and Layout

3.2.1 Excel VBA Overview

STEP was created using Excel Visual Basic for Applications (VBA). Excel VBA was selected as the programming language to ensure that the program is user-friendly and easy to navigate. Engineers are generally familiar with Excel and are not required to learn a completely new program to use the STM software. VBA is the coding language within Microsoft® products that allows users to automate tasks. When using the STM computer program, designers input information directly into an Excel spreadsheet. The users then click a “Run Program” button for the program to perform the calculations. Creating the program in Excel allowed for graphical outputs of the strut-and-tie model as well as tables that display the results of the design checks that were performed.

In VBA, there are two main procedures: functions and subroutines. A function is a type of code that receives inputs through arguments and performs calculations to output a result. A subroutine also performs calculations but does not have specific input arguments and does not output a result. Functions are used within a cell in Excel. Common predefined functions already included in Excel are SUM(), PI(), AVERAGE(), etc. Subs can be run using a button that is placed within a worksheet or from the Developer tab located in the top ribbon in Excel. Subs are utilized in this software and are run using a button. All subs are written in modules accessed through the Developer tab.

A variable is used within VBA to store values or information. In VBA, variables must be dimensioned before being used in the code. There any many different dimension categories, called data types. Common data types include string, integer, Boolean, double, and worksheet. Any variable dimensioned as a string variable is stored as text. Integer variables can only be numbers that are integers. A Boolean variable can only be true or false. Double variables, the most common data type utilized in STEP, are stored as decimal numbers. For a double variable, 15-16 significant digits are stored in Excel. Variables defined as worksheets are used throughout the program to create new sheets. There are several more types of variable data types in VBA; however, those listed here are the key dimensions used in the STM computer program.

The scope of a variable in VBA determines what modules can access the variable and is defined when the variable is dimensioned. There are three different scopes for variables: procedure, module, or public. Variables defined for a procedure can only be accessed within that specific procedure, which is either the sub or function in which the variables are dimensioned. These variables are dimensioned at the beginning of the sub or function. Variables defined for a module can be accessed by a sub or function included in that module. Multiple procedures can be included within a module, and Excel stores the module-defined variable to be used in any of these procedures. These variables must be dimensioned before any of the subs or functions in that module. A public variable is similar to a module variable but can be recognized by any module in the workbook. Public variables are dimensioned before any other code in the workbook.

Procedure and public variables are utilized in the STM computer program. By definition, the public variables are dimensioned at the beginning of the first module. Some examples of public variables in the program include user inputs, node locations, and member forces. Procedure modules are dimensioned at the beginning of the sub in which they are used. Some examples of procedure variables include fixed end moments used within the moment distribution method for continuous beam analysis, the value of h_{stm}/tan(25°), and the available length for bar anchorage.

3.2.2 Functionality of the Computer Program

Early in the project, the INDOT Study Advisory Committee decided that the computer program should create a strut-and-tie model and perform STM design procedures for a structural component with details that are input by the user. The program does not perform a complete structural design and determine member geometry, the required area of reinforcement, and appropriate reinforcement details. Because many variables impact these aspects of design, they are left to the discretion of the engineer. Therefore, the designer using the program is responsible for developing parameters for the structural component, such as geometric conditions, material properties, and reinforcement information. The designer must also determine the critical load cases. The program will use these parameters to develop a strut-and-tie model and perform design procedures according to Article 5.8.2 of AASHTO LRFD (2017). The program will inform the user whether the design input into STEP satisfies the design checks performed by the program or if the design does not meet code requirements.

The computer program consists of the three primary components presented in Figure 3.1. The program initially reads inputs provided by the designer. This information is then used to perform analysis according to the STM. After the analysis is run, the strut-and-tie model is output along with a summary of the design procedures that were performed. Each of these components is explained in more detail in this chapter. It is the responsibility of the designer to ensure the accuracy of the results and interpret whether the design is structurally sound. The computer program only performs design procedures required by the STM provisions in Article 5.8.2 of AASHTO LRFD (2017). The designer may need to perform additional design checks not specified in the STM provisions.

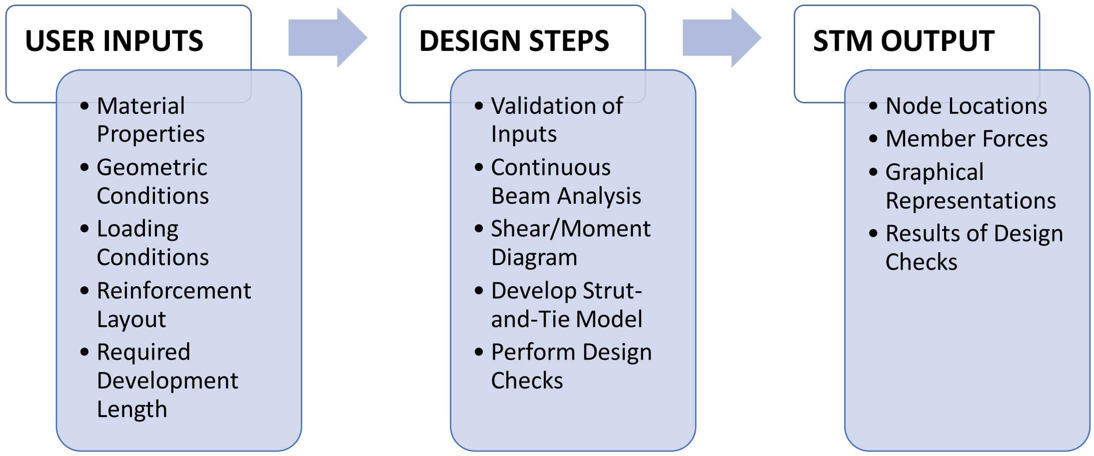


Figure 3.1: Computer Program Procedure Overview

Due to the difference in procedures for pier caps and end bent caps, two different Excel workbooks have been created for the computer program. The first workbook, titled *Pier Cap STM Design*, is to be used for the design of multi-column bent caps and straddle bent caps. The second workbook, titled *End Bent Cap STM Design*, is to be used for the design of integral and semi-integral end bent caps.

3.2.3 Computer Program Layout and Formatting

When the user initially opens either workbook, only two sheets (i.e., worksheets) will appear. The first sheet is the *Instructions* sheet which includes limitations and assumptions the user should review prior to using the program. These limitations and assumptions are also listed in detail in this guidebook in Section 4.3. The second sheet, the *Inputs* sheet, is where users should input information before running the program. Cells where the user will input information have a yellow background. The user has a choice of units for certain inputs. This is indicated by unit cells displayed in a purple color. These cells have a dropdown menu within the cell to choose units. Input values with units displayed in black text cannot be changed and the input value must be entered with the corresponding units. Any other cells should not be changed.

Throughout the document, any cell with blue text contains a note with additional information. If the designer clicks on this cell, the note will appear. Red bolded text within sheets that summarize design checks indicates a design check that was not satisfied. The designer should pay special attention to these cells and resolve the problem in order to develop a safe design.

After clicking the “Run Program” button in the *Inputs* sheet, the program will perform a series of calculations. Once the program is finished running, a window will appear displaying “Program Run Complete.” To run the program again, the designer should first click on the “Reset Program” button in the *Inputs* sheet. When the “Reset Program” button is clicked, all sheets except the *Inputs* sheet and the *Instructions* sheet will be deleted. If the designer wishes to compare various designs, he or she should use two or more workbooks, one for each design.

3.3 User Inputs

After reviewing the information on the *Instructions* sheet, the designer should open the *Inputs* sheet and input details of the bridge element being designed. Figure 3.2 and Figure 3.3 show the *Inputs* sheet when the program is initially opened for a pier cap and an end bent cap, respectively. In the following subsections, the various user inputs required by the program will be described. For more information and examples for specific bridge components, see Chapters 4 and 5.

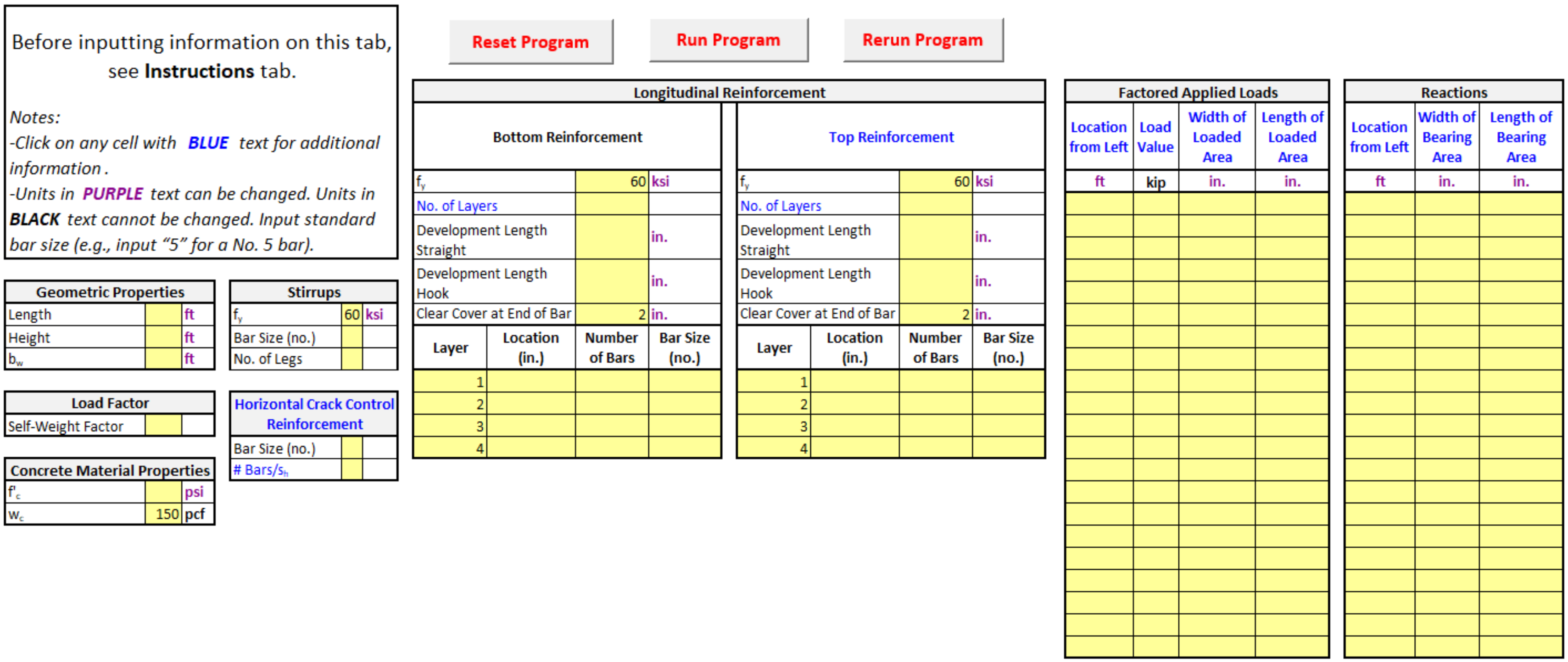


Figure 3.2: Pier Cap STM Design Workbook Inputs Sheet

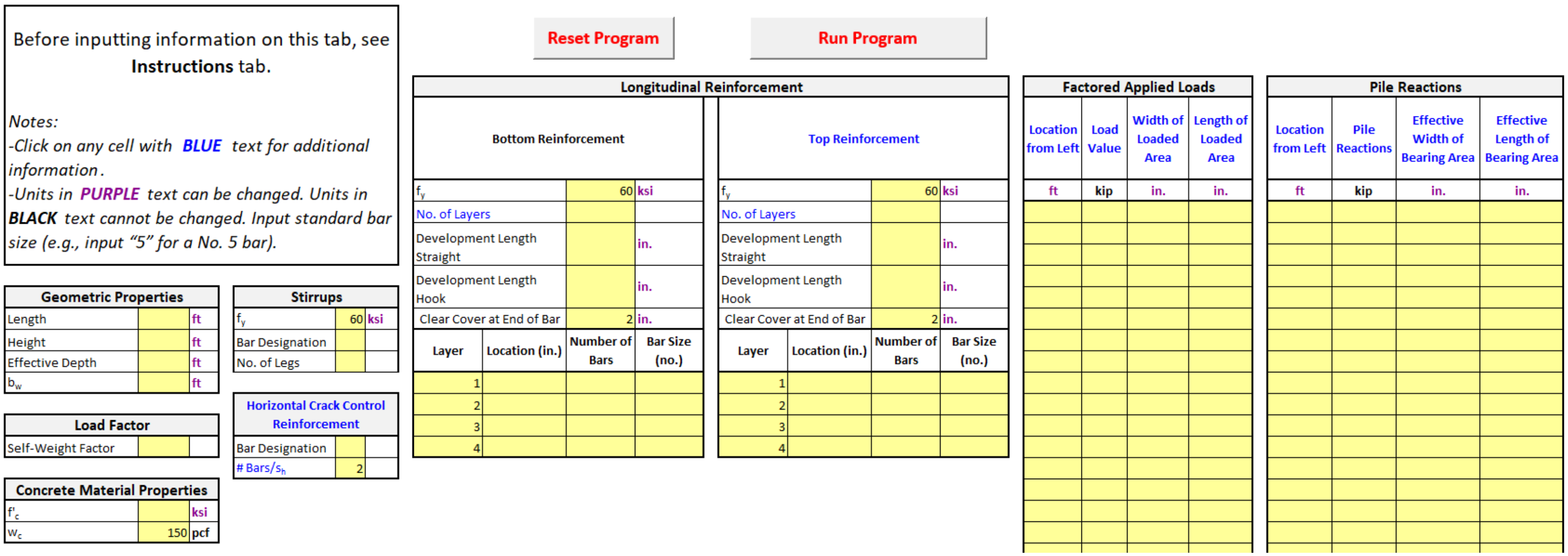


Figure 3.3: End Bent Cap STM Design Workbook Inputs Sheet

3.3.1 Geometric Properties

The length, height, and width of the beam should be input in the "Geometric Properties" table, shown in Figure 3.4. For end bent caps, the effective height of the cap is an additional input (see Section 5.2.3). Designers can input the section properties with units of inches or feet and should choose the appropriate units from the dropdown menu in the units cell. It is not required that the geometric properties and other inputs explained below all be entered in the same units.

Geometric Properties		
Length		ft
Height		ft
b_w		ft

(a)

Geometric Properties		
Length		ft
Height		ft
Effective Depth		ft
b_w		ft

(b)

Figure 3.4: "Geometric Properties" Input Table (a) for Pier Caps STM Design Workbook; (b) for End Bent Caps STM Design Workbook

3.3.2 Load Factor and Concrete Material Properties

The self-weight load factor is input by the designer in the "Load Factor" table, shown in Figure 3.5, and is used by the program for determining the magnitudes of the self-weight forces that will be applied to the strut-and-tie model. As described in Section 2.6, self-weight based on tributary volumes is added to the applied loads acting on the structural component. The self-weight load factor input into the program should match the load factor specified for *DC* loads in AASHTO LRFD for the particular load combination under consideration. If the engineer desires to distribute the self-weight along the structural component in a different manner (i.e., using smaller tributary areas), the load factor should be input as zero. The designer must then input the factored self-weight as applied point loads in the "Factored Applied Loads" table (see Section 3.3.5).

Load Factor		
Self-Weight Factor		

Figure 3.5: "Load Factor" Input Table

If the designer distributes self-weight prior to entering loads in the "Factored Applied Loads" table, each self-weight point load must be entered in a separate row in the table, even if the self-weight point load acts at the same location as a girder load. For these loads, the width and length of the loaded area should be input as zero. Within the strut-and-tie model that will be developed, a node will be located under each load input in this table. However, the self-weight loads not associated with a loaded area will be treated as smeared nodes and strength checks will not be performed on these nodes. Self-weight point loads acting at the same location as a girder load will not be considered when performing bearing face checks since these loads are not acting on the bearing area. The program distinguishes bearing loads and self-weight loads based on whether non-zero loaded area dimensions are provided for the load.

Concrete compressive strength and density are input in the "Concrete Material Properties" table, shown in Figure 3.6. The compressive strength can be input with units of ksi or psi. The units should be selected from the dropdown menu in the units cell. Concrete density is used for calculating self-weight and should be input with units of pcf.

Concrete Material Properties		
f'_c		psi
w_c	150	pcf

Figure 3.6: "Concrete Material Properties" Input Table

3.3.3 Stirrups and Horizontal Crack Control Reinforcement

In the "Stirrups" table, shown in Figure 3.7, the yield strength of stirrups, the bar size (designation) used for the stirrups, and the number of stirrup legs provided through the width of the member are input by the designer. The yield strength can be input with units of psi or ksi. The bar size is selected from a list of standard bar sizes in a dropdown menu. The number of legs should be input as an integer.

Stirrups		
f_y	60	ksi
Bar Size (no.)		
No. of Legs		

Figure 3.7: "Stirrups" Input Table

Horizontal crack control reinforcement, also known as skin reinforcement, is an additional required input. This is input in the "Horizontal Crack Control Reinforcement" table shown in Figure 3.8. The program requires the user to input the bar size of the skin reinforcement and the number of bars through the width of the member within spacing s_h, as illustrated in Figure 3.9. For the thin member in the figure, two bars are provided through the member width. For the thick member, four bars are provided through the width. The yield strength of the reinforcement used for horizontal crack control is not a required input because the reinforcement design is based on the reinforcement ratio, not the yield strength.

Horizontal Crack Control Reinforcement		
Bar Size (no.)		
# Bars/s_h		

Figure 3.8: "Horizontal Crack Control Reinforcement" Input Table

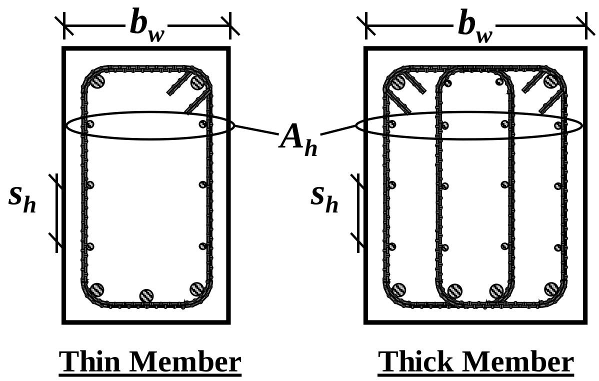


Figure 3.9: Horizontal Crack Control Reinforcement for a Thin and Thick Member (adapted from AASHTO LRFD, 2017)

3.3.4 Longitudinal Reinforcement

The next table on the *Inputs* sheet is the "Longitudinal Reinforcement" table, as shown in Figure 3.10. The bottom reinforcement information and top reinforcement information are entered separately, but both follow the same format. The yield strength of the bars and the required development length for a straight bar and a hooked bar are input first. The required development lengths should be calculated in accordance with AASHTO LRFD development length equations. If different sizes of bars are used in the same region (top or bottom) of the member (i.e. a layer of four No. 10 bars and another layer of four No. 8 bars are both provided along the bottom of the member), the governing development length considering the various bar sizes should be input into the program. If the user only wishes to consider hooked bars or straight bars, the input cell for the development length of the other condition can be left blank. The cover at the end of the bars is also input by the designer and is used for calculating the available length for the development of longitudinal reinforcement. The value is set by default to 2 in.; however, the designer can change the value in the cell.

Longitudinal Reinforcement								
Bottom Reinforcement					Top Reinforcement			
f_y		60	ksi		f_y		60	ksi
No. of Layers					No. of Layers			
Development Length Straight			in.		Development Length Straight			in.
Development Length Hook			in.		Development Length Hook			in.
Clear Cover at End of Bar		2	in.		Clear Cover at End of Bar		2	in.
Layer	Location (in.)	Number of Bars	Bar Size (no.)		Layer	Location (in.)	Number of Bars	Bar Size (no.)
1					1			
2					2			
3					3			
4					4			

Figure 3.10: "Longitudinal Reinforcement" Input Table

Next, the number of layers of reinforcement are input. If there is no top reinforcement in the member (i.e., if no negative moment region exists along the length of the member), the number of layers should be input as zero. The location, number of bars, and bar size (designation) are needed for each layer of reinforcement provided in the member. The location of each layer can be input with units of inches or feet and is measured from the bottom surface of the member. The bar size is selected from a dropdown menu. The user must input the information for each layer. The "Longitudinal Reinforcement" table provides space for four layers of both top and bottom reinforcement. If additional layers of reinforcement are provided in the member, the user can add the information for the additional layers under the existing table. The program will read the inputs and will reformat the table (i.e., add appropriate cell borders) after the user runs the program.

3.3.5 Factored Applied Loads and Reactions

The last input required by the program is a load case as well as locations of support reactions. All loads must be input as factored point loads. The program does not automatically factor any loads with the exception of applying the load factor input by the user for determining self-weight values.

For each point load, the location of the load measured from the left end of the member and the magnitude of the load are input into the “Factored Applied Loads” table, shown in Figure 3.11. The location can be input with units of inches or feet. The units should be selected from the dropdown menu in the units cell. All locations for applied loads must be input using the same units. One limitation of the program is that all applied loads are assumed to act downwards. The user should input the magnitude of each load with units of kip. The width and length of the loaded areas are also needed. The width of the loaded area is measured in the same direction as the width of the member. The length of the loaded area is measured along the longitudinal axis of the member. These dimensions can be input with units of inches or feet. It is recommended that circular loaded areas be input as equivalent squares with the same area as the actual loaded area.

Factored Applied Loads			
Location from Left	Load Value	Width of Loaded Area	Length of Loaded Area
ft	kip	in.	in.

Figure 3.11: “Factored Applied Loads" Input Table

Reactions are input in the same manner as the loads. However, the “Reactions” input table for pier caps is different from the table for end bent caps, as shown in Figure 3.12. For pier caps, the “Reactions” table allows the user to input the locations of the support reactions and the dimensions of the bearing areas. The computer program will analyze the structural component using a continuous beam analysis to determine the magnitudes of the reactions. For end bent caps, the table includes an additional column for the user to input the magnitudes of the reactions. If these cells are left blank, the program will determine the reactions of the end bent cap using a continuous beam analysis. If the designer would like to use reactions based on an alternative analysis method (e.g., pile group analysis), the reactions can be directly input into the "Reactions" table (see Section 5.2.3). These values will then be used when analyzing the strut-and-tie model for the end bent cap. Again, circular bearing areas should be input as equivalent squares. For end bent caps supported on H-piles, an effective bearing area should be input.

Reactions		
Location from Left	Width of Bearing Area	Length of Bearing Area
ft	in.	in.

Figure 3.12: "Reactions" Input Table

3.4 Program Outputs

After the program is run, more information will appear in various sheets within the workbook. Each of these sheets contains important information for the designer to review. The strut-and-tie model and a summary of the design procedures performed by the program are included in the sheets. Other structural analysis calculations are also included. Output sheets are arranged with the sheets relevant to the STM provided first, and the preliminary steps taken before creating the strut-and-tie model provided last.

3.4.1 Strut-and-Tie Model

Once the program is finished running, the *Strut-and-Tie Model* sheet will automatically open. The strut-and-tie model developed by the program is displayed on this worksheet. Nodes are labeled with letters starting at the top left with "A." The top chord is labeled first, and then the bottom chord is labeled from left to right as well. Struts are represented by dashed green lines and ties are represented by solid red lines. Forces in members are displayed next to the member. Loads and reactions are also displayed in the graphic. This graphic is editable, but it is not recommended that users make any changes to the graphic. If a user wishes to change components of the model, modifications should be made in the *Nodes and Members* sheet. This process is explained in Section 4.2.9.

3.4.2 Nodes and Members

The *Nodes and Members* sheet follows the *Strut-and-Tie Model* sheet. The sheet includes four tables. The first table, the "Nodes" table, lists each node in the strut-and-tie model. If the node is subdivided in order to perform the strength checks, the location of each portion of the subdivided node is presented in the "Subdivided Nodes" table. Both tables include the label for each node (or portion of a node) and its location in an xy-coordinate system. The origin of the coordinate system is the bottom left corner of the structural component. Next, the "Members" table lists each member in the strut-and-tie model. Each member is labeled based on the two nodes it connects. The table includes the force in each member and indicates if it is a strut or a tie. A negative force represents the compressive force in a strut, and a positive force represents the tensile force in a tie.

The "Nodes with Combined Forces and Updated Angles" table in the *Nodes and Members* sheet includes the details of all nodes that are not smeared nodes. These nodes have been considered for the nodal strength checks. For nodes that have been subdivided, a row is included in the table for each portion of the node. In each row, the force and orientation, or angle, of each member that intersects at the node (or portion of the node) is provided. Struts entering a node from the same direction have been combined together (see Section 2.11.2). The angle of each member listed in the table is measured from the positive x-axis of the same xy-coordinate system used in the "Nodes" table. If an angle has been adjusted based on the subdivision of a node, the updated angle is displayed (see Section 2.11.2).

3.4.3 Node Figures

The next sheet, *Node Figures,* contains graphical representations of each node (or portion of a node) in the "Nodes with Combined Forces and Updated Angles" table on the *Nodes and Members* sheet. Each node (or portion of a node) for which nodal strength checks have been performed is displayed. The graphics reflect the members intersecting at a node after adjacent struts have been combined together and angles have been updated based on the subdivision of nodes. The load or reaction that is applied at each node is also included in the graphics.

3.4.4 Nodal Checks

The *Nodal Checks* sheet includes a detailed summary of the strength checks that were performed for each node that is not a smeared node. Each row in the table includes all strength checks associated with a particular node (or portion of a node). In each row, the type of node (i.e., CCC, CCT, or CTT) and confinement modification factor, *m*, are first provided. Next, the length of each face (i.e., bearing, back, and strut-to-node interface) is given. The strength checks for each face are then displayed. For each face, the factored load acting on the face, P_u; the concrete efficiency factor, ν; the limiting compressive stress, f_{cu}; and the factor resistance, ϕP_n, is provided. The "Pass?" column included for each nodal face indicates if the factored resistance is greater than the factored load acting at the face by displaying "OK" or "NG" (no good). If the strength of the nodal face is insufficient, the corresponding cells will contain red bolded text.

If no direct compressive force acts on the back face of a node, the back face check is unnecessary (see Article 5.8.2.5.3b of AASHTO LRFD (2017)). In this case, "Check N/A" is displayed in the cells for the back face check. For nodes that are subdivided, the bearing face check is performed on the entire bearing surface of the original node prior to subdividing. Therefore, only one bearing strength check is displayed for the portions of a subdivided node. Similarly, the portions of a subdivided node share a common back face, and therefore, only one back face strength check is performed.

3.4.5 Reinforcement Design

The next sheet in the program is the *Reinforcement* sheet. The information presented in the sheet includes the following:

- A summary of the strength checks performed for the longitudinal reinforcement.
- The required spacing of stirrups based on vertical tie forces and vertical crack control reinforcement requirements.
- The required spacing of horizontal crack control reinforcement (i.e., skin reinforcement).
- A summary of anchorage checks conducted by the program.

First, the "Longitudinal Reinforcement" table provides a summary of the strength checks for the bottom and top reinforcement. The factored resistances, $\phi A_{st} f_y$, of the reinforcement are provided near the top of the table. Each horizontal tie within the strut-and-tie model is listed along with the corresponding tensile force. The ties are checked to ensure sufficient strength is provided by the reinforcement. Any tie that has a demand that is greater than the factored resistance will be indicated by red bolded text and "NG" (no good) will be displayed in the "Pass?" column. If the factored resistance of the reinforcement is greater than the demand, "OK" will be displayed in the "Pass?" column.

The next table, the "Crack Control Reinforcement" table, includes the maximum allowable spacing that satisfies crack control reinforcement code provisions for both the horizontal skin reinforcement and the stirrups (see Section 2.12).

The "Stirrups" table is provided next and includes a list of each vertical tie in the strut-and-tie model and the corresponding force in the tie. Next, the assumed width of each tie is displayed. The width is used to determine the maximum stirrup spacing as described in Section 2.10. This spacing is indicated in the "Req'd Tie Spacing (in.)" column. The required spacing of the stirrups to satisfy crack control reinforcement code provisions is provided next. The last column in the table gives the governing spacing, which is taken as the lesser of the spacing required to carry the force in the vertical ties and the spacing required for crack control reinforcement.

The last table in the *Reinforcement* sheet is the "Anchorage Check" table. Each node at which anchorage was checked is listed. For each check, the corresponding required development length(s) for a straight and/or a hooked bar previously input by the user is displayed. The available length within the structural component for developing the bar (see Section 2.13) is also provided. The required development length(s) is compared to the calculated available length, and the program indicates whether a straight and/or hooked bar will provide adequate anchorage. If the available length is less than the required development length, the corresponding cells will contain red bolded text.

3.4.6 Structural Analysis Outputs

The remaining output sheets that are created after running the computer program include preliminary steps that are performed prior to developing the strut-and-tie model. The "Cont. Beam Analysis" sheet contains the continuous beam analysis that was conducted using the moment distribution method to determine the reactions at the supports. The next sheet, *Shear Diagram*, includes the shear diagram for the structural component. Because the orientation of diagonal struts is directly related to the sign (positive or negative) of the shear force in the corresponding region of the member, the shear diagram is used by the program to correctly orient the diagonal struts in the strut-and-tie model. The moment diagram, displayed in the *Moment Diagram* sheet, is used to determine the critical moment along the length of the member that is used to optimize the height of the strut-and-tie model when there is no negative moment region along the length of the component (see Section 0). The last sheet, *Shear and Moment Diagram Calcs*, contains the values calculated to plot the shear and moment diagrams displayed on the preceding sheets.

3.5 Summary

STEP was created in Excel using VBA as the coding language. The intent of the computer program is for engineers to use it as a tool to perform design checks on specific bridge components. Engineers input geometric conditions, material properties, loading conditions, and reinforcement information for the bridge component directly into an Excel sheet. Using this information, the program will develop a strut-and-tie model of the component and perform design checks. Within the output sheets created by the program, engineers should pay special attention to any text in red because it indicates an insufficient aspect of the design. The engineer must modify the design to satisfy all design checks.

The following chapters include examples of implementing the computer program for a multi-column bent cap and integral and semi-integral end bent caps.

CHAPTER 4. COMPONENT 1 - MULTI-COLUMN BENT CAP

4.1 Overview

Multi-column bent caps, such as the one shown in Figure 4.1, are common bridge substructure elements in Indiana. This chapter presents the procedures used by the computer program to perform design checks for a multi-column bent cap in accordance with strut-and-tie method (STM) code provisions. An example that demonstrates these procedures is also presented. The STM design of the same multi-column bent cap used in the example was described in detail in Chapter 4 of Williams et al. (2012). Implementation of the computer program to aid in the design of the bent cap is presented in the current chapter. The design procedure followed in the example can be applied to other multi-column bent caps provided that the element satisfies the limitations and assumptions that are presented herein.

Figure 4.1: Multi-Column Bent Cap

4.2 Computer Program Procedure

A general overview of the computer program layout and required user inputs was presented in Chapter 3. The current chapter includes a more detailed procedure specific to multi-column bent caps that is used by the computer program in addition to an explanation of the computer code that accomplishes the design tasks. Straddle bent caps can be designed using the same workbook as that used for multi-column bent caps. Appendix A contains the procedure used by the computer program for multi-column bent caps in an outline format and can be referenced alongside this chapter.

4.2.1 Instructions

When the user first opens the STEP workbook titled *Pier Cap STM Design*, an *Instructions* sheet will appear. This sheet, shown in Figure 4.2, includes basic instructions about which cells are used for inputting information and which cells should be left unchanged. The two buttons located in the *Inputs* sheet are also explained (see Section 4.2.3). Furthermore, limitations and assumptions for the program are included in this sheet. Although the *Instructions* sheet is meant to provide basic information to the designer in order to begin utilizing the program, detailed information and instructions about how to ensure proper operation of STEP is included in this thesis.

INSTRUCTIONS

To use STEP, designers should begin by entering required user inputs on the Inputs worksheet in any cell with a yellow background. Cells that do not have a yellow background should not be edited. When inputting values, ensure that appropriate units are selected. Units in purple text can be changed, while units in black text cannot be changed. It is not necessary to format the Inputs worksheet. The sheet will be automatically reformatted when the program is run. Once the designer has correctly input all required information, the "Run Program" button should be clicked to start the STM design procedure. If the program has been run previously and not yet reset, a message will appear informing the user that running the program will cause the previous results to be deleted. The user should click "Yes" to allow the program to run. Another message will appear when the program run is complete. The "Reset Program" button deletes the results of a previous run of the program (i.e., all worksheets other than this Instructions sheet and the Inputs sheet will be deleted when the button is clicked). The "Rerun Program" button should only be used after running the program once. Detailed information about when to use the "Rerun Program" button is included in the STEP Guidebook.

For more information about how to use STEP, reference the STEP Guidebook.

DISCLAIMER

STEP is a tool developed to aid engineers with implementing the strut-and-tie method for the design of bridge substructure components. The engineer should be familiar with the limitations and the assumptions of the program prior to using it as a design tool. It is the responsibility of the engineer to ensure the accuracy of the results and interpret whether the design is safe and serviceable. The developers of STEP made every effort to create a program that will generate viable strut-and-tie models and accurately perform design procedures for most typical multi-column bent caps and straddle bent caps . Nevertheless, some particular design situations may have been inadvertently overlooked, which may cause the program to output erroneous results. Ultimately, the engineer is responsible for verifying the viability and accuracy of the strut-and-tie model and validating all results output by the program.

Limitations

1. The top and bottom chords of the strut-and-tie model are located at constant depths along the length of the member. In other words, the location of the top and bottom chords does not change along the length of the bent cap. If there is a negative moment region, the top chord will be located at the centroid of the top longitudinal reinforcement. Otherwise, if there is no negative moment, the location of the top chord will be based on an optimized design.
2. The amount of top and bottom longitudinal reinforcement is constant along the length of the member.
3. A positive moment region must exist at some location along the length of the member.
4. All loads are applied downward (in the direction of gravity), and all support reactions act upward.
5. All supports are assumed to provide only vertical restraint.
6. Self-weight is distributed by the program to locations where external loads are applied to the member. If further refinement is desired, the user should apply self-weight as several external applied loads along the length of the member. In this case, the self-weight load factor should be input as zero.
7. The program is limited to a strut-and-tie model with 104 nodes or less.
8. If a node is subdivided into two or three parts due to the subdivision of an applied load or support reaction, the classification of each part of the node as a CCC, CCT, or CTT node is determined based on the classification of the original node before subdividing.
9. The program is designed to satisfy the strut-and-tie method design provisions in Article 5.8.2 of AASHTO LRFD (2017). Other code provisions may not be satisfied by the design considered in the program.
10. Only prismatic members are considered.
11. The program assumes that the entire bent cap is defined by D-regions. If any B-regions exist, they will also be modeled with a strut-and-tie model. The designer is responsible for performing appropriate design procedures based on a sectional model for any B-regions along the length of the member (see Article 5.5.1.2.2 of AASHTO LRFD (2017)).
12. Anchorage checks will only be performed near the ends of members (i.e., for the outermost longitudinal ties). If the designer terminates bars at intermediate locations along the length of the member, the designer must check anchorage at those locations.
13. The program performs a bearing check for nodes based on the bearing area input by the user. If girder loads are combined together before being input into the program, the designer must check the bearing strength for each girder individually.
14. For cases in which a node is subdivided into three parts, the angle of the vertical strut will likely change slightly. The program neglects this change of angle and considers the strut to be vertical. If the diagonal struts at the node have vastly different magnitudes, the designer may need to consider a revised angle for the vertical strut at that node.
15. For the back face of a CCC node, the contribution of any nonprestressed compressive reinforcement is not considered. If consideration of the contribution of compressive reinforcement is desired, the designer should include it in his/her own calculations in accordance with Article 5.8.2.5.1 of AASHTO LRFD (2017). The designer must also check that the reinforcement is fully developed in compression.
16. A diagonal strut that changes orientation due to the subdivision of a node cannot be handled by the program. For example, if a diagonal strut is originally oriented from the upper left to the lower right and subdividing a node will cause it to be oriented from the upper right to the lower left, the program will not run.
17. Consideration of built-up bearing seats is not included in the program.

Assumptions

1. The strength of a back face is only checked if compression acts on the face. Article 5.8.2.5.3b of AASHTO LRFD states that "[b]ond stresses resulting from the force in a developed tie[…]need not be applied to the back face of the CCT node." However, this phenomenon may also occur at CTT nodes. (See Node P of Example 1 in "Strut-and-Tie Model Design Examples for Bridges." The horizontal component of the diagonal strut must transfer to the longitudinal reinforcement as a bond stress, changing the force in the horizontal tie at the node.) The program considers this observation and only checks the strength of a back face when it is subjected to direct compressive stresses.
2. Because bent caps do not fall under the category of slabs or footings, horizontal and vertical crack control reinforcement in accordance with Article 5.8.2.6 of AASHTO LRFD must be provided. Therefore, the efficiency factors in Table 5.8.2.5.3a-1 of AASHTO LRFD are used for all nodal strength calculations.
3. The width of each vertical tie is assumed to equal the smaller length of the two adjacent "truss panels" of the strut-and-tie model and is centered on the tie.
4. Longitudinal reinforcement distributed through the width of the pier cap is considered effective no matter the width of the member relative to the width of loaded areas or supports. Additional research is needed to determine the limits for distributing longitudinal reinforcement through the member width.

Figure 4.2: Instructions Sheet within Computer Program

4.2.2 Subroutines

A total of 20 different subroutines (subs) are used to analyze a multi-column bent cap in the computer program. The procedures in each sub will be described in the following sections. The subs are as follows:

- Run
- Inputs
- Validate_Inputs
- Self_Weight_Distribution
- Structural_Analysis
- Shear_Moment_Diagrams
- Label_Shear_Moment
- Top_Chord_Optimization
- Node_Locations
- Create_Members
- Method_of_Joints
- Plot_Model
- Design_Reinforcement
- Combine_Struts_Subdivide_Nodes
- Nodal_Checks
- Plot_Node_Figures
- Anchorage
- Formatting
- Rerun
- Reset

4.2.3 User Inputs

The next worksheet after the *Instructions* sheet is the *Inputs* sheet in which all information provided by the user is stored. A blank version of this sheet is shown in Figure 4.3. Section 3.3 provides an overview of the required inputs and the corresponding units. Once the designer inputs all required values and clicks the “Run Program” button, the code will start to run. The “Run

Program” button is linked to the first sub called “Run.” This sub calls all the remaining subs in order, except the “Rerun” and “Reset” subs. The “Reset Program” button runs the “Reset” sub (see Section 4.2.8). The “Rerun” sub is explained in Section 4.2.9. After all subs have run, a window will pop up that says, “Program Run Complete.”

The first sub that is run after clicking the “Run Program” button is the “Inputs” sub which imports all values input by the user and store them as variables to be used in later steps. The scope of all input variables is public to allow the variables to be used in multiple modules. The “Inputs” sub starts by importing section properties, the self-weight factor, and concrete material properties. The variables for these inputs were dimensioned before running the sub because they are public variables. The sub assigns values to each variable according to the value input by the user. Information in the “Stirrups” and “Horizontal Crack Control Reinforcement” tables is imported next. Again, the variables were previously dimensioned, and the user input values are stored for each input.

The information included in the “Longitudinal Reinforcement,” “Factored Applied Loads,” and “Reactions” tables is stored as matrix variables in the program. The “Longitudinal Reinforcement” table is imported first. Two public variables were created to store information about the top and bottom longitudinal reinforcement, called “top_reinf” and “bot_reinf,” respectively. The number of reinforcement layers input by the user for the bottom and top reinforcement is used by the program to determine how many rows are needed in the matrices for these variables. The matrix variables stored for use in future calculations match the tables on the *Inputs* sheet and contain the location, number of bars, and bar size (designation) for each layer of reinforcement in the top and bottom of the member. After importing the reinforcement information, the centroid of the reinforcement at both the top and bottom of the member are calculated. The centroids are used as the locations of the top and bottom chords in the strut-and-tie model and are defined in the “Inputs” sub. If no top reinforcement is input by the user, the program will only calculate the centroid for the bottom reinforcement and use this to place the bottom chord of the model. The location of the top chord will be determined later based on an optimized height of the strut-and-tie model after the shear diagram is created.

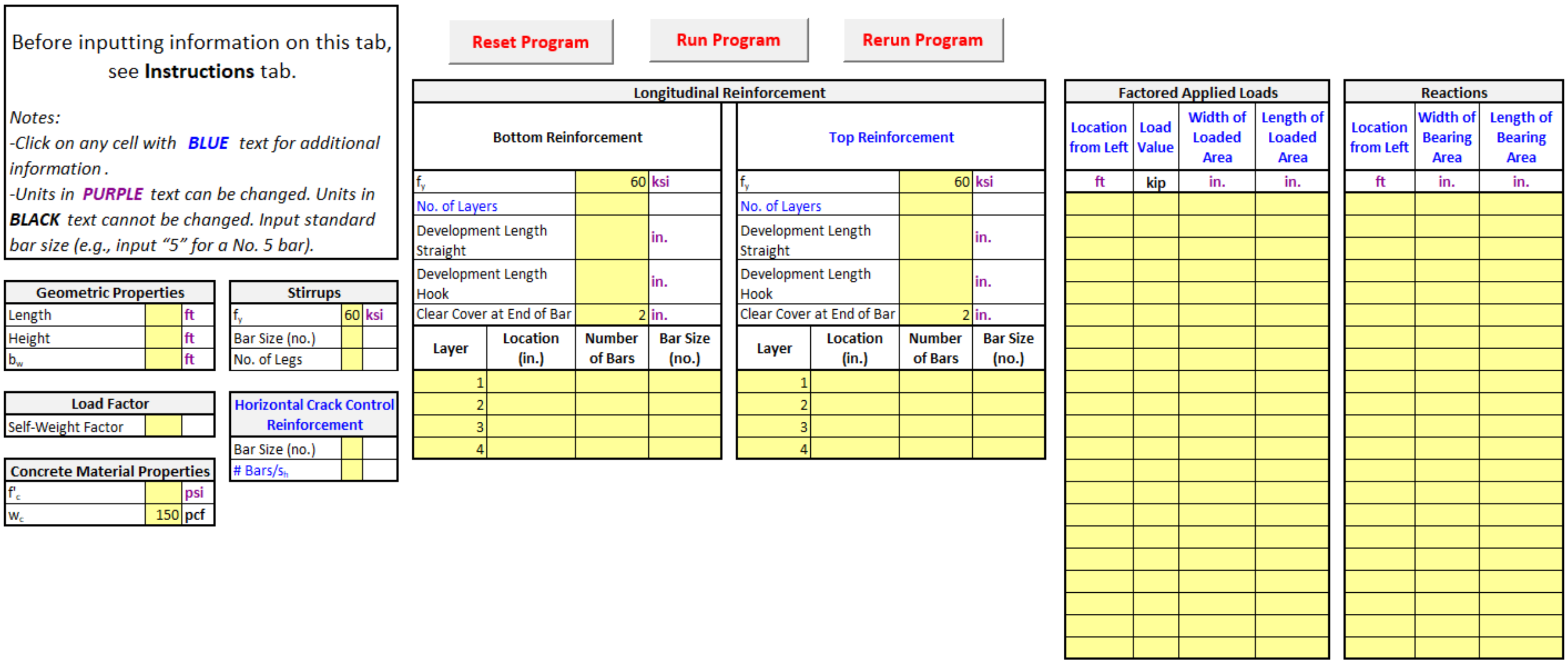


Figure 4.3: Inputs Sheet for Multi-Column Bent Cap

After importing the information in the “Longitudinal Reinforcement” table, the program counts the number of rows that contain information in the “Factored Applied Loads” table. This number is stored as an integer and used to determine how many rows to include in the “Loads” matrix variable. The same process is followed for the “Reactions” table. All information included in the “Factored Applied Loads” table is stored as one matrix, as is the information in the “Reactions” table. The second column of the “Reactions” matrix variable is left empty initially. After a continuous beam analysis is performed, the reaction values calculated will be stored in the second column of the “Reactions” matrix variable.

The “Validate” sub is called after the “Inputs” sub and is used to ensure that all user inputs are valid. If any inputs are entered as negative numbers, the program will stop running and notify the user. The user is required to correct the inputs and run the program again. The user will also be notified if any material properties do not satisfy the requirements in Article 5.8.2.1 of AASHTO LRFD (2017). In this case, a message will appear that gives the material property that violates AASHTO LRFD provisions. The user will be given the option to continue running the program or stop the program and enter different material properties.

4.2.4 Analyze Structural Component

After all user inputs are imported into the program and validated, the bridge element is analyzed to determine the reactions, and the shear and moment diagrams are developed. Four different subs are involved in this process: “Self_Weight_Distribution,” “Structural_Analysis,” “Shear_Moment_Diagram,” and “Label_Shear_Moment.” Each sub is explained in more detail in the following subsections.

4.2.4.1 Self-Weight Distribution

The first step in the analysis process is performed through the “Self_Weight_Distribution” sub which distributes self-weight as point loads along the element. One limitation of the program is that self-weight is only distributed to locations where external loads are applied to the member. If the user desires to distribute the self-weight along the member in a different manner, he or she must input the factored self-weight as external point loads in the *Inputs* sheet (see Sections 2.6 and 3.3.2). In this case, the user should input the self-weight factor as zero. The user should also input the loaded area dimensions for all self-weight load values as zero. The nodes at locations containing only self-weight will be treated as smeared nodes and will not be checked for strength. If a self-weight load is acting at the same location as a girder load, the bearing check will only consider the girder load.

If the user allows the program to distribute the self-weight, the self-weight will be added to each applied load based on the tributary width on either side of the applied load. The tributary width is multiplied by the density of the concrete and the beam width to calculate a point load value for self-weight. This value is factored by multiplying it by the self-weight factor input by the user. This factored load is then added to the magnitude of the corresponding applied load that was input by the user in the "Factored Applied Loads" table. Loads displayed in the graphics that are output by the program include the factored loads input by the user plus the factored self-weight distributed by the program.

4.2.4.2 Structural Analysis

After self-weight has been distributed, the "Structural_Analysis" sub is called. This sub analyzes the member as a continuous beam using the moment distribution method to determine reaction values. A new sheet, *Cont. Beam Analysis,* is created to display the calculations of the analysis. The moment distribution method begins by treating spans between supports as individual beams fixed at both ends. The moments at these fixed ends are then distributed throughout the member based on distribution and carry-over factors.

A table is created in the *Cont. Beam Analysis* sheet for the moment distribution calculations. The first row of the table displays each index listed as "Reaction 1," "Reaction 2," and so on for the number of reactions input by the user. The "Member" row includes labels for the columns in the table that correspond to the indices. The column supports of the bent cap are assumed to behave as pinned supports. The distribution factor is therefore assumed to be 0.5 at all supports except when considering the cantilever overhang at an exterior support. Then, the corresponding distribution factor at the exterior support is 1.0. Furthermore, the carry-over factors are taken as 0.5.

The fixed end moment (FEM) at each support is calculated based on applied loads. When an overhang extends past an exterior support, the overhang is analyzed as a cantilever beam fixed at the support. The FEM at the support due to each load on the overhang is calculated as shown in Figure 4.4(a). For all interior spans of the bent cap between two supports, the span is analyzed as a fixed-fixed beam, and the FEM at each support due to each point load is calculated as presented in Figure 4.4(b). Figure 4.4 shows the different variables and equations used to calculate moments for the cantilever beam condition and the fixed-fixed beam condition.

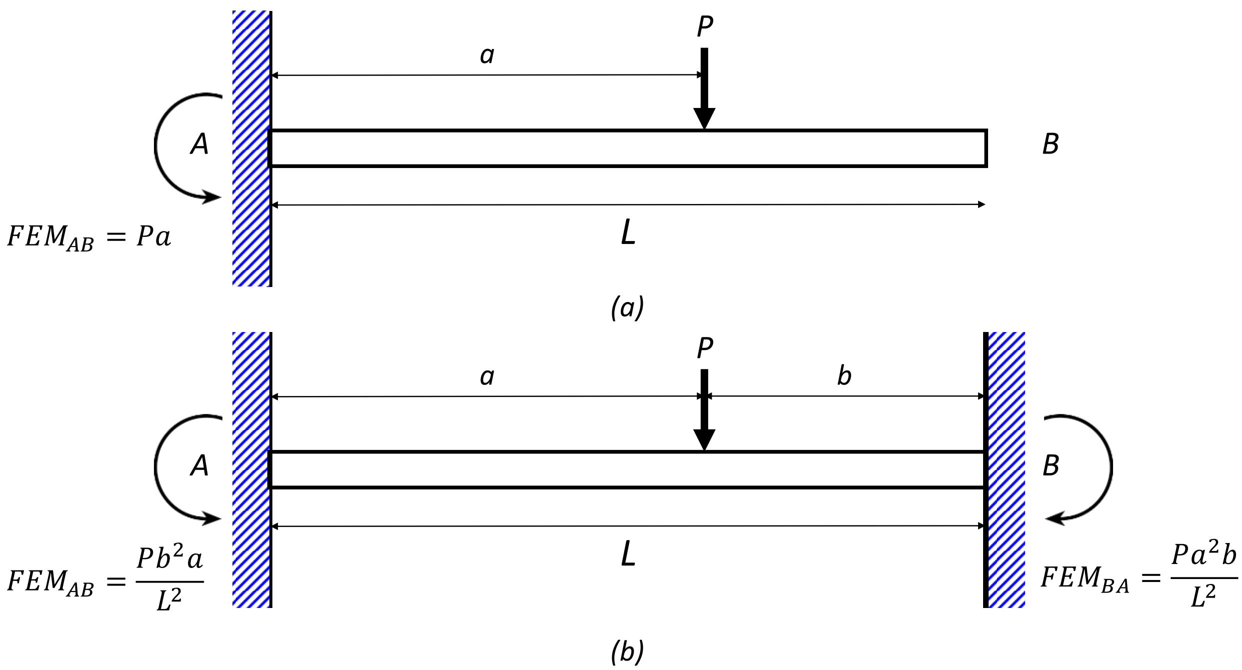


Figure 4.4: Fixed-End-Moment Calculations - (a) Cantilever Beam Condition; (b) Fixed-Fixed Beam Condition

After calculating the fixed end moments for each support, the program begins distributing and carrying over these values in accordance with the moment distribution method. To distribute moments in the first distribution ("DIST") row, the fixed end moments on each side of a given support are added together, the sign of the resulting sum is changed (i.e., the sum of the FEMs is multiplied by -1), and the resulting value is multiplied by the distribution factor. Then, in the carry-over ("CO") row, the distributed moments are multiplied by the carry-over factor and transferred, or carried over, to the support on the opposite end of that span. For example, the value in the first carry-over row for Member 1-2 is equal to the carry-over factor times the value in the first distribution row for Member 2-1. The program will continue distributing and carrying over moments in this manner until all values in a distribution ("DIST") row are less than the specified tolerance. The tolerance value specified in the program is 10^{-7}.

Once the program has completed the process of distributing and carrying over moments, all the moment values in each column are added. Then, each span of the bent cap is analyzed to calculate the shear force at each support due to the point loads. The shear forces at the supports due to the unbalanced moments acting on each span are calculated next. The shear from the point loads and the shear from the unbalanced moments are then added to determine the total shear force at each support. The shear values at each support are added together to get the reaction at the column. Any point loads acting directly above the support are added to this value to get the total reaction at each support. These are listed in the last row of the table as "Solved Reactions." The program then stores these values within the "Reactions" matrix variable in the column that was previously left blank (see Section 4.2.3).

4.2.4.3 Develop Shear and Moment Diagrams

The "Shear_Moment_Diagram" sub is run next and creates shear and moment diagrams for the bent cap. This sub begins by creating a matrix that includes both load and reaction values and locations in one variable. Applied loads are negative while reactions are positive. Any loads and reactions acting at the same point are added together. The matrix is sorted by location of the loads and reactions along the length of the beam.

After all the external loads are stored in one variable, the calculations for the shear diagram are performed. The shear calculations are straightforward since all loads are point loads. The results of the calculations are stored in the *Shear and Moment Diagram Calcs* sheet, and shear values are plotted in the *Shear Diagram* sheet.

After completing the shear calculations and graphing the shear diagram, this information is used to develop the moment diagram. The slope of the moment diagram at each location is calculated based on the value of the shear diagram. The results are displayed in a table within the *Shear and Moment Diagrams Calcs* sheet. These data are then plotted in the *Moment Diagram* sheet.

The next sub that is run by the program is the "Label_Shear_Moment" sub which labels the shear values on the shear diagram, and local maximums and minimums on the moment diagram. Regions of zero shear are not labeled on the shear diagram. This is the last step in analyzing the structural component.

4.2.5 Develop Strut-and-Tie Model

After the structural analysis of the bent cap is complete, the strut-and-tie model is developed. Five different subs are run to develop the model: "Top_Chord_Optimization," "Node_Locations," "Create_Members," "Method_of_Joints," and "Plot_Model." Each of these is explained in further detail below.

4.2.5.1 Optimize Top Chord Location

If there is only positive moment along the length of the member, the program will optimize the height of the strut-and-tie model to find location of the top horizontal chord of the model. More information on the reasons to optimize the height of the model can be found in Section 0. The program begins this procedure by identifying the location of the most critical positive moment. To accomplish this, the program considers the shear and moment values at each applied load and reaction location. If the shear changes signs at the location of an applied load, the node that will be placed along the top chord of the strut-and-tie model and abut the loaded area (e.g., load plate) will be a CCC node. Otherwise, the node will be a CCT node. The concrete efficiency factor, ν, depends on the type of node and is equal to 0.85 for a CCC node and 0.70 for a CCT node. The moment at each location is divided by the efficiency factor at the location. The most critical moment exists at the location with the largest moment-to-efficiency factor ratio.

After finding the critical moment, this value is used to determine the location of the top chord of the strut-and-tie model. The following equation is used to calculate a, the depth of the horizontal strut along the top chord:

$$M = \phi \nu f_c' b_w a * (d - {}^{a}/_{2}) \tag{4.1}$$

where:

M = critical moment

ϕ = resistance factor (= 0.70 according to Article 5.5.4.2 of AASHTO LRFD (2017))

ν = concrete efficiency factor (= 0.85 for CCC nodes and 0.70 for CCT nodes)

f_c' = specified compressive strength of concrete

b_w = width of member

a = depth of horizontal strut along the top chord of the strut-and-tie model

d = distance from top of member to centroid of bottom longitudinal reinforcement

The top chord, which represents the centroid of the horizontal strut at the top of the beam, will be placed a distance of $a/2$ from the top surface of the member. The bottom chord of the model coincides with the centroid of the bottom longitudinal reinforcement. If there is negative moment at some location along the length of the member, the top chord of the member will coincide with the centroid of the top longitudinal reinforcement, rather than at a distance of $a/2$ from the top surface of the member.

4.2.5.2 Node Locations

The "Node_Locations" sub determines the location of all nodes in the model and assigns a letter to each node. Nodes are placed along the top chord below every load point and along the bottom chord above every reaction. The program will only develop strut-and-tie models for which the locations of the top and bottom chord do not change along the length of the member. The x-coordinate of each node is set equal to the x-coordinate of the corresponding load or reaction. The y-coordinate of each node is set equal to the y-coordinate of the corresponding chord (top or bottom) of the strut-and-tie model.

After assigning nodes at these locations, the shear force along the member is used to place additional nodes. At any node located at a load or reaction where the shear force does not change signs, an additional node with the same x-coordinate is located on the chord opposite to the existing node. Although it is likely that the shear force changes signs at each reaction, the program considers the shear force for both loads and reactions to capture unique cases. The program will not attempt to place an additional node where both a load and a reaction occur at the same x-coordinate along the member because nodes will already exist on both the top and bottom chords at this location. If a node is located at a load or reaction where the shear force changes signs, an additional node on the opposite chord is not needed.

The program considers one exception when adding nodes at locations where the shear force in the member does not change sign. For any node that will be added opposite a load or reaction where the shear force does not change signs, a check is performed to determine if the new node falls within the dimensions of a bearing area (if the node is added on the bottom chord) or a loaded area (if the node is added on the top chord). When a node falls within these dimensions, the program deems the node unnecessary and the node will not be created. An example of this scenario is illustrated in Figure 4.5. If Node F, which falls within the bearing area of the column, is included in the strut-and-tie model (Figure 4.5(a)), a vertical tie is added at the node location. The resulting model implies an indirect load transfer from the load at Node C and the column reaction at Node E. A more realistic model is developed when Node F is not included in the model and a single diagonal strut transfers the load from Node C to Node E at the column reaction (Figure 4.5(b)). Without the unnecessary vertical tie, the resulting model is also more efficient (Williams at al., 2012).

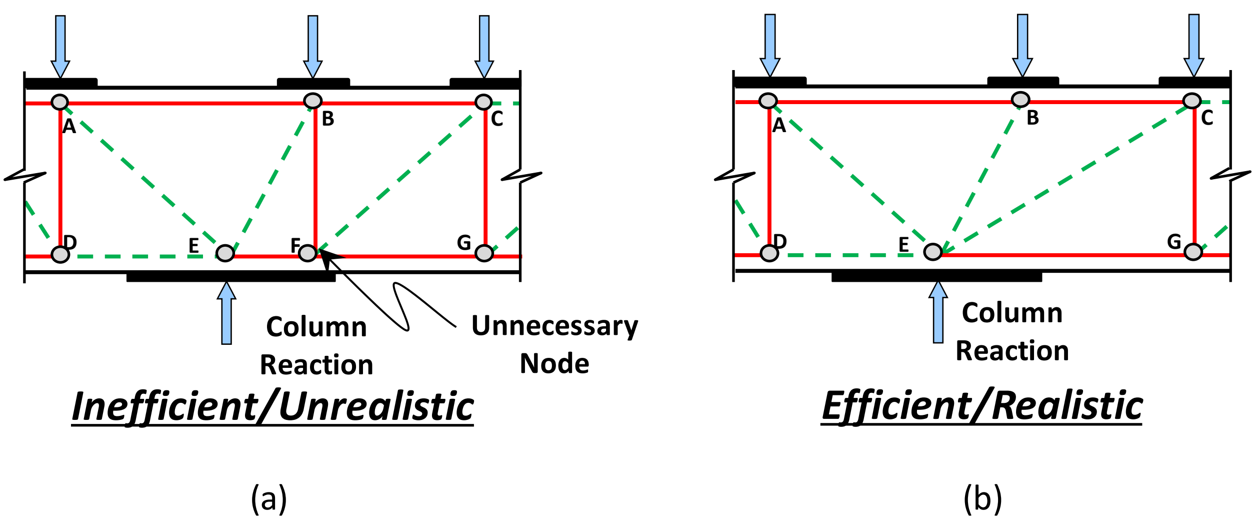
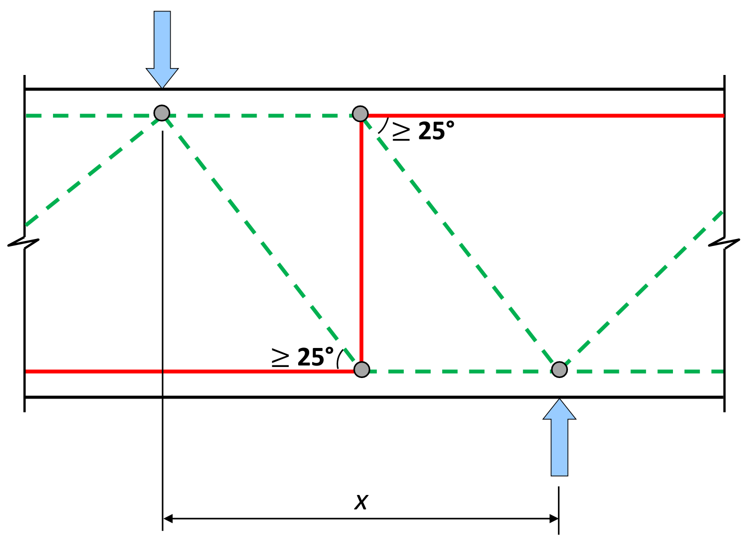


Figure 4.5: Strut-and-Tie Model Near a Column (adapted from Williams et al., 2012)

After adding nodes at load or reaction locations where the shear force in the member does not change signs, the program adds nodes when the horizontal distance between two adjacent nodes, irrespective of whether the nodes are on the top or bottom chords, is greater than h_{stm}/tan(25°), where h_{STM} is the height of the strut-and-tie model (i.e., the internal moment arm, *jd*). The STM provisions in AASHTO LRFD (2017) require that no angle between a strut and tie entering the same node be less than 25°. Adding additional nodes when the distance between adjacent nodes is greater than h_{stm}/tan(25°) ensures this 25-degree rule is satisfied. The program begins with the node at the top left of the component (i.e., Node A) and determines the horizontal distance to the next node. If this distance is greater than h_{stm}/tan(25°), the minimum number of truss panels between the two nodes that is needed to satisfy the 25° angle limit between a strut and tie (see Section 0) is determined. If the horizontal distance between the two adjacent nodes is between h_{stm}/tan(25°) and 2(h_{stm}/tan(25°)), two truss panels are necessary to satisfy the 25° angle limit, as shown in Figure 4.6. If the horizontal distance between the two adjacent nodes is between 2(h_{stm}/tan(25°)) and 3(h_{stm}/tan(25°)), three truss panels are necessary to satisfy the angle limit, and so on.



$$h_{STM}/\tan(25°) < x < 2(h_{STM}/\tan(25°))$$

Figure 4.6: Additional Nodes to Satisfy Angle Limit- 2 Panels

After determining the number of truss panels that are required, nodes are added along the top and bottom chords between the two adjacent nodes. The horizontal distance between the nodes is subdivided into the number of equal panel sections that were deemed necessary, and nodes are placed at the division points. This process is repeated for any two adjacent nodes in the strut-and-tie model separated by a horizontal distance greater than h_{stm}/tan(25°).

One last check is performed on the nodes that have been created to ensure the 25-degree rule of the STM provisions is satisfied. The angles between horizontal ties and diagonal struts that will exist within the strut-and-tie model will all be greater than 25° after adding nodes where the horizontal distance between any two adjacent nodes is greater than h_{stm}/tan(25°), as described in the preceding paragraphs. However, angles between vertical ties and diagonal struts still need to be checked at locations where nodes were added because the shear force at a load or reaction does not change signs. To perform the necessary check, the computer program first identifies nodes that will be connected by vertical ties. As described in Section 4.2.5.3, a vertical tie will connect the two nodes located under load points or above reactions where the shear force does not change signs. For example, Tie DY in Figure 4.7(a) is a vertical tie above a reaction (R_2) where the sign of the shear force does not change. Such ties are found by identifying two nodes that have the same x-coordinate and that also correspond with the x-coordinate of a load or reaction. If both a load and a reaction have the same x-coordinate as the nodes, the vertical member will be a strut, and the check is not performed. If there is only a load or a reaction at the x-coordinate, the location of a vertical tie of interest has been identified. Here, the angles between the tie and adjacent diagonal struts are checked. The program performs this check by first identifying panels adjacent to the tie. If the distance to the adjacent node is less than h_{stm}·tan(25°), the angle between the diagonal strut and the vertical tie is too small. In Figure 4.7(a), the distance between Node D and Node E is less than h_{stm}·tan(25°). Therefore, the angle θ is less than 25°. In this case, the node that was previously added because the shear force does not change signs at the reaction, Node D, will be deleted, and the strut-and-tie model shown in Figure 4.7(b) results.

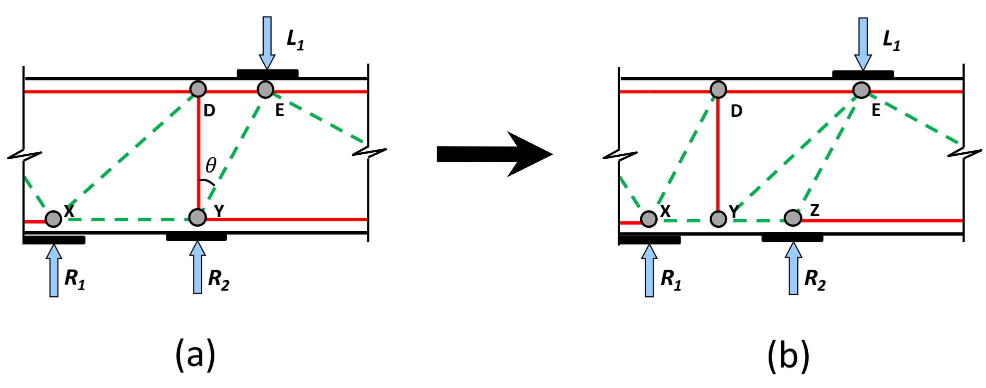


Figure 4.7: Satisfying the Angle Limit Between Vertical Ties and Diagonal Struts

After adding all nodes to the strut-and-tie model, the x- and y-coordinates of each node are displayed in a table on the *Nodes and Members* sheet, and each node is assigned a letter. A number index is also assigned to each node starting with Node A indexed as 1. This index is used throughout the program in FOR loops as a row index. The program is limited to a strut-and-tie model with 104 nodes or less. If the structural component requires a strut-and-tie model with more than 104 nodes, the program will stop running and the designer will be notified.

The last procedure in the "Node_Locations" sub is the creation of a variable containing the applied loads and reactions at each load or reaction point. The variable is a vector with a row for every node. If there is an applied load or reaction at the node, the value of the load or reaction is stored in the corresponding row of the matrix. If no load or reaction acts at the node, zero is stored in the corresponding row of the matrix. This variable is used when solving for the member forces within the strut-and-tie model using the method of joints.

4.2.5.3 Create Members

After assigning node locations, the "Create_Members" sub is run. This sub is used to locate members (struts and ties) in the strut-and-tie model by connecting nodes. Horizontal members are created first, followed by vertical members. Diagonal members are generated last. Each member is given a label based on the two nodes it connects, and the type of member is stored with a designation of either "H-top," "H-bottom," "V," or "D." The designation is used later in the program when proportioning ties and determining which struts should be combined together for performing nodal strength checks.

To create horizontal members, the number of nodes with a y-coordinate along the top chord is determined along with the number of nodes with a y-coordinate along the bottom chord. Then, all nodes along the top chord are connected with horizontal members and are given the designation "H-top." The members are labeled from left to right along the top chord. For example, the left-most member will be given the label "A-B." The same procedure is followed to create the horizontal members along the bottom chord, but these members are given the designation "H-bottom."

A vertical member is created by connecting a node located along the top chord with a node located along the bottom chord that has a matching x-coordinate. Vertical members are given the designation “V,” and are labeled based on the letter assigned to the node located on the top chord followed by the letter assigned to the node located on the bottom chord. Vertical struts will exist between a load and a reaction that have the same x-coordinate. Vertical ties will exist at the locations of loads or reactions where the shear force does not change signs or between nodes that were added to satisfy the 25° angle limit (see Section 4.2.5.2). However, the members are not initially assigned as struts or ties. The type of member is determined later based on the sign of the axial force in the member.

Diagonal members are created next. The manner in which diagonal members are oriented is determined by the shear in the member, as shown in Figure 4.8. In negative shear regions, diagonal struts are oriented from the upper left to the lower right. Member M-GG in Figure 4.8 demonstrates this. In positive shear regions, the members are oriented from the upper right to the lower left, as seen by Member M-FF in Figure 4.8. Using this convention, each node along the top chord is connected to the nearest node located along the bottom chord that does not have a matching x-coordinate. Diagonal members are generated starting with the left-most node along the top chord. Like vertical members, they are labeled based on the letter assigned to the node located on the top chord followed by the letter assigned to the node located on the bottom chord. The members are given the designation “D.”

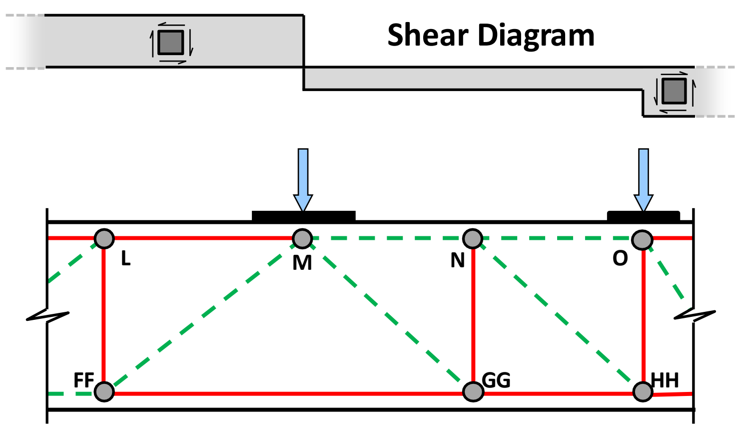


Figure 4.8: Diagonal Strut Orientation According to Shear Diagram

After generating all members, the labels are stored in a “Members” matrix and displayed on the *Nodes and Members* sheet of the program.

4.2.5.4 Method of Joints

The method of joints is used to calculate the force in each member. To perform the method of joints for a strut-and-tie model with many nodes, the program creates a variable named “Solving Matrix.” For each node, the “Solving Matrix” includes coefficients for each member that intersects at the node. The members intersecting at a particular node are determined by identifying members with a label that includes the letter that matches the label of the node. There are two rows in the matrix for every node, one for horizontal components of forces acting at the node and one for vertical components of forces. The matrix includes a column for every member in the model. Coefficients are determined based on the members entering each node and the horizontal and vertical components of each member force. For example, within the row corresponding with the horizontal components of forces at a given node, coefficients will be included for any member that has a horizontal component acting at that node. The coefficients in that row will be equal to the fraction of the member forces that define the horizontal components of the forces.

An example submatrix taken from a large “Solving Matrix” variable is shown in Table 4.1. The submatrix represents the two rows of the “Solving Matrix” that correspond to Node W, illustrated in Figure 4.9. The three members that intersect at the node are represented by the three columns in Table 4.1. All other columns in the “Solving Matrix” that are not shown in the table correspond to the other members of the strut-and-tie model and contain a value of zero because the other members do not intersect at Node W. Coefficients in the matrix are used when enforcing horizontal and vertical equilibrium at each node and are equal to the fraction of the member forces that define the horizontal and vertical components.

Table 4.1: Solving Submatrix for Node W

	Member A-W	Member B-W	Member W-X
Node W Horizontal Component	$\cos(51.70) = 0.620$	$\cos(31.19) = 0.855$	$\cos(0) = 1$
Node W Vertical Component	$\sin(51.70) = 0.785$	$\sin(31.19) = 0.520$	$\sin(0) = 0$

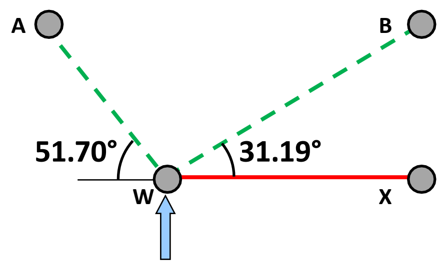


Figure 4.9: Members Intersecting at Node W

The program will determine how many members are entering a node, and how many forces of those members are unknown. It will begin by solving equilibrium equations for nodes at which only two members intersect. Then, the program will loop through the matrix and will solve for member forces at nodes at which only two unknown member forces remain. This process continues until all member forces are known. At the beginning of the analysis, all members are assumed to be in tension. After solving for member forces, positive forces indicate ties while negative forces indicate struts.

Member forces are stored in a matrix that also contains the label and designation of the member (i.e. “H-top,” “H-bottom,” “V,” or “D”) as well as whether the member is a strut or a tie. The member label, force, and strut or tie designation are displayed in the “Members” table on the *Nodes and Members* sheet of the program. Any members determined to have a force of zero are deleted from the “Members” table as well as the matrix containing the member forces because these members are unnecessary.

4.2.5.5 Plot Model

After determining all details of the strut-and-tie model, a graphical representation of the model is created. The *Strut-and-Tie Model* sheet contains a plotted graph of the model. The user will be automatically transferred to this sheet after running the program. To create the graph, the *Strut-and-Tie Model* sheet is generated and gridlines and headings are removed from the sheet. All nodes are plotted as a scatter plot without gridlines, axes, or chart titles. The area of the chart is sized based on the geometric conditions of the structural component. The top left corner of the sheet will always be displayed when it is first opened. The designer should zoom in and out and scroll as needed to view the entire model.

A data set is used to label nodes on the plot. A data set with the name "Nodes" is created that places a label next to each node. For nodes along the top chord of the model, the labels are placed above each node. For nodes along the bottom chord, the labels are placed below each node. Points on the scatter plot that represent nodes and the labels at these points are then formatted. Text for node labels is bolded, and the points are displayed as a gray circle with a black outline.

After labeling nodes, each member is plotted as a data set. A line representing each member is plotted based on the two nodes the member connects. If the member is a strut, it is plotted as a dashed green line. If the member is a tie, it is plotted as a solid red line. The force in the member is labeled next to the corresponding line. Next, loads and reactions are added to the figure. An arrow is plotted for every load and reaction and is created by graphing three lines. The arrows are labeled with the corresponding load or reaction value. Lastly, a simple legend is added to the figure to remind the user of the convention used for representing struts, ties, and nodes.

4.2.6 Perform Design Checks

Following the development of the strut-and-tie model, design checks are performed in accordance with the strut-and-tie method. The design steps are conducted using five subs: "Design_Reinforcement," "Combine_Struts_Subdivide_Nodes," "Nodal_Checks," "Plot_Node_Figures," and "Anchorage." Each sub is explained in the following subsections.

4.2.6.1 Design Reinforcement

The first design steps that are performed focus on the reinforcement within the structural component. The "Design Reinforcement" sub begins by determining the required spacing for the horizontal and vertical crack control reinforcement. The upper limit for the spacing of crack control reinforcement as defined by AASHTO LRFD (2017) is calculated first. Article 5.8.2.6 of the specifications states that the spacing of the bars used for crack control "shall not exceed the smaller of *d*/4 and 12.0 in." (AASHTO LRFD, 2017). The program considers both positive and negative moment regions when calculating *d* for the spacing limit. In other words, the distance from the top of the member to the centroid of the bottom reinforcement as well as the distance from the bottom of the member to the centroid of the top reinforcement is considered. The smaller value of these two distances is used when determining the crack control spacing limit. This spacing is calculated and stored.

The required spacing for horizontal and vertical crack control reinforcement is determined next. The maximum reinforcement ratio for crack control reinforcement in both directions is 0.003 as specified by Article 5.8.2.6 of AASHTO LRFD (2017). The required spacing is calculated using the reinforcement information input by the user for the horizontal skin reinforcement and the vertical reinforcement (i.e., stirrups) (see Sections 2.12 and 3.4.5). This required spacing is compared to the upper limit of $d/4$ or 12.0 in. to determine the maximum spacing that can be provided for the stirrups and skin reinforcement. This maximum allowable spacing is displayed in the "Crack Control Reinforcement" table on the *Reinforcement* sheet. All spacing values displayed on this sheet are rounded down to the nearest tenth of an inch.

If the maximum allowable spacing of the crack control reinforcement is calculated to be less than 3 in., the stirrups and/or horizontal reinforcing bars used for crack control are determined to be inadequate. A spacing less than 3 in. is deemed to be unreasonable for construction. In this case, the word "*Inadequate" will be displayed in the "Crack Control Reinforcement" table on the *Reinforcement* sheet and a note will appear informing the user that a different reinforcement detail is required. Red bolded text will be used to note the inadequate design. The designer should either increase the bar size of the stirrups and/or horizontal crack control reinforcement or increase the number of bars provided through the width of the member (e.g., include additional stirrup legs) to ensure the required spacing is a more reasonable value.

After the required spacing of the crack control reinforcement is calculated, the horizontal ties of the strut-and-tie model are considered. Strength checks are performed to determine if the longitudinal reinforcement input by the user is adequate to carry the forces in the ties. If a negative moment region exists within the member, strength checks are performed for both the top and bottom chords of the model. Otherwise, strength checks are conducted for only the bottom chord. For each chord being considered, the factored resistance of the ties is calculated using Equation (4.2):

$$\phi P_n = \phi A_{st} f_y \tag{4.2}$$

where:

P_n = nominal resistance of the tie (kip)

ϕ = resistance factor (= 0.90 according to Article 5.5.4.2 of AASHTO LRFD (2017))

A_{st} = total area of longitudinal nonprestressed reinforcement (in.^{2})

f_y = yield strength of longitudinal nonprestressed reinforcement (ksi)

The factored resistance is calculated separately for the top longitudinal reinforcement and the bottom longitudinal reinforcement, and the values are displayed for each chord in the "Longitudinal Reinforcement" table on the *Reinforcement* sheet. The program assumes that the top and bottom longitudinal reinforcement input by the user is constant along the length of the member. The program is unable to consider varying amounts of reinforcement (e.g., cut-off bars) along the member length. Therefore, the factored resistance is constant along each chord of the strut-and-tie model. After determining factor resistance(s), the ties along each chord are identified and the force in each tie is compared to the capacity. The program loops through the "Members" matrix (see Section 4.2.5.3) and identifies members designated as "H-top" or "H-bottom" to compare the member forces to the corresponding factored resistance. Members with the designation "H-bottom" are displayed on the left side of the "Longitudinal Reinforcement" table, and members with the designation "H-top" are listed on the right side. For each horizontal tie, the member label, member force, and whether the tie has adequate strength is presented. Within the "Pass?" columns, "OK" is displayed if the factored resistance is greater than the force in the tie. Otherwise, "NG" (no good) is displayed. Red bolded text is used to indicate any ties that do not satisfy the strength check.

Vertical ties are proportioned last. Each member designated as "V" is displayed in the "Stirrups" table on the *Reinforcement* sheet. The width of each vertical tie, w_t, is defined as the smaller width of the two adjacent truss panels of the strut-and-tie model (see Section 2.10). The stirrups considered to carry the force in the tie are assumed to be centered on the vertical tie in the model (Williams et al., 2012). The required spacing of stirrups that are used to carry the force in a vertical tie is calculated using Equation (4.3)

$$spacing \leq \frac{\phi A_v * f_y * w_t}{P_u} \tag{4.3}$$

where:

ϕ = resistance factor (= 0.90 according to Article 5.5.4.2 of AASHTO LRFD (2017))

A_v = transverse area of reinforcement (in.^{2})

f_y = yield strength of stirrups (ksi)

w_t = width of tie (in.)

P_u = tensile force in tie (kip)

The value of A_v is the area of transverse reinforcement (i.e., all stirrup legs) within one stirrup spacing. The stirrup spacing required to carry the tie forces are listed in the "Req'd Tie Spacing (in.)" column in the "Stirrups" table. In the next column, the spacing that was previously calculated to satisfy the crack control reinforcement requirements is listed. This value is the same for every row of the table because it is based on a constant reinforcement ratio of 0.003.

The last column in the "Stirrups" table is the governing spacing. This spacing is the lesser of the spacing required to carry the force in the ties and the spacing required for crack control reinforcement. If either of these values is calculated to be less than 3 in., the stirrups are deemed to be inadequate, and a different reinforcement detail (i.e., larger stirrup bars or additional stirrup legs) is needed. In this case, the word "*Inadequate" will be displayed in the column for the governing spacing and a corresponding note will appear.

4.2.6.2 Combine Struts and Subdivide Nodes

After horizontal ties are checked and the required reinforcement spacing is determined, several steps are executed through the "Combine_Struts_Subdivide_Nodes" sub to prepare the nodes before performing nodal strength checks. First, smeared nodes are identified. Smeared nodes within the strut-and-tie model for a multi-column bent cap are any nodes that do not abut a load or bearing plate and therefore do not have clearly defined geometries (see Section 2.11.4). The program considers all nodes and compares the x-coordinate of each node to the x-coordinates of applied loads and reactions. If there is no applied load or reaction at the x-coordinate of the node, the node is defined as a smeared node.

As previously explained (see Sections 3.3.2 and 4.2.4.1), if the designer chooses to distribute self-weight by manually inputting self-weight values as externally applied point loads in the "Factored Applied Loads" table on the *Inputs* sheet, any load that only consists of self-weight will not have corresponding loaded area dimensions within the table. In addition to nodes not associated with an external load or reaction, the program will also consider the nodes that are placed at the locations of manually-input self-weight loads to be smeared nodes. When considering the location of each node, if a node has the same x-coordinate as a load input into the "Factored Applied Loads" table but the cells for the loaded area dimensions are zero, the program defines the node as a smeared node.

All smeared nodes are stored in a matrix. Before executing any calculations related to the nodes, the program loops through the nodes in the model. For each node, the program first determines if it has been defined as a smeared node. If the node is smeared, no calculations will be performed, and the loop will continue to the next node.

After defining smeared nodes, the program combines adjacent struts when two or more struts enter a node from the same side. Node M in Figure 4.10(a) has Strut 1 and Strut 2 entering the node from the right side. Therefore, these two struts will be combined. In addition, if a node only has struts that enter from one side in addition to a vertical strut that enters the node, all the struts will be combined together. Node N in Figure 4.10(b) has two struts entering the node. Strut 1 is a vertical strut and Strut 2 enters from the right side of the node. No other struts enter the node. Therefore, Strut 1 and Strut 2 will be combined. To determine the struts to combine at a node, if any, the program first creates a "Members at Node" matrix specific to that node. This matrix contains the label of each member entering the node, the force in each member, and the angle at which it enters the node (i.e., the orientation of the member). This matrix is used to count the number of struts entering each side of the node.

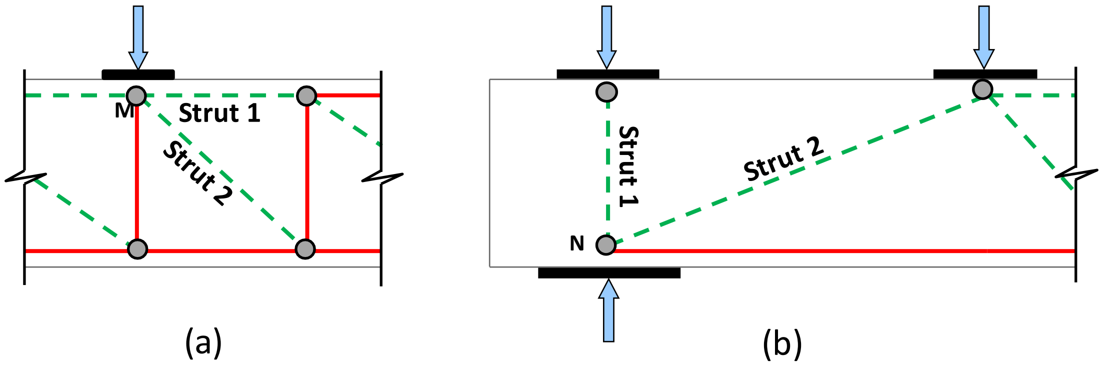


Figure 4.10: Nodes Where Struts are Combined

If two or more struts enter the node, the number of struts entering from the left side of the node is counted separately from the struts entering the right side. The program also determines if a vertical strut enters the node. If two or more struts enter the node from either side, the program combines the struts on that side together. The magnitude and angle of the resultant force are stored in the "Members at Node" matrix as one strut, replacing the struts that were combined. The same procedure is followed for struts that enter the node from only one side in addition to a vertical strut. If a node has a single strut entering the node from each side, no combination of struts is needed.

After combining struts at a node as necessary, the program determines if the node should be subdivided. Nodes are subdivided when diagonal struts enter the node from both sides. This occurs when the shear force in the structural component changes signs at the location of the node. In this case, the node is typically subdivided into two parts, shown in Figure 4.11. Once exception may occur. If an applied load is located directly above a support, a vertical strut will enter the node that is located directly above the support reaction. If diagonal struts also enter the node from both sides, the node is subdivided into three parts, as shown in Figure 4.12.

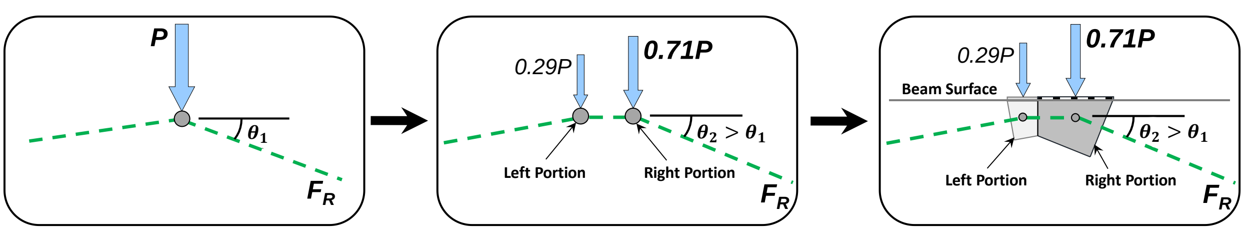


Figure 4.11: Subdividing a Node into Two Parts (adapted from Williams et al., 2012)

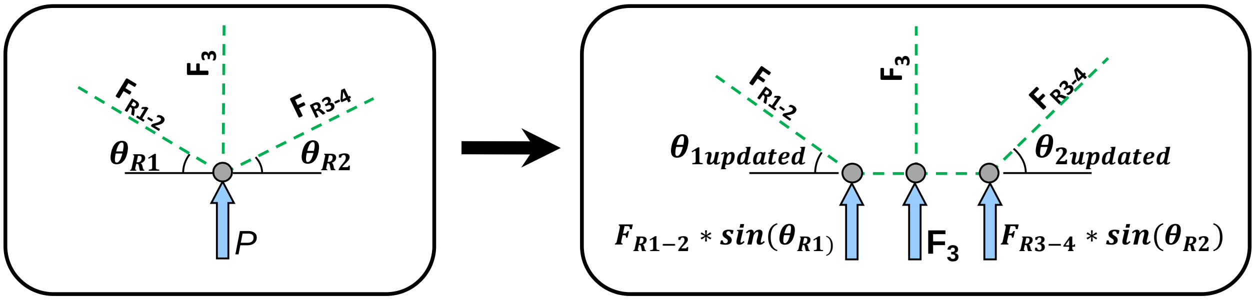


Figure 4.12: Subdividing a Node into Three Parts (adapted from Williams et al., 2012)

If the program determines a node should be subdivided, it begins by determining the x-coordinate of each part of the subdivided node. The node is subdivided based on the shear force on each side of the node, which is equal to the vertical component of the diagonal strut entering on each side. The process for subdividing a node into two parts is illustrated in Figure 4.11. Here, the vertical component of the strut on the left (i.e., the shear force on the left) is equal to 29% of the applied load acting at the node. Similarly, the vertical component of the strut on the right is equal to 71% of the applied load. The applied load is divided based on these percentages. The portion of the load plate corresponding with the bearing face of each of the two parts of the subdivided note also corresponds to these percentages. The left part of the node will have a bearing dimension that is 29% of the entire length of the load plate. The portion of the applied load acting on the left part of the node, 0.29P, will be centered on this bearing dimension.

A similar process is used to position the portion of the applied load acting on the right part of the node. By positioning the loads in this manner, uniform pressure is maintained over the load plate. The x-coordinate of each part of the subdivided node corresponds to the line of action of each portion of the applied load. The same procedure depicted in Figure 4.11 is followed for nodes subdivided into three parts. However, the applied load or reaction will also be divided into three parts, shown in Figure 4.12. The center part of the subdivided node will correspond to a vertical strut entering the node (Williams et al., 2012).

Subdividing a node causes the force and the angle (i.e., orientation) of the diagonal struts entering the node to change. Typically, the resulting changes are small. The angle at which a strut enters the node will always increase and the force in the strut will always decrease when a node is subdivided. A parametric study was performed to consider numerous potential nodes and the resulting strut forces and angles if the nodes were subdivided. It was determined that considering the angle change but neglecting the force change leads to a reasonably conservative result. Therefore, the program updates the angles of diagonal struts after subdividing a node, as indicated in Figure 4.11, but does not update the strut force.

For nodes subdivided into three parts, the program does not update the angle of the vertical strut that enters the node. While this angle will likely change slightly due to the subdivision of the node, the change is assumed to be negligible, and allowing the strut to remain vertical simplifies the geometry of the node. The slight angle change for the vertical strut in the node depicted in Figure 4.12 is shown in Figure 4.13. The node at the F_3 reaction will be located slightly off center, and the strut will be at a slight angle. However, if the diagonal struts entering the node, represented by $F_{R1\text{-}2}$ and $F_{R3\text{-}4}$, have vastly different magnitudes, the node at the F_3 reaction could be located such that the designer may need to consider the angle change of the vertical strut by performing additional calculations.

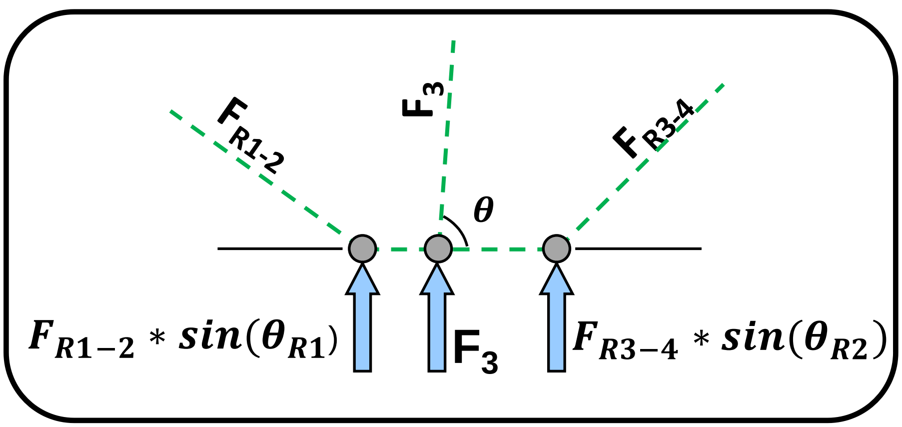


Figure 4.13: Slight Change in Angle for Vertical Strut when Node is Subdivided into Three Parts

After determining the x-coordinates of each portion of a subdivided node, the angles at which the diagonal struts enter the node are updated based on the new x-coordinates. The updated angle for each strut is stored in the “Members at Node” matrix. The procedure for subdividing nodes is summarized in Article C5.8.2.2 of AASHTO LRFD (2017). A detailed example of combining struts and subdividing a node is provided in Section 4.4.4 of Williams et al. (2012).

After combining struts and subdividing nodes, the updated force and angle information for each node is displayed in the "Nodes with Combined Forces and Updated Angles" table on the *Nodes and Members* sheet. The values presented in this table reflect resultant struts and subdivided nodes. All nodes except smeared nodes are displayed in this table, even if no struts entering the node are combined and the node is not subdivided. Each force acting at a node and its angle (i.e., orientation) are provided. Positive force values represent tensile forces, and negative force values represent compressive forces. The angles displayed are measured from the positive x-axis.

It should be noted that the change in the angle of a diagonal strut due to the subdivision of a node also affects the orientation at which the strut enters the node located at its opposite end. The strut angle listed for the node located at the opposite end of the member should also be updated in the "Nodes with Combined Forces and Updated Angles" table. To accomplish this, the program iterates through each node in the table again. If the angle of a strut entering a node has changed due to the subdivision of a node at its opposite end, the angle is updated accordingly and displayed in the table.

4.2.6.3 Plot Node Figures

The "Plot_Node_Figures" sub creates a new sheet, *Node Figures*, and hides gridlines on the sheet. The sub plots graphs, called charts, for each node included in the "Nodes with Combined Forces and Updated Angles" table. These are displayed on the *Node Figures* sheet in rows with three charts per row. The sub follows the same procedure to create the figure for each node. First, the node label is added as the chart title, the gridlines are removed from the chart, and the chart is sized. The node is then plot at the center of the chart and labeled and the horizontal axis is plotted. Each force entering the node is then plotted based on the angle at which it enters the node and formatted based on whether it is a compressive or tensile force. The force value and angle at which it enters are labeled. Lastly, any applied loads or reactions acting at the node are plotted.

4.2.6.4 Nodal Strength Checks

After the magnitude and orientation of the forces acting at each node are determined, the "Nodal Checks" sub is used to establish the geometries of the nodes and check the strength of each nodal face. Nodal geometries and strength checks are summarized on the *Nodal Checks* sheet of the program. For each node, the program first determines the node type (CCC, CCT, or CTT) based on the members that intersect at the node. For a node that has been subdivided, the node type of each part of the node (left, right, or middle) is determined separately based on the unique forces acting on that part.

Next, the geometry of the node (or portion of the node) is determined, as shown in Figure 4.14. Each node has three faces: a bearing face, a back face, and a strut-to-node interface. The program determines the length of each of these faces for every node that is not a smeared node. For nodes that are not subdivided, the length of the bearing face, l_b, is taken as the length of the load plate or bearing area. For subdivided nodes, the length of the bearing face for each part of the node is the dimension of the corresponding portion of the load plate or bearing area that was previously stored (see Section 4.2.6.2). The length of the back face, h_a, is taken as twice the distance from the longitudinal chord of the strut-and-tie model along which the node is located to either the top or bottom surface of the structural component. If the node is located along the top chord, the distance to the top surface of the member is used, and if the node is located along the bottom chord, the distance to the bottom surface of the member is used. The length of the strut-to-node interface, w_s, is calculated using Equation (4.4):

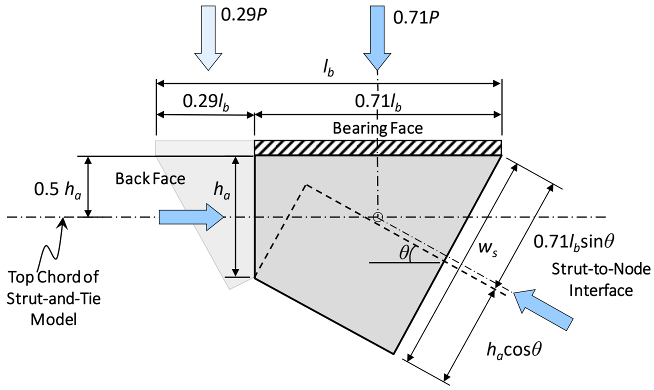
$$w_s = l_b sin\theta + h_a cos\theta \quad (4.4)$$

where:

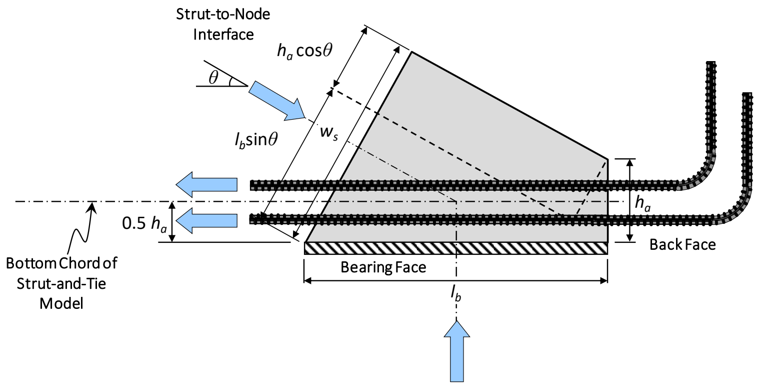
l_b = length of the bearing face (in.)

θ = angle between the diagonal strut and the longitudinal axis of the member

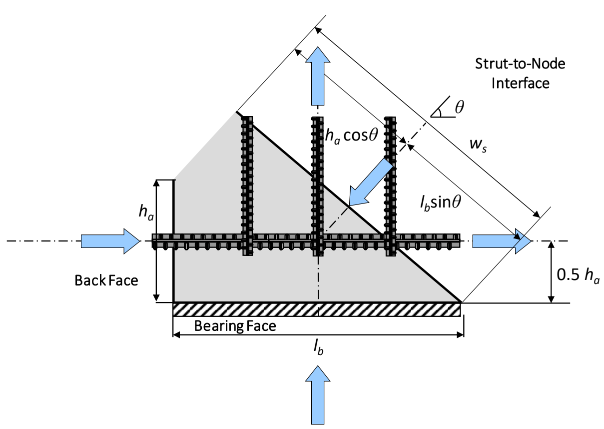
h_a = length of the back face (in.)



(a)



(b)



(c)

Figure 4.14: Proportions of Three Node Types

All nodes that are not smeared nodes within the strut-and-tie model for a multi-column bent cap abut a load plate or bearing area. Therefore, the width of all three faces of a node (or portion of a node) measured along the transverse width of the structural component (i.e., the width of the node measured into the page in Figure 4.15) is taken as the width of the load plate or bearing area. The confinement modification factor, *m,* is then calculated for the node using Equation (4.5) where A_1 and A_2 are defined according to Figure 4.15 (see Section 2.11.5):

$$m = \sqrt{\frac{A_2}{A_1}} \leq 2.0 \qquad (4.5)$$

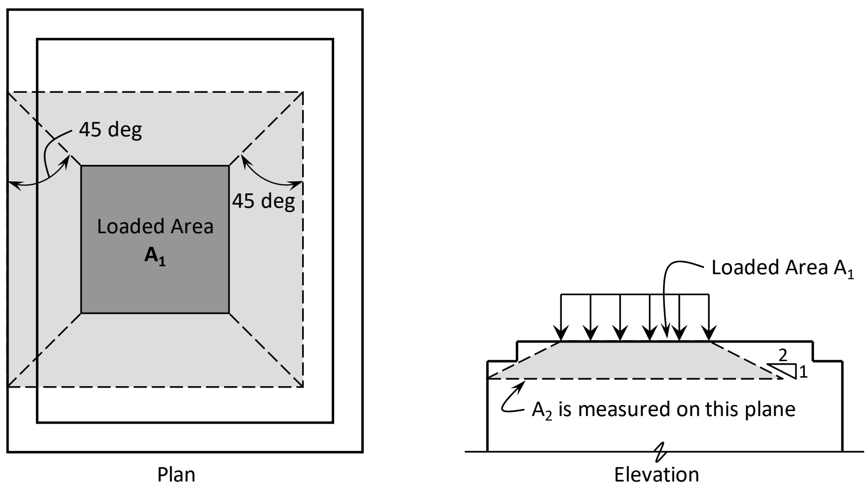


Figure 4.15: Bearing Area Definitions for Confinement Modification Factor (adapted from Williams et al., 2012, and AASHTO LRFD, 2017)

The program uses the loaded area or bearing area dimensions input by the user to calculate the area A_1. The area A_2 is calculated next in accordance with Figure 4.15. If the loaded area or bearing area is square, A_2 is defined by a square with dimensions equal to the width of the structural element.

Next, the factored design strength of each face of the node is calculated. The appropriate efficiency factor, ν, is selected for each face based on the node type. The efficiency factors specified in Article 5.8.2.5.3a of AASHTO LRFD (2017) are provided in Table 4.2.

Table 4.2: Concrete Efficiency Factors, ν (from AASHTO LRFD, 2017)

	Node Type		
Face	**CCC**	**CCT**	**CTT**
Bearing Face	0.85	0.7	$0.85 - \frac{f_c'}{20\ ksi}$ $0.45 \leq \nu \leq 0.65$
Back Face			
Strut-to-Node Interface	$0.85 - \frac{f_c'}{20\ ksi}$ $0.45 \leq \nu \leq 0.65$	$0.85 - \frac{f_c'}{20\ ksi}$ $0.45 \leq \nu \leq 0.65$	

The factored nominal resistance, ϕP_n, of each face of the node is calculated using Equation (4.6).

$$\phi P_n = \phi * f_{cu} * A_{cn} \tag{4.6}$$

where:

ϕ = resistance factor (= 0.70 according to Article 5.5.4.2 of AASHTO LRFD (2017))

$f_{cu} =$ limiting compressive stress at the face of the node (ksi)

$A_{cn} =$ effective cross-sectional area of the nodal face (in.^{2})

The value of f_{cu} is calculated using Equation (4.7):

$$f_{cu} = m * \nu * f_c' \tag{4.7}$$

where:

m = effective confinement modification factor

ν = concrete efficiency factor

f_c' = specified compressive strength of the concrete (ksi)

The strength of the bearing face of the node is calculated first. Only one bearing check is performed per node, even if the node is subdivided. The entire loaded or bearing area as input by the designer on the *Inputs* sheet is used to calculate strength rather than the bearing dimensions of the subdivided portions of the node. The factored design strength is later compared to the entire factored load or support reaction acting at the node. If girder loads are combined together before entering the loads on the *Inputs* sheet, the designer is responsible for ensuring the bearing strength is adequate for each individual girder.

The strength of the back face of the node is determined next, if applicable. It is only necessary to check the strength of a back face if a direct compressive force acts on that face. Article 5.8.2.5.3b of AASHTO LRFD (2017) states that bond stresses resulting from the force developed in a tie do not need to be applied to the back face of a CCT node. Even though not explicitly mentioned in the specifications, the same phenomenon may occur at CTT nodes as well. The program assumes that bond stresses do not need to be applied to the back face of either node type. The strength of the back face of a node will only be checked if a direct compressive force acts on that face. If the strength check is not performed, "Check N/A" will be displayed in the *Nodal Checks* sheet for the back face.

When a node is subdivided, the back face is shared between two parts of the node, shown in Figure 4.16. The back face on both sides of the subdivided node experiences the same force demand. Even if a node is subdivided into three parts, the back faces that separate the three parts will have the same dimensions and the same force demand, assuming that the vertical strut entering the node remains vertical, as shown in Figure 4.17. Therefore, only one back face check is performed for a subdivided node.

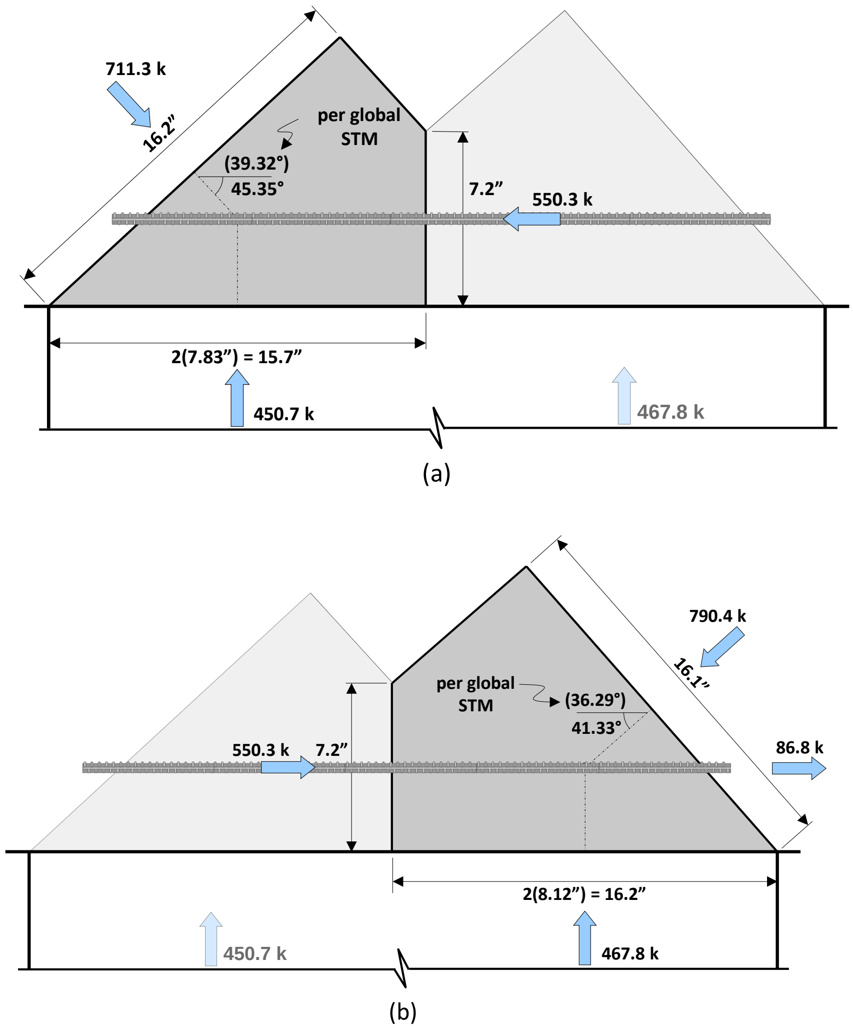


Figure 4.16: Node Subdivided into Two Parts – (a) Left Portion of Node; (b) Right Portion of Node (from Williams et al., 2012)

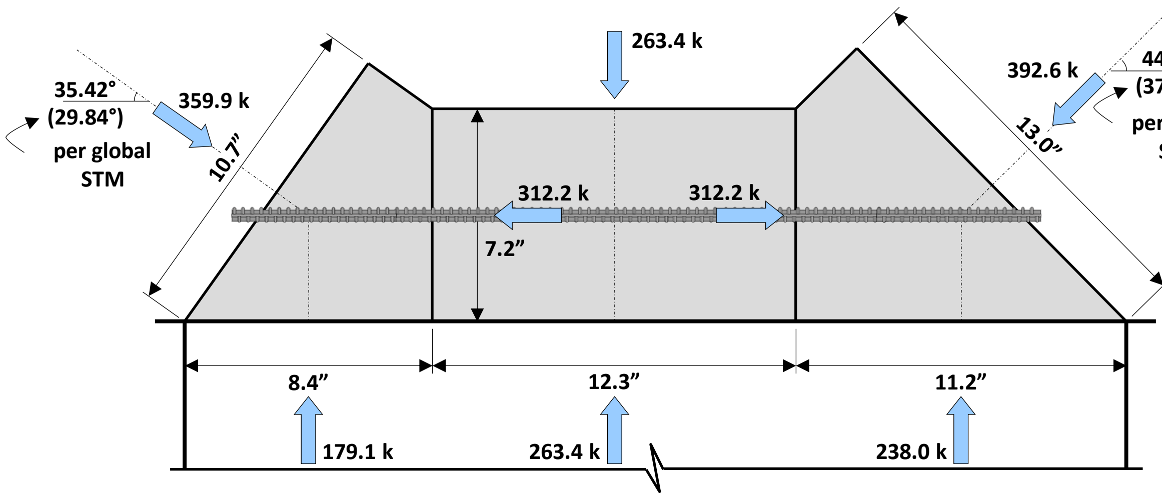


Figure 4.17: Node Subdivided into Three Parts (from Williams et al., 2012)

The strength of the strut-to-node interface is calculated last. The force in the vertical or diagonal strut entering the node is the demand on the strut-to-node interface. Unlike the bearing and back face checks, a separate strut-to-node interface strength check is needed for each portion of a subdivided node.

After calculating the factored nominal resistance, ϕP_n, and determining the demand, P_u, on each face of the node, the values are compared. For the design strengths to be adequate, Equation (4.8) must be satisfied:

$$\phi P_n \geq P_u \tag{4.8}$$

The result of the strength check on each nodal face is displayed in the corresponding "Pass?" column on the *Nodal Checks* sheet. If the design strength of the nodal face is satisfactory, "OK" will be displayed in this column. If the strength is inadequate, "NG" (no good) will be displayed, and red bolded text will indicate the insufficient strength.

4.2.6.5 Anchorage Checks

The last design step is performed using the "Anchorage" sub, which determines if the longitudinal reinforcement can be properly developed near the ends of the structural component. The required development length(s) for a straight and/or hooked bar is input by the user for the top and bottom longitudinal reinforcement. At the appropriate nodal regions, the computer program calculates the available length in the member for development of the bars and compares this value to the required length.

To perform the anchorage check, the same procedure is followed for each end of the structural component. If no negative moment region exists along the component, only the bottom reinforcement is checked. Otherwise, the reinforcement along both the top and bottom chords of the member is considered.

At each end of the member and for each applicable chord, the program first identifies the outermost horizontal tie in the strut-and-tie model. Once the tie is identified, the node at which anchorage will be checked is determined. This node is located at the end of the tie that is closest to the end of the structural component.

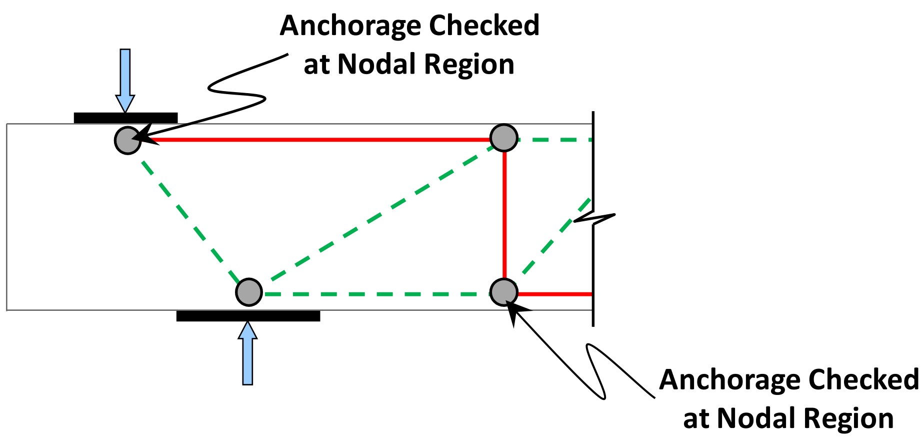


Figure 4.18: Nodal Regions where Anchorage Checks are Performed

At this node, the available length for development is the distance from the end of the member, minus cover, to the point at which the centroid of the reinforcement exits the extended nodal zone (see Figure 2.17). Within the “Anchorage” sub, the clear cover input by the user is subtracted from the distance from the end of the member to the inside edge of the load plate or support, labeled as *x* in Figure 4.19. To calculate the total available length, the resulting value is added to the distance from the inside edge of the load plate or support to the point at which centroid of the reinforcement exits the extended nodal zone, labeled as *y* in Figure 4.19.

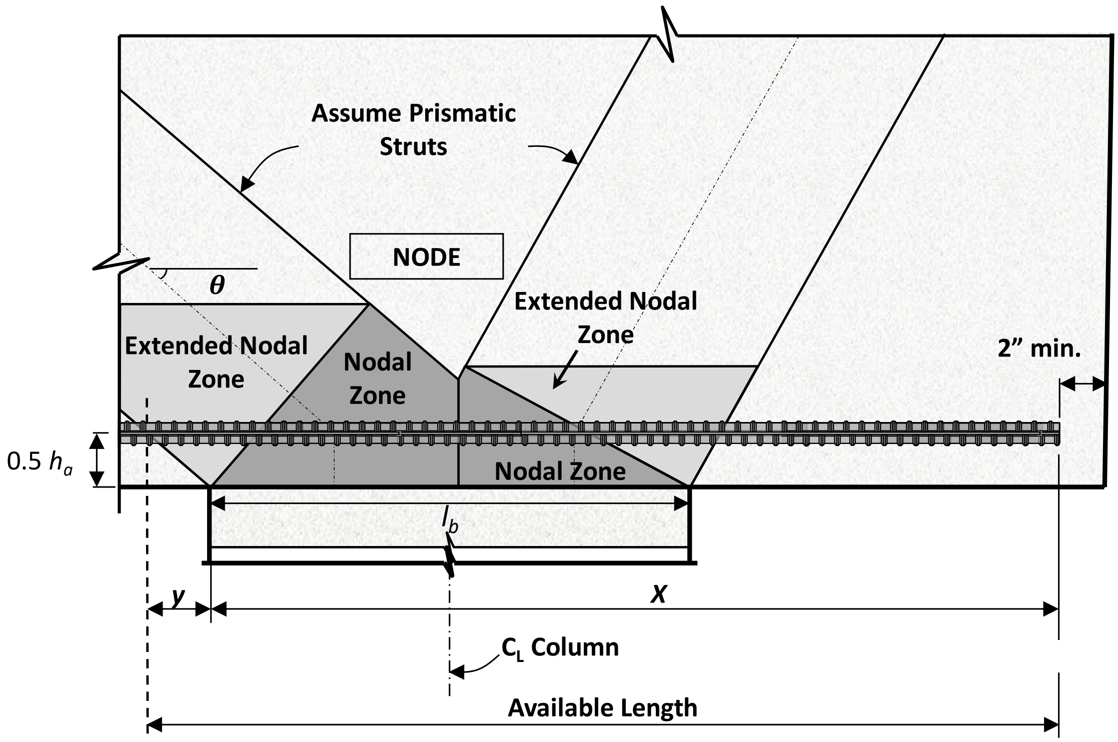


Figure 4.19: Available Length Calculation for Pier Caps Performed by the STM Program (adapted from Williams et al., 2012)

After calculating the available length for bar development at each end of the structural member, the values are compared to the required development length(s) input by the user. The anchorage checks are summarized on the "Anchorage for Ties" table on the *Reinforcement* sheet. Each node at which anchorage was considered is listed along with the corresponding available length and the required development length(s) input by the user. For each type of bar, straight or hooked, for which the user input a development length, a "Pass?" column indicates whether sufficient length is available. If the available length is greater than the required length, "OK" is displayed in the "Pass?" column. If insufficient length is available to develop the bar, "NG" (no good) is displayed, and red bolded text is used to indicate the deficient design.

If any bars are terminated along the length of the structural component, the user is responsible for ensuring the bars that are cut off are adequately developed at the locations they are needed for carrying tie forces. The program does not perform anchorage checks at intermediate points along the length of the member. Anchorage is only checked for the outermost ties near each end of the structural component.

4.2.7 Formatting

After completing all analysis and design steps in accordance with STM specifications, the *Inputs* sheet and the various output sheets are formatted to improve the visual appearance of the information and to enable the designer to easily interpret the results. This is accomplished through the "Formatting" sub. The *Inputs* sheet is formatted based on the number of applied loads and reactions defined by the user. The sizes of the "External Loads" and "Reactions" tables are adjusted to correspond to the number of inputs. Furthermore, table borders are added to the content in each output sheet. Titles for tables and columns are bolded and centered. The output tabs are arranged to ensure that worksheets with information directly relevant to the STM design steps precede the worksheets that include details of the structural analysis. Columns are autofit to ensure that all text is visible. Cell A1 is set as the active cell on each sheet so that the top left of each worksheet is displayed when a sheet is opened.

4.2.8 Resetting the Code

A button titled "Reset Program" is included on the *Inputs* sheet. Clicking this button runs the "Reset" sub which deletes all worksheets except the *Instructions* and *Inputs* sheets. In other words, all program outputs from a previous STM analysis and design will be deleted. The code must always be reset before running the program again using the "Run Program" button. If the user does not first reset the program, a message will appear asking the user to either reset the program and proceed running the code or cancel the run. If the user wishes to compare multiple outputs for various designs, multiple workbooks should be utilized. Users should keep in mind that all output sheets will be deleted as soon as the "Reset Program" button is clicked.

4.2.9 Rerunning the Code

In addition to the "Run Program" and "Reset Program" buttons, a button titled "Rerun Program" is included on the *Inputs* sheet. This button starts the "Rerun" sub and should only be used after running the program once. The program should not be reset before using this procedure. The "Rerun" sub is used when the engineer believes the strut-and-tie model developed by the program contains unrealistic nodes or a better model can be developed. In this case, the engineer should delete the nodes he or she feels are unnecessary or unrealistic from the "Nodes" table in the *Nodes and Members* sheet. The contents in the entire row of the table should be deleted. In other words, the contents in the “Label” cell, “x (ft)” cell, and the “y (ft)” cell should all be deleted for the node the designer wants to remove from the strut-and-tie model. Only smeared nodes should be removed from the model. Any node at an applied load or reaction should never be deleted from the model.

After deleting nodes in the table that are believed to be unnecessary, the engineer should not make any other adjustments to any sheets within the program. The user should only click the “Rerun Program” button on the *Inputs* sheet. When the button is clicked, the “Rerun” sub starts. This sub uses the “Nodes” table in the *Nodes and Members* sheet as the locations for all nodes within the program. It relabels and indexes the nodes, starting with a label of A and an index of 1 for the first node in the table.

When the program is finished running the first time, all previously stored variables are cleared. Therefore, when the program is rerun, user inputs must be stored as variables again. The same user inputs from the first run of the program are read into the program using the “Inputs” sub. The “Self-Weight Distribution” sub is also run again. However, the *Shear Diagram, Moment Diagram,* and *Shear and Moment Diagram Calcs* sheets are not deleted when the program is rerun, and the information stored on these sheets can still be accessed by the program. Therefore, the “Structural_Analysis” sub, the “Shear_Moment_Diagrams” sub, and the “Label_Shear_Moment” sub are not run again. Furthermore, because the location of the top chord in the strut-and-tie model is already known, the “Top_Chord_Optimization” sub is not run again. Lastly, since the locations of nodes are determined from the table on the *Nodes and Members* sheet, the program does not need to run the “Node_Locations” sub.

The remaining output sheets are deleted from the workbook and replaced with output sheets based on the new strut-and-tie model. All subs from the “Create_Members” sub to the “Formatting” sub listed in Section 4.2.2 are run again to develop an updated model and perform the STM design steps. These subs update the *Nodes and Members* sheet and create new *Strut-and-Tie Model, Nodes Figures, Reinforcement,* and *Nodal Checks* sheets.

4.3 Additional Limitations and Assumptions

The computer program contains additional limitations that have not been mentioned in previous sections. The limitations are not expected to affect the design of most typical multi-column bent caps. Moreover, some assumptions were required during the development of the program. It is important for the designer to understand all limitations and assumptions associated with the computer program to ensure that the structural elements being designed fit within the intended use of the program.

One limitation of the program is that it cannot correctly handle a diagonal strut that will change from being oriented from the upper left to the lower right to being oriented from the upper right to the lower left, or vice versa, due to the subdivision of a node(s). A case in which a strut changes orientation in this manner due to the subdivision of a node is illustrated in Figure 4.20. The diagonal strut connecting Nodes M and X is originally oriented from the upper right to the lower left. After subdividing Node X, the strut is oriented from the upper left to the lower right. The program is unable to perform STM design checks for a model in which this occurs.

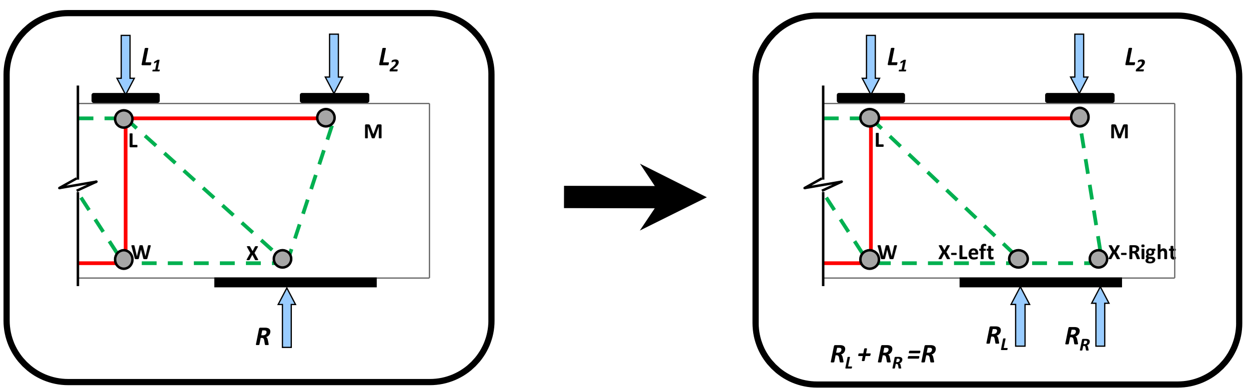


Figure 4.20: Example of Limitation within the Program – Changing Strut Orientation due to Subdivision of a Node

Furthermore, the program does not consider the contribution of any nonprestressed compressive reinforcement when performing strength checks for the back face of a CCC node. Article 5.8.2.5.1 of AASHTO LRFD (2017) states that "[w]here the back face of a CCC node contains nonprestressed compressive reinforcement the resistance…may be augmented with the yield resistance of the nonprestressed reinforcement" (AASHTO LRFD, 2017). If the engineer wishes to consider this contribution, he or she must perform the necessary calculations. The engineer should also check that the reinforcement is properly developed in compression.

The program has the following additional limitations that have not been discussed in previous sections:

- Supports only provide upward vertical restraint. The program cannot analyze the structural component or develop a strut-and-tie model if supports provide additional restraint, or if the reaction is downward.
- The program can only be used for prismatic members.
- Built-up bearing seats are not considered in the program. Suggestions for developing nodal geometries when built-up bearing seats are present are provided in Williams et al. (2012).

The following assumptions that have not been introduced in previous sections are made within the program:

- The procedure developed for the computer program assumes that a positive moment region exists at some location along the length of the bridge element.
- The program assumes the entire bridge element is defined by D-regions. If any B-regions exist along the length of the member, they will be designed using the STM. The engineer is responsible for performing any additional design checks based on sectional design procedures as required in AASHTO LRFD.

The engineer should also be aware that the program only performs design checks related to the STM provisions presented in Article 5.8.2 of AASHTO LRFD (2017). Using the program to aid in design steps of the strut-and-tie method does not ensure that other code provisions are satisfied. It is the responsibility of the engineer to ensure that all code provisions are satisfied and that the structural element being designed meets the limitations and assumptions of the computer program.

4.4 Interpretation of Program Outputs

As with any structural analysis software, it is essential that the engineer properly interprets the outputs that are provided after running the program. The program will not automatically output a design that satisfies STM design requirements. Any strength checks that are not satisfied are displayed in red bolded text. The engineer is responsible for updating the details of the bent cap to develop a design that satisfies all requirements. Several parameters can be adjusted to develop an adequate design. For example, the concrete strength can be increased, reinforcement can be added, or the geometric conditions of the member can be changed. It is important for the engineer to be aware that if a negative moment region exists at some location along the length of the element, the top chord of the strut-and-tie model will be placed to coincide with the centroid of the longitudinal reinforcement along the top of the member. If the back face of a node located along the top chord at a positive member region does not have the required strength, the program does not allow the horizontal strut at this node to be shifted downward to provide a larger back face. Instead, the depth of the member will need to be increased, or the location of top longitudinal reinforcement can be shifted downward. If a more refined model that considers a top chord with a varying location along the length of the member (i.e., a model with a varying internal moment arm *jd*) is desired, it must be developed by the engineer outside of the program.

4.5 Design Example: Five-Column Bent Cap

The design example presented in the following subsections is intended to familiarize engineers with the implementation of the computer program to aid in performing STM design procedures for a multi-column bent cap. The computer program completes the design steps and creates the outputs that are presented by the procedures described in Section 4.2. A detailed STM design of the same five-column bent cap used in the following example is the focus of Chapter 4 of Williams et al. (2012). The following subsections specifically focus on the use of the computer program to aid in the design of the member. The reader should refer to Williams at al. for the details of each step of the design using the strut-and-tie method.

4.5.1 Bent Cap Geometry, Material Properties, and Loading

The layout of the five-column bent cap is presented in Figure 4.21 and Figure 4.22. Five 3-ft diameter circular columns support the bent cap. The longitudinal column bars are anchored in the cap using straight bars. For the design, each column is assumed to act as a pinned support. Only rectangular bearing area dimensions can be input into the computer program. Therefore, the circular cross-sections of the columns are replaced by square bearings with the same area. These equivalent square bearing areas are 31.9 in. by 31.9 in.

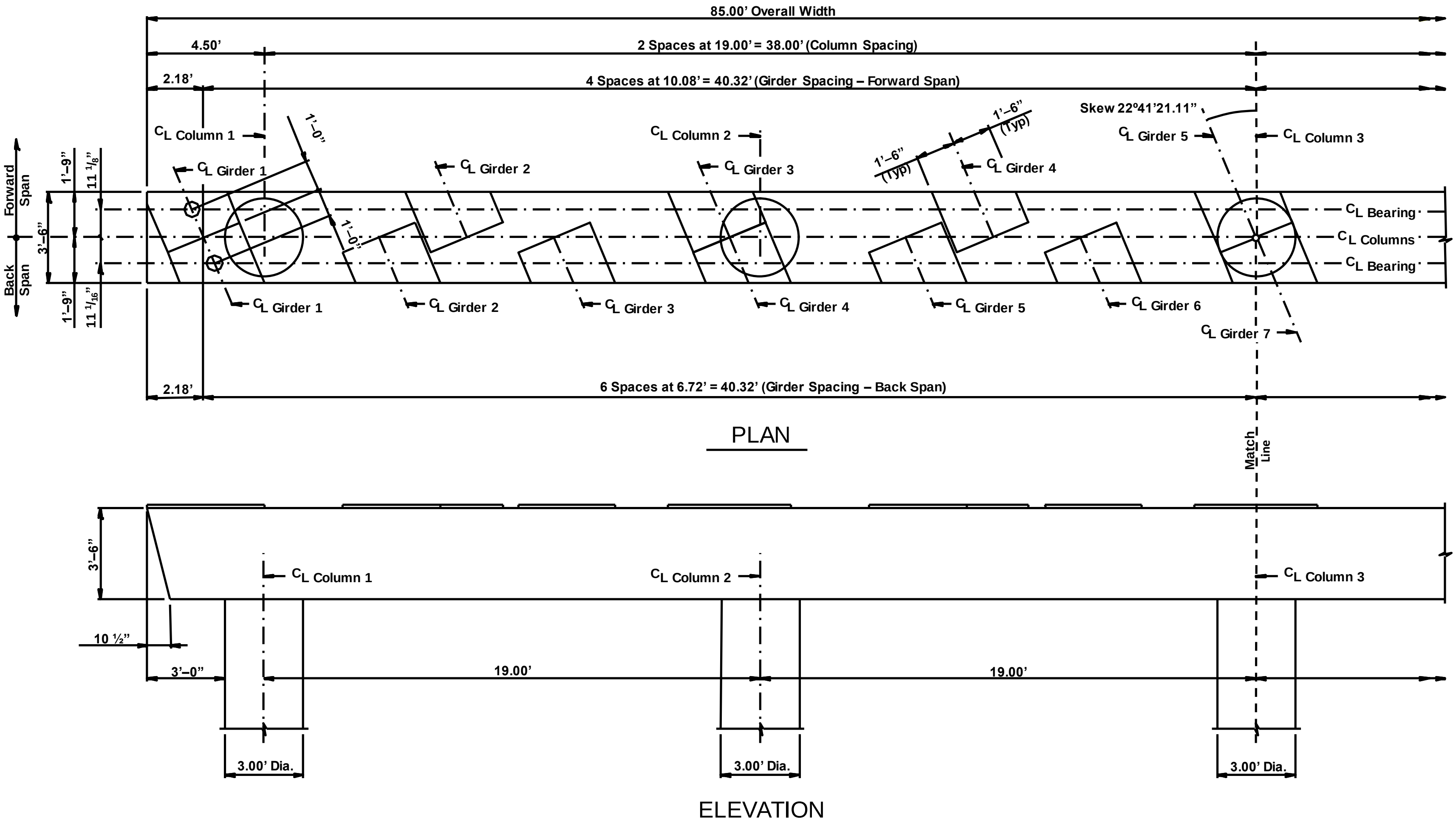


Figure 4.21: Plan and Elevation Views of Bent Cap – Left (from Williams et al., 2012)

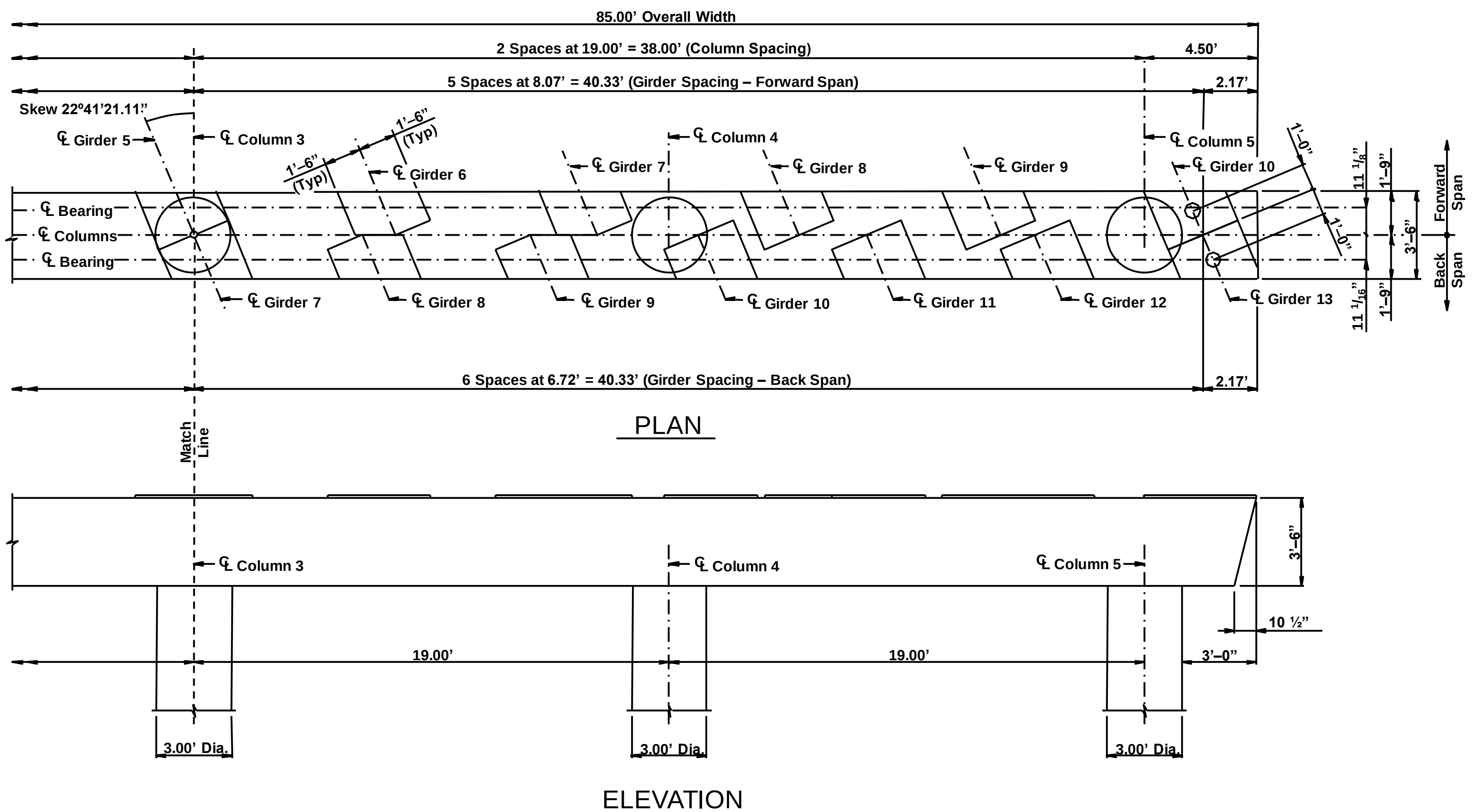


Figure 4.22: Plan and Elevation Views of Bent Cap – Right (from Williams et al., 2012)

A total of 10 prestressed concrete girders from the forward span are supported on the bent cap along with 13 prestressed concrete girders from the back span. The cross-sections of the superstructure for both spans are shown in Figure 4.23. Due to the skew of the bridge spans, the bearing pads supporting the prestressed girders are also oriented at a skew. A procedure to convert the skewed bearing pads to square bearing areas centered on the longitudinal axis of the girder is described within the original design example in Williams et al. (2012). The procedure performed by Williams et al. also considers the effect of built-up bearing seats. Furthermore, within the original design example, loads that act on the bent cap in close proximity to one another are combined together to allow for a reasonable strut-and-tie model to be developed. The consideration of bearing seats and the simplification of the loading conditions results in 16.2-in. by 16.2-in. loaded areas for single girder loads and 23.0-in. by 23.0-in. loaded areas for two girder loads that have been combined, as illustrated in Figure 4.24. These same loaded areas are used for the design example presented here. The reader can therefore compare the design results in this document with the example in Williams et al.

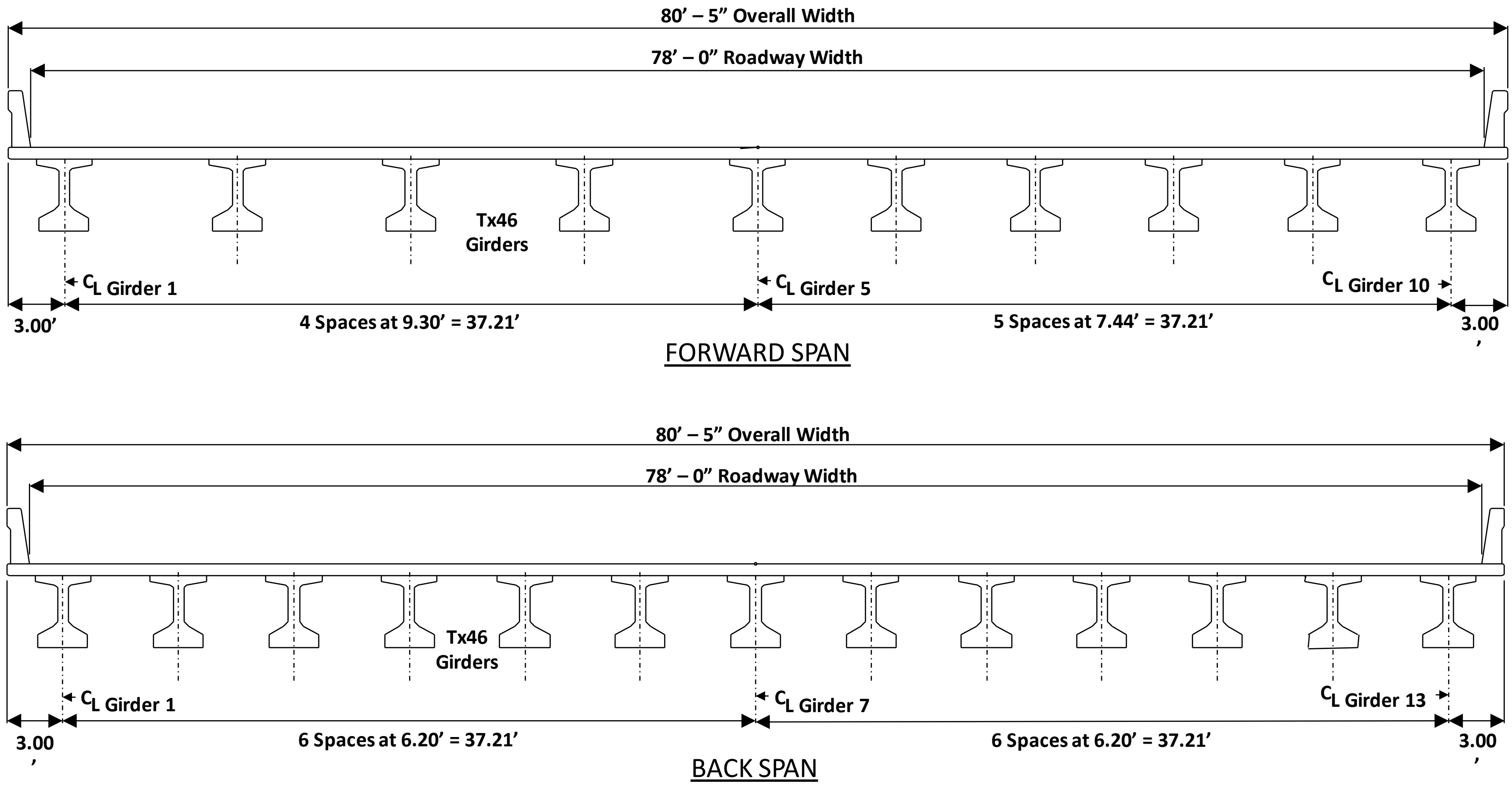


Figure 4.23: Superstructure Cross-Sections (from Williams et al., 2012)

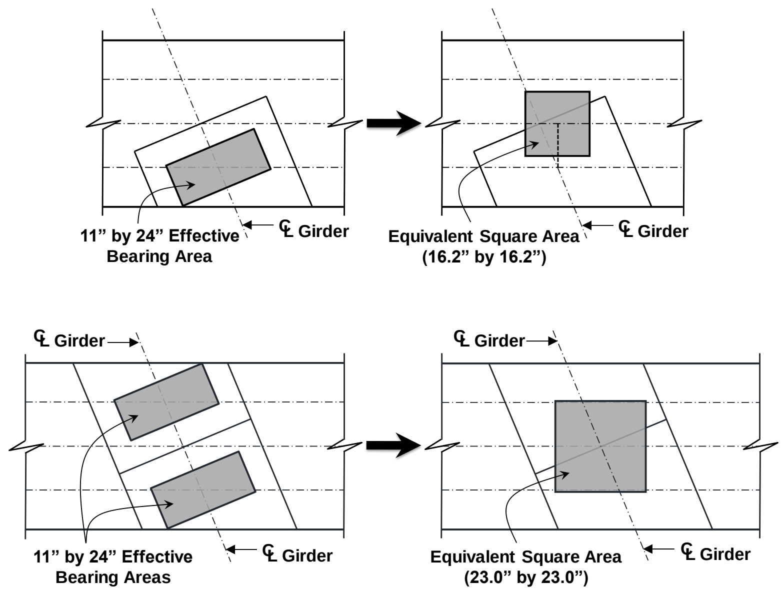


Figure 4.24: Assumed Bearing Area for Girder Loads

The computer program can only consider one load case for a single run of the program. The load case considered in the current design example matches the load case used for the design of the bent cap in Williams et al. (2012). The factored loads acting on the bent cap presented in Williams et al. and used in the current example are shown in Figure 4.25. The factored self-weight based on tributary widths has already been added to the applied girder loads. As stated in Section 4.3, the program can only be used for prismatic members. The tapered ends of the bent cap as shown in Figure 4.21 and Figure 4.22 will be neglected in the computer program, and the cap will be treated as an 85-ft long prismatic member. Despite this small difference in geometry, the self-weight loads will be kept the same as those applied to the structure in Williams et al. to allow direct comparisons between the design presented in that document and the outputs from the computer program. Therefore, the applied loads input into the program will include the factored self-weight, consistent with Figure 4.25.

Within the design example, the specified concrete compressive strength, f_c', is taken as 4 ksi and the specified yield strength, f_y, of all reinforcement is 60 ksi. Concrete is normalweight.

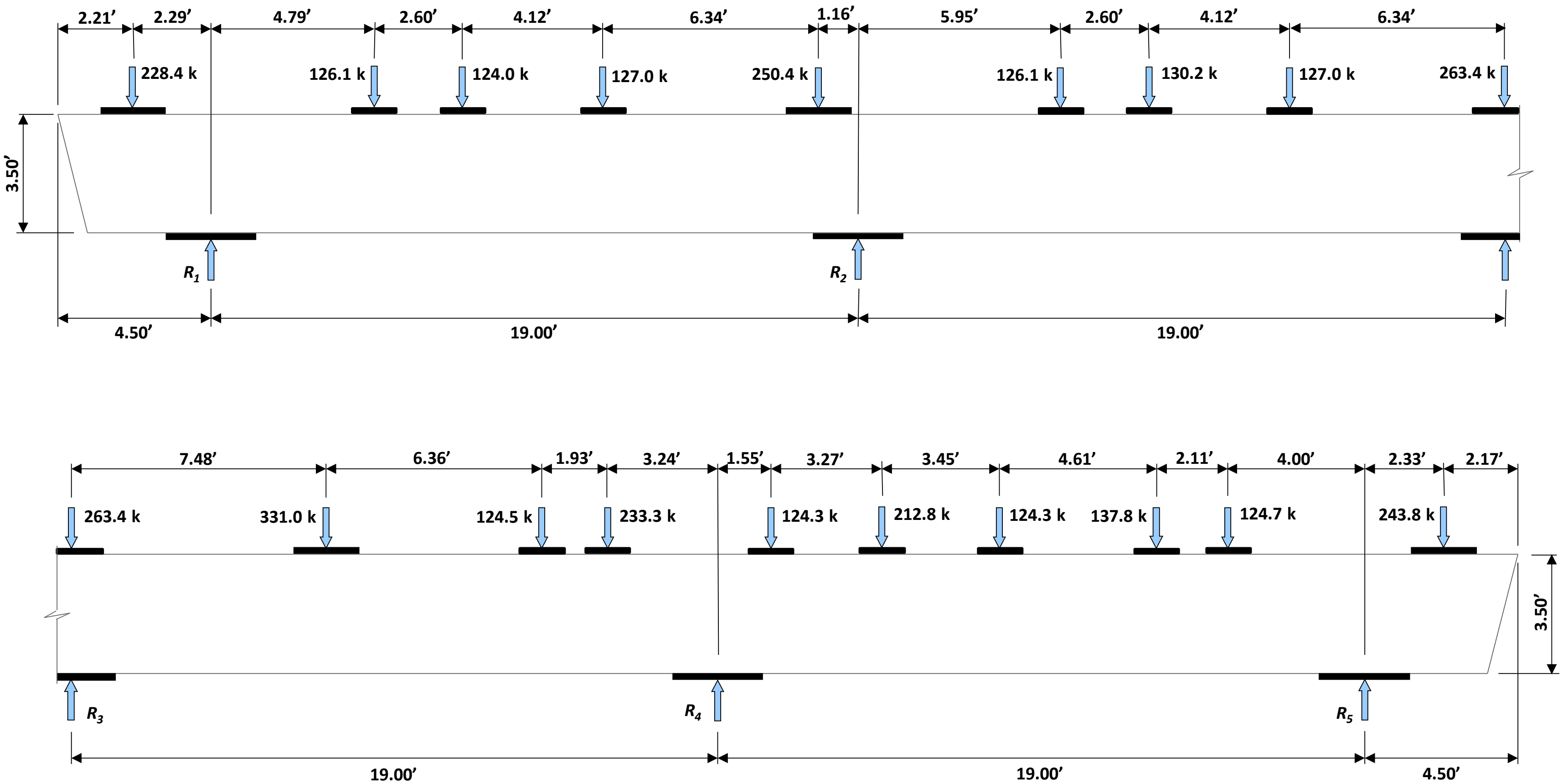


Figure 4.25: Factored Loads and Locations of Reactions (adapted from Williams et al., 2012)

4.5.2 User Inputs

To begin using the computer program, information about the structural component and the load case being considered is entered into the tables on the *Inputs* sheet. The *Inputs* sheet with all values properly entered for the five-column bent cap is shown in Figure 4.26. In the "Geometric Properties" table, the overall dimensions of the prismatic member are input: a length of 85 ft, a height of 3.5 ft, and a width of 3.5 ft. The engineer has the option of entering these values in units of feet or inches. Units of feet are used for this design example and the "ft" dropdown is selected from the menu in the units cell. In the "Load Factor" table, the self-weight factor is input as zero because the self-weight will be manually insert as an applied load. Next, the specified compressive strength of concrete is input into the "Concrete Material Properties" table. The unit weight of concrete, w_c, is kept as the default value of 150 pcf for normalweight concrete. However, because the self-weight factor is input as zero, the unit weight of concrete will not be used within the program.

Information for the stirrups and horizontal crack control reinforcement (i.e., skin reinforcement) is entered into the corresponding tables. As shown in Figure 4.26, the specified yield strength of reinforcement is kept as the default value of 60 ksi in both tables. No. 5 bars are used for both stirrups and horizontal crack control reinforcement. For the final design of the bent cap in Williams et al. (2012), two stirrup legs are provided along the length of the member except for a small portion where four stirrup legs are needed. For the current design example, two legs are assumed along the entire length of the cap. Furthermore, for skin reinforcement, the default of two horizontal bars, one near each side face, are provided through the width of the member within spacing s_h.

Information for the longitudinal reinforcement in the top and bottom of the member is also entered into corresponding tables. As shown in Figure 4.26, the default value of 60 ksi for specified yield strength of reinforcement is kept. No. 11 bars are used for both top and bottom longitudinal reinforcement. Four No. 11 bars are used along the bottom of the member, and seven No. 11 bars are used along the top of the member. Development length for a straight and hooked No. 11 bar is calculated according to Article 5.10.8.2.1a and Article 5.10.8.2.4a of AASHTO LRFD, respectively (2017). For straight bars along the top of the member, the reinforcement location factor, λ_{rl}, is equal to 1.3. The reinforcement confinement factor, λ_{rc}, is calculated and applied to the development length equation for both straight and hooked bars along the bottom of the member. For straight bars, λ_{rc} is equal to 0.4 where K_{tr} is assumed to be zero and c_b is 3.58 in. For hooked bars, λ_{rc} is equal to 0.8. The rest of the factors are assumed to be 1. The development lengths calculated for straight and hooked bars along the top and bottom of the member are input into the "Longitudinal Reinforcement" table. For a straight bar, the required development length for the reinforcement along the top of the member is 52.8 in., and the required development length for the reinforcement along the bottom of the bar is 40.6 in. The required development length of a hooked bar is 21.4 in. The clear cover at the end of the bar is assumed to be 2 in. for both top and bottom longitudinal reinforcement.

The applied load and reaction information is entered into corresponding tables. The "Factored Applied Loads" table includes applied loads from the girders with loaded area dimensions and self-weight loads entered without loaded area dimensions. The self-weight loads are added in a separate row in order to ensure the bearing checks performed by the computer program are accurate. Reaction locations and equivalent square bearing areas are entered into the "Reactions" table. Locations of the reactions are shown in Figure 4.25. The equivalent square bearing area for each column is 31.9 in by 31.9 in as discussed in Section 4.5.1. Figure 4.26 shows the *Inputs* sheet with applied load and reaction information entered.

Before inputting information on this tab, see **Instructions** tab.

Notes:
-Click on any cell with ***BLUE*** *text for additional information .*
-Units in ***PURPLE*** *text can be changed. Units in* ***BLACK*** *text cannot be changed. Input standard bar size (e.g., input "5" for a No. 5 bar).*

Reset Program | **Run Program** | **Rerun Program**

Geometric Properties		
Length	85	ft
Height	3.5	ft
b_w	3.5	ft

Stirrups		
f_y	60	ksi
Bar Size (no.)	5	
No. of Legs	4	

Load Factor		
Self-Weight Factor		

Horizontal Crack Control Reinforcement		
Bar Size (no.)	5	
# Bars/s_h	2	

Concrete Material Properties		
f'_c	4	ksi
w_c	150	pcf

Longitudinal Reinforcement

Bottom Reinforcement			
f_y		60	ksi
No. of Layers		1	
Development Length Straight		40.6	in.
Development Length Hook		21.4	in.
Clear Cover at End of Bar		2	in.
Layer	**Location (in.)**	**Number of Bars**	**Bar Size (no.)**
1	3.58	4	11
2			
3			
4			

Top Reinforcement			
f_y		60	ksi
No. of Layers		1	
Development Length Straight		52.8	in.
Development Length Hook		21.4	in.
Clear Cover at End of Bar		2	in.
Layer	**Location (in.)**	**Number of Bars**	**Bar Size (no.)**
1	38.42	7	11
2			
3			
4			

Factored Applied Loads

Location from Left	Load Value	Width of Loaded Area	Length of Loaded Area
ft	kip	in.	in.
2.21	216.3	23.0	23.0
9.29	115.0	16.2	16.2
11.89	116.3	16.2	16.2
16.01	115.0	16.2	16.2
22.34	235.0	23.0	23.0
29.45	115.0	16.2	16.2
32.05	122.5	16.2	16.2
36.17	115.0	16.2	16.2
42.50	247.6	23.0	23.0
49.98	315.0	23.0	23.0
56.33	115.0	16.2	16.2
58.26	225.5	16.2	16.2
63.05	115.0	16.2	16.2
66.32	205.0	16.2	16.2
69.78	115.0	16.2	16.2
74.39	130.1	16.2	16.2
76.50	115.0	16.2	16.2
82.83	232.5	23.0	23.0
2.21	12.2	0.0	0.0
9.29	11.1	0.0	0.0
11.89	7.7	0.0	0.0
16.01	12.0	0.0	0.0
22.34	15.4	0.0	0.0
29.45	11.1	0.0	0.0
32.05	7.7	0.0	0.0
36.17	12.0	0.0	0.0
42.50	15.9	0.0	0.0
49.98	15.9	0.0	0.0
56.33	9.5	0.0	0.0
58.26	7.7	0.0	0.0
63.05	9.3	0.0	0.0
66.32	7.7	0.0	0.0
69.78	9.3	0.0	0.0
74.39	7.7	0.0	0.0
76.50	9.7	0.0	0.0
82.83	11.3	0.0	0.0

Reactions

Location from Left	Width of Bearing Area	Length of Bearing Area
ft	in.	in.
4.5	31.9	31.9
23.5	31.9	31.9
42.5	31.9	31.9
61.5	31.9	31.9
80.5	31.9	31.9

Figure 4.26: Inputs Sheet for Design Example (Bent Cap)

4.5.3 Computer Program Outputs

After all user inputs are entered, the program is run by clicking the “Run Program” button on the *Inputs* sheet. The computer program performs the analysis and design steps in accordance with the strut-and-tie method and outputs the results on various sheets. In the following subsections, the information on each of the output sheets will be discussed, and the final design of the bent cap for the load case under consideration will be presented.

4.5.3.1 Strut-and-Tie Model

After running the program, the *Strut-and-Tie Model* sheet will automatically open and display the strut-and-tie model for the bent cap. The strut-and-tie model output by the program based on the inputs described in Section 4.5.2 is shown in Figure 4.27. On the worksheet, the entire model is presented as a single unit. In Figure 4.27, the model is divided into two parts to allow it to fit on one page. The model developed by the computer program is the almost same model that was developed for the design example in Williams et al. (2012). However, at Column 4, an extra node is placed under Node Q. Section 4.4.2 of Williams et al. discusses why this node is unnecessary and how removing the node is more efficient and realistic. Therefore, the decision is made to remove the node and rerun the program. Node KK is deleted from the “Nodes” table in the *Nodes and Members* sheet and the “Rerun Program” button is clicked. When this is done, the model in Figure 4.28 is developed, which matches the model developed for the design example in Williams et al. (2012). The remaining results presented are based on the model in Figure 4.28.

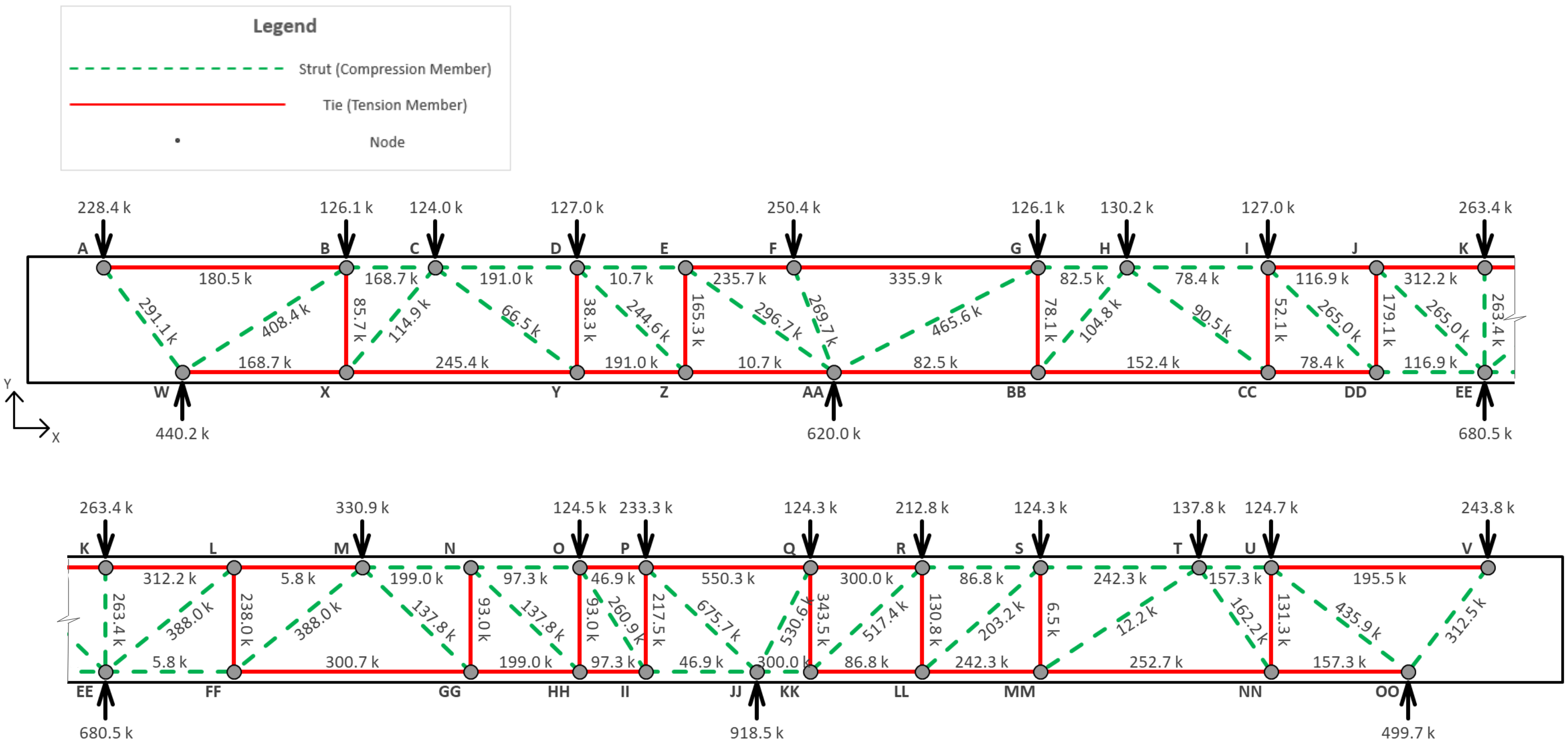


Figure 4.27: Initial Strut-and-Tie Model from Program Output (Bent Cap)

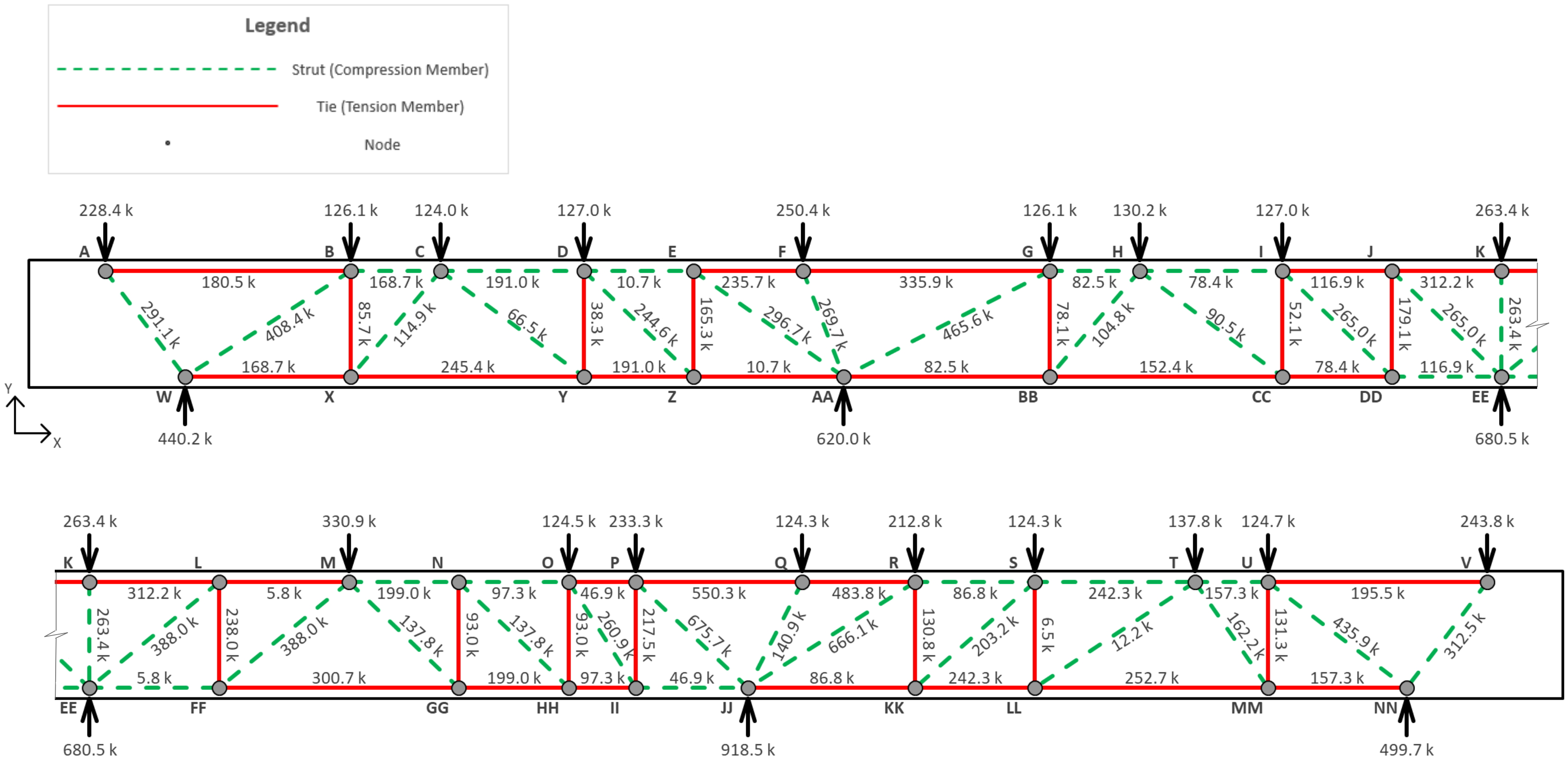


Figure 4.28: Strut-and-Tie Model from Program Output after Removing Unnecessary Node (Bent Cap)

4.5.3.2 Nodes and Members

The *Nodes and Members* sheet, shown in Figure 4.29, contains information about the strut-and-tie model in a table format. The "Nodes" table lists each node and the corresponding x- and y-coordinates measured from the bottom left corner of the member. If a node has been subdivided, the "Subdivided Nodes" table lists each subdivided node label and the corresponding x- and y-coordinates measured from the bottom left corner of the member. The "Members" table contains the label and force for each strut and tie in the model. A total of 40 nodes, 9 of which were subdivided, and 77 members comprise the strut-and-tie model. Of the 9 subdivided nodes, 8 of these were subdivided into two parts, and one node was subdivided into three parts. The "Nodes with Combined Forces Updated Angles" table includes 33 nodes (or portions of nodes) that undergo strength checks. The force acting at each node and its angle (i.e., orientation) is also presented.

Nodes		
Label	x (ft)	y (ft)
A	2.21	3.20
B	9.29	3.20
C	11.89	3.20
D	16.01	3.20
E	19.17	3.20
F	22.34	3.20
G	29.45	3.20
H	32.05	3.20
I	36.17	3.20
J	39.33	3.20
K	42.50	3.20
L	46.24	3.20
M	49.98	3.20
N	53.16	3.20
O	56.33	3.20
P	58.26	3.20
Q	63.05	3.20
R	66.32	3.20
S	69.78	3.20
T	74.39	3.20
U	76.50	3.20
V	82.83	3.20
W	4.50	0.30
X	9.29	0.30
Y	16.01	0.30
Z	19.17	0.30
AA	23.50	0.30
BB	29.45	0.30
CC	36.17	0.30
DD	39.33	0.30
EE	42.50	0.30
FF	46.24	0.30
GG	53.16	0.30
HH	56.33	0.30
II	58.26	0.30
JJ	61.50	0.30
KK	66.32	0.30
LL	69.78	0.30
MM	76.50	0.30
NN	80.50	0.30

Subdivided Nodes		
Label	x (ft)	y (ft)
C Left	11.68	3.20
C Right	12.35	3.20
H Left	31.77	3.20
H Right	32.45	3.20
M Left	49.71	3.20
M Right	50.67	3.20
T Left	73.75	3.20
T Right	74.42	3.20
W Left	3.86	0.30
W Right	5.19	0.30
AA Left	23.06	0.30
AA Right	24.39	0.30
EE Left	41.57	0.30
EE Middle	42.08	0.30
EE Right	43.20	0.30
JJ Left	60.82	0.30
JJ Right	62.15	0.30
NN Left	79.85	0.30
NN Right	81.18	0.30

Members		
Label	Force	Strut or Tie
A-B	180.5	Tie
B-C	-168.7	Strut
C-D	-191.0	Strut
D-E	-10.7	Strut
E-F	235.7	Tie
F-G	335.9	Tie
G-H	-82.5	Strut
H-I	-78.4	Strut
I-J	116.9	Tie
J-K	312.2	Tie
K-L	312.2	Tie
L-M	5.8	Tie
M-N	-199.0	Strut
N-O	-97.3	Strut
O-P	46.9	Tie
P-Q	550.3	Tie
Q-R	483.8	Tie
R-S	-86.8	Strut
S-T	-242.3	Strut
T-U	-157.3	Strut
U-V	195.5	Tie
W-X	168.7	Tie
X-Y	245.4	Tie
Y-Z	191.0	Tie
Z-AA	10.7	Tie
AA-BB	82.5	Tie
BB-CC	152.4	Tie
CC-DD	78.4	Tie
DD-EE	-116.9	Strut
EE-FF	-5.8	Strut
FF-GG	300.7	Tie
GG-HH	199.0	Tie
HH-II	97.3	Tie
II-JJ	-46.9	Strut
JJ-KK	86.8	Tie
KK-LL	242.3	Tie
LL-MM	252.7	Tie
MM-NN	157.3	Tie
B-X	85.7	Tie
D-Y	38.3	Tie
E-Z	165.3	Tie
G-BB	78.1	Tie
I-CC	52.1	Tie
J-DD	179.1	Tie
K-EE	-263.4	Strut
L-FF	238.0	Tie
N-GG	93.0	Tie
O-HH	93.0	Tie
P-II	217.5	Tie
R-KK	130.8	Tie
S-LL	6.5	Tie
U-MM	131.3	Tie
A-W	-291.1	Strut
B-W	-408.4	Strut
C-X	-114.9	Strut
C-Y	-66.5	Strut
D-Z	-244.6	Strut
E-AA	-296.7	Strut
F-AA	-269.7	Strut
G-AA	-465.6	Strut
H-BB	-104.8	Strut
H-CC	-90.5	Strut
I-DD	-265.0	Strut
J-EE	-265.0	Strut
L-EE	-388.0	Strut
M-FF	-388.0	Strut
M-GG	-137.8	Strut
N-HH	-137.8	Strut
O-II	-260.9	Strut
P-JJ	-675.7	Strut
Q-JJ	-140.9	Strut
R-JJ	-666.1	Strut
S-KK	-203.2	Strut
T-LL	-12.2	Strut
T-MM	-162.2	Strut
U-NN	-435.9	Strut
V-NN	-312.5	Strut

Nodes with Combined Forces and Updated Angles								
Node	Force (kip)	Angle (deg)	Force (kip)	Angle (deg)	Force (kip)	Angle (deg)	Force (kip)	Angle (deg)
A	180.5	0.00	-291.1	299.68				
B	180.5	180.00	-168.7	0.00	85.7	270.00	-408.4	215.32
C Left	-259.9	199.70	-245.4	0.00				
C Right	-248.3	350.90	-245.4	180.00				
D	-191.0	180.00	-252.6	319.12	38.3	270.00		
F	235.7	180.00	335.9	0.00	-269.7	284.00		
G	335.9	180.00	-82.5	0.00	78.1	270.00	-465.6	209.86
H Left	-171.3	208.29	-152.4	0.00				
H Right	-161.1	340.25	-152.4	180.00				
I	-78.4	180.00	116.9	0.00	52.1	270.00	-265.0	317.48
K	312.2	180.00	312.2	0.00	-263.4	270.00		
M Left	5.8	180.00	-388.0	219.92	-300.7	0.00		
M Right	-314.8	341.55	-300.7	180.00				
O	-97.3	180.00	46.9	0.00	93.0	270.00	-260.9	303.55
P	46.9	180.00	550.3	0.00	217.5	270.00	-675.7	311.47
Q	550.3	180.00	483.8	0.00	-140.9	252.75		
R	483.8	180.00	-86.8	0.00	130.8	270.00	-666.1	214.84
S	-275.4	208.35	-242.3	0.00	6.5	270.00		
T Left	-252.8	181.48	-252.7	0.00				
T Right	-284.7	332.41	-252.7	180.00				
U	-157.3	180.00	195.5	0.00	131.3	270.00	-435.9	319.12
V	195.5	180.00	-312.5	240.42				
W Left	-291.1	119.68	-180.5	0.00				
W Right	168.7	0.00	-408.4	35.32	-180.5	180.00		
AA Left	10.7	180.00	-541.3	124.33	-335.9	0.00		
AA Right	82.5	0.00	-465.6	29.86	-335.9	180.00		
EE Left	-360.0	144.58	-312.2	0.00				
EE Middle	-263.4	90.00	-312.2	0.00	-312.2	180.00		
EE Right	-392.6	44.59	-312.2	180.00				
JJ Left	-711.4	134.65	-550.3	0.00				
JJ Right	86.8	0.00	-790.5	41.32	-550.3	180.00		
NN Left	157.3	180.00	-435.9	139.12	-195.5	0.00		
NN Right	-312.5	60.42	-195.5	180.00				

Figure 4.29: Nodes and Members Sheet from Program Output (Bent Cap)

4.5.3.3 Node Figures

The *Node Figures* sheet displays figures representing the 33 nodes (or portions of nodes) listed in the "Nodes with Combined Forces and Updated Angles" table on the *Nodes and Members* sheet. For each node, a graphic is provided that illustrates all members that enter the node. Three of the graphics generated for the five-column bent cap are shown in Figure 4.30. If any struts were combined at the node, the resultant of the struts is displayed. The angles labeled in the figures reflect the updated angles resulting from the subdivision of nodes. By providing a visual representation of each node, the graphics are meant to aid designers with understanding the nodal strength checks that are performed by the program.

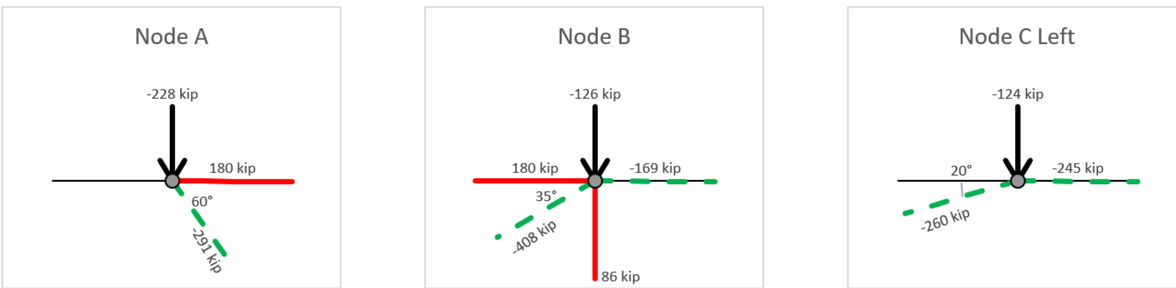


Figure 4.30: Examples of Node Figures from Program Output (Bent Cap)

4.5.3.4 Nodal Checks

The *Nodal Checks* sheet provides a summary of the strength checks for the 33 nodes (or portions of nodes) included in the *Node Figures* sheet. For each node, the factored design strengths of the bearing face, back face, and strut-to-node interface are calculated, as applicable, and compared to the factored force acting on each face. The summary of the nodal strength checks for the five-column bent cap as displayed on the *Nodal Checks* sheet is provided in Figure 4.31. If the words "Check N/A" appear for a back face check, no direct compressive force was applied to the face. None of the strength checks are displayed in red bolded text, and the cells in the "Pass?" columns display "OK," as opposed to "NG," for each node, indicating that all nodal faces have adequate strength.

Node	Type	Triaxial Confinement Factor, m	Face Length (in.)			Strength Checks														
						Bearing					Back					Strut-to-Node				
			Bearing	Back	Strut-to-Node	F_u (kip)	Efficiency (v)	f_{cu} (ksi)	ϕF_n (kip)	Pass?	F_u (kip)	Efficiency (v)	f_{cu} (ksi)	ϕF_n (kip)	Pass?	F_u (kip)	Efficiency (v)	f_{cu} (ksi)	ϕF_n (kip)	Pass?
A	CCT	1.8	23.0	7.2	23.5	228.4	0.70	5.1	1893.4	OK	Check N/A					291.1	0.65	4.7	1798.5	OK
B	CTT	2.0	16.2	7.2	15.2	126.1	0.65	5.2	955.3	OK	168.7	0.65	5.2	422.2	OK	408.4	0.65	5.2	896.8	OK
C Left	CCC	2.0	11.2	7.2	10.5											259.9	0.65	5.2	620.0	OK
C Right	CCC	2.0	5.0	7.2	7.9	124.0	0.85	6.8	1249.2	OK	245.4	0.85	6.8	552.1	OK	248.3	0.65	5.2	463.6	OK
D	CCT	2.0	16.2	7.2	16.0	127.0	0.70	5.6	1028.8	OK	191.0	0.70	5.6	454.7	OK	252.6	0.65	5.2	944.5	OK
F	CCT	1.8	23.0	7.2	24.0	250.4	0.70	5.1	1893.4	OK	Check N/A					269.7	0.65	4.7	1838.3	OK
G	CTT	2.0	16.2	7.2	14.3	126.1	0.65	5.2	955.3	OK	82.5	0.65	5.2	422.2	OK	465.6	0.65	5.2	841.8	OK
H Left	CCC	2.0	9.7	7.2	10.9											171.3	0.65	5.2	643.3	OK
H Right	CCC	2.0	6.5	7.2	8.9	130.2	0.85	6.8	1249.2	OK	152.4	0.85	6.8	552.1	OK	161.1	0.65	5.2	526.5	OK
I	CTT	2.0	16.2	7.2	16.2	127.0	0.65	5.2	955.3	OK	78.4	0.65	5.2	422.2	OK	265.0	0.65	5.2	956.8	OK
K	CCT	1.8	23.0	7.2	23.0	263.4	0.70	5.1	1893.4	OK	Check N/A					263.4	0.65	4.7	1758.1	OK
M Left	CCT	1.8	16.5	7.2	16.1											388.0	0.65	4.7	1231.0	OK
M Right	CCC	1.8	6.5	7.2	8.8	330.9	0.70	5.1	1893.4	OK	300.7	0.70	5.1	589.4	OK	314.8	0.65	4.7	675.5	OK
O	CTT	2.0	16.2	7.2	17.5	124.5	0.65	5.2	955.3	OK	97.3	0.65	5.2	422.2	OK	260.9	0.65	5.2	1029.5	OK
P	CTT	2.0	16.2	7.2	16.9	233.3	0.65	5.2	955.3	OK	Check N/A					675.7	0.65	5.2	995.4	OK
Q	CCT	2.0	16.2	7.2	17.6	124.3	0.70	5.6	1028.8	OK	Check N/A					140.9	0.65	5.2	1037.5	OK
R	CTT	2.0	16.2	7.2	15.1	212.8	0.65	5.2	955.3	OK	86.8	0.65	5.2	422.2	OK	666.1	0.65	5.2	892.3	OK
S	CCT	2.0	16.2	7.2	14.0	124.3	0.70	5.6	1028.8	OK	242.3	0.70	5.6	454.7	OK	275.4	0.65	5.2	825.2	OK
T Left	CCC	2.0	0.8	7.2	7.2											252.8	0.65	5.2	423.2	OK
T Right	CCC	2.0	15.4	7.2	13.5	137.8	0.85	6.8	1249.2	OK	252.7	0.85	6.8	552.1	OK	284.7	0.65	5.2	795.7	OK
U	CTT	2.0	16.2	7.2	16.0	124.7	0.65	5.2	955.3	OK	157.3	0.65	5.2	422.2	OK	435.9	0.65	5.2	944.4	OK
V	CCT	1.8	23.0	7.2	23.5	243.8	0.70	5.1	1893.4	OK	Check N/A					312.5	0.65	4.7	1799.1	OK
W Left	CCC	1.3	16.6	7.2	17.9											291.1	0.65	3.4	1370.4	OK
W Right	CCT	1.3	15.3	7.2	14.7	440.2	0.70	3.7	2626.0	OK	180.5	0.70	3.7	589.4	OK	408.4	0.65	3.4	1124.7	OK
AA Left	CCT	1.3	21.4	7.2	21.7											541.3	0.65	3.4	1659.0	OK
AA Right	CCT	1.3	10.5	7.2	11.4	620.0	0.70	3.7	2626.0	OK	335.9	0.70	3.7	589.4	OK	465.6	0.65	3.4	874.6	OK
EE Left	CCC	1.3	8.4	7.2	10.7											360.0	0.65	3.4	818.0	OK
EE Middle	CCC	1.3	12.3	7.2	12.3											263.4	0.65	3.4	943.9	OK
EE Right	CCC	1.3	11.2	7.2	12.9	680.5	0.85	4.5	3188.7	OK	312.2	0.85	4.5	715.7	OK	392.6	0.65	3.4	988.4	OK
JJ Left	CCC	1.3	15.7	7.2	16.2											711.4	0.65	3.4	1235.9	OK
JJ Right	CCT	1.3	16.2	7.2	16.1	918.5	0.70	3.7	2626.0	OK	550.3	0.70	3.7	589.4	OK	790.5	0.65	3.4	1231.0	OK
NN Left	CCT	1.3	16.3	7.2	16.1											435.9	0.65	3.4	1231.3	OK
NN Right	CCC	1.3	15.6	7.2	17.1	499.7	0.70	3.7	2626.0	OK	195.5	0.70	3.7	589.4	OK	312.5	0.65	3.4	1304.6	OK

Figure 4.31: Summary of Nodal Strength Checks from Program Output (Bent Cap)

4.5.3.5 Reinforcement

The *Reinforcement* sheet displays a summary of the adequacy of the longitudinal reinforcement, the spacing requirements crack control reinforcement, stirrup spacing requirements based on vertical tie forces, and anchorage checks. A summary of the strength checks for the longitudinal reinforcement is provided first. The table displayed on the *Reinforcement* sheet is shown in Figure 4.32. On the *Inputs* sheet, four No. 11 bars were entered as the longitudinal reinforcement along the bottom of the bent cap, and seven No. 11 bars were entered as the reinforcement along the top of the member. The factored nominal resistance, $\phi A_{st} f_y$, for both the top and bottom reinforcement is displayed near the top of the table. Four No. 11 bars with a yield strength of 60 ksi have a factored nominal capacity of 337.0 kip, and seven No. 11 bars with a yield strength of 60 ksi have a factored nominal capacity of 589.7 kip. Each horizontal tie along the bottom and top chords of the strut-and-tie model and the force in each tie are then listed in the table. The factored force demand for each tie is compared to the factored nominal capacity. As indicated by “OK” displayed for each tie in the “Pass?” columns, the top and bottom longitudinal reinforcement provided in the bent cap have adequate capacity to carry the tensile forces in all horizontal ties along the top and bottom chords, respectively.

Longitudinal Reinforcement					
Bottom Reinforcement			**Top Reinforcement**		
$\phi A_s F_y$	337.0	kip	$\phi A_s F_y$	589.7	kip
Member	**Force (kip)**	**Pass?**	**Member**	**Force (kip)**	**Pass?**
W-X	168.7	OK	A-B	180.5	OK
X-Y	245.4	OK	E-F	235.7	OK
Y-Z	191.0	OK	F-G	335.9	OK
Z-AA	10.7	OK	I-J	116.9	OK
AA-BB	82.5	OK	J-K	312.2	OK
BB-CC	152.4	OK	K-L	312.2	OK
CC-DD	78.4	OK	L-M	5.8	OK
FF-GG	300.7	OK	O-P	46.9	OK
GG-HH	199.0	OK	P-Q	550.3	OK
HH-II	97.3	OK	Q-R	483.8	OK
JJ-KK	86.8	OK	U-V	195.5	OK
KK-LL	242.3	OK			
LL-MM	252.7	OK			
MM-NN	157.3	OK			

Figure 4.32: Summary of Longitudinal Reinforcement Strength Checks from Program Output (Bent Cap)

Crack control reinforcement requirements are displayed after the “Longitudinal Reinforcement” table. The program output is provided in Figure 4.33. The maximum spacing required to satisfy horizontal and vertical crack control reinforcement specifications are shown. No. 5 bars were input for horizontal crack control reinforcement and for stirrups (i.e., vertical crack control reinforcement). The maximum spacing that can be provided for both horizontal and vertical crack control reinforcement is 4.9 in.

Crack Control Reinforcement		
Horizontal Reinforcement		
Max Spacing	4.9	in.
Vertical Reinforcement		
Max Spacing	4.9	in.

Figure 4.33: Crack Control Reinforcement from Program Output (Bent Cap)

The summary of stirrup spacing requirements is displayed next on the *Reinforcement* sheet, shown in Figure 4.34. In the “Stirrups” table, each vertical tie in the strut-and-tie model is listed along with the force in the tie. No. 5 stirrups with two legs were input into the program for the bent cap. The width of each tie given in the table is calculated following the procedure described in Section 4.2.6.1. The spacing that is required to carry the force in the ties is provided in the next column. The procedure for calculating this value is also described in Section 4.2.6.1. The required vertical crack control reinforcement spacing previously provided in the “Crack Control Reinforcement” table is also displayed in the “Stirrups” table. For each tie, the required spacing to carry the tie force is compared to the required spacing for crack control reinforcement, and the governing value is displayed in the last column of the table. For the five-column bent cap, the spacing required for crack control governs along the length of the member except within the width of Tie P-II. Here, the required stirrup spacing to carry the force in the tie is 3.5 in. This spacing is very close for stirrups. If the designer wanted to use 4 legs instead of 2 legs for stirrups in this region, he or she could test this in the program by changing the input in the “Stirrups” table on the *Inputs* sheet to 4 legs and running the program again. The output for this is shown in Figure 4.35. When 4 legs are included, the governing spacing for Tie P-II is 7.1 inches. In this example, none of the tie forces require a stirrup spacing less than 3 in. If a spacing less than 3 in. were required, “*Inadequate” will be displayed in the table and a different stirrup detail is required. If no vertical ties were present in the strut-and-tie model, the “Stirrups” table would not contain any information under the headings. Instead, a note will be displayed under the headings that says “No vertical ties exist in the strut-and-tie model. Use required spacing for vertical crack control reinforcement throughout the member.”

Stirrups					
Member	Force (kip)	Vertical Tie Width (in.)	Req'd Tie Spacing (in.)	Req'd Spacing for Crack Control (in.)	Governing Spacing (in.)
B-X	85.7	31.2	12.1	4.9	*4.9*
D-Y	38.3	38.0	33.1	4.9	*4.9*
E-Z	165.3	38.0	7.6	4.9	*4.9*
G-BB	78.1	31.2	13.3	4.9	*4.9*
I-CC	52.1	38.0	24.4	4.9	*4.9*
J-DD	179.1	38.0	7.1	4.9	*4.9*
L-FF	238.0	44.9	6.3	4.9	*4.9*
N-GG	93.0	38.1	13.7	4.9	*4.9*
O-HH	93.0	23.1	8.3	4.9	*4.9*
P-II	217.5	23.1	3.5	4.9	*3.5*
R-KK	130.8	39.2	10	4.9	*4.9*
S-LL	6.5	41.4	213.2	4.9	*4.9*
U-MM	131.3	25.3	6.4	4.9	*4.9*

Figure 4.34: Stirrup Spacing Requirements from Program Output with 2 Stirrup Legs Input (Bent Cap)

Stirrups					
Member	Force (kip)	Vertical Tie Width (in.)	Req'd Spacing for Shear (in.)	Req'd Spacing for Crack Control (in.)	Governing Spacing (in.)
B-X	85.7	31.2	24.3	9.8	*9.8*
D-Y	38.3	38.0	66.3	9.8	*9.8*
E-Z	165.3	38.0	15.3	9.8	*9.8*
G-BB	78.1	31.2	26.7	9.8	*9.8*
I-CC	52.1	38.0	48.8	9.8	*9.8*
J-DD	179.1	38.0	14.2	9.8	*9.8*
L-FF	238.0	44.9	12.6	9.8	*9.8*
N-GG	93.0	38.1	27.4	9.8	*9.8*
O-HH	93.0	23.1	16.6	9.8	*9.8*
P-II	217.5	23.1	7.1	9.8	*7.1*
R-KK	130.8	39.2	20	9.8	*9.8*
S-LL	6.5	41.4	426.4	9.8	*9.8*
U-MM	131.3	25.3	12.9	9.8	*9.8*

Figure 4.35: Stirrup Spacing Requirements from Program Output with 4 Stirrup Legs Input (Bent Cap)

A summary of the check to ensure proper anchorage of the longitudinal reinforcement is provided in the last table on the *Reinforcement* sheet and is shown in Figure 4.36. Because at least one negative moment region exists along the length of the bent cap, anchorage is checked for both the top and bottom chords of the strut-and-tie model. More specifically, the space available for the reinforcement carrying the forces in the outermost ties of each chord is checked to ensure that the reinforcement can be fully developed. Nodes A and V are the nodes located at the outermost ends of the outermost ties at each end of the top chord. Similarly, Nodes W and NN are the nodes located at the outermost ends of the outermost ties at each end of the bottom chord. The required development length for a straight No. 11 bar and a hooked No. 11 bar were entered into the *Inputs* sheet before running the program. The available length for the development of the reinforcement at Nodes A, V, W, and NN are calculated by the program and displayed in the "Anchorage for Ties" table. The example included in Williams et al. (2012) considered the tapered end of the member when calculating available length. The computer program does not include this within the calculations for available length. If the designer wishes to consider the effect from the tapered end of the member, this contribution should be added to the available length given in the program. The available length at all four nodes is sufficient for a hooked bar to be fully developed. However, the available length for the top layer of steel is not adequate to accommodate a straight bar. Therefore, hooked bars should be used to anchor the top layer of reinforcement in the bent cap. The available length in the bottom chord is long enough to accommodate both straight and hooked bars. Straight bars are chosen for the bottom layer of reinforcement.

Anchorage for Ties					
	Lengths (in.)				
Node	**Available Length**	**Reqd Hook**	**Pass?**	**Reqd Straight**	**Pass?**
A	38.0	21.4	OK	**52.8**	**NG**
V	37.6	21.4	OK	**52.8**	**NG**
W	73.0	21.4	OK	40.6	OK
NN	72.1	21.4	OK	40.6	OK

Figure 4.36: Anchorage Check from Program Output (Bent Cap)

4.5.3.6 Continuous Beam Analysis

Following the *Reinforcement* sheet, the remaining output sheets include information that was used to develop the strut-and-tie model for the bent cap. The *Cont. Beam Analysis* sheet provides the full set of calculations based on the moment distribution method that were performed to determine the column reactions. The calculations as presented on the sheet are shown in Figure 4.37. The five column reactions are displayed in the last row of the table.

Column Index		Reaction 1		Reaction 2		Reaction 3		Reaction 4		Reaction 5	
Member			**1-2**	**2-1**	**2-3**	**3-2**	**3-4**	**4-3**	**4-5**	**5-4**	
Distribution Factor			1	0.5	0.5	0.5	0.5	0.5	0.5	1	
Carry-Over Factor			0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	
Fixed-End-Moment (kip-ft)		-524.03	923.95	-936.87	869.71	-794.20	1144.55	-1452.03	1328.406877	-1160.086642	567.7077838
	DIST		-399.9136	33.5805	33.5805	-175.1707	-175.1707	61.8096	61.8096	592.3789	
	CO		16.7903	-199.9568	-87.5854	16.7903	30.9048	-87.5854	296.1894	30.9048	
	DIST		-16.7903	143.7711	143.7711	-23.8475	-23.8475	-104.3020	-104.3020	-30.9048	
	CO		71.8855	-8.3951	-11.9238	71.8855	-52.1510	-11.9238	-15.4524	-52.1510	
	DIST		-71.8855	10.1595	10.1595	-9.8673	-9.8673	13.6881	13.6881	52.1510	
	CO		5.0797	-35.9428	-4.9336	5.0797	6.8440	-4.9336	26.0755	6.8440	
	DIST		-5.0797	20.4382	20.4382	-5.9619	-5.9619	-10.5709	-10.5709	-6.8440	
	CO		10.2191	-2.5399	-2.9809	10.2191	-5.2855	-2.9809	-3.4220	-5.2855	
	DIST		-10.2191	2.7604	2.7604	-2.4668	-2.4668	3.2015	3.2015	5.2855	
	CO		1.3802	-5.1096	-1.2334	1.3802	1.6007	-1.2334	2.6427	1.6007	
	DIST		-1.3802	3.1715	3.1715	-1.4905	-1.4905	-0.7047	-0.7047	-1.6007	
	CO		1.5857	-0.6901	-0.7452	1.5857	-0.3523	-0.7452	-0.8004	-0.3523	
	DIST		-1.5857	0.7177	0.7177	-0.6167	-0.6167	0.7728	0.7728	0.3523	
	CO		0.3588	-0.7929	-0.3084	0.3588	0.3864	-0.3084	0.1762	0.3864	
	DIST		-0.3588	0.5506	0.5506	-0.3726	-0.3726	0.0661	0.0661	-0.3864	
	CO		0.2753	-0.1794	-0.1863	0.2753	0.0330	-0.1863	-0.1932	0.0330	
	DIST		-0.2753	0.1829	0.1829	-0.1542	-0.1542	0.1898	0.1898	-0.0330	
	CO		0.0914	-0.1377	-0.0771	0.0914	0.0949	-0.0771	-0.0165	0.0949	
	DIST		-0.0914	0.1074	0.1074	-0.0932	-0.0932	0.0468	0.0468	-0.0949	
	CO		0.0537	-0.0457	-0.0466	0.0537	0.0234	-0.0466	-0.0474	0.0234	
	DIST		-0.0537	0.0461	0.0461	-0.0385	-0.0385	0.0470	0.0470	-0.0234	
	CO		0.0231	-0.0268	-0.0193	0.0231	0.0235	-0.0193	-0.0117	0.0235	
	DIST		-0.0231	0.0231	0.0231	-0.0233	-0.0233	0.0155	0.0155	-0.0235	
	CO		0.0115	-0.0115	-0.0116	0.0115	0.0077	-0.0116	-0.0118	0.0077	
	DIST		-0.0115	0.0116	0.0116	-0.0096	-0.0096	0.0117	0.0117	-0.0077	
	CO		0.0058	-0.0058	-0.0048	0.0058	0.0058	-0.0048	-0.0039	0.0058	
	DIST		-0.0058	0.0053	0.0053	-0.0058	-0.0058	0.0043	0.0043	-0.0058	
	CO		0.0026	-0.0029	-0.0029	0.0026	0.0022	-0.0029	-0.0029	0.0022	
	DIST		-0.0026	0.0029	0.0029	-0.0024	-0.0024	0.0029	0.0029	-0.0022	
	CO		0.0015	-0.0013	-0.0012	0.0015	0.0015	-0.0012	-0.0011	0.0015	
	DIST		-0.0015	0.0013	0.0013	-0.0015	-0.0015	0.0011	0.0011	-0.0015	
	CO		0.0006	-0.0007	-0.0007	0.0006	0.0006	-0.0007	-0.0007	0.0006	
	DIST		-0.0006	0.0007	0.0007	-0.0006	-0.0006	0.0007	0.0007	-0.0006	
	CO		0.0004	-0.0003	-0.0003	0.0004	0.0004	-0.0003	-0.0003	0.0004	
	DIST		-0.0004	0.0003	0.0003	-0.0004	-0.0004	0.0003	0.0003	-0.0004	
	CO		0.0002	-0.0002	-0.0002	0.0002	0.0001	-0.0002	-0.0002	0.0001	
	DIST		-0.0002	0.0002	0.0002	-0.0002	-0.0002	0.0002	0.0002	-0.0001	
	CO		9.092E-05	-7.713E-05	-7.528E-05	9.092E-05	9.102E-05	-7.528E-05	-7.343E-05	9.102E-05	
	DIST		-9.092E-05	7.621E-05	7.621E-05	-9.097E-05	-9.097E-05	7.436E-05	7.436E-05	-9.102E-05	
	CO		3.810E-05	-4.546E-05	-4.549E-05	3.810E-05	3.718E-05	-4.549E-05	-4.551E-05	3.718E-05	
	DIST		-3.810E-05	4.547E-05	4.547E-05	-3.764E-05	-3.764E-05	4.550E-05	4.550E-05	-3.718E-05	
	CO		2.274E-05	-1.905E-05	-1.882E-05	2.274E-05	2.275E-05	-1.882E-05	-1.859E-05	2.275E-05	
	DIST		-2.274E-05	1.894E-05	1.894E-05	-2.274E-05	-2.274E-05	1.870E-05	1.870E-05	-2.275E-05	
	CO		9.468E-06	-1.137E-05	-1.137E-05	9.468E-06	9.352E-06	-1.137E-05	-1.137E-05	9.352E-06	
	DIST		-9.468E-06	1.137E-05	1.137E-05	-9.410E-06	-9.410E-06	1.137E-05	1.137E-05	-9.352E-06	
	CO		5.685E-06	-4.734E-06	-4.705E-06	5.685E-06	5.687E-06	-4.705E-06	-4.676E-06	5.687E-06	
	DIST		-5.685E-06	4.720E-06	4.720E-06	-5.686E-06	-5.686E-06	4.691E-06	4.691E-06	-5.687E-06	
	CO		2.360E-06	-2.842E-06	-2.843E-06	2.360E-06	2.345E-06	-2.843E-06	-2.843E-06	2.345E-06	
	DIST		-2.360E-06	2.843E-06	2.843E-06	-2.353E-06	-2.353E-06	2.843E-06	2.843E-06	-2.345E-06	
	CO		1.421E-06	-1.180E-06	-1.176E-06	1.421E-06	1.422E-06	-1.176E-06	-1.173E-06	1.422E-06	
	DIST		-1.421E-06	1.178E-06	1.178E-06	-1.421E-06	-1.421E-06	1.174E-06	1.174E-06	-1.422E-06	
	CO		5.890E-07	-7.107E-07	-7.107E-07	5.890E-07	5.872E-07	-7.107E-07	-7.108E-07	5.872E-07	
	DIST		-5.890E-07	7.107E-07	7.107E-07	-5.881E-07	-5.881E-07	7.107E-07	7.107E-07	-5.872E-07	
	CO		3.553E-07	-2.945E-07	-2.941E-07	3.553E-07	3.554E-07	-2.941E-07	-2.936E-07	3.554E-07	
	DIST		-3.553E-07	2.943E-07	2.943E-07	-3.554E-07	-3.554E-07	2.938E-07	2.938E-07	-3.554E-07	
	CO		1.471E-07	-1.777E-07	-1.777E-07	1.471E-07	1.469E-07	-1.777E-07	-1.777E-07	1.469E-07	
	DIST		-1.471E-07	1.777E-07	1.777E-07	-1.470E-07	-1.470E-07	1.777E-07	1.777E-07	-1.469E-07	
	CO		8.884E-08	-7.357E-08	-7.352E-08	8.884E-08	8.884E-08	-7.352E-08	-7.346E-08	8.884E-08	
	DIST		-8.884E-08	7.355E-08	7.355E-08	-8.884E-08	-8.884E-08	7.349E-08	7.349E-08	-8.884E-08	
Sum of Moments (kip-ft)		-524.03	524.03	-975.18	975.18	-906.56	906.56	-1597.81	1597.81	-567.71	567.71
Shear from Point Loads (kip)		228.44	235.52	392.00	200.64	182.73	274.35	414.35	413.58	310.18	243.75
Shear from Unbalanced Moment (kip)			-23.74	23.74	3.61	-3.61	-36.38	36.38	54.22	-54.22	
Shear at Column (kip)		228.44	211.77	415.75	204.26	179.12	237.97	450.73	467.80	255.97	243.75
Point Loads at Column Location (kip)							15.86				
Solved Reactions (kip)		*Column 1*	440.2	*Column 2*	620.0	*Column 3*	680.5	*Column 4*	918.5	*Column 5*	499.7

Figure 4.37: Continuous Beam Analysis from Program Output (Bent Cap)

4.5.3.7 Shear and Moment Diagrams

The shear and moment diagrams are displayed on the next two sheets of the program. The last sheet includes the information needed to develop these diagrams. The shear diagram is used to determine the location of nodes and the orientation of diagonal struts in the strut-and-tie model. The moment diagram was not used for this bent cap. However, for members with no negative moment regions, the moment diagram is used to optimize the location of the top chord of the model. The shear diagram, moment diagram, and corresponding information needed to development the diagrams are shown in Figure 4.38, Figure 4.39, and Figure 4.40 respectively.

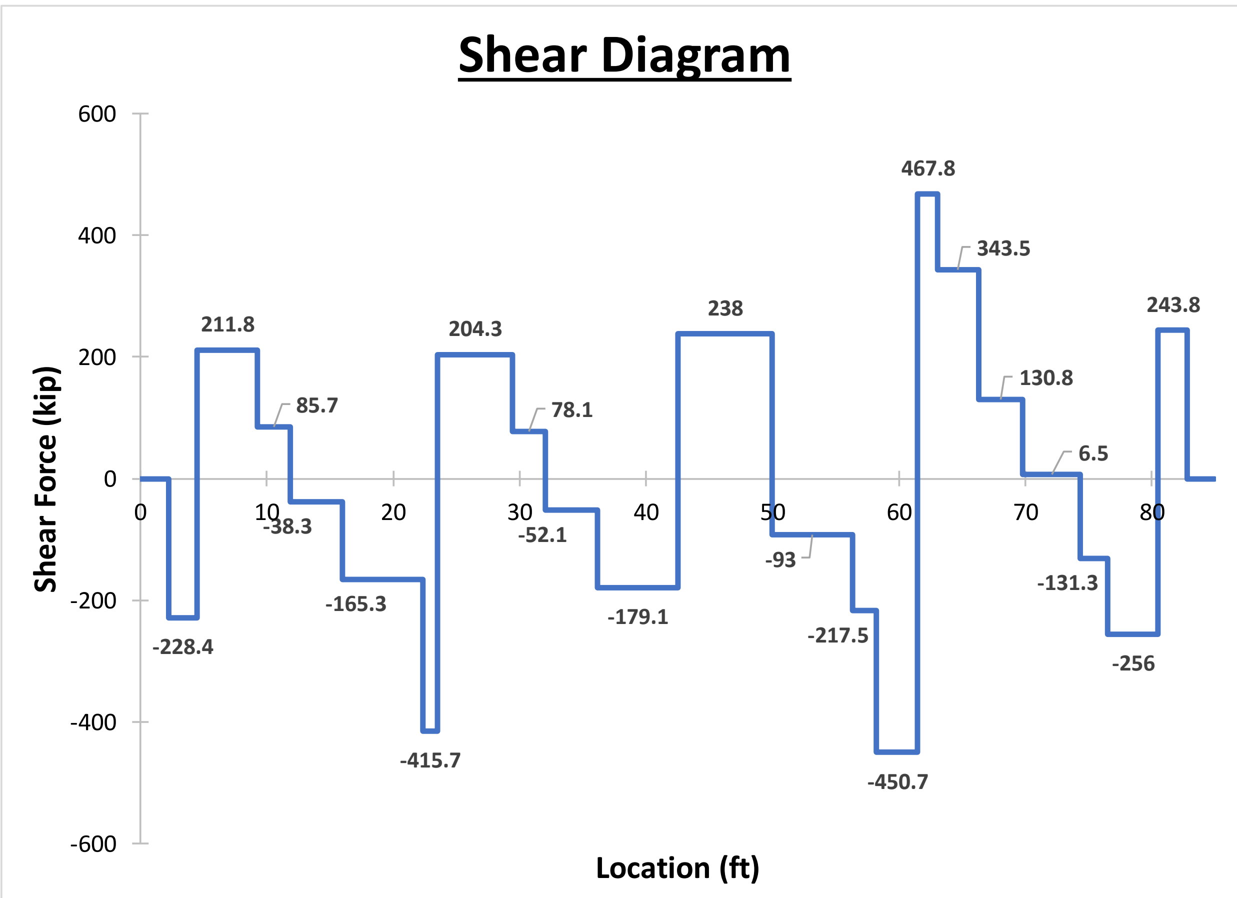


Figure 4.38: Shear Diagram from Program Output (Bent Cap)

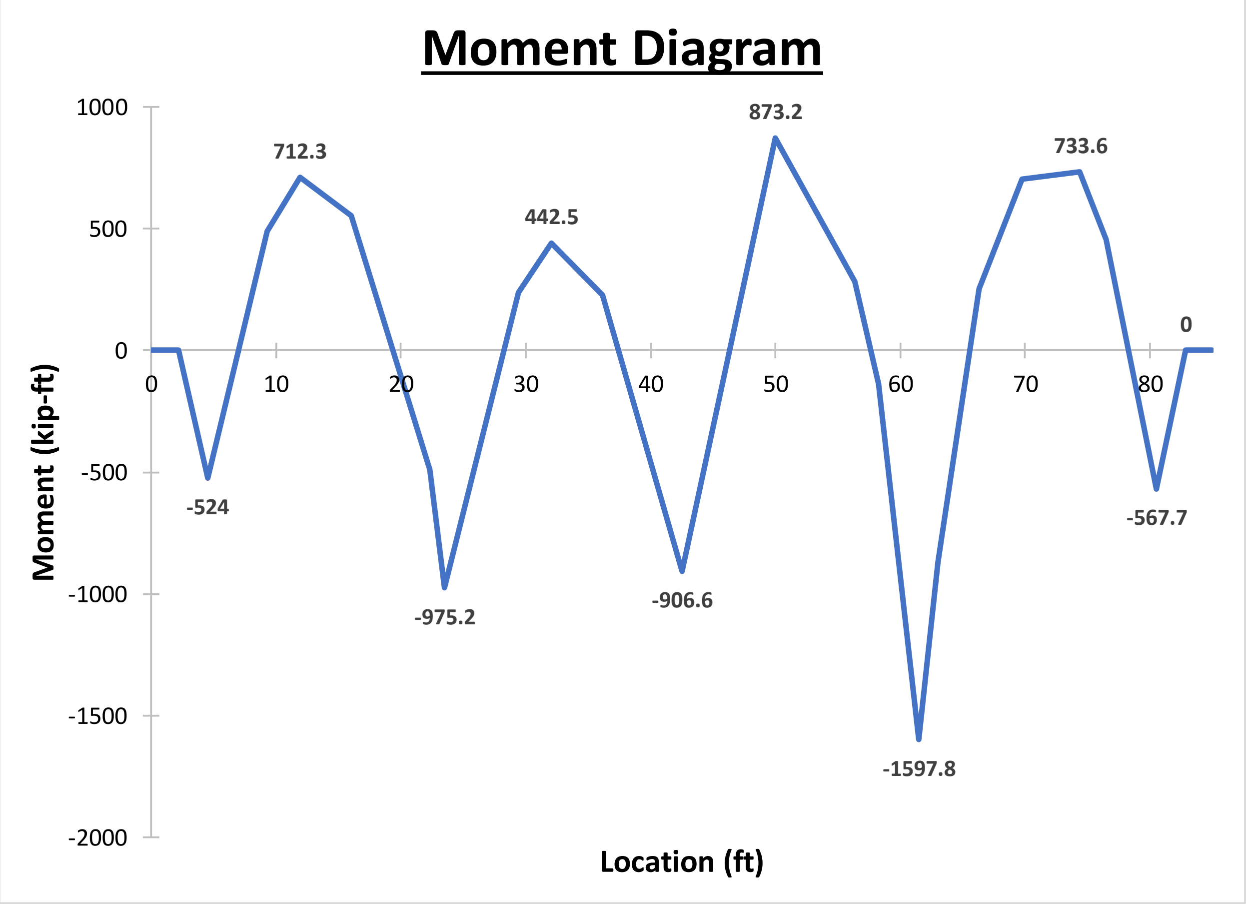


Figure 4.39: Moment Diagram from Program Output (Bent Cap)

Forces	
x-Location (ft)	Load (kip)
2.21	-228.4
4.50	440.2
9.29	-126.1
11.89	-124.0
16.01	-127.0
22.34	-250.4
23.50	620.0
29.45	-126.1
32.05	-130.2
36.17	-127.0
42.50	417.1
49.98	-330.9
56.33	-124.5
58.26	-233.3
61.50	918.5
63.05	-124.3
66.32	-212.8
69.78	-124.3
74.39	-137.8
76.50	-124.7
80.50	499.7
82.83	-243.8

Shear Diagram	
x-Location (ft)	Shear (kip)
0.00	0.0
2.21	0.0
2.21	-228.4
4.50	-228.4
4.50	211.8
9.29	211.8
9.29	85.7
11.89	85.7
11.89	-38.3
16.01	-38.3
16.01	-165.3
22.34	-165.3
22.34	-415.7
23.50	-415.7
23.50	204.3
29.45	204.3
29.45	78.1
32.05	78.1
32.05	-52.1
36.17	-52.1
36.17	-179.1
42.50	-179.1
42.50	238.0
49.98	238.0
49.98	-93.0
56.33	-93.0
56.33	-217.5
58.26	-217.5
58.26	-450.7
61.50	-450.7
61.50	467.8
63.05	467.8
63.05	343.5
66.32	343.5
66.32	130.8
69.78	130.8
69.78	6.5
74.39	6.5
74.39	-131.3
76.50	-131.3
76.50	-256.0
80.50	-256.0
80.50	243.8
82.83	243.8
82.83	0.0
85.00	0.0

Moment Diagram	
x-Location (ft)	Moment (kip-ft)
0.00	0.0
2.21	0.0
4.50	-524.0
9.29	489.8
11.89	712.3
16.01	554.4
22.34	-492.1
23.50	-975.2
29.45	239.6
32.05	442.5
36.17	227.7
42.50	-906.6
49.98	873.2
56.33	282.6
58.26	-136.1
61.50	-1597.8
63.05	-871.1
66.32	252.1
69.78	703.6
74.39	733.6
76.50	456.8
80.50	-567.7
82.83	0.0
85.00	0.0

Figure 4.40: Shear and Moment Diagram Calculations from Program Output (Bent Cap)

4.5.3.8 Final Design

The program outputs listed for the bent cap design example provide summaries of the design steps performed by the program based on the details input by the user. It is the responsibility of the engineer to interpret the results and determine if the design is adequate. For the multi-column bent cap the longitudinal reinforcement input into the program had adequate strength and sufficient space for the development of hooked bars. Furthermore, reasonable values for spacing were output for the required crack control reinforcement. Nodal checks also concluded that the nodes in the model have adequate strength.

The reinforcement layout that satisfies design requirements as determined by the program is presented in Figure 4.41 and Figure 4.42. After reviewing all program outputs, a stirrup spacing of 4.5 in. is chosen along the length of the member except in the region near Tie P-II. In this region, stirrups are spaced at 7 in. Horizontal crack control reinforcement is spaced at approximately 4.5 in.

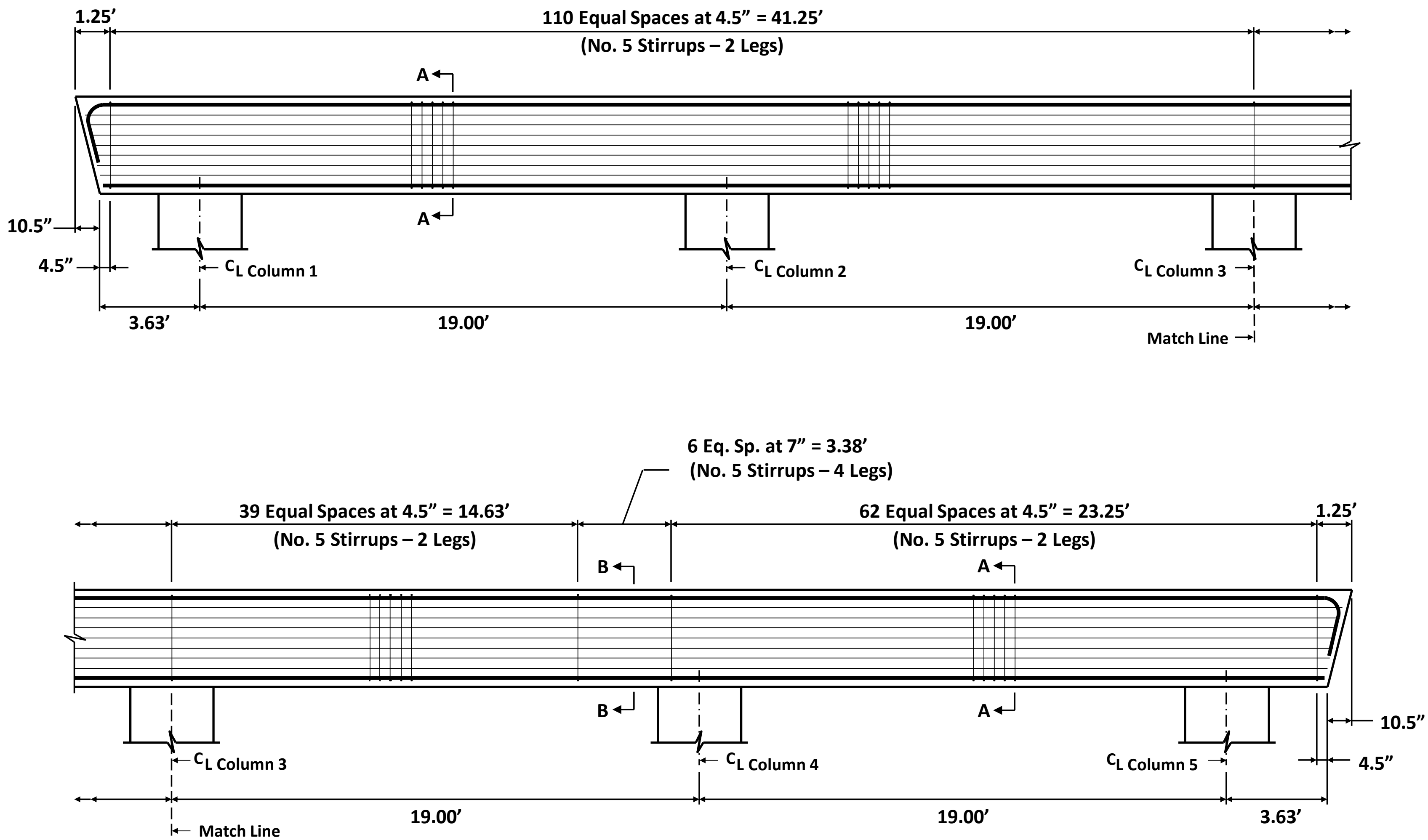
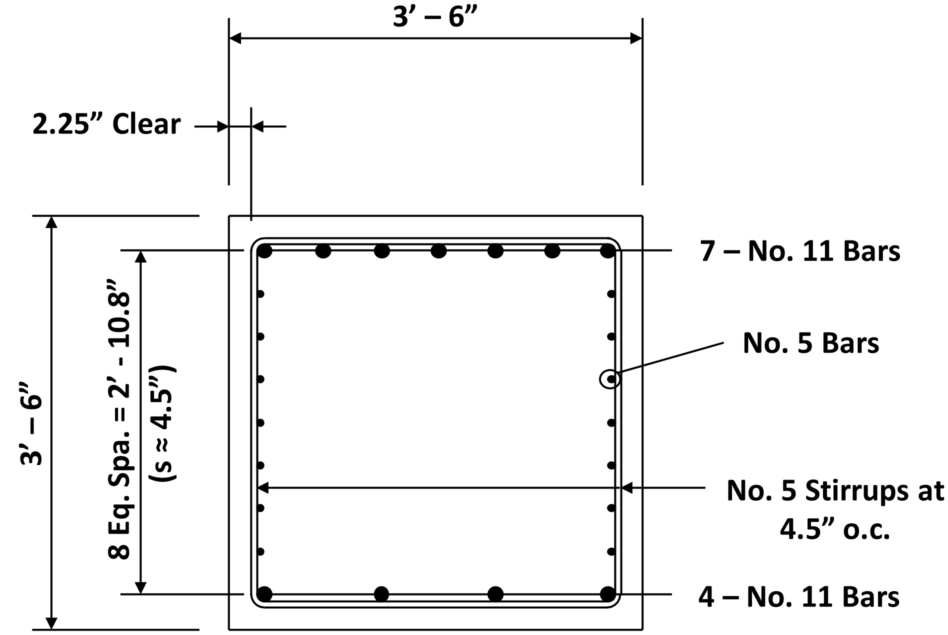
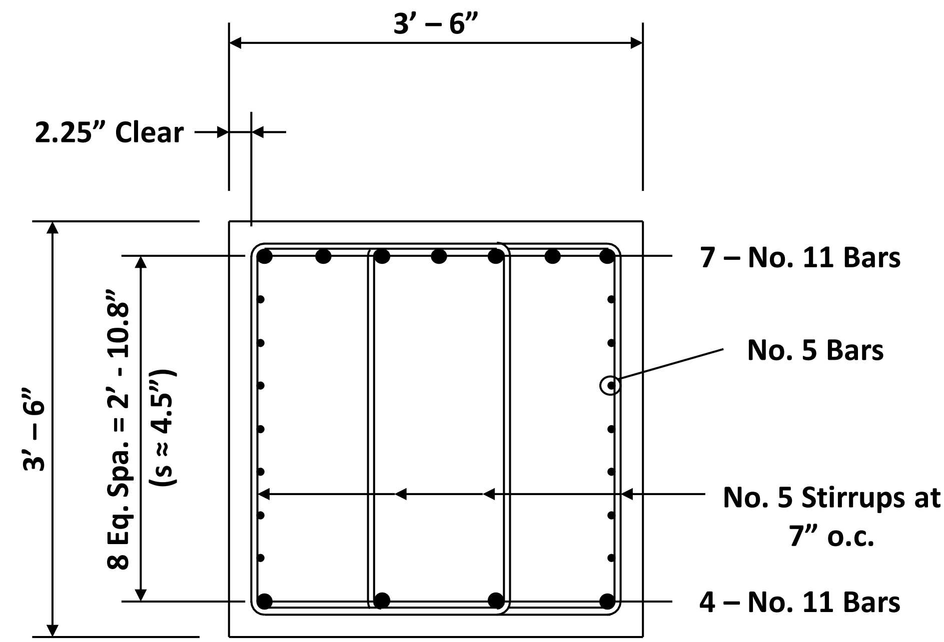


Figure 4.41: Elevation with Reinforcement Details based on Program Output- Bent Cap (from Williams et al., 2012)



Section A-A



Section B-B

Figure 4.42: Section with Reinforcement Design based on Program Outputs – Bent Cap (from Williams et al., 2012)

4.6 Summary

The procedures used by the computer program to perform STM design steps for multi-column bent caps was presented in this chapter. Limitations and assumption for the program were detailed. A design example of a five-column bent cap resisting loads from bulb-tee girders was demonstrated. This example can be used as a reference for engineers designing other multi-column bent caps with the assistance of the STM computer program. The next chapter will present procedures used by the computer program for integral and semi-integral end bent caps.

CHAPTER 5. COMPONENT 2 – INTEGRAL AND SEMI-INTEGRAL END BENT CAPS

5.1 Overview

Integral and semi-integral end bent caps are common bridge substructure elements in Indiana because the end bents allow the elimination of the expansion joint at the end of a bridge, enhancing long-term durability. This chapter presents the procedures used by the computer program to perform STM design steps for an integral or semi-integral end bent cap. Only the procedures that differ from those presented in Chapter 4 for multi-column bent caps are included. An example that demonstrates the use of the program to assist with the STM design of an integral end bent using these procedures is also presented. The design procedure followed in this example can be applied to other end bent caps provided that the element satisfies the limitations and assumptions that are presented herein.

5.2 Computer Program Procedure

Integral and semi-integral end bent caps present unique challenges for the implementation of the STM. Although end bent caps are often primarily composed of D-regions and justify the use of the STM, no known references are available that detail the application of the STM for the design of end bent caps. Due to the manner in which stresses are transferred through the member to the embedded piles, simplifying assumptions are necessary to develop a viable strut-and-tie model. The procedure followed by the computer program for a multi-column bent cap was presented in Section 4.2. The procedure used by the computer program for end bent caps is similar with a few modifications. The following subsections highlight the differences in the two procedures. Appendix B contains the procedure used by the computer program for end bent caps in an outline format and can be referenced alongside this chapter.

Because of the simplifying assumptions necessary to develop a strut-and-tie model for end bent caps, it is recommended that traditional analysis and design methods be performed alongside the STM design for these substructure components. Design checks associated with traditional methods should be satisfied along with standard detailing practices for end bents.

5.2.1 Instructions

Similar to the *Pier Cap STM Design* workbook for the multi-column bent cap, the *Instructions* sheet will appear when the user first opens the *End Bent Cap STM Design* workbook. The basic instructions and limitations included in this workbook are the same as those for the multi-column bent cap. However, assumptions for the program for end bent caps are different than those for the pier caps. The assumptions for end bent caps are included on the *Instructions* sheet.

5.2.2 Subroutines

The same 20 subroutines (subs) presented in Section 4.2.2 are used to perform the STM design steps for end bent caps. Only the subs in the *End Bent Cap STM Design* workbook that contain differences from those in the *Pier Cap STM Design* workbook are detailed in the following subsections. The remaining subs are identical to those used for the pier caps.

5.2.3 User Inputs

Two required user inputs for end bent caps differ from those required for pier caps, and these differences are reflected within the “Inputs” sub of the program. First, the end bent cap program requires the engineer to input the effective depth of the cap within the “Geometric Properties” table. Here, the effective depth is defined as the portion of the member that is considered effective in resisting factored load demand. The effective depth is measured from the top surface of the end bent cap, as shown in Figure 5.1. Because the ends of supporting piles are embedded within the concrete of end bent caps, a designer may assume that the effective depth is less than the actual depth of the member. For example, if only the portion of the member above the end of the piles is assumed to be effective, the bottom horizontal chord of the resulting strut-and-tie model will be placed above the piles. Other designers may assume that the entire depth of the member is effective, placing primary longitudinal reinforcement along the bottom of the member. The strut-and-tie model will be developed within this effective depth input by the designer. The program assumes that the location at which the pile reactions are introduced to the member coincides with the bottom of the effective depth (i.e., the pile reactions are assumed to be applied at an effective depth below the top surface of the end bent cap). Regardless, of the effective depth that is assumed, the entire height of the member is still used for self-weight distribution calculations.

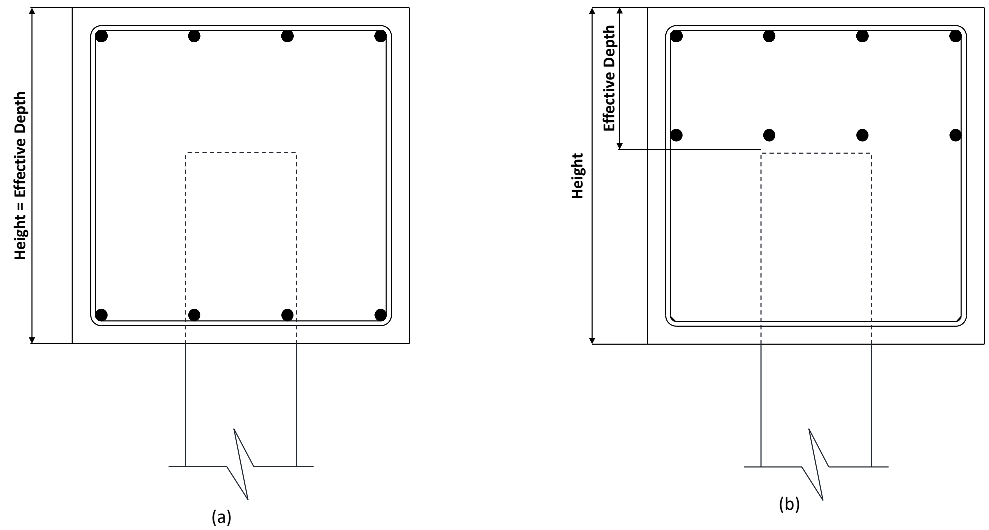


Figure 5.1: Effective Depth Measurement for End Bent Cap – (a) entire member depth is effective; (b) portion of member depth is effective

The second difference between the inputs for the end bent caps and pier caps is related to the manner in which the program handles support reactions. The “Pile Reactions” table on the *Inputs* sheet for an end bent cap replaces the “Reactions” table that is provided for pier caps. The “Pile Reactions” table is shown in Figure 5.2. For end bent caps, the designer has the option of inputting pile reactions based on an analysis performed outside of the program. This option is provided to allow the designer to calculate pile reactions using a pile group analysis and input calculated values into the “Pile Reactions” column in the table. In this analysis, the designer considers the stiffness of each pile and the moment is distributed to different piles based on the moment of inertia of the pile group.

Pile Reactions			
Location from Left	Pile Reactions	Effective Width of Bearing Area	Effective Length of Bearing Area
ft	kip	ft	ft

Figure 5.2: "Pile Reactions" Input Table for End Bent Caps

If the “Pile Reactions” column is left blank, the computer program will calculate reaction values using the same continuous beam analysis described in Section 4.2.4.2. Within the “Inputs” sub, the program defines a variable, “cont_analysis,” as true or false based on whether values are input into the “Pile Reactions” column. The “cont_analysis” variable will be set to true if no pile reactions are input. If pile reactions are provided, the “cont_analysis” variable will be set to false. Regardless of the type of analysis that is desired, values must be input into the other three columns in the table (i.e., reaction locations and effective bearing area dimensions) for each pile supporting the end bent cap.

5.2.4 Structural Analysis

The “Structural_Analysis” sub within the *End Bent Cap STM Design* workbook contains an IF statement that is dependent on the “cont_analysis” variable defined in the “Inputs” sub. If the “cont_analysis” variable is true, the sub will calculate reaction values using the moment distribution method described in Section 4.2.4.2. If the “cont_analysis” variable is false, the program will perform force and moment equilibrium checks to ensure the pile reactions entered by the user are in equilibrium with the loads entered into the “Factored Applied Loads” table. In this case, the *Cont. Beam Analysis* sheet will not be created because no continuous beam analysis is performed. The force equilibrium check adds all load values input in the “Factored Applied Loads” table and all reaction values input in the “Pile Reactions” table and evaluates the difference in the two resulting values. The moment equilibrium check sums moments about the left end of the member due to the applied loads and reactions.

If the magnitude of the difference between the sum of the applied loads and the sum of the reactions is less than 0.5 kip and the magnitude of the sum of moments is less than 0.5 kip-ft, the program assumes the structural component is in equilibrium and proceeds with developing the strut-and-tie model. If the magnitudes are greater than the limits of 0.5 kip and/or 0.5 kip-ft, a window appears displaying the following message: "Force and moment equilibrium have been checked. Sum of forces equals ___ kip and sum of moments equals ___ kip-ft. Small deviations from zero can be a result of rounding error. Do you wish to proceed?" The blanks in the message represent the values calculated by the program for the sum of forces and the sum of moments. If the designer is confident the deviations from zero are a result of rounding error, he or she should click the "Yes" button below the message and the program will continue to run. If the designer believes incorrect pile reactions have been input, he or she should click the "No" button. When the "No" button is clicked, the program stops running, and a message is displayed that states, "The program has stopped. Please correct pile reaction values and rerun the program." The designer should update the pile reaction values, click on the "Reset Program" button, and then click on the "Run Program" button to perform the design procedures for the end bent cap.

5.2.5 Nodal Checks

The nodal strength checks performed for nodes along the top chord of the strut-and-tie model for end bent caps follow the same procedure as those for pier caps. However, along the bottom chord, the strengths of the bearing faces of nodes located at pile reactions are not checked. Because reaction forces are transferred to a pile through both bearing at the end of the pile and friction along the portion of the pile embedded in the concrete, the bearing face of the nodes is not considered. The "Nodal_Checks" sub reflects this difference. In the *Nodal Checks* sheet, the program displays "Node located at H-pile. Check N/A" for bearing strength checks at nodes located along the bottom chord of the strut-and-tie model. If the designer desires to perform a bearing check at the piles based on experience or standard practice, the check must be performed outside the program. Furthermore, the confinement modification factor, *m*, is not considered in strength calculations for nodes located at piles. The benefits of confinement produced by a bearing plate on the surface of a member are not expected to exist at the nodes located at pile locations. The confinement modification factor is therefore displayed as 1.00 for the nodes along the bottom chord in the *Nodal Checks* sheet.

5.2.6 Anchorage Checks

The anchorage check for end bent caps performed using the “Anchorage” sub is similar to the check used for pier caps. However, an extended nodal zone is not included in the calculation of the available length for development of reinforcement located along the bottom chord of an end bent cap. Due to the uncertainty of the geometries of the diagonal struts entering the nodes located at pile supports, the inside face of the pile is assumed to be the critical section for bar development.

5.3 Additional Limitations and Assumptions

The additional limitations and assumptions in Section 4.3 also apply to the *End Bent Cap STM Design* workbook. All limitations and assumptions are also listed in Appendix B. The engineer should review these before using the computer program to ensure the structural element being designed meets the limitations and assumptions of the computer program.

5.4 Design Example: Integral End Bent

The design example presented in the following subsections is intended to familiarize engineers with the implementation of the computer program to aid in performing STM design procedures for an integral or semi-integral end bent cap. The example presented is for an integral end bent cap, but the same procedures apply to a semi-integral end bent cap. The following subsections focus on the use of the computer program to aid in the design of the structural component.

5.4.1 End Bent Cap Geometry, Material Properties, and Loading

The layout of the integral end bent cap is presented in Figure 5.3 and Figure 5.4. The integral end bent cap is supported by seven HP14x89 steel H-piles spaced at 6 ft-8 in. along the cap. For the design, each support is assumed to act as a pin. For the design, each support is assumed to act as a pin. Equivalent bearing area dimensions for the H-piles need to be entered into the program. The equivalent bearing area is assumed to be a rectangular area tracing the outside of the pile. The

“Effective Width of the Bearing Area” is assumed to be 14.7 in. and the “Effective Length of the Bearing Area” is assumed to be 13.8 in. based on the dimensions of an HP14x89 H-pile.

A total of four bulb-tee prestressed concrete girders are supported on the integral end bent cap and are spaced at 11 ft-9 in. along the cap. The elevation view of the superstructure is shown in Figure 5.4. The bridge has no skew, and loaded areas are already centered on the longitudinal axis of the cap. The loaded areas for the girder reactions that are applied to the end bent cap are 9 in. by 33 in.

The computer program can only consider one load case for a single run of the program. The load case used for this design example is shown in Figure 5.5. The distributed self-weight has not been included in these applied girder loads. The loads shown in the figure have already been factored based on the Strength I load combination in AASHTO LRFD (2012).

Within the design example, the specified concrete compressive strength, f_c', is taken as 4 ksi and the specified yield strength, f_y, of all reinforcement is 60 ksi. Concrete is normalweight.

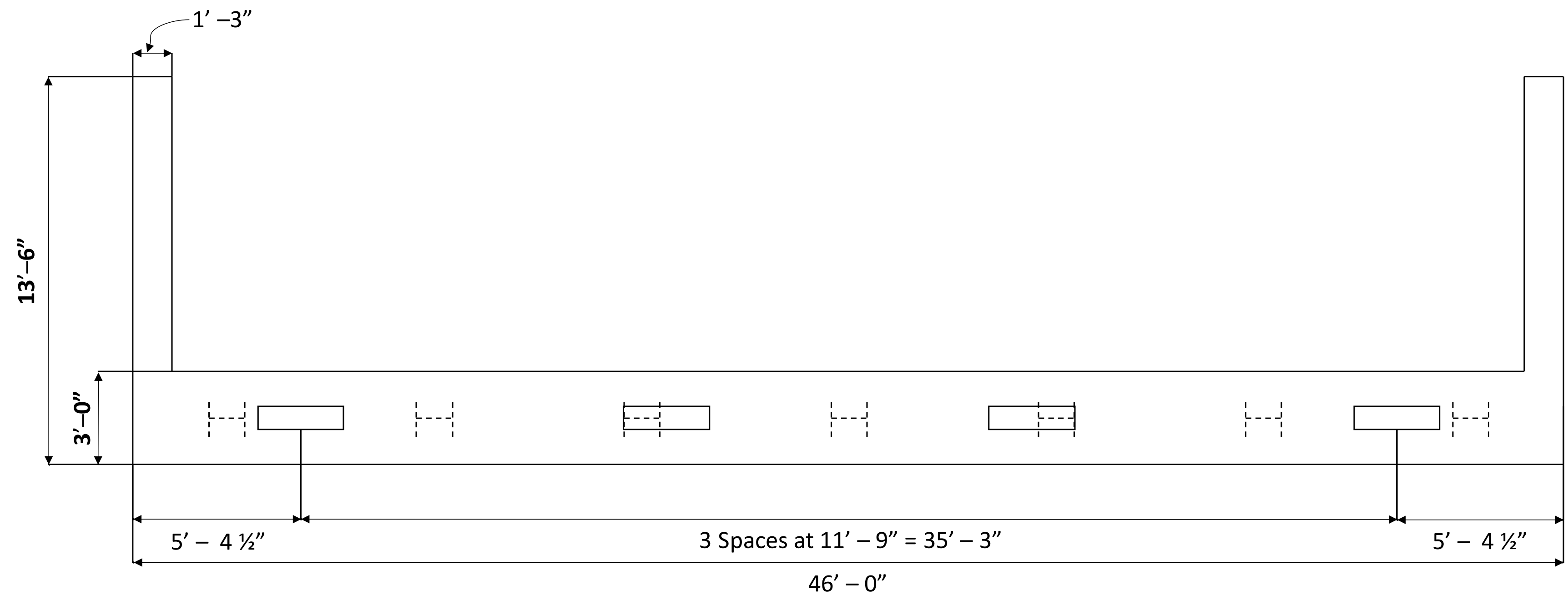


Figure 5.3: Plan View of Integral End Bent Cap

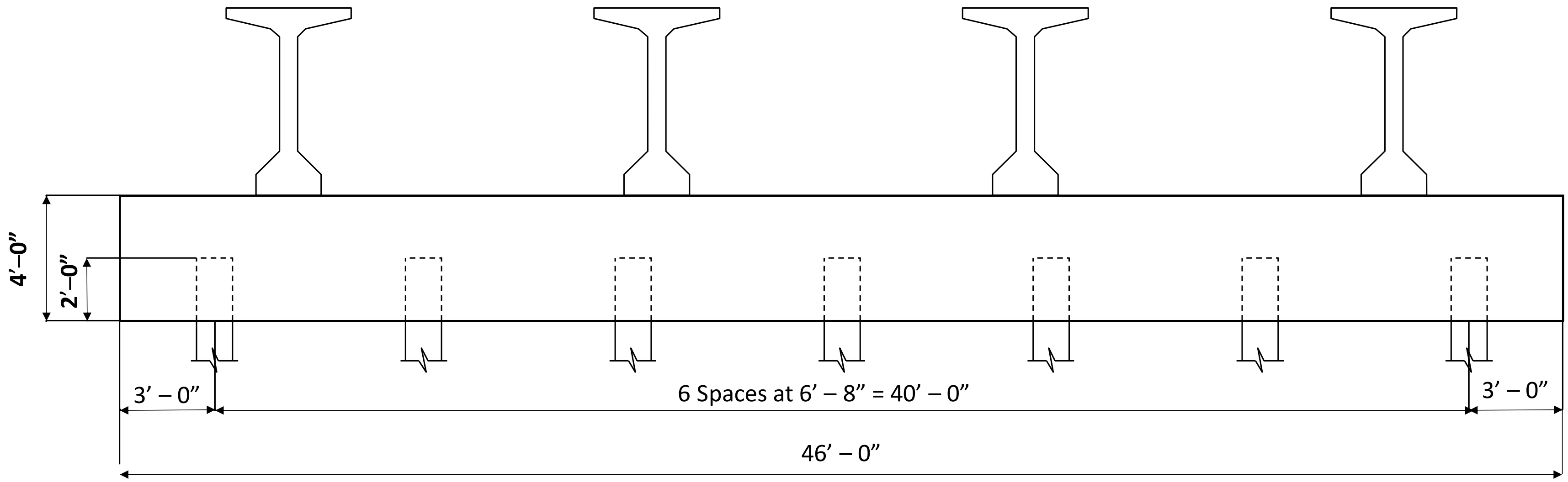


Figure 5.4: Elevation View of Integral End Bent

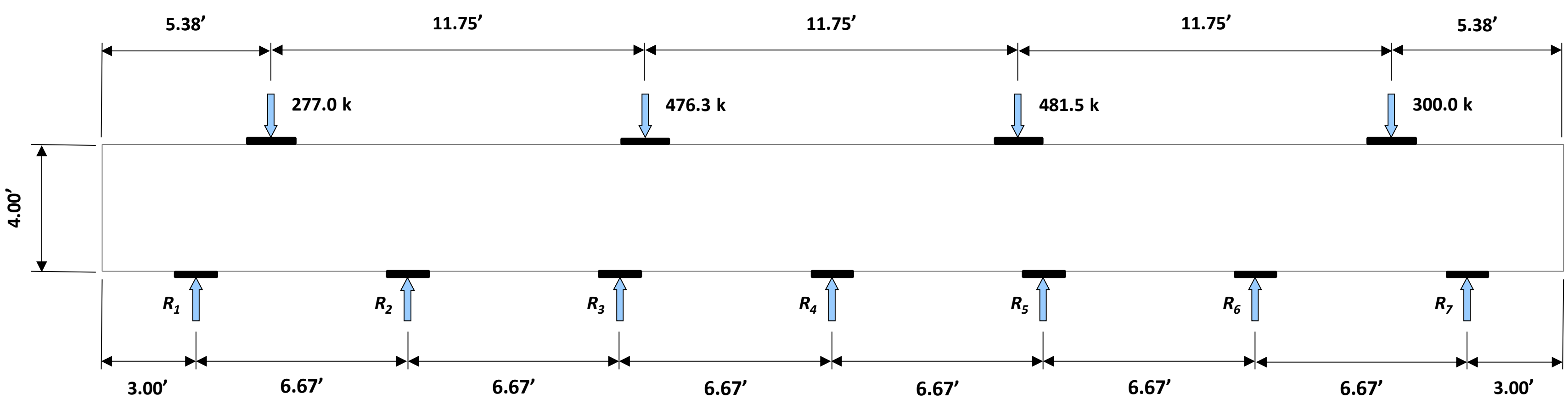


Figure 5.5: Factored Loads and Locations of Reactions

5.4.2 User Inputs

To begin using the computer program, information about the structural component and the load case being considered is entered into the tables on the *Inputs* sheet. The *Inputs* sheet with all values properly entered for the integral end bent cap is shown in Figure 5.6. The “Geometric Properties” table includes the dimensions of the member: a length of 46 ft, a height of 4 ft, an effective depth of 4 ft, and a width of 3 ft. The longitudinal reinforcement carrying the forces in the horizontal ties located along the bottom chord of the strut-and-tie model (see Section 0) is placed near the bottom surface of the member. Therefore, the entire height of the member is used as the effective depth, and the inputs for the member height and effective depth are the same. The geometric properties can be input into the program with units of inches or feet. Units of feet was chosen, and the “ft” option was selected from the dropdown menu in the units cell for all geometric properties. In the “Load Factor” table, the self-weight distribution factor is input as the maximum value of 1.25 for *DC* loads in the Strength I load combination in AASHTO LRFD. Next, the specified compressive strength of concrete is input into the “Concrete Material Properties” table. The unit weight of concrete, w_c, is kept as the default value of 150 pcf for normalweight concrete.

Information for the stirrups and horizontal crack control reinforcement (i.e., skin reinforcement) is entered into the corresponding tables. As shown in Figure 5.6, the specified yield strength of stirrups is kept as the default value of 60 ksi. No. 6 bars are used for stirrups, and No. 7 bars are used for horizontal crack control reinforcement. Two stirrup legs are provided along the entire length of the cap. Furthermore, for skin reinforcement, the default value of two horizontal bars, one near each side face, are provided through the width of the member within spacing s_h.

Before inputting information on this tab, see **Instructions** tab.

Notes:
-Click on any cell with ***BLUE*** *text for additional information.*
-Units in ***PURPLE*** *text can be changed. Units in* ***BLACK*** *text cannot be changed. Input standard bar size (e.g., input "5" for a No. 5 bar).*

Reset Program | **Run Program** | **Rerun Program**

Geometric Properties		
Length	46	ft
Height	4	ft
Effective Depth	4	ft
b_w	3	ft

Load Factor	
Self-Weight Factor	1.25

Concrete Material Properties		
f'_c	4	ksi
w_c	150	pcf

Stirrups		
f_y	60	ksi
Bar Designation	6	
No. of Legs	2	

Horizontal Crack Control Reinforcement	
Bar Designation	7
# Bars/s_h	2

Longitudinal Reinforcement

Bottom Reinforcement			
f_y		60	ksi
No. of Layers		1	
Development Length Straight		94.5	in.
Development Length Hook		19.95	in.
Clear Cover at End of Bar		3	in.
Layer	Location (in.)	Number of Bars	Bar Size (no.)
1	3.19	4	7
2			
3			
4			

Top Reinforcement			
f_y		60	ksi
No. of Layers		1	
Development Length Straight		122.9	in.
Development Length Hook		19.95	in.
Clear Cover at End of Bar		3	in.
Layer	Location (in.)	Number of Bars	Bar Size (no.)
1	44.81	4	7
2			
3			
4			

Factored Applied Loads			
Location from Left	Load Value	Width of Loaded Area	Length of Loaded Area
ft	kip	in.	in.
5.38	277.0	9.0	33.0
17.13	467.3	9.0	33.0
28.88	481.5	9.0	33.0
40.63	300.0	9.0	33.0

Pile Reactions			
Location from Left	Pile Reactions	Effective Width of Bearing Area	Effective Length of Bearing Area
ft	kip	in.	in.
3		14.7	13.8
9.6666667		14.7	13.8
16.333333		14.7	13.8
23		14.7	13.8
29.666667		14.7	13.8
36.333333		14.7	13.8
43		14.7	13.8

Figure 5.6: Inputs Sheet for Design Example (End Bent Cap)

Information for the longitudinal reinforcement located along the top and bottom of the member is also entered into the corresponding tables. The default value of 60 ksi for the specified yield strength of reinforcement is used. Four No. 7 bars are provided for both the top and bottom longitudinal reinforcement. The development length for a straight and a hooked No. 7 bar is calculated according to Article 5.10.8.2.1a and Article 5.10.8.2.4a of AASHTO LRFD (2017), respectively. The Indiana Design Manual requires that reinforcement in end bent caps be epoxy coated. Therefore, the coating factor, λ_{cf} or $\lambda_{cw,}$ is equal to 1.5 for straight bars 1.2 for hooked bars. For straight bars along the top of the member, the reinforcement location factor, λ_{rl}, is equal to 1.3. The remaining modification factors are assumed to be 1.0. The development lengths calculated for straight and hooked bars along the top and bottom of the member are input into the "Longitudinal Reinforcement" table as shown in Figure 5.6Figure 5.6.

The "Factored Applied Loads" table and "Pile Reactions" table are filled with the information in Figure 5.5. The locations and values of applied loads as well as loaded area dimensions are entered in the "Factored Applied Loads" table. Pile reaction locations and equivalent bearing area dimensions are entered in the "Pile Reactions" table. The equivalent bearing area for an HP14x89 H-pile is assumed to be a rectangle drawn around the H-pile. In other words, the area used to define the bottom face of a node located at a pile is taken as the smallest rectangle in which the H-pile can fit. Therefore, the effective width of the bearing area is 14.7 in., and the effective length of the bearing area is 13.8 in. The designer should be reminded that the program does not check bearing stresses at the location of a pile (see Section5.2.5). For this example, a continuous beam analysis is used to solve for pile reactions. Therefore, the "Pile Reactions" column of the table is left blank.

5.4.3 Computer Program Outputs

After all user inputs are entered, the program is run by clicking on the "Run Program" button on the *Inputs* tab. The computer program performs the analysis and design steps in accordance with the strut-and-tie method and outputs the results on various sheets. In the following subsections, the information on each of the output sheets will be discussed, and the final design of the end bent cap for the load case under consideration will be presented.

5.4.3.1 Strut-and-Tie Model

After running the program, the *Strut-and-Tie Model* sheet will automatically open and display the strut-and-tie model for the end bent cap. The strut-and-tie model developed by the computer program for the integral end bent cap is shown in Figure 5.7. Because the effective depth and the total height of the member are the same, no dashed line appears on the plot. However, if the effective depth was different from the total height of the member, a dashed horizontal line would be shown to represent the effective depth of the member.

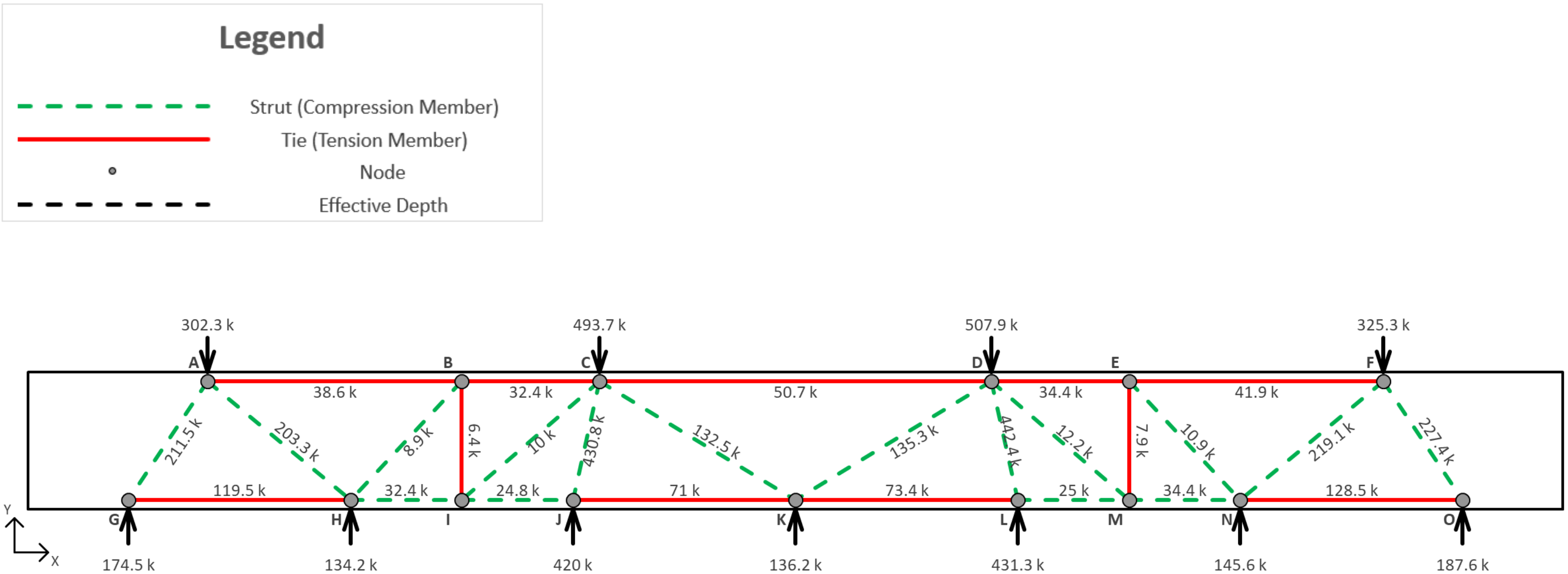


Figure 5.7: Strut-and-Tie Model from Program Output (End Bent Cap)

5.4.3.2 Nodes and Members

The *Nodes and Members* sheet contains information about the model in a table format as shown in Figure 5.8. The “Nodes” table includes the x- and y-coordinates of each node measured from the bottom left corner of the member. If a node has been subdivided, the coordinates of each part of the subdivided node are provided in the “Subdivided Nodes” table. The “Members” table lists each member (i.e., each strut and tie) and the force within the member. The strut-and-tie model for the integral end bent cap contains 15 nodes, 7 of which are subdivided into two parts. The model also contains 27 members. The “Nodes with Combined Forces and Updated Angles” table contains the 18 nodes (or portions of nodes) that undergo strength checks. The forces acting at each node and the angle (i.e., orientation) of each force are listed for these 18 nodes (or portions of nodes).

Nodes		
Label	x (ft)	y (ft)
A	5.38	3.73
B	13.00	3.73
C	17.13	3.73
D	28.88	3.73
E	33.00	3.73
F	40.63	3.73
G	3.00	0.27
H	9.67	0.27
I	13.00	0.27
J	16.33	0.27
K	23.00	0.27
L	29.67	0.27
M	33.00	0.27
N	36.33	0.27
O	43.00	0.27

Subdivided Nodes		
Label	x (ft)	y (ft)
A Left	5.22	3.73
A Right	5.59	3.73
C Left	17.07	3.73
C Right	17.45	3.73
D Left	28.55	3.73
D Right	28.93	3.73
F Left	40.41	3.73
F Right	40.78	3.73
H Left	9.64	0.27
H Right	10.24	0.27
K Left	22.68	0.27
K Right	23.30	0.27
N Left	35.74	0.27
N Right	36.37	0.27

Members		
Label	Force	S or T
A-B	37.9	Tie
B-C	32.0	Tie
C-D	49.9	Tie
D-E	33.9	Tie
E-F	41.2	Tie
G-H	117.5	Tie
H-I	-32.0	Strut
I-J	-24.7	Strut
J-K	70.2	Tie
K-L	72.6	Tie
L-M	-24.9	Strut
M-N	-33.9	Strut
N-O	126.5	Tie
B-I	6.2	Tie
E-M	7.6	Tie
A-G	-208.0	Strut
A-H	-199.8	Strut
B-H	-8.5	Strut
C-I	-9.6	Strut
C-J	-426.4	Strut
C-K	-131.0	Strut
D-K	-133.8	Strut
D-L	-438.0	Strut
D-M	-11.8	Strut
E-N	-10.5	Strut
F-N	-215.6	Strut
F-O	-223.9	Strut

Nodes with Combined Forces and Updated Angles						
Node	Force (kip)	Angle (deg)	Force (kip)	Angle (deg)	Force (kip)	Angle (deg)
A Left	-208.0	237.42	-117.5	0.00		
A Right	37.9	0.00	-199.8	319.60	-117.5	180.00
C Left	32.0	180.00	-434.1	257.18	-70.2	0.00
C Right	49.9	0.00	-131.0	328.00	-70.2	180.00
D Left	49.9	180.00	-133.8	212.00	-65.3	0.00
D Right	33.9	0.00	-447.5	282.98	-65.3	180.00
F Left	41.2	180.00	-215.6	220.40	-126.5	0.00
F Right	-223.9	302.58	-126.5	180.00		
G	117.5	0.00	-208.0	57.42		
H Left	117.5	180.00	-199.8	141.80	-40.1	0.00
H Right	-38.4	9.47	-40.1	180.00		
J	-24.7	180.00	70.2	0.00	-426.4	77.95
K Left	70.2	180.00	-131.0	148.29	-42.8	0.00
K Right	72.6	0.00	-133.8	33.40	-42.8	180.00
L	72.6	180.00	-24.9	0.00	-438.0	102.06
N Left	-41.9	169.22	-41.2	0.00		
N Right	126.5	0.00	-215.6	40.50	-41.2	180.00
O	126.5	180.00	-223.9	122.58		

Figure 5.8: Nodes and Members Sheet from Program Output (End Bent Cap)

5.4.3.3 Node Figures

The *Node Figures* sheet displays figures representing the 18 nodes (or portions of nodes) listed in the “Nodes with Combined Forces and Updated Angles” table on the *Nodes and Members* sheet. For each node, a graphic is provided that illustrates all members that enter the node. Three of the graphics generated for the end bent cap are shown in Figure 5.9. If any struts were combined at the node, the resultant of the struts is displayed. The angles labeled in the figures reflect the updated angles resulting from the subdivision of nodes. By providing a visual representation of each node, the graphics are meant to aid designers with understanding the nodal strength checks that are performed by the program.

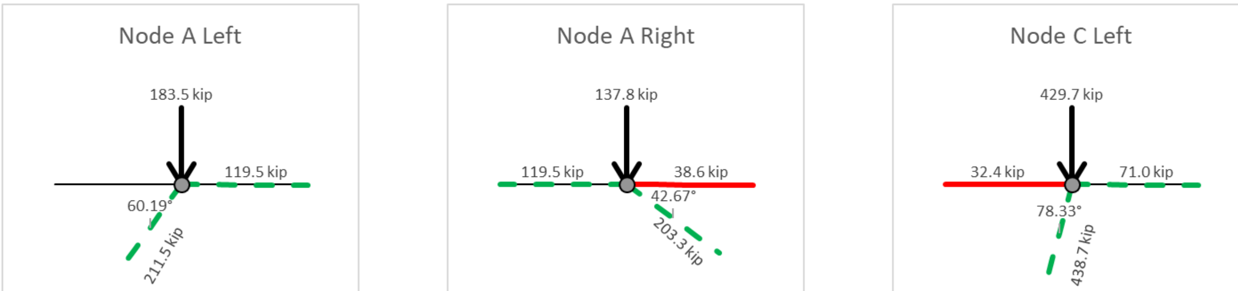


Figure 5.9: Examples of Node Figures from Program Output (End Bent Cap)

5.4.3.4 Nodal Checks

The *Nodal Checks* sheet provides a summary of the strength checks for the 18 nodes (or portions of nodes) included in the *Node Figures* sheet. For each node, the factored design strengths of the bearing face, back face, and strut-to-node interface are calculated, as applicable, and compared to the factored force acting on each face. The summary of the nodal strength checks for the end bent cap as displayed on the *Nodal Checks* sheet is provided in Figure 5.10. If the words “Check N/A” appear for a back face check, no direct compressive force was applied to the face. As indicated by the “NG” in the “Pass?” column and the red bolded text in Figure 5.10, the strut-to-node interfaces of Node C-Left, D-Right, Node J, and Node K do not have adequate strength. When strength checks are not satisfied, the engineer should use his or her discretion to correct the design of the structural component accordingly. Two of the nodes that do not satisfy the strength checks are along the top chord at girder locations and two of the nodes are at the location of H-piles along the bottom chord of the strut-and-tie model. Admittedly, the strut-and-tie model simplifies the transfer of force to the embedded piles. Engineering judgement can therefore be used to determine the solution to the strength checks that are not satisfied. In this design example, the loaded area width is changed to 12 in. to provide adequate strength at the two strut-to-node interfaces along the top chord, and the specified concrete compressive strength, f_c', is increased to 5 ksi to provide adequate strength at the two strut-to-node interfaces along the bottom chord.

Node	Type	Triaxial Confinement Factor, m	Face Length (in.)			Strength Checks														
						Bearing					Back					Strut-to-Node				
			Bearing	Back	Strut-to-Node	F_u (kip)	Efficiency (ν)	f_{cu} (ksi)	ϕF_n (kip)	Pass?	F_u (kip)	Efficiency (ν)	f_{cu} (ksi)	ϕF_n (kip)	Pass?	F_u (kip)	Efficiency (ν)	f_{cu} (ksi)	ϕF_n (kip)	Pass?
A Left	CCC	2.0	5.2	6.4	7.8											211.5	0.65	5.2	255.9	OK
A Right	CCT	2.0	3.8	6.4	7.3	302.3	0.70	5.6	1164.2	OK	119.5	0.70	5.6	824.7	OK	203.3	0.65	5.2	239.8	OK
C Left	CCT	2.0	7.8	6.4	9.0											**438.7**	**0.65**	**5.2**	**294.7**	**NG**
C Right	CCT	2.0	1.2	6.4	6.1	493.7	0.70	5.6	1164.2	OK	71.0	0.70	5.6	824.7	OK	132.5	0.65	5.2	198.4	OK
D Left	CCT	2.0	1.2	6.4	6.1											135.3	0.65	5.2	198.3	OK
D Right	CCT	2.0	7.8	6.4	9.0	507.9	0.70	5.6	1164.2	OK	65.8	0.70	5.6	824.7	OK	**452.2**	**0.65**	**5.2**	**295.3**	**NG**
F Left	CCT	2.0	3.8	6.4	7.3											219.1	0.65	5.2	239.9	OK
F Right	CCC	2.0	5.2	6.4	7.8	325.3	0.70	5.6	1164.2	OK	128.5	0.70	5.6	824.7	OK	227.4	0.65	5.2	255.7	OK
G	CCT	1.0	13.8	6.4	15.1	Node located at H-pile. Check N/A.					Check N/A					211.5	0.65	2.6	402.9	OK
H Left	CCT	1.0	14.4	6.4	14.2											203.3	0.65	2.6	380.4	OK
H Right	CCC	1.0	0.7	6.4	6.4	Node located at H-pile. Check N/A.					35.3	0.70	2.8	183.7	OK	39.1	0.65	2.6	171.3	OK
J	CCT	1.0	13.8	6.4	14.8	Node located at H-pile. Check N/A.					24.8	0.70	2.8	183.7	OK	**430.8**	**0.65**	**2.6**	**396.7**	**NG**
K Left	CCT	1.0	7.6	6.4	9.5											132.5	0.65	2.6	254.1	OK
K Right	CCT	1.0	7.7	6.4	9.6	Node located at H-pile. Check N/A.					41.3	0.70	2.8	183.7	OK	135.3	0.65	2.6	256.4	OK
L	CCT	1.0	13.8	6.4	14.8	Node located at H-pile. Check N/A.					25.0	0.70	2.8	183.7	OK	**442.4**	**0.65**	**2.6**	**396.7**	**NG**
N Left	CCC	1.0	0.8	6.4	6.4											42.6	0.65	2.6	171.5	OK
N Right	CCT	1.0	14.3	6.4	14.2	Node located at H-pile. Check N/A.					41.9	0.70	2.8	183.7	OK	219.1	0.65	2.6	378.8	OK
O	CCT	1.0	13.8	6.4	14.9	Node located at H-pile. Check N/A.					Check N/A					227.4	0.65	2.6	397.9	OK

Figure 5.10 Summary of Nodal Check Program Outputs (End Bent Cap)

5.4.3.5 Reinforcement

The *Reinforcement* sheet displays a summary of the adequacy of the longitudinal reinforcement, spacing requirements for crack control reinforcement, stirrup spacing requirements based on vertical tie forces, and anchorage checks. A table that summarizes the strength checks for the longitudinal reinforcement is displayed first. This table is provided in Figure 5.11. On the *Inputs* sheet, four No. 7 bars were entered as the longitudinal reinforcement along both the top and bottom of the end bent cap, resulting in a factored nominal resistance, $\phi A_{st} f_y$, of 129.6 kip for each chord as displayed in the table. Each horizontal tie along the bottom and top chords of the strut-and-tie model and the force in each tie are also listed in the table. The factored force demand for each tie is compared to the factored nominal capacity. As indicated by "OK" displayed for each tie in the "Pass?" columns, the top and bottom longitudinal reinforcement provided in the bent cap have adequate capacity to carry the tensile forces in all horizontal ties along the top and bottom chords, respectively.

Longitudinal Reinforcement					
Bottom Reinforcement			**Top Reinforcement**		
$\phi A_s F_y$	129.6	kip	$\phi A_s F_y$	129.6	kip
Member	**Force (kip)**	**Pass?**	**Member**	**Force (kip)**	**Pass?**
G-H	119.5	OK	A-B	38.6	OK
J-K	71.0	OK	B-C	32.4	OK
K-L	73.4	OK	C-D	50.7	OK
N-O	128.5	OK	D-E	34.4	OK
			E-F	41.9	OK

Figure 5.11: Summary of Longitudinal Reinforcement Strength Checks from Program Output (End Bent Cap)

Crack control reinforcement requirements are displayed next on the *Reinforcement* sheet. The program output is provided in Figure 5.12. The maximum spacing required to satisfy horizontal and vertical crack control reinforcement specifications are shown. No. 7 bars were input for horizontal crack control reinforcement, and No. 6 bars were input for stirrups (i.e., vertical crack control reinforcement). The maximum spacing that can be provided for horizontal crack control reinforcement is 11.1 in. The maximum spacing that can be provided for vertical crack control reinforcement is 8.1 in.

Crack Control Reinforcement		
Horizontal Reinforcement		
Max Spacing	11.1	in.
Vertical Reinforcement		
Max Spacing	8.1	in.

Figure 5.12: Crack Control Reinforcement Requirements from Program Output (End Bent Cap)

The summary of stirrup spacing requirements is provided after the table for crack control reinforcement. The "Stirrups" table on the Reinforcement *sheet for the integral end bent cap is shown in*

. In the table, each vertical tie in the strut-and-tie model is listed along with the force in the tie. Only two vertical ties are present in the strut-and-tie model for the end bent cap. No. 6 stirrups with two legs were input into the program. The width of each tie given in the table is calculated following the procedure described in Section 4.2.6.1. The spacing that is required to carry the force in the ties is provided in the next column. The procedure for calculating this value is also described in Section 4.2.6.1. The required vertical crack control reinforcement spacing previously provided in the "Crack Control Reinforcement" table is also displayed in the "Stirrups" table. For each tie, the required spacing to carry the tie force is compared to the required spacing for crack control reinforcement, and the governing value is displayed in the last column of the table. For the end bent cap, the forces in the vertical ties are small, and the spacing required for crack control governs for both ties. Therefore, a stirrup spacing no greater than 8.1 in. should be provided along the length of the member. If no vertical ties were present in the strut-and-tie model, the "Stirrups" table would not contain any information under the headings. Instead, a note will be displayed under the headings that says "No vertical ties exist in the strut-and-tie model. Use required spacing for vertical crack control reinforcement throughout the member."

Stirrups					
Member	Force (kip)	Vertical Tie Width (in.)	Req'd Spacing for Shear (in.)	Req'd Spacing for Crack Control (in.)	Governing Spacing (in.)
B-I	6.4	40.0	296.3	8.1	*8.1*
E-M	7.9	40.0	241.6	8.1	*8.1*

Figure 5.13: Stirrup Spacing Requirements from Program Output (End Bent Cap)

A summary of the check to ensure proper anchorage of the longitudinal reinforcement is provided in the last table on the *Reinforcement* sheet and is shown in Figure 5.14. Because at least one negative moment region exists along the length of the bent cap, anchorage is checked for both the top and bottom chords of the strut-and-tie model. More specifically, the space available for the reinforcement carrying the forces in the outermost ties of each chord is checked to ensure that the reinforcement can be fully developed. Nodes A and F are the nodes located at the outermost ends of the outermost ties at each end of the top chord. Similarly, Nodes G and O are the nodes located at the outermost ends of the outermost ties at each end of the bottom chord. The required development length for a straight No. 7 bar and a hooked No. 7 bar were entered into the *Inputs* sheet before running the program. The available length for the development of the reinforcement at Nodes A, F, G, and O are calculated by the program and displayed in the "Anchorage for Ties" table. The available length at all four nodes is not sufficient for a straight bar to be developed. However, the available length at all four nodes is sufficient for a hooked bar to be fully developed. Therefore, hooked bars should be used to anchor the top and bottom layers of reinforcement in the end bent cap. The bars will be extended to the ends of the member, leaving clear cover.

Anchorage for Ties					
	Lengths (in.)				
Node	Available Length	Reqd Hook	Pass?	Reqd Straight	Pass?
A	69.7	20.0	Yes	**122.9**	**No**
F	69.7	20.0	Yes	**122.9**	**No**
G	40.4	20.0	Yes	**94.5**	**No**
O	40.4	20.0	Yes	**94.5**	**No**

Figure 5.14: Anchorage Check from Program Output (End Bent Cap)

5.4.3.6 Continuous Beam Analysis

After the *Reinforcement* sheet, the remaining sheets output by the program include information related to preliminary steps that were taken before developing the strut-and-tie model and performing the STM design procedure. The first of these sheets is the *Cont. Beam Analysis* sheet. Pile reactions were not entered into the *Inputs* sheet for this design example. Therefore, the program calculated the pile reactions using the moment distribution method. These calculations, displayed in the *Cont. Beam Analysis* sheet, are shown in Figure 5.15. The seven pile reactions are displayed in the last row of the table.

Column Index	Reaction 1		Reaction 2		Reaction 3		Reaction 4		Reaction 5		Reaction 6		Reaction 7	
Member		1-2	2-1	2-3	3-2	3-4	4-3	4-5	5-4	5-6	6-5	6-7	7-6	
Distribution Factor		1	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	1	
Carry-Over Factor		0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	
Fixed-End-Moment (kip-ft)	0.00	292.56	-161.90	0.00	0.00	300.30	-40.47	41.64	-309.03	0.00	0.00	174.4314807	-315.2007458	0
DIST		-292.5634	80.9520	80.9520	-150.1519	-150.1519	-0.5882	-0.5882	154.5170	154.5170	-87.2157	-87.2157	315.2007	
CO		40.4760	-146.2817	-75.0759	40.4760	-0.2941	-75.0759	77.2585	-0.2941	-43.6079	77.2585	157.6004	-43.6079	
DIST		-40.4760	110.6788	110.6788	-20.0909	-20.0909	-1.0913	-1.0913	21.9510	21.9510	-117.4294	-117.4294	43.6079	
CO		55.3394	-20.2380	-10.0455	55.3394	-0.5456	-10.0455	10.9755	-0.5456	-58.7147	10.9755	21.8039	-58.7147	
DIST		-55.3394	15.1417	15.1417	-27.3969	-27.3969	-0.4650	-0.4650	29.6302	29.6302	-16.3897	-16.3897	58.7147	
CO		7.5709	-27.6697	-13.6984	7.5709	-0.2325	-13.6984	14.8151	-0.2325	-8.1949	14.8151	29.3574	-8.1949	
DIST		-7.5709	20.6841	20.6841	-3.6692	-3.6692	-0.5583	-0.5583	4.2137	4.2137	-22.0862	-22.0862	8.1949	
CO		10.3420	-3.7854	-1.8346	10.3420	-0.2792	-1.8346	2.1068	-0.2792	-11.0431	2.1068	4.0974	-11.0431	
DIST		-10.3420	2.8100	2.8100	-5.0314	-5.0314	-0.1361	-0.1361	5.6611	5.6611	-3.1021	-3.1021	11.0431	
CO		1.4050	-5.1710	-2.5157	1.4050	-0.0681	-2.5157	2.8306	-0.0681	-1.5511	2.8306	5.5216	-1.5511	
DIST		-1.4050	3.8434	3.8434	-0.6685	-0.6685	-0.1574	-0.1574	0.8096	0.8096	-4.1761	-4.1761	1.5511	
CO		1.9217	-0.7025	-0.3342	1.9217	-0.0787	-0.3342	0.4048	-0.0787	-2.0880	0.4048	0.7755	-2.0880	
DIST		-1.9217	0.5184	0.5184	-0.9215	-0.9215	-0.0353	-0.0353	1.0834	1.0834	-0.5902	-0.5902	2.0880	
CO		0.2592	-0.9608	-0.4607	0.2592	-0.0176	-0.4607	0.5417	-0.0176	-0.2951	0.5417	1.0440	-0.2951	
DIST		-0.2592	0.7108	0.7108	-0.1208	-0.1208	-0.0405	-0.0405	0.1564	0.1564	-0.7929	-0.7929	0.2951	
CO		0.3554	-0.1296	-0.0604	0.3554	-0.0202	-0.0604	0.0782	-0.0202	-0.3964	0.0782	0.1475	-0.3964	
DIST		-0.3554	0.0950	0.0950	-0.1676	-0.1676	-0.0089	-0.0089	0.2083	0.2083	-0.1129	-0.1129	0.3964	
CO		0.0475	-0.1777	-0.0838	0.0475	-0.0044	-0.0838	0.1042	-0.0044	-0.0564	0.1042	0.1982	-0.0564	
DIST		-0.0475	0.1307	0.1307	-0.0215	-0.0215	-0.0102	-0.0102	0.0304	0.0304	-0.1512	-0.1512	0.0564	
CO		0.0654	-0.0237	-0.0108	0.0654	-0.0051	-0.0108	0.0152	-0.0051	-0.0756	0.0152	0.0282	-0.0756	
DIST		-0.0654	0.0173	0.0173	-0.0301	-0.0301	-0.0022	-0.0022	0.0403	0.0403	-0.0217	-0.0217	0.0756	
CO		0.0086	-0.0327	-0.0151	0.0086	-0.0011	-0.0151	0.0202	-0.0011	-0.0109	0.0202	0.0378	-0.0109	
DIST		-0.0086	0.0239	0.0239	-0.0038	-0.0038	-0.0026	-0.0026	0.0060	0.0060	-0.0290	-0.0290	0.0109	
CO		0.0119	-0.0043	-0.0019	0.0119	-0.0013	-0.0019	0.0030	-0.0013	-0.0145	0.0030	0.0054	-0.0145	
DIST		-0.0119	0.0031	0.0031	-0.0053	-0.0053	-0.0006	-0.0006	0.0079	0.0079	-0.0042	-0.0042	0.0145	
CO		0.0015	-0.0060	-0.0027	0.0015	-0.0003	-0.0027	0.0039	-0.0003	-0.0021	0.0039	0.0072	-0.0021	
DIST		-0.0015	0.0043	0.0043	-0.0006	-0.0006	-0.0006	-0.0006	0.0012	0.0012	-0.0056	-0.0056	0.0021	
CO		0.0022	-0.0008	-0.0003	0.0022	-0.0003	-0.0003	0.0006	-0.0003	-0.0028	0.0006	0.0011	-0.0028	
DIST		-0.0022	0.0005	0.0005	-0.0009	-0.0009	-0.0001	-0.0001	0.0016	0.0016	-0.0008	-0.0008	0.0028	
CO		0.0003	-0.0011	-0.0005	0.0003	-6.970E-05	-0.0005	0.0008	-6.970E-05	-0.0004	0.0008	0.0014	-0.0004	
DIST		-0.0003	0.0008	0.0008	-0.0001	-0.0001	-0.0002	-0.0002	0.0002	0.0002	-0.0011	-0.0011	0.0004	
CO		0.0004	-0.0001	-5.078E-05	0.0004	-7.977E-05	-5.078E-05	0.0001	-7.977E-05	-0.0005	0.0001	0.0002	-0.0005	
DIST		-0.0004	9.360E-05	9.360E-05	-0.0002	-0.0002	-3.485E-05	-3.485E-05	0.0003	0.0003	-0.0002	-0.0002	0.0005	
CO		4.680E-05	-0.0002	-7.627E-05	4.680E-05	-1.743E-05	-7.627E-05	0.0002	-1.743E-05	-8.165E-05	0.0002	0.0003	-8.165E-05	
DIST		-4.680E-05	0.0001	0.0001	-1.469E-05	-1.469E-05	-3.989E-05	-3.989E-05	4.954E-05	4.954E-05	-0.0002	-0.0002	8.165E-05	
CO		6.717E-05	-2.340E-05	-7.343E-06	6.717E-05	-1.994E-05	-7.343E-06	2.477E-05	-1.994E-05	-0.0001	2.477E-05	4.082E-05	-0.0001	
DIST		-6.717E-05	1.537E-05	1.537E-05	-2.361E-05	-2.361E-05	-8.713E-06	-8.713E-06	6.350E-05	6.350E-05	-3.280E-05	-3.280E-05	0.0001	
CO		7.686E-06	-3.359E-05	-1.181E-05	7.686E-06	-4.356E-06	-1.181E-05	3.175E-05	-4.356E-06	-1.640E-05	3.175E-05	5.353E-05	-1.640E-05	
DIST		-7.686E-06	2.270E-05	2.270E-05	-1.665E-06	-1.665E-06	-9.971E-06	-9.971E-06	1.038E-05	1.038E-05	-4.264E-05	-4.264E-05	1.640E-05	
CO		1.135E-05	-3.843E-06	-8.323E-07	1.135E-05	-4.986E-06	-8.323E-07	5.189E-06	-4.986E-06	-2.132E-05	5.189E-06	8.199E-06	-2.132E-05	
DIST		-1.135E-05	2.338E-06	2.338E-06	-3.181E-06	-3.181E-06	-2.178E-06	-2.178E-06	1.315E-05	1.315E-05	-6.694E-06	-6.694E-06	2.132E-05	
CO		1.169E-06	-5.674E-06	-1.591E-06	1.169E-06	-1.089E-06	-1.591E-06	6.576E-06	-1.089E-06	-3.347E-06	6.576E-06	1.066E-05	-3.347E-06	
DIST		-1.169E-06	3.632E-06	3.632E-06	-3.984E-08	-3.984E-08	-2.493E-06	-2.493E-06	2.218E-06	2.218E-06	-8.618E-06	-8.618E-06	3.347E-06	
CO		1.816E-06	-5.844E-07	-1.992E-08	1.816E-06	-1.246E-06	-1.992E-08	1.109E-06	-1.246E-06	-4.309E-06	1.109E-06	1.673E-06	-4.309E-06	
DIST		-1.816E-06	3.022E-07	3.022E-07	-2.849E-07	-2.849E-07	-5.445E-07	-5.445E-07	2.778E-06	2.778E-06	-1.391E-06	-1.391E-06	4.309E-06	
CO		1.511E-07	-9.081E-07	-1.424E-07	1.511E-07	-2.723E-07	-1.424E-07	1.389E-06	-2.723E-07	-6.956E-07	1.389E-06	2.155E-06	-6.956E-07	
DIST		-1.511E-07	5.253E-07	5.253E-07	6.060E-08	6.060E-08	-6.232E-07	-6.232E-07	4.839E-07	4.839E-07	-1.772E-06	-1.772E-06	6.956E-07	
CO		2.626E-07	-7.554E-08	3.030E-08	2.626E-07	-3.116E-07	3.030E-08	2.420E-07	-3.116E-07	-8.859E-07	2.420E-07	3.478E-07	-8.859E-07	
DIST		-2.626E-07	2.262E-08	2.262E-08	2.449E-08	2.449E-08	-1.361E-07	-1.361E-07	5.987E-07	5.987E-07	-2.949E-07	-2.949E-07	8.859E-07	
CO		1.131E-08	-1.313E-07	1.224E-08	1.131E-08	-6.807E-08	1.224E-08	2.994E-07	-6.807E-08	-1.474E-07	2.994E-07	4.429E-07	-1.474E-07	
DIST		-1.131E-08	5.954E-08	5.954E-08	2.838E-08	2.838E-08	-1.558E-07	-1.558E-07	1.078E-07	1.078E-07	-3.711E-07	-3.711E-07	1.474E-07	
CO		2.977E-08	-5.655E-09	1.419E-08	2.977E-08	-7.790E-08	1.419E-08	5.388E-08	-7.790E-08	-1.856E-07	5.388E-08	7.372E-08	-1.856E-07	
DIST		-2.977E-08	-4.267E-09	-4.267E-09	2.407E-08	2.407E-08	-3.403E-08	-3.403E-08	1.317E-07	1.317E-07	-6.380E-08	-6.380E-08	1.856E-07	
CO		-2.134E-09	-1.488E-08	1.203E-08	-2.134E-09	-1.702E-08	1.203E-08	6.587E-08	-1.702E-08	-3.190E-08	6.587E-08	9.279E-08	-3.190E-08	
DIST		2.134E-09	1.425E-09	1.425E-09	9.575E-09	9.575E-09	-3.895E-08	-3.895E-08	2.446E-08	2.446E-08	-7.933E-08	-7.933E-08	3.190E-08	
Sum of Moments (kip-ft)	0.00	0.00	-131.47	131.47	-90.47	90.47	-147.70	147.70	-92.26	92.26	-142.95	142.95	0.00	0.00
Shear from Point Loads (kip)	0.00	191.35	105.90	0.00	0.00	430.45	58.00	59.69	442.96	0.00	0.00	114.09	206.16	0.00
Shear from Unbalanced Moment (kip)		-19.72	19.72	6.15	-6.15	-8.58	8.58	8.32	-8.32	-7.60	7.60	21.44	-21.44	
Shear at Column (kip)	0.00	171.63	125.62	6.15	-6.15	421.86	66.59	68.01	434.64	-7.60	7.60	135.53	184.72	0.00
Point Loads at Column Location (kip)														
Solved Reactions (kip)	*Column 1*	171.6	*Column 2*	131.8	*Column 3*	415.7	*Column 4*	134.6	*Column 5*	427.0	*Column 6*	143.1	*Column 7*	184.7

Figure 5.15: Continuous Beam Analysis from Program Output (End Bent Cap)

5.4.3.7 Shear and Moment Diagrams

The shear and moment diagrams are displayed on the next two sheets of the program. The last sheet includes information needed to develop these diagrams. The shear diagram is used to determine the location of nodes and the orientation of diagonal struts in the strut-and-tie model. The moment diagram was not used for this end bent cap. However, for members with no negative moment regions, the moment diagram is used to optimize the location of the top chord of the model. The shear diagram, moment diagram, and corresponding information needed to develop the diagrams are shown in Figure 5.16, Figure 5.17, and Figure 5.18, respectively.

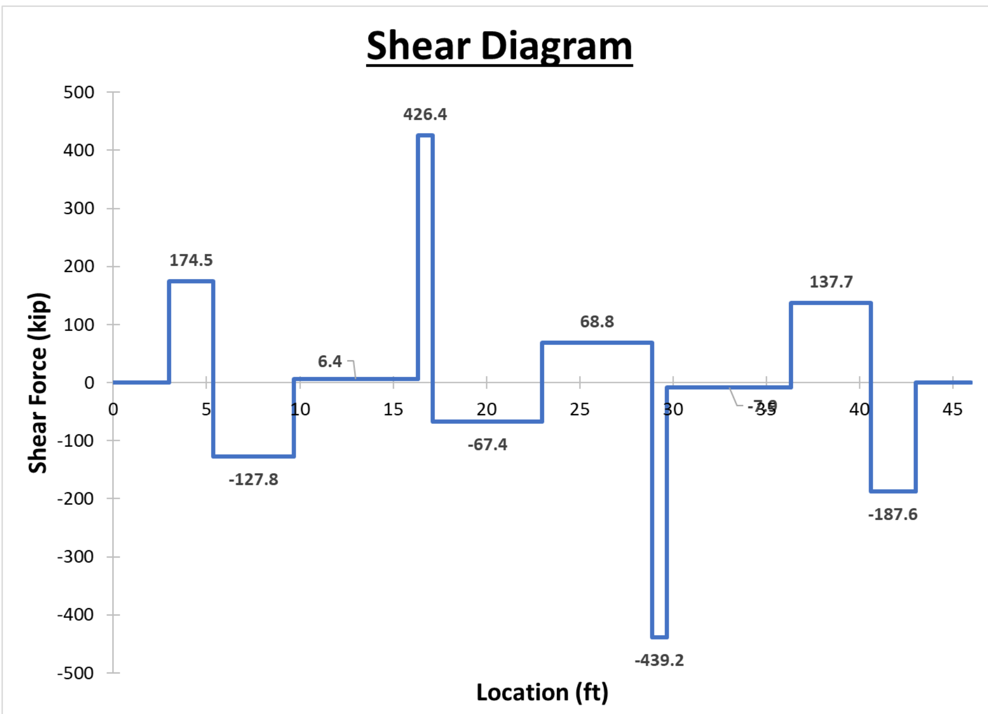


Figure 5.16: Shear Diagram from Program Output (End Bent Cap)

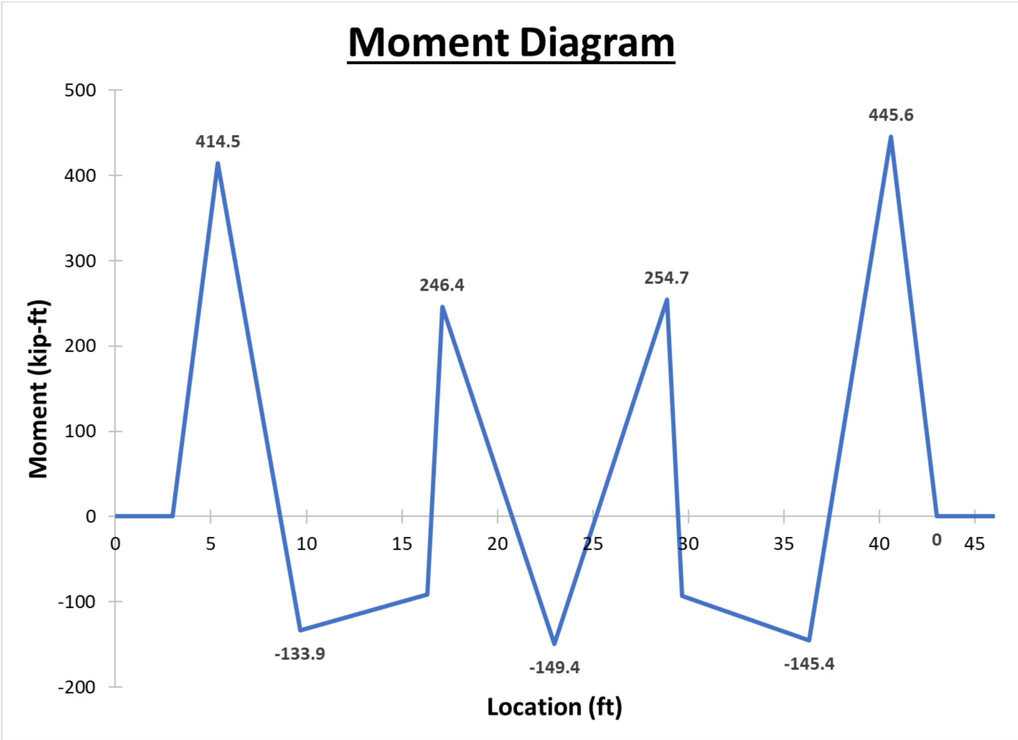


Figure 5.17: Moment Diagram from Program Output (End Bent Cap)

Forces	
x-Location (ft)	Load (kip)
3.00	174.5
5.38	-302.3
9.67	134.2
16.33	420.0
17.13	-493.7
23.00	136.2
28.88	-507.9
29.67	431.3
36.33	145.6
40.63	-325.3
43.00	187.6

Shear Diagram	
x-Location (ft)	Shear (kip)
0.00	0.0
3.00	0.0
3.00	174.5
5.38	174.5
5.38	-127.8
9.67	-127.8
9.67	6.4
16.33	6.4
16.33	426.4
17.13	426.4
17.13	-67.4
23.00	-67.4
23.00	68.8
28.88	68.8
28.88	-439.2
29.67	-439.2
29.67	-7.9
36.33	-7.9
36.33	137.7
40.63	137.7
40.63	-187.6
43.00	-187.6
43.00	0.0
46.00	0.0

Moment Diagram	
x-Location (ft)	Moment (kip-ft)
0.00	0.0
3.00	0.0
5.38	414.5
9.67	-133.9
16.33	-91.2
17.13	246.4
23.00	-149.4
28.88	254.7
29.67	-93.0
36.33	-145.4
40.63	445.6
43.00	0.0
46.00	0.0

Figure 5.18: Shear and Moment Diagram Calculations from Program Output (End Bent Cap)

5.4.3.8 Final Design

The program outputs for the end bent cap design example provide summaries of the design steps performed by the program based on the details input by the user. It is the responsibility of the engineer to interpret the results and determine if the design is adequate. For the integral end bent cap, the longitudinal reinforcement input into the program had adequate strength and sufficient space for the development of hooked bars. Furthermore, reasonable values for spacing were output for the required crack control reinforcement. However, the strut-to-node interfaces of four nodes, or portions of nodes, (Node C-Left, Node D-Right, Node J, and Node K) did not have adequate strength. The loaded area width was increased to 12.0 in. and the specified concrete compressive strength was increased to 5 ksi to sufficiently increase the strengths of these faces.

The reinforcement layout that satisfies design requirements as determined by the program is presented in Figure 5.19 and Figure 5.20. After reviewing all program outputs, a stirrup spacing of 8 in. is specified along the member. Horizontal crack control reinforcement is spaced at approximately 11 in.

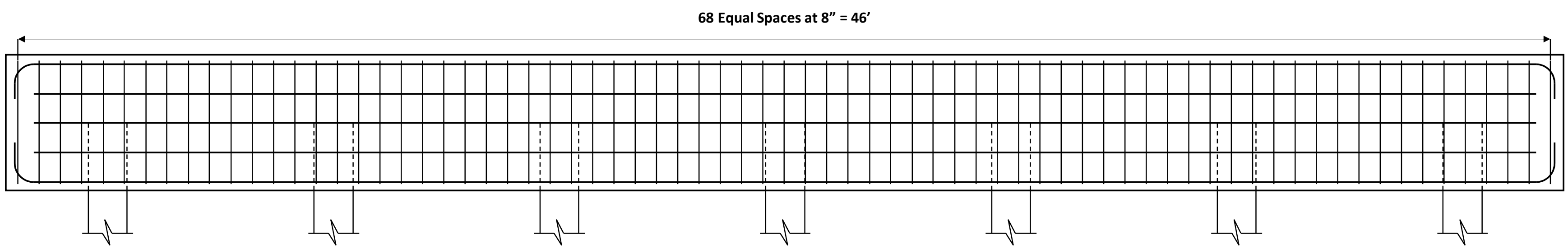


Figure 5.19: Elevation with Reinforcement Details based on Program Output (End Bent Cap)

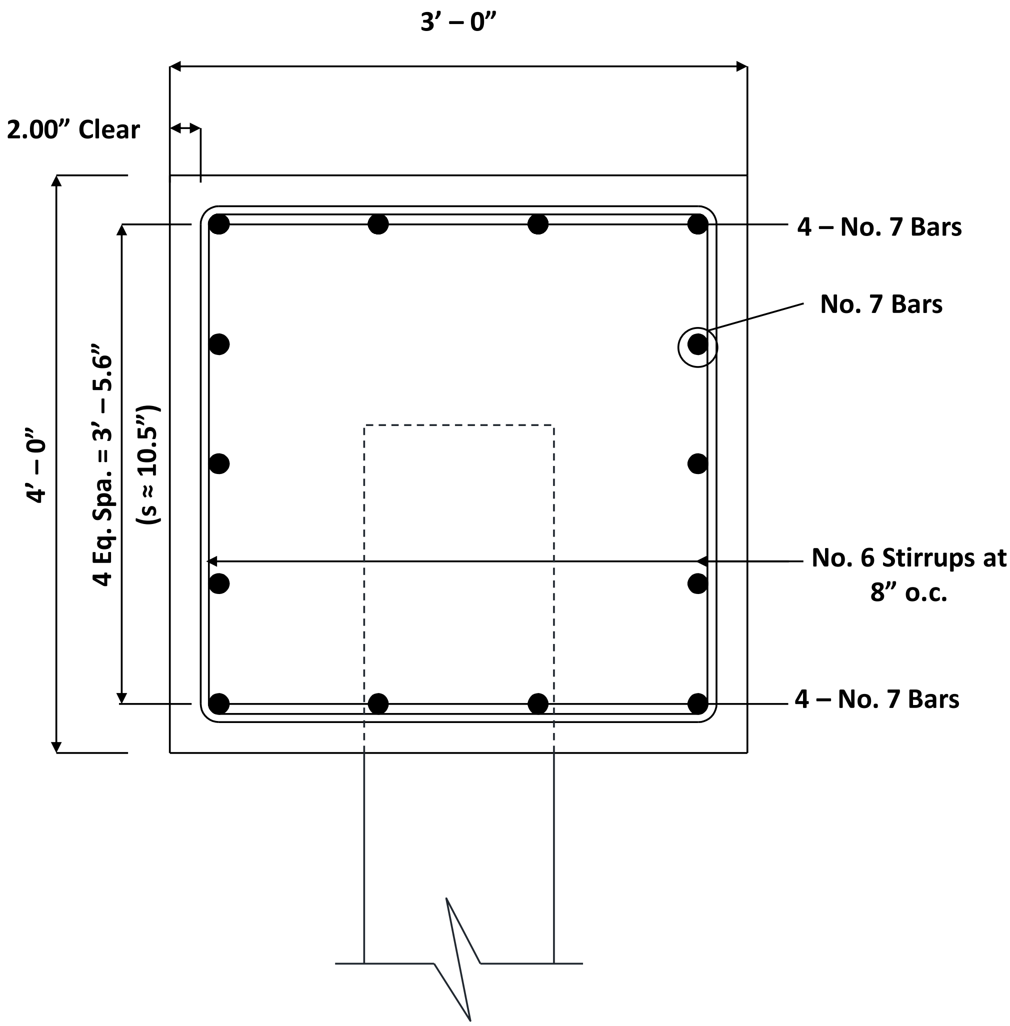


Figure 5.20: Section with Reinforcement Design based on Program Outputs (End Bent Cap)

5.5 Summary

The procedures used by the computer program to perform STM design steps for integral and semi-integral end bent caps was presented in this chapter. Specifically, differences between the procedures for end bent caps and pier caps were highlighted, and assumptions applied specifically to end bent caps were described. A design example of an integral end bent cap supported on seven H-piles and resisting loads from four bulb-tee girders was demonstrated. This example can be used as a reference for engineers designing other integral or semi-integral end bent caps with the assistance of the STM computer program.

CHAPTER 6. SUMMARY AND CONCLUDING REMARKS

6.1 Summary

The strut-and-tie method (STM) is a powerful design tool for D-regions of reinforced concrete structures. The STM is a lower-bound design theory that represents forces within a structural component as a truss model, known as a strut-and-tie model. This model includes struts (compression members), ties (tension members), and nodes where the members intersect. Due to the STM being a lower-bound theory, multiple correct strut-and-tie models can be correct for a given structural component. Many bridge substructure components consist of D-regions and require the use of the STM design.

Due to engineers often being unfamiliar with the STM design process, INDOT identified the need for a tool to assist with the STM design of typical bridge substructure components. STEP, a computer program created using Excel VBA, was developed with the objective of fulfilling this need. The program can be used to assist in the design of multi-column bent caps, straddle bent caps, and integral and semi-integral end bents. Two Excel workbooks were created for the program due to the different procedures required for designing a pier cap using the STM compared to designing an end bent cap. The first workbook, titled *Pier Cap STM Design*, is to be used for the design of multi-column bent caps and straddle bents caps. The second workbook, titled *End Bent Cap STM Design*, is to be used for the design of integral and semi-integral end bent caps. Despite the complexity of the STM procedure, rules can be used to develop a strut-and-tie model and perform required design steps. STEP uses a set of rules to develop a viable strut-and-tie model.

Background information on the STM was presented in Chapter 2 to familiarize or remind engineers of the design procedure. An introduction to Excel VBA and an overview of the computer program layout and procedures were then presented in Chapter 3. The procedure used by the computer program for pier caps was detailed in Chapter 4 along with an example of using STEP to assist in the design of a five-column bent cap. Chapter 5 included the STM design procedure used by the computer program for end bent caps, specifically focusing on procedures that differ from those presented in Chapter 4. An example of using STEP to design an integral end bent cap was then presented.

This thesis includes background on the STM and design examples using STEP. Williams et al. (2012) is another reference that can be used by engineers when designing using the STM. Williams et al. is a guidebook that includes a comprehensive introduction to the STM and five step-by-step design examples for bridge substructure components. In addition, Birrcher et al. (2009) and Larson et al. (2013) provide insight into the research that influenced the current STM provisions in AASHTO LRFD (2017).

6.2 Concluding Remarks

STEP is a tool developed to aid engineers with implementing the STM for the design of bridge substructure components. The engineer should be familiar with the limitations and the assumptions of the program prior to using it as a design tool. Engineers input geometric properties, material properties, reinforcement information, and a factored load case into STEP. The program then uses this information to develop a strut-and-tie model and perform STM design procedures. The results of the strength checks are output by the program for the engineer to review.

It is the responsibility of the engineer to ensure the accuracy of the results and interpret whether the design is safe and serviceable. The developers of STEP made every effort to create a program that will generate viable strut-and-tie models and accurately perform design procedures for most typical multi-column bent caps, straddle bent caps, and integral and semi-integral end bent caps. Nevertheless, some particular design situations may have been inadvertently overlooked, which may cause the program to output erroneous results. Ultimately, the engineer is responsible for verifying the viability and accuracy of the strut-and-tie model and validating all results output by the program. In addition, STEP only considers the STM specifications in Article 5.8.2 of AASHTO LRFD (2017). Other code requirements must be checked by the engineer.

This document should be referenced when using STEP to assist with the design of bridge substructure components. While STEP was designed with reference to typical bridge substructure components in Indiana, the program is versatile as long as the engineer is familiar with the limitations and assumptions of the program. It is the goal of this project that STEP helps to save time in the design office and increase understanding of the STM design process.

APPENDIX A. PIER CAP STM DESIGN PROCEDURE

Multi-Column Bent Caps

1. User Input

- ***a.*** The user inputs the following information into the program:
  - ***i.*** Overall member geometry.
  - ***ii.*** Material properties.
  - ***iii.*** Location of applied loads and support reactions.
  - ***iv.*** Dimensions of loaded areas and supports. Circular supports should be input as equivalent square supports.
  - ***v.*** Magnitude of loads.
  - ***vi.*** The bar size, number of bars, and location of each layer of reinforcement for the top and bottom longitudinal reinforcement. The cover at the ends of the bars is also input. The user is able to modify the exact layout of the top and bottom longitudinal reinforcement by performing multiple iterations of the program.
  - ***vii.*** The bar size of the stirrups and number of legs.
  - ***viii.*** The bar size of horizontal crack control reinforcement and number of bars through member width within spacing s_h.
  - ***ix.*** Development lengths for longitudinal reinforcement. The designer can input development lengths for straight bars and/or standard hooks.
  - ***x.*** Load factor for self-weight

2. Analyze Structural Component

- ***a.*** Determine centroid of longitudinal reinforcement (top and bottom chords, as appropriate). Refer to Item 1.a.vi (Limitation 2).

- ***b.*** Apply self-weight based on tributary volumes to each applied load along the top of the member (Limitation 6).

- ***c.*** Analyze member as a continuous beam to find support reactions using the moment distribution method (Limitation 5).

- ***d.*** Develop shear and bending moment diagrams for the member.

3. Locations

- ***a.*** Bottom Chord (Limitation 1):
  - ***i.*** The bottom chord is located at the centroid of the bottom longitudinal reinforcement (Limitations 2 and 3).
  - ***ii.*** The bottom chord extends from the two outermost bottom nodes.

- ***b.*** Top Chord (Limitation 1):
  - ***i.*** If a negative moment region exists, the top chord is located at the centroid of the top longitudinal reinforcement (Limitation 2).
  - ***ii.*** If only positive moment regions exist, the top chord is located to optimize the height of the strut-and-tie model.

1. Examine the location of each node to find the location of the most critical positive moment.

a. Determine if each node will be a CCC or CCT node. A CCT node will exist if the shear diagram does not change signs at the location of a load point. The exception of Item 3.f.i.1.a applies.

b. Divide the moment by the appropriate efficiency factor to determine how critical it is.

2. Use simple sectional analysis with the appropriate efficiency factor to determine the location of the top chord.

iii. The top chord extends from the two outermost top nodes.

c. Nodes – Horizontal Coordinates:

i. For each load or reaction, a node is given the corresponding horizontal coordinate.

ii. Additional nodes are located along the bottom chord directly under loads where the shear does not change signs. A similar situation can occur for the top chord if the shear does not change signs at a reaction. An additional node is not necessary where the exception of Item 3.f.i.1.a occurs.

iii. Additional nodes are also located where the horizontal distance between two adjacent nodes is greater than h_{STM}/tan(25°), where h_{STM} is the height of the strut-and-tie model. If the horizontal distance between two adjacent nodes is greater than h_{STM}/tan(25°) but less than or equal to 2[h_{STM}/tan(25°)], two nodes, one located along each chord, are given a horizontal coordinate half-way between the two original adjacent nodes. If the horizontal distance between two original adjacent nodes is greater than 2[h_{STM}/tan(25°)] but less than or equal to 3[h_{STM}/tan(25°)], four nodes, two located along each chord, are given horizontal coordinates defined by the third points between the two original adjacent nodes. If the horizontal distance between the two original adjacent nodes is even greater than 3[h_{STM}/tan(25°)], the same process applies.

d. Nodes – Vertical Coordinates:

i. The vertical coordinate of each node corresponds with the location of the top and bottom chords of the strut-and-tie model.

e. Horizontal Members:

i. Horizontal members connect the nodes along the top chord and nodes along the bottom chord. As previously stated, both chords extend from the two outermost nodes.

f. Vertical Members:

i. Vertical ties:

1. A vertical tie exists at a load or reaction where the shear diagram does not change signs. However, there is one exception to this rule:

a. If the vertical tie will fall within the length of the support or loaded area opposite the load or reaction at which the shear diagram does not change signs, the tie will not be created. Without the tie, nearby diagonal struts will extend to the node located at the support or loaded area opposite the load or reaction at which the shear diagram does not change signs.

2. Vertical ties also exist between each pair of nodes that is added to a model when two original nodes are separated by a horizontal distance greater than h_{STM}/tan(25°).

ii. Vertical struts:

1. A vertical strut only exists if a load is located directly above a support reaction (Limitation 14).

g. Diagonal Members (Diagonal Struts):

i. In negative shear regions, struts are drawn from the upper left to the lower right. In positive shear regions, struts are drawn from the upper right to the lower left.

ii. Using this convention, a diagonal strut exists between every load point and the nearest node along the bottom chord of the strut-and-tie model. An exception can occur for a load point that already has a vertical strut below it due to Item 3.f.ii.

4. Developing and Analyzing the Strut-and-Tie Model

a. Developing the Strut-and-Tie Model:

i. The strut-and-tie model is developed as outlined in Item 3 above.

ii. Nodes are labeled with letters based upon their location along the top and bottom chords. The top chord is labeled first from left to right, followed by the bottom chord from left to right. Struts and ties are named based on the nodes they "connect."

b. Analyzing the Strut-and-Tie Model:

i. Using the method of joints, vertical and horizontal equilibrium is satisfied at each node.

1. The number of members (struts and ties) entering a node is determined by the number of members with a coordinate matching the coordinate of the node.

2. For each node, use vertical and horizontal equilibrium to solve for the unknown strut and tie forces. Assume all members are in tension. Positive forces will indicate ties, and negative forces will indicate struts. Nodes with only two members and an applied load or support reaction are considered first followed by other nodes with only two unknown forces entering the node. Last, nodes with only one remaining unknown force are considered.

5. Proportioning Ties

a. Ties along Top and Bottom Chords:

i. Check each tie along both the top chord and bottom chord. Compare the factored tie strength to the appropriate tension force value to determine if the area of reinforcement is OK or NG. Indicate any ties that do not have a sufficient amount of reinforcement. Use the size of bars input by the user to calculate strength. Refer to Item 1.a.vi (Limitation 2; Assumption D).

b. Vertical Ties:

i. For each vertical tie, assume the width of the tie is equal to the smaller length of the two adjacent "truss panels" and is centered on the tie (Assumption C).

ii. Determine the spacing of stirrups needed to satisfy the tie force. Refer to Item 1.a.vii.

1. Use the bar size of stirrups and number of legs input by the user. If the required spacing is less than 3 in., a message appears to notify the user.

6. *Checking Nodal Strengths*

a. Nodes to Consider:

i. Consider all nodes except smeared nodes. For multi-column bent caps, smeared nodes are those that do not abut a loaded area or reaction area.

b. Modify Nodes:

i. If two or more struts enter the node from the same side (diagonal or horizontal struts), the struts are combined using vector addition to form one strut. Furthermore, if struts only enter a node from one side in addition to a vertical strut, the struts are combined to form one strut.

ii. Subdivide nodes:

1. At each load or support where the sign of the shear diagram changes, subdivide the node and the applied load or reaction.

2. Subdivide a node at a support into three parts if a load is located directly above the support and flows directly into it while forces also transfer to the support from diagonal struts on both the right and the left. Otherwise, subdivide the node into two parts. Although unlikely, the need to subdivide a node into three parts can also occur at an applied load (Limitation 14).

3. Position each portion of the load or reaction (and center of each part of the subdivided node) along the loaded area based on the percentage of shear force relative to the total load or reaction that exists on each side of the original load or reaction. Update the angles of diagonal struts based on the locations of each part of the subdivided node.

c. Proportion Nodes:

i. For the bearing face, the length is taken as the length of the loaded area or support if a node that has not been subdivided is located at that particular loaded area or support. Otherwise, the length of the face is taken as the portion of the loaded area or support associated with each part of a subdivided node. The width of the bearing face is taken as the width of the loaded area or support.

ii. For the back face, the length is taken as double the distance from the center of the node (i.e., location of the corresponding chord) to the top or bottom surface of the member. The width of the back face is taken as the width of the loaded area or support.

iii. For the strut-to-node interface, the length formula for strut-to-node interfaces is used. The width is taken as the width of the loaded area or support.

d. Calculate Nodal Strengths and Compare to Factored Demands:

i. Determine if the node is a CCC, CCT, or CTT node by considering the number of members entering the node with tension (positive) force values. If only two members are ties and both are along the horizontal chord of the member, the node is a CCT node.

ii. Calculate the confinement modification factor, *m*, if the width of the member is larger than the width of the loaded area or support.

iii. Calculate strength of nodal faces:

1. Check all bearing faces and strut-to-node interfaces. Check back faces when a direct compressive force acts on the face (Limitation 15; Assumption A).

2. For each face, select the appropriate efficiency factor, ν, based on the type of node and the nodal face under consideration. The chosen efficiency factors assume that horizontal and vertical crack control reinforcement is provided in accordance with Article 5.8.2.6 of AASHTO LRFD (2017) (Assumption B).
3. For each face, calculate the factored nodal strength using the equations in Article 5.8.2.5 of AASHTO LRFD (2017).

iv. For each face, compare the factored nodal strength to the appropriate compressive force value to determine if the strength of the nodal face is OK or NG.

7. Proportion Crack Control Reinforcement

a. The required crack control reinforcement is the same along the length of the member since the web width b_w is constant (Limitation 10).

b. Design Vertical and Horizontal Crack Control Reinforcement:

i. Use the bar size of stirrups input by the user to determine the spacing required to satisfy Eq. 5.8.2.6-1 of AASHTO LRFD (2017). Refer to Item 1.a.vii.

ii. Use the bar size of horizontal crack control reinforcement input by the user to determine the required spacing to satisfy Eq. 5.8.2.6-2 of AASHTO LRFD (2017). Refer to Item 1.a.viii.

iii. For both stirrups and horizontal reinforcement used for crack control, any spacing less than 3 in. is considered to be too close. Moreover, ensure the spacing satisfies the maximum spacing limit given in Article 5.8.2.6 of AASHTO LRFD (2017) (d/4 or 12.0 in.). The value of d is taken as the minimum d considering both positive and negative moment regions.

c. Within vertical tie widths, determine if the spacing requirements for the vertical crack control reinforcement or the tie reinforcement of Item 5.b.ii governs. Give the required stirrup spacing along the length of the bent cap.

8. Provide Necessary Anchorage for Ties

a. Consider the outermost ends of the outermost ties at both ends of the top and bottom chords of the strut-and-tie model (Limitations 2 and 12).

b. Calculate Available Length:

i. Calculate the available length for development for the outermost ends of the outermost ties at both ends of the top and bottom chords in accordance with Article C5.8.2.4.2 of AASHTO LRFD (2017). Leave sufficient space for required cover at the end of the bars. Refer to Item 1.a.vi.

ii. This methodology of Article C5.8.2.4.2 of AASHTO LRFD (2017) assumes that a diagonal strut enters the node from the side of the node opposite from where the bars are being anchored. If a different case occurs, the inside face of the loaded area or support is considered to be the critical section for bar development in this rare scenario.

c. For each outermost tie, compare the calculated available length to the required development length input by the user. State if sufficient space is available for straight bar anchorage and/or for hooked anchorage. Refer to Item 1.a.ix.

Limitations:

1. The top and bottom chords of the strut-and-tie model are located at constant depths along the length of the member. In other words, the location of the top and bottom chords does not change along the length of the bent cap. If there is a negative moment region, the top chord will be located at the centroid of the top longitudinal reinforcement. Otherwise, if there is no negative moment, the location of the top chord will be based on an optimized design.
2. The amount of top and bottom longitudinal reinforcement is constant along the length of the member.
3. A positive moment region must exist at some location along the length of the member.
4. All loads are applied downward (in the direction of gravity), and all support reactions act upward.
5. All supports are assumed to provide only vertical restraint.
6. Self-weight is distributed by the program to locations where external loads are applied to the member. If further refinement is desired, the user should apply self-weight as several external applied loads along the length of the member. In this case, the self-weight load factor should be input as zero.
7. The program is limited to a strut-and-tie model with 104 nodes or less.
8. If a node is subdivided into two or three parts due to the subdivision of an applied load or support reaction, the classification of each part of the node as a CCC, CCT, or CTT node is determined based on the classification of the original node before subdividing.
9. The program is designed to satisfy the strut-and-tie method design provisions in Article 5.8.2 of AASHTO LRFD (2017). Other code provisions may not be satisfied by the design considered in the program.
10. Only prismatic members are considered.
11. The program assumes that the entire bent cap is defined by D-regions. If any B-regions exist, they will also be modeled with a strut-and-tie model. The designer is responsible for performing appropriate design procedures based on a sectional model for any B-regions along the length of the member (see Article 5.5.1.2.2 of AASHTO LRFD (2017)).
12. Anchorage checks will only be performed near the ends of members (i.e., for the outermost longitudinal ties). If the designer terminates bars at intermediate locations along the length of the member, the designer must check anchorage at those locations.
13. The program performs a bearing check for nodes based on the bearing area input by the user. If girder loads are combined together before being input into the program, the designer must check the bearing strength for each girder individually.
14. For cases in which a node is subdivided into three parts, the angle of the vertical strut will likely change slightly. The program neglects this change of angle and considers the strut to be vertical. If the diagonal struts at the node have vastly different magnitudes, the designer may need to consider a revised angle for the vertical strut at that node.
15. For the back face of a CCC node, the contribution of any nonprestressed compressive reinforcement is not considered. If consideration of the contribution of compressive reinforcement is desired, the designer should include it in his/her own calculations in accordance with Article 5.8.2.5.1 of AASHTO LRFD (2017). The designer must also check that the reinforcement is fully developed in compression.

16. A diagonal strut that changes orientation due to the subdivision of a node cannot be handled by the program. For example, if a diagonal strut is originally oriented from the upper left to the lower right and subdividing a node will cause it to be oriented from the upper right to the lower left, the program will not run.
17. Consideration of built-up bearing seats is not included in the program.

Note 1: As stated in Limitation 1, if there is any negative moment region, the top chord will be located at the centroid of the top reinforcement. If a back face of a node along the top chord does not have the required strength, the horizontal strut at this location cannot simply be shifted to provide a larger back face. Instead, the depth of the member will need to be increased. The designer can also shift the location of the top chord reinforcement defining the length of the back face. If a more refined strut-and-tie model is desired that incorporates a top chord that does not have a constant location (i.e., y-coordinate) along the length of the member, it must be developed outside of the program.

Note 2: If any of the strength checks are not satisfied, the user must update the details of the bent cap to find a design that is valid. The program will not automatically output a design that satisfies STM code requirements.

Assumptions:

A. The strength of a back face is only checked if compression acts on the face. Article 5.8.2.5.3b of AASHTO LRFD states that "[b]ond stresses resulting from the force in a developed tie…need not be applied to the back face of the CCT node." However, this phenomenon may also occur at CTT nodes. The program considers this observation, and the strength of a back face is only checked when it is subjected to direct compressive stresses.
B. Because bent caps do not fall under the category of slabs or footings, horizontal and vertical crack control reinforcement in accordance with Article 5.8.2.6 of AASHTO LRFD (2017) should be provided. Therefore, the efficiency factors in Table 5.8.2.5.3a-1 of AASHTO LRFD (2017) are used for all nodal strength calculations.
C. The width of each vertical tie is assumed to equal the smaller length of the two adjacent "truss panels" of the strut-and-tie model and is centered on the tie.
D. Longitudinal reinforcement distributed through the width of the pier cap is considered effective no matter the width of the member relative to the width of loaded areas or supports. Additional research is needed to determine the limits for distributing longitudinal reinforcement through the member width.

APPENDIX B. END BENT CAP STM DESIGN PROCEDURE

Integral and Semi-Integral End Bent Caps

1. User Input

- ***a.*** The user inputs the following information into the program:
  - ***i.*** Overall member geometry, and the effective depth of the cap. The effective depth is the portion of the cap considered effective in resisting loads (Assumption E).
  - ***ii.*** Material properties.
  - ***iii.*** Location of applied loads.
  - ***iv.*** Location of piles.
  - ***v.*** Dimensions of loaded areas.
  - ***vi.*** Effective dimensions of piles.
  - ***vii.*** Magnitude of loads.
  - ***viii.*** Pile reactions if the designer desires to use reaction values from a different analysis method (e.g., pile group analysis) instead of from a continuous beam analysis.
  - ***ix.*** The bar size, number of bars, and location of each layer of reinforcement for the top and bottom longitudinal reinforcement. The cover at the ends of the bars is also input. The longitudinal reinforcement is located within the effective depth of the cap. The user is able to modify the exact layout of the top and bottom longitudinal reinforcement by performing multiple iterations of the program.
  - ***x.*** The bar size of the stirrups and number of legs.
  - ***xi.*** The bar size of horizontal crack reinforcement and number of bars through member width within spacing s_h.
  - ***xii.*** Development lengths for longitudinal reinforcement. The designer can input development lengths for straight bars and/or standard hooks.
  - ***xiii.*** Load factor for self-weight.

2. Analyze Structural Component

- ***a.*** Determine centroid of longitudinal reinforcement (top and bottom chords, as appropriate). Refer to Item 1.a.ix (Limitation 2; Assumption E).

- ***b.*** Apply self-weight based on tributary volumes calculated using the full height of the member to each applied load along the top of the member (Limitation 6).

- ***c.*** If the user does not input pile reactions, analyze member as a continuous beam using the moment distribution method (Limitation 5). If pile reactions are input, check to ensure the member is in equilibrium.
  - ***i.*** The user is given the option of inputting pile reactions calculated using pile group analysis or other analysis methods. However, the program defaults to solving for pile reactions using a continuous beam analysis, unless reactions are provided.

- ***d.*** Develop shear and bending moment diagrams for the member.

3. Locations

- ***a.*** Bottom Chord (Limitation 1):
  - ***i.*** The bottom chord is located at the centroid of the bottom longitudinal reinforcement (Limitations 2 and 3).
  - ***ii.*** The bottom chord extends from the two outermost bottom nodes.

- ***b.*** Top Chord (Limitation 1):
  - ***i.*** If a negative moment region exists, the top chord is located at the centroid of the top longitudinal reinforcement (Limitation 2).
  - ***ii.*** If only positive moment regions exist, the top chord is located to optimize the height of the STM.
    - ***1.*** Examine the location of each node to find the location of the most critical positive moment.
      - ***a.*** Determine if each node will be a CCC or CCT node. A CCT node will exist if the shear diagram does not change signs at the location of a load point. The exception of Item 3.f.i.1.a applies.
      - ***b.*** Divide the moment by the appropriate efficiency factor to determine how critical it is.
    - ***2.*** Use simple sectional analysis with the appropriate efficiency factor to determine the location of the top chord.
  - ***iii.*** The top chord extends from the two outermost top nodes.

- ***c.*** Nodes – Horizontal Coordinates:
  - ***i.*** For each load or reaction, a node is given the corresponding horizontal coordinate.
  - ***ii.*** Additional nodes are located along the bottom chord directly under loads where the shear does not change signs. A similar situation can occur for the top chord if the shear does not change signs at a reaction. An additional node is not necessary where the exception of Item 3.f.i.1.a occurs.
  - ***iii.*** Additional nodes are also located where the horizontal distance between two adjacent nodes is greater than h_{STM}/tan(25°), where h_{STM} is the height of the strut-and-tie model. If the horizontal distance between two adjacent nodes is greater than h_{STM}/tan(25°) but less than or equal to 2[h_{STM}/tan(25°)], two nodes, one located along each chord, are given a horizontal coordinate half-way between the two original adjacent nodes. If the horizontal distance between two original adjacent nodes is greater than 2[h_{STM}/tan(25°)] but less than or equal to 3[h_{STM}/tan(25°)], four nodes, two located along each chord, are given horizontal coordinates defined by the third points between the two original adjacent nodes. If the horizontal distance between the two original adjacent nodes is even greater than 3[h_{STM}/tan(25°)], the same process applies.

- ***d.*** Nodes – Vertical Coordinates:
  - ***i.*** The vertical coordinate of each node corresponds with the location of the top and bottom chords of the STM.

- ***e.*** Horizontal Members:
  - ***i.*** Horizontal members connect the nodes along the top chord and nodes along the bottom chord. As previously stated, both chords extend from the two outermost nodes.

f. Vertical Members:

- ***i.*** Vertical ties:
  - ***1.*** A vertical tie exists at a load or reaction where the shear diagram does not change signs. However, there is one exception to this rule:
    - ***a.*** If the vertical tie will fall within the length of the effective pile dimensions or loaded area opposite the load or reaction at which the shear diagram does not change signs, the tie will not be created. Without the tie, nearby diagonal struts will extend to the node located at the supporting pile or loaded area opposite the load or reaction at which the shear diagram does not change signs.
  - ***2.*** Vertical ties also exist between each pair of nodes that is added to a model when two original nodes are separated by a horizontal distance greater than $h_{STM}/\tan(25°)$.
- ***ii.*** Vertical struts:
  - ***1.*** A vertical strut only exists if a load is located directly above a support reaction (Limitation 14).

g. Diagonal Members (Diagonal Struts):

- ***i.*** In negative shear regions, struts are drawn from the upper left to the lower right. In positive shear regions, struts are drawn from the upper right to the lower left.
- ***ii.*** Using this convention, a diagonal strut exists between every load point and the nearest node along the bottom chord of the strut-and-tie model. An exception can occur for a load point that already has a vertical strut below it due to Item 3.f.ii.

4. Developing and Analyzing the Strut-and-Tie Model

a. Developing the Strut-and-Tie Model:

- ***i.*** The strut-and-tie model is developed as outlined in Item 3 above.
- ***ii.*** Nodes are labeled with letters based upon their location along the top and bottom chords. The top chord is labeled first from left to right, followed by the bottom chord from left to right. Struts and ties are named based on the nodes they “connect.”

b. Analyzing the Strut-and-Tie Model:

- ***i.*** Using the method of joints, vertical and horizontal equilibrium is satisfied at each node.
  - ***1.*** The number of members (struts and ties) entering a node is determined by the number of members with a coordinate matching the coordinate of the node.
  - ***2.*** For each node, use vertical and horizontal equilibrium to solve for the unknown strut and tie forces. Assume all members are in tension. Positive forces will indicate ties, and negative forces will indicate struts. Nodes with only two members and an applied load or support reaction are considered first followed by other nodes with only two unknown forces entering the node. Last, nodes with only one remaining unknown force are considered.

5. Proportioning Ties

a. Ties along Top and Bottom Chords:

- ***i.*** Check each tie along both the top chord and bottom chord. Compare the factored tie strength to the appropriate tension force value to determine if the area of reinforcement is OK or NG. Indicate any ties that do not have a sufficient amount of reinforcement.

Use the size of bars input by the user to calculate strength. Refer to Item 1.a.ix (Limitation 2; Assumption D).

b. Vertical Ties:

- ***i.*** For each vertical tie, assume the width of the tie is equal to the smaller length of the two adjacent "truss panels" and is centered on the tie (Assumption C).
- ***ii.*** Determine the spacing of stirrups needed to satisfy the tie force. Refer to Item 1.a.x.
  - ***1.*** Use the bar size of stirrups and number of legs input by the user. If the required spacing is less than 3 in., a message appears to notify the user.

6. Checking Nodal Strengths

a. Nodes to Consider:

- ***i.*** Consider all nodes except smeared nodes. For integral and semi-integral end bent caps, smeared nodes are those that do not abut a loaded area or are not located at a pile.

b. Modify Nodes:

- ***i.*** If two or more struts enter the node from the same side (diagonal or horizontal struts), the struts should be combined using vector addition to form one strut. Furthermore, if struts only enter a node from one side in addition to a vertical strut, the struts should be combined to form one strut.
- ***ii.*** Subdivide nodes:
  - ***1.*** At each load or support where the sign of the shear diagram changes, subdivide the node and the applied load or reaction.
  - ***2.*** Subdivide a node at a support into three parts if a load is located directly above the support and flows directly into it while forces also transfer to the support from diagonal struts on both the right and the left. Otherwise, subdivide the node into two parts. The need to subdivide a node into three parts can also occur at an applied load (Limitation 14).
  - ***3.*** Position each portion of the load or reaction (and center of each part of the subdivided node) along the loaded area based on the percentage of shear force relative to the total load or reaction that exists on each side of the original load or reaction. Update the angles of diagonal struts based on the locations of each part of the subdivided node.

c. Proportion Nodes:

- ***i.*** Nodes along top chord:
  - ***1.*** For the bearing face, the length is taken as the length of the loaded area if a node that has not been subdivided is located at that particular loaded area. Otherwise, the length of the face is taken as the portion of the loaded area associated with each part of a subdivided node. The width of the bearing face is taken as the width of the loaded area.
  - ***2.*** For the back face, the length is taken as double the distance from the center of the node (i.e., location of the top chord) to the top surface of the member. The width of the back face is taken as the width of the loaded area.
  - ***3.*** For the strut-to-node interface, the length formula for strut-to-node interfaces is used. The width is taken as the width of the loaded area.

ii. Nodes along bottom chord:

1. Bearing stresses are not checked for piles (Assumption F).

2. For the back face, the length is taken as double the distance from the center of the node (i.e., location of the bottom chord) to the bottom of the effective member depth. The width of the back face is taken as the width of the effective pile dimensions.

3. For the strut-to-node interface, the length formula for strut-to-node interfaces is used. The width is taken as the width of the effective pile dimensions.

d. Calculate Nodal Strengths and Compare to Factored Demands:

i. Determine if the node is a CCC, CCT, or CTT node by considering the number of members entering the node with tension (positive) force values. If only two members are ties and both are along the horizontal chord of the member, the node is a CCT node.

ii. Calculate the confinement modification factor, *m*, if the width of the member is larger than the width of the loaded area. Confinement modification factors are not applied to nodes associated with piles (Assumption I).

iii. Calculate strength of nodal faces:

1. Check all bearing faces associated with applied loads and all strut-to-node interfaces. Check back faces when a direct compressive force acts on the face (Limitation 15; Assumptions A and F).

2. For each face, select the appropriate efficiency factor, ν, based on the type of node and the nodal face under consideration. The chosen efficiency factors assume that horizontal and vertical crack control reinforcement is provided in accordance with Article 5.8.2.6 of AASHTO LRFD (2017) (Assumption B).

3. For each face, calculate the factored nodal strength using the equations in Article 5.8.2.5 of AASHTO LRFD (2017).

iv. For each face, compare the factored nodal strength to the appropriate compressive force value to determine if the strength of the nodal face is OK or NG.

7. Proportion Crack Control Reinforcement

a. The required crack control reinforcement is the same along the length of the member since the web width b_w is constant (Limitation 10).

b. Design Vertical and Horizontal Crack Control Reinforcement:

i. Use the bar size of stirrups input by the user to determine the required spacing to satisfy Eq. 5.8.2.6-1 of AASHTO LRFD (2017). Refer to Item 1.a.x.

ii. Use the bar size of horizontal crack control reinforcement input by the user to determine the required spacing to satisfy Eq. 5.8.2.6-2 of AASHTO LRFD (2017). Refer to Item 1.a.xi.

iii. For both stirrups and horizontal reinforcement used for crack control, any spacing less than 3 in. is considered to be too close. Moreover, ensure the spacing satisfies the maximum spacing limit given in Article 5.8.2.6 of AASHTO LRFD (2017) (d/4 or 12.0 in.). The value of d is taken as the minimum d considering both positive and negative moment regions. Note that the vertical reinforcement is also limited by 12 in. by INDOT suggested integral end bent details (INDOT 2018).

c. Within vertical tie widths, determine if the spacing requirements for the vertical crack control reinforcement or the tie reinforcement of Item 5.b.ii governs. Give the required stirrup spacing along the length of the bent cap.

8. Provide Necessary Anchorage for Ties

a. Consider the outermost ends of the outermost ties at both ends of the top and bottom chords of the strut-and-tie model (Limitations 2 and 12).

b. Calculate Available Length:

i. Calculate the available length for development for the outermost ends of the outmost ties at both ends of the top and bottom chords. Because of the uncertainty of the geometries of the diagonal struts entering the nodes located at pile supports along the bottom of the cap, consider the inside face of the pile as the critical section for bar development at the bottom chord (Assumption J). For the top chord, calculate the available development length in accordance with Article C5.8.2.4.2 of AASHTO LRFD (2017). Leave sufficient space for required cover at the end of the bars. If multiple layers of hooks are used, consider the length of the bars with the inner hooks. Refer to Item 1.a.ix.

ii. For the top chord, the methodology of Article C5.8.2.4.2 of AASHTO LRFD (2017) assumes that a diagonal strut enters the node from the side of the node opposite from where the bars are being anchored. If a different case occurs, the inside face of the loaded area is considered to be the critical section for bar development in this rare scenario.

c. For each outermost tie, compare the calculated available length to the required development length input by the user. State if sufficient space is available for straight bar anchorage and/or for hooked anchorage. Refer to Item 1.a.xii.

Limitations:

1. The top and bottom chords of the strut-and-tie model are located at constant depths along the length of the member. In other words, the location of the top and bottom chords does not change along the length of the bent cap. If there is a negative moment region, the top chord will be located at the centroid of the top longitudinal reinforcement. Otherwise, if there is no negative moment, the location of the top chord will be based on an optimized design.
2. The amount of top and bottom longitudinal reinforcement is constant along the length of the member.
3. A positive moment region must exist at some location along the length of the member.
4. All loads are applied downward (in the direction of gravity), and all support reactions act upward.
5. All supports are assumed to provide only vertical restraint.
6. Self-weight is distributed by the program to locations where external loads are applied to the member. If further refinement is desired, the user should apply self-weight as several external applied loads along the length of the member. In this case, the self-weight load factor should be input as zero.
7. The program is limited to a strut-and-tie model with 104 nodes or less.

8. If a node is subdivided into two or three parts due to the subdivision of an applied load or support reaction, the classification of each part of the node as a CCC, CCT, or CTT node is determined based on the classification of the original node before subdividing.
9. The program is designed to satisfy the strut-and-tie method design provisions in Article 5.8.2 of AASHTO LRFD (2017). Other code provisions may not be satisfied by the design considered in the program.
10. Only prismatic members are considered.
11. The program assumes that the entire bent cap is defined by D-regions. If any B-regions exist, they will also be modeled with a strut-and-tie model. The designer is responsible for performing appropriate design procedures based on a sectional model for any B-regions along the length of the member (see Article 5.5.1.2.2 of AASHTO LRFD (2017)).
12. Anchorage checks will only be performed near the ends of members (i.e., for the outermost longitudinal ties). If the designer terminates bars at intermediate locations along the length of the member, the designer must check anchorage at those locations.
13. The program performs a bearing check for nodes based on the bearing area input by the user. If girder loads are combined together before being input into the program, the designer must check the bearing strength for each girder individually.
14. For cases in which a node is subdivided into three parts, the angle of the vertical strut will likely change slightly. The program neglects this change of angle and considers the strut to be vertical. If the diagonal struts at the node have vastly different magnitudes, the designer may need to consider a revised angle for the vertical strut at that node.
15. For the back face of a CCC node, the contribution of any nonprestressed compressive reinforcement is not considered. If consideration of the contribution of compressive reinforcement is desired, the designer should include it in his/her own calculations in accordance with Article 5.8.2.5.1 of AASHTO LRFD (2017). The designer must also check that the reinforcement is fully developed in compression.
16. A diagonal strut that changes orientation due to the subdivision of a node cannot be handled by the program. For example, if a diagonal strut is originally oriented from the upper left to the lower right and subdividing a node will cause it to be oriented from the upper right to the lower left, the program will not run.
17. Consideration of built-up bearing seats is not included in the program.

Note 1: As stated in Limitation 1, if there is any negative moment region, the top chord will be located at the centroid of the top reinforcement. If a back face of a node along the top chord does not have the required strength, the horizontal strut at this location cannot simply be shifted to provide a larger back face. Instead, the depth of the member will need to be increased. The designer can also shift the location of the top chord reinforcement defining the length of the back face. If a more refined strut-and-tie model is desired that incorporates a top chord that does not have a constant location (i.e., y-coordinate) along the length of the member, it must be developed outside of the program.

Note 2: If any of the strength checks are not satisfied, the user must update the details of the bent cap to find a design that is valid. The program will not automatically output a design that satisfies STM code requirements.

Assumptions:

A. The strength of a back face is only checked if compressive stresses acts on the face. Article 5.8.2.5.3b of AASHTO LRFD (2017) states that "[b]ond stresses resulting from the force in a developed tie…need not be applied to the back face of the CCT node." However, this phenomenon may also occur at CTT nodes. The program considers this observation, and the strength of a back face is only checked when it is subjected to direct compressive stresses.
B. Because bent caps do not fall under the category of slabs or footings, horizontal and vertical crack control reinforcement in accordance with Article 5.8.2.6 of AASHTO LRFD (2017) is assumed to be provided. Therefore, the efficiency factors in Table 5.8.2.5.3a-1 of AASHTO LRFD (2017) are used for all nodal strength calculations.
C. The width of each vertical tie is assumed to equal the smaller length of the two adjacent "truss panels" of the strut-and-tie model and is centered on the tie.
D. Longitudinal reinforcement distributed through the width of the end bent cap is considered effective no matter the width of the member relative to the width of piles or loaded areas. Additional research is needed to determine the limits for distributing longitudinal reinforcement through the member width.
E. The depth at which the resultant of the pile reaction is introduced to the member is assumed to be at the location of the bottom chord of the strut-and-tie model.
F. The strength of the member to resist bearing stresses at piles is not checked because force is transferred by both friction along the portion of the pile embedded in concrete and end bearing. The designer must perform any checks of pile bearing capacity outside of the program.
G. Despite the uncertainty of the geometries of the struts entering the nodes located at pile supports along the bottom of the cap, the geometry of the back faces and strut-to-node interfaces of nodes located at piles is defined in the same manner as for typical nodes.
H. Any contribution of the diaphragm at end bents to the strength of end bent caps is neglected.
I. The confinement modification factor, m, is not applied to nodes associated with piles.
J. Because of the uncertainty of the geometries of the diagonal struts entering the nodes located at pile supports along the bottom of the cap, the critical section for bar development along the bottom chord is assumed to be the inside face of the pile (i.e., the location at which the centroid of the bars exits the extended nodal zone is not considered).

Note 3: Integral and semi-integral end bent caps present unique challenges for the implementation of the strut-and-tie method. Although the members, or portions of the members, are often classified as D-regions, little documentation of the use of the strut-and-tie method for their design is available. Due to the manner in which stresses are transferred from the embedded piles to the member, simplifying assumptions are necessary to develop a viable strut-and-tie model. The developers of STEP incorporated such simplifying assumptions into the program. The designer should understand these assumptions prior to using STEP. The designer is encouraged to consider other assumptions outside of the program if he/she believes that the assumptions incorporated into the program are not valid for the member being designed. Furthermore, for integral end bent caps, the diaphragm may contribute to the overall strength of the member. Due to the uncertainty of its contribution, however, the diaphragm has been neglected.

Because of the simplifying assumptions necessary to develop a strut-and-tie model for end bent caps, it is recommended that traditional analysis and design methods be performed alongside the STM design for these substructure components. Design checks associated with traditional methods should be satisfied along with standard detailing practices for end bents.

APPENDIX C. CTR REPORT PERMISSION

FW: CTR Report Related to Current Project at Purdue - Question
Williams, Christopher S
Thu 5/23/2019 8:17 PM
Andi,
We have proper permission to use the information from the Texas report in your thesis. See below.
Best,
Chris Williams
-----Original Message-----
From: Barnes, Kevyn A
Sent: Thursday, May 23, 2019 2:46 PM
To: Williams, Christopher S
Subject: RE: CTR Report Related to Current Project at Purdue - Question
Hi Chris,
Thank you for following up and being diligent. Yes, the permissions do extend to the thesis and future
reports, as long as the publication is cited as the source.
Best wishes,
Kevyn
KEVYN BARNES
she/her/hers
2019 Chair, Transportation Division | Special Libraries Association Manager, Library Services | TxDOT
-----Original Message-----
From: Williams, Christopher S
Sent: Thursday, May 23, 2019 1:23 PM
To: Barnes, Kevyn A
Subject: RE: CTR Report Related to Current Project at Purdue - Question
Hello Kevyn,
I wrote to you last month concerning permission to use information from my past CTR report for a
report being written for research sponsored by the Indiana DOT. Just to be sure that I have proper
permission, material from the CTR report will also be used in a student's master's thesis. Please let me
know if there is any problem with also using the information in the thesis with proper attribution to the
CTR report.
Thanks,
Chris

Christopher S. Williams, Ph.D.
Assistant Professor
Purdue University

-----Original Message-----
From: Williams, Christopher S
Sent: Wednesday, April 17, 2019 9:16 AM
To: 'Barnes, Kevyn A'
Subject: RE: CTR Report Related to Current Project at Purdue - Question
Hello Kevyn,
Thank you for the prompt reply. I appreciate you giving permission to use the content in the CTR report.
As with the first report, I hope this new report will be of interest to other state DOTs, including TxDOT.
Best,
Chris

Christopher S. Williams, Ph.D.
Assistant Professor
Purdue University
-----Original Message-----
From: Barnes, Kevyn A
Sent: Tuesday, April 16, 2019 11:32 PM
To: Williams, Christopher S
Subject: Re: CTR Report Related to Current Project at Purdue - Question
Hi Chris,
Thank you for contacting the library. I am glad to hear you will be building on this work; there was high
interest from other state DOTs when it was published.
This email serves as permission to use any content that is original to the work (FHWA/TX-12/5-5253-01-
1, Strut-and-Tie Model Design Examples for Bridges: Final Report) with proper attribution. Permissions
for any content in the report that is cited from other sources should be sought from that separate copyright holder.
Please let me know if you have any questions.
All the Best,
Kevyn

(Ms.) Kevyn Barnes
Manager, Library Services
The University of Texas at Austin | Center for Transportation Research | 512.232.3130 |
____
From: Williams, Christopher S
Sent: Tuesday, April 16, 2019 2:21:09 PM
To: Barnes, Kevyn A
Subject: CTR Report Related to Current Project at Purdue - Question
Hello Kevyn,
I am a faculty member at Purdue University. If the questions I pose below should be directed toward
someone else, could you please let me know who I should ask?
I received my MS and PhD degrees at UT Austin while working on TxDOT projects that resulted in CTR
research reports. I was the first author of FHWA/TX-12/5-5253-01-1 Strut-and-Tie Model Design

Examples for Bridges: Final Report. I am currently working on a related research project for the Indiana
DOT. For the report that will result from this current project, I hope to utilize content in the CTR report.
This will include the use of figures and some explanations from the CTR report. I should note that I was
the creator of the figures in the report. Proper citations will be given to the CTR report throughout the
document that will result from my ongoing research project. The ongoing project is not repeating the
work of the CTR report but is building upon it.
To be more specific, the CTR report contains several engineering design examples and explains to
engineers how to carry out the designs using the strut-and-tie method. The objective of the current
project is to develop a computer program that will perform designs using the strut-and-tie method. In
order to explain the background of the computer program and show how it works, it would be very
helpful to demonstrate its capabilities by referencing a design example that already exists in the CTR
report. In fact, I expect that the new report will cause many readers to seek out the CTR report.
Can I have proper permission to reference the CTR report as mentioned, including the use of figures
from the report, if proper citation is given throughout? I believe the result will be advantageous to all.
Thanks,
Chris

Christopher S. Williams, Ph.D.
Assistant Professor
Purdue University

REFERENCES

AASHTO LRFD, *Bridge Design Specifications*, 1994, First Edition, American Association of State Highway and Transportation Officials, Washington, DC, 1994.

AASHTO LRFD, *Bridge Design Specifications*, 2017, Eighth Edition, American Association of State Highway and Transportation Officials, Washington, DC, September 2017.

ACI Committee 318, "Building Code Requirements for Structural Concrete (ACI 318-02) and Commentary," American Concrete Institute, Farmington Hills, MI, 2002.

ACI Committee 318, "Building Code Requirements for Structural Concrete (ACI 318-14) and Commentary," American Concrete Institute, Farmington Hills, MI, 2014.

AASHTO LRFD, *Bridge Design Specifications*, 2014, Seventh Edition with 2016 Interim Revisions, American Association of State Highway and Transportation Officials, Washington, DC, 2014.

Birrcher, D., Tuchscherer, R., Huizinga, M., Bayrak, O., Wood, S., and Jirsa, J., *Strength and Serviceability Design of Reinforced Concrete Deep Beams*, Rep. No. FHWA/TX-09/0-5253-1, Center for Transportation Research, The University of Texas at Austin, Austin, TX, 2009.

INDOT, "Indiana Design Manual 2013," Indiana Department of Transportation, Indianapolis, IN, Nov. 2018.

Larsen, Ronald, W., "Engineering with Excel," 2013, Fourth Edition, Pearson Education, Inc., 2013.

Larson, N., Gómez, E. F., Garber, D., Bayrak, O., and Ghannoum, W., Strength and Serviceability Design of Reinforced Concrete Inverted-T Beams, Rep. No. FHWA/TX-13/0-6416-1, Center for Transportation Research, The University of Texas at Austin, Austin, TX, 2013.

Schlaich, J., Schäfer, K., and Jennewein, M., "Toward a Consistent Design of Structural Concrete." *PCI Journal*, Vol. 32, No. 3, May-June 1987, pp. 74-150.

Williams, C. S., Deschenes, D. J., and Bayrak, O., *Strut-and-Tie Model Design Examples for Bridges*, Rep. No. FHWA/TX-12/5-5253-01-1, Center for Transportation Research, The University of Texas at Austin, Austin, TX, 2012.