

# **NCHRP**

## **REPORT 485**

**NATIONAL  
COOPERATIVE  
HIGHWAY  
RESEARCH  
PROGRAM**

### **Bridge Software—Validation Guidelines and Examples**

**TRANSPORTATION RESEARCH BOARD**

**NATIONAL RESEARCH COUNCIL**

## **TRANSPORTATION RESEARCH BOARD EXECUTIVE COMMITTEE 2003 (Membership as of January 2003)**

### **OFFICERS**

**Chair:** Genevieve Giuliano, Director and Professor, School of Policy, Planning, and Development, University of Southern California, Los Angeles

**Vice Chair:** Michael S. Townes, Executive Director, Transportation District Commission of Hampton Roads, Hampton, VA

**Executive Director:** Robert E. Skinner, Jr., Transportation Research Board

### **MEMBERS**

WILLIAM D. ANKNER, Director, Rhode Island DOT

THOMAS F. BARRY, JR., Secretary of Transportation, Florida DOT

MICHAEL W. BEHRENS, Executive Director, Texas DOT

JOSEPH H. BOARDMAN, Commissioner, New York State DOT

SARAH C. CAMPBELL, President, TransManagement, Inc., Washington, DC

E. DEAN CARLSON, Secretary of Transportation, Kansas DOT

JOANNE F. CASEY, President, Intermodal Association of North America

JAMES C. CODELL III, Secretary, Kentucky Transportation Cabinet

JOHN L. CRAIG, Director, Nebraska Department of Roads

BERNARD S. GROSECLOSE, JR., President and CEO, South Carolina State Ports Authority

SUSAN HANSON, Landry University Professor of Geography, Graduate School of Geography, Clark University

LESTER A. HOEL, L. A. Lacy Distinguished Professor, Department of Civil Engineering, University of Virginia

ADIB K. KANAFANI, Cahill Professor and Chairman, Department of Civil and Environmental Engineering, University of California at Berkeley

RONALD F. KIRBY, Director of Transportation Planning, Metropolitan Washington Council of Governments

HERBERT S. LEVINSON, Principal, Herbert S. Levinson Transportation Consultant, New Haven, CT

MICHAEL D. MEYER, Professor, School of Civil and Environmental Engineering, Georgia Institute of Technology

JEFF P. MORALES, Director of Transportation, California DOT

KAM MOVASSAGHI, Secretary of Transportation, Louisiana Department of Transportation and Development

DAVID PLAVIN, President, Airports Council International, Washington, DC

JOHN REBENDSORF, Vice President, Network and Service Planning, Union Pacific Railroad Co., Omaha, NE

CATHERINE L. ROSS, Executive Director, Georgia Regional Transportation Agency

JOHN M. SAMUELS, Senior Vice President-Operations Planning & Support, Norfolk Southern Corporation, Norfolk, VA

PAUL P. SKOUTELAS, CEO, Port Authority of Allegheny County, Pittsburgh, PA

MARTIN WACHS, Director, Institute of Transportation Studies, University of California at Berkeley

MICHAEL W. WICKHAM, Chairman and CEO, Roadway Express, Inc., Akron, OH

MIKE ACOTT, President, National Asphalt Pavement Association (ex officio)

MARION C. BLAKEY, Federal Aviation Administrator, U.S.DOT (ex officio)

REBECCA M. BREWSTER, President and CEO, American Transportation Research Institute, Atlanta, GA (ex officio)

THOMAS H. COLLINS (Adm., U.S. Coast Guard), Commandant, U.S. Coast Guard (ex officio)

JENNIFER L. DORN, Federal Transit Administrator, U.S.DOT (ex officio)

ELLEN G. ENGLEMAN, Research and Special Programs Administrator, U.S.DOT (ex officio)

ROBERT B. FLOWERS (Lt. Gen., U.S. Army), Chief of Engineers and Commander, U.S. Army Corps of Engineers (ex officio)

HAROLD K. FORSEN, Foreign Secretary, National Academy of Engineering (ex officio)

EDWARD R. HAMBERGER, President and CEO, Association of American Railroads (ex officio)

JOHN C. HORSLEY, Executive Director, American Association of State Highway and Transportation Officials (ex officio)

MICHAEL P. JACKSON, Deputy Secretary of Transportation, U.S.DOT (ex officio)

ROBERT S. KIRK, Director, Office of Advanced Automotive Technologies, U.S. Department of Energy (ex officio)

RICK KOWALEWSKI, Acting Director, Bureau of Transportation Statistics, U.S.DOT (ex officio)

WILLIAM W. MILLAR, President, American Public Transportation Association (ex officio)

MARY E. PETERS, Federal Highway Administrator, U.S.DOT (ex officio)

SUZANNE RUDZINSKI, Director, Office of Transportation and Air Quality, U.S. Environmental Protection Agency (ex officio)

JEFFREY W. RUNGE, National Highway Traffic Safety Administrator, U.S.DOT (ex officio)

ALLAN RUTTER, Federal Railroad Administrator, U.S.DOT (ex officio)

ANNETTE M. SANDBERG, Deputy Administrator, Federal Motor Carrier Safety Administration, U.S.DOT (ex officio)

WILLIAM G. SCHUBERT, Maritime Administrator, U.S.DOT (ex officio)

ROBERT A. VENEZIA, Earth Sciences Applications Specialist, National Aeronautics and Space Administration (ex officio)

## **NATIONAL COOPERATIVE HIGHWAY RESEARCH PROGRAM**

### **Transportation Research Board Executive Committee Subcommittee for NCHRP**

GENEVIEVE GIULIANO, University of Southern California,  
Los Angeles (Chair)

E. DEAN CARLSON, Kansas DOT

LESTER A. HOEL, University of Virginia

JOHN C. HORSLEY, American Association of State Highway and  
Transportation Officials

MARY E. PETERS, Federal Highway Administration

ROBERT E. SKINNER, JR., Transportation Research Board

MICHAEL S. TOWNES, Transportation District Commission  
of Hampton Roads, Hampton, VA

---

---

**NCHRP REPORT 485**

---

---

**Bridge Software—  
Validation Guidelines  
and Examples**

**MICHAEL BAKER JR., INC.**  
Moon Township, PA

in association with

**BRIDGETECH, INC.**  
Laramie, WY

**MODJESKI AND MASTERS, INC.**  
Mechanicsburg, PA

and

**PAUL D. THOMPSON**  
Castle Rock, CO

**SUBJECT AREAS**

Bridges, Other Structures, and Hydraulics and Hydrology

---

Research Sponsored by the American Association of State Highway and Transportation Officials  
in Cooperation with the Federal Highway Administration

---

**TRANSPORTATION RESEARCH BOARD**

WASHINGTON, D.C.  
2003  
[www.TRB.org](http://www.TRB.org)

## **NATIONAL COOPERATIVE HIGHWAY RESEARCH PROGRAM**

Systematic, well-designed research provides the most effective approach to the solution of many problems facing highway administrators and engineers. Often, highway problems are of local interest and can best be studied by highway departments individually or in cooperation with their state universities and others. However, the accelerating growth of highway transportation develops increasingly complex problems of wide interest to highway authorities. These problems are best studied through a coordinated program of cooperative research.

In recognition of these needs, the highway administrators of the American Association of State Highway and Transportation Officials initiated in 1962 an objective national highway research program employing modern scientific techniques. This program is supported on a continuing basis by funds from participating member states of the Association and it receives the full cooperation and support of the Federal Highway Administration, United States Department of Transportation.

The Transportation Research Board of the National Academies was requested by the Association to administer the research program because of the Board's recognized objectivity and understanding of modern research practices. The Board is uniquely suited for this purpose as it maintains an extensive committee structure from which authorities on any highway transportation subject may be drawn; it possesses avenues of communications and cooperation with federal, state and local governmental agencies, universities, and industry; its relationship to the National Research Council is an insurance of objectivity; it maintains a full-time research correlation staff of specialists in highway transportation matters to bring the findings of research directly to those who are in a position to use them.

The program is developed on the basis of research needs identified by chief administrators of the highway and transportation departments and by committees of AASHTO. Each year, specific areas of research needs to be included in the program are proposed to the National Research Council and the Board by the American Association of State Highway and Transportation Officials. Research projects to fulfill these needs are defined by the Board, and qualified research agencies are selected from those that have submitted proposals. Administration and surveillance of research contracts are the responsibilities of the National Research Council and the Transportation Research Board.

The needs for highway research are many, and the National Cooperative Highway Research Program can make significant contributions to the solution of highway transportation problems of mutual concern to many responsible groups. The program, however, is intended to complement rather than to substitute for or duplicate other highway research programs.

---

**Note:** The Transportation Research Board of the National Academies, the National Research Council, the Federal Highway Administration, the American Association of State Highway and Transportation Officials, and the individual states participating in the National Cooperative Highway Research Program do not endorse products or manufacturers. Trade or manufacturers' names appear herein solely because they are considered essential to the object of this report.

## **NCHRP REPORT 485**

Project C12-50 FY 1998

ISSN 0077-5614

ISBN 0-309-06811-8

Library of Congress Control Number 2003104152

© 2003 Transportation Research Board

**Price \$40.00**

## **NOTICE**

The project that is the subject of this report was a part of the National Cooperative Highway Research Program conducted by the Transportation Research Board with the approval of the Governing Board of the National Research Council. Such approval reflects the Governing Board's judgment that the program concerned is of national importance and appropriate with respect to both the purposes and resources of the National Research Council.

The members of the technical committee selected to monitor this project and to review this report were chosen for recognized scholarly competence and with due consideration for the balance of disciplines appropriate to the project. The opinions and conclusions expressed or implied are those of the research agency that performed the research, and, while they have been accepted as appropriate by the technical committee, they are not necessarily those of the Transportation Research Board, the National Research Council, the American Association of State Highway and Transportation Officials, or the Federal Highway Administration, U.S. Department of Transportation.

Each report is reviewed and accepted for publication by the technical committee according to procedures established and monitored by the Transportation Research Board Executive Committee and the Governing Board of the National Research Council.

Published reports of the

## **NATIONAL COOPERATIVE HIGHWAY RESEARCH PROGRAM**

are available from:

Transportation Research Board  
Business Office  
500 Fifth Street, NW  
Washington, DC 20001

and can be ordered through the Internet at:

<http://www.national-academies.org/trb/bookstore>

Printed in the United States of America

# THE NATIONAL ACADEMIES

## *Advisers to the Nation on Science, Engineering, and Medicine*

The **National Academy of Sciences** is a private, nonprofit, self-perpetuating society of distinguished scholars engaged in scientific and engineering research, dedicated to the furtherance of science and technology and to their use for the general welfare. On the authority of the charter granted to it by the Congress in 1863, the Academy has a mandate that requires it to advise the federal government on scientific and technical matters. Dr. Bruce M. Alberts is president of the National Academy of Sciences.

The **National Academy of Engineering** was established in 1964, under the charter of the National Academy of Sciences, as a parallel organization of outstanding engineers. It is autonomous in its administration and in the selection of its members, sharing with the National Academy of Sciences the responsibility for advising the federal government. The National Academy of Engineering also sponsors engineering programs aimed at meeting national needs, encourages education and research, and recognizes the superior achievements of engineers. Dr. William A. Wulf is president of the National Academy of Engineering.

The **Institute of Medicine** was established in 1970 by the National Academy of Sciences to secure the services of eminent members of appropriate professions in the examination of policy matters pertaining to the health of the public. The Institute acts under the responsibility given to the National Academy of Sciences by its congressional charter to be an adviser to the federal government and, on its own initiative, to identify issues of medical care, research, and education. Dr. Harvey V. Fineberg is president of the Institute of Medicine.

The **National Research Council** was organized by the National Academy of Sciences in 1916 to associate the broad community of science and technology with the Academy's purposes of furthering knowledge and advising the federal government. Functioning in accordance with general policies determined by the Academy, the Council has become the principal operating agency of both the National Academy of Sciences and the National Academy of Engineering in providing services to the government, the public, and the scientific and engineering communities. The Council is administered jointly by both the Academies and the Institute of Medicine. Dr. Bruce M. Alberts and Dr. William A. Wulf are chair and vice chair, respectively, of the National Research Council.

The **Transportation Research Board** is a division of the National Research Council, which serves the National Academy of Sciences and the National Academy of Engineering. The Board's mission is to promote innovation and progress in transportation by stimulating and conducting research, facilitating the dissemination of information, and encouraging the implementation of research results. The Board's varied activities annually engage more than 4,000 engineers, scientists, and other transportation researchers and practitioners from the public and private sectors and academia, all of whom contribute their expertise in the public interest. The program is supported by state transportation departments, federal agencies including the component administrations of the U.S. Department of Transportation, and other organizations and individuals interested in the development of transportation. [www.TRB.org](http://www.TRB.org)

[www.national-academies.org](http://www.national-academies.org)

## COOPERATIVE RESEARCH PROGRAMS STAFF FOR NCHRP REPORT 485

ROBERT J. REILLY, *Director, Cooperative Research Programs*  
CRAWFORD F. JENCKS, *Manager, NCHRP*  
DAVID B. BEAL, *Senior Program Officer*  
EILEEN P. DELANEY, *Managing Editor*  
HILARY FREER, *Associate Editor II*

### NCHRP PROJECT C12-50 PANEL Field of Design—Area of Bridges

LOREN RISCH, *Kansas DOT (Chair)*  
RICHARD BEST, *Illinois DOT*  
HADLY EISENBEISZ, *South Dakota DOT*  
EDWARD FAIN, *Arkansas SHTD*  
SUDHAKAR R. KULKARNI, *Michigan DOT*  
BRIAN MCCAFFREY, *New York State DOT*  
RICHARD PICKINGS, *Bridgesight Software, Rescue, CA*  
WALLACE W. SANDERS, *Iowa State University*  
MURUGESU VINAYAGAMOORTHY, *California DOT*  
HAMID GHASEMI, *FHWA Liaison Representative*  
KURT JOHNSON, *AASHTO Liaison Representative*  
STEPHEN F. MAHER, *TRB Liaison Representative*

### AUTHOR ACKNOWLEDGMENTS

The research reported herein was performed under NCHRP Project 12-50 by Michael Baker Jr., Inc.; BridgeTech, Inc.; Modjeski and Masters, Inc.; and Paul D. Thompson. Michael Baker Jr., Inc. was the prime contractor for this study. The work undertaken by BridgeTech, Inc.; Modjeski and Masters, Inc.; and Paul D. Thompson was performed under a subcontract with Michael Baker Jr., Inc.

Mark Mlynarski, Project Manager, Michael Baker Jr., Inc., and Jay A. Puckett, Ph.D., Professor of Civil Engineering, University of Wyoming, and President, BridgeTech, Inc., were the co-principal investigators. The other authors of this report are Chad M. Clancy, Associate, Modjeski and Masters, Inc.; Mark C. Jablin, Engineer, BridgeTech, Inc.; and Paul D. Thompson.

The work was done under the general supervision of Mr. Mlynarski and Dr. Puckett. The work at Michael Baker Jr., Inc., was done under the supervision of Mr. Mlynarski with the assistance of William L. Smyers and Kenneth E. Wilson. The work at BridgeTech, Inc., was under the supervision of Dr. Puckett with the assistance of Mark C. Jablin and Brian L. Goodrich, Engineer. The work at Modjeski and Masters, Inc., was under the supervision of Wagdy G. Wassef, Ph.D., Associate, with the assistance of Chad M. Clancy. The research team would also like to acknowledge James H. Garrett, Jr., Ph.D. of Carnegie Mellon University for his review of a draft of the final report and for his suggestions on the development of the final file formats.

## FOREWORD

By David B. Beal  
Staff Officer  
Transportation Research  
Board

*NCHRP Report 485: Bridge Software—Validation Guidelines and Examples* contains the findings of a study to develop a process for validating bridge design and analysis software. The report describes the research effort leading to the recommended process and discusses the application of this process to software evaluation, design comparisons, and specification review. The material in this report will be of immediate interest to software developers, bridge-design specification writers, and bridge engineers.

---

Computer software is needed to permit a full adoption of the AASHTO *LRFD Bridge Design Specifications* as the standard specification. AASHTO is addressing this issue through joint development of AASHTOWare BridgeWare Products, and others are also developing software systems. Because of the complexity of the specifications, designers and bridge owners need a way to ensure that the software is valid and reliable. A standard validation process is needed to achieve this goal.

The objective of this project was to develop a methodology for bridge-design software validation. This objective has been achieved with the development of “Process 12-50,” a systematic method of comparing and evaluating bridge design and analysis software. Process 12-50 also provides a standardized report format for presenting and comparing results for a specific bridge design and a powerful method for formally reviewing specification changes.

Michael Baker Jr., Inc. was assisted by BridgeTech, Inc.; Modjeski and Masters; and Paul D. Thompson to perform the research. The report fully documents the methodology used to develop the validation process and provides specific instructions for implementing the process. The accompanying CD-ROMs contain the testbed of bridges with well-defined parametric inputs and outputs developed in this research. All common tables necessary to implement Process 12-50 and software to aid in viewing the databases also are included on the CD-ROMs.

<b>1</b>	<b>SUMMARY</b>
<b>3</b>	<b>CHAPTER 1 Introduction and Research Approach</b>
	Introduction, 3
	Research Objectives and Results, 3
<b>4</b>	<b>CHAPTER 2 Findings</b>
	Introduction, 4
	Survey Findings, 4
	DOT Survey, 4
	Developer Survey (Phase I), 4
	Developer Survey (Phase II), 4
	Findings from Process Development, 4
	Findings from Demonstration/Testing of Process 12-50, 5
	Findings from Implementation Details, 5
	Findings from Distribution of Data Investigation, 5
	Use Cases, 5
	Process 12-50 Overview, 5
<b>9</b>	<b>CHAPTER 3 Interpretation, Appraisal, and Applications</b>
	Introduction, 9
	Software Developers, 9
	Software Users, 9
	Specification Developers and Researchers, 9
	Acceptance Criteria, 9
	Implementation Illustration—AASHTO LRFD Subdomains, 9
	Full Automation, 9
	Subdomain Definitions and Parameterization, 10
	Problem Generation, 11
	Performing Computations, 11
	Examination of Results, 14
	Application of the Process—Program Limitations, 18
	Application of the Process—Confirmation of Program Modifications, 18
	Distribution of the Data, 18
<b>20</b>	<b>CHAPTER 4 Conclusion and Suggested Research</b>
	Limitations, 20
	Additional Testing Possibilities, 20
	Design Comparisons, 20
	Specification Review, 20
	Software Validation, 20
	Suggested Research, 21
	Short-Term Research/Action Items, 21
	Long-Term Research, 21
<b>22</b>	<b>GLOSSARY</b>
<b>A-1</b>	<b>APPENDIX A Data Flow and Processing</b>
<b>B-1</b>	<b>APPENDIX B ASCII Table Formats</b>
<b>C-1</b>	<b>APPENDIX C Error Detection and Identification Example</b>
<b>D-1</b>	<b>APPENDIX D Example of Non-Automated Data Comparison Using Process 12-50</b>
<b>E-1</b>	<b>APPENDIX E Subdomain for Process 12-50 and Report IDs</b>
<b>F-1</b>	<b>APPENDIX F Developer Documentation</b>
<b>G-1</b>	<b>APPENDIX G Data Organization</b>



# BRIDGE SOFTWARE—VALIDATION GUIDELINES AND EXAMPLES

## SUMMARY

The automation of bridge computations is an integral part of bridge design and rating practice and will continue to expand. Additionally, the migration to the AASHTO LRFD Bridge Design Specifications and their annual revisions cause engineers to modify their computational processes (CPs) and associated computer-based algorithms. The surveys of the DOTs performed as part of this research indicated that most DOTs have implemented, or plan to implement, the LRFD Specifications within their departments by the Year 2000. Moreover, the AASHTO vote (by the Subcommittee on Bridges and Structures, 1999) to discontinue maintenance of the Standard Specifications accelerates this migration.

NCHRP Project 12-50 was designed to investigate current software validation procedures and to provide an improved method of verifying bridge design and analysis software. Referred to hereafter as Process 12-50, this methodology has resulted in a test-bed of bridges with well-defined parametric inputs and outputs. This test-bed (or portions thereof) is readily usable by developers, end users, and others, and is available on the accompanying CD-ROMs.

Using Process 12-50, two or more software analysis packages and/or hand calculations with the same data set may be compared in tabular and/or graphical format. Process 12-50 permits drilling-down in the results to show how computations were performed and to clearly reveal differences between processes clearly.

With the aid of Process 12-50, a designer easily can compare the results of alternative CPs (e.g., programs and spreadsheets). The results from the processes can be imported into a common viewer for comparison so that differences will be apparent.

Specification writing committees can benefit by having specific test computations on real and derived bridges prior to considering specification changes, both large and small. By comparing different versions on a large set of bridges, specification writers can determine whether or not changes accomplish the desired objectives and can prevent problems from arising because of changes. Similarly, bridge engineers can see the consequences of the changes on current engineering practices before they are put into practice and are found to be problematic.

In theory, software should be free of errors—yet perfect software remains elusive. The number of bridge types, geometric configurations, materials, and loadings creates a large solution domain that must be addressed. Current practice indicates that hand computations and comparisons are common, but provide only limited checking. The developer surveys performed as part of this research indicate that the methods for, and extent of, testing vary. The end user generally has little information indicating the extent of this testing for any given release of a program. Process 12-50 provides a means of developing comprehensive and economical methods that have been demonstrated to be both practical and effective.

Process 12-50 is simple. The bridge analysis and design process is divided into manageable computational domains (the set of all values produced by bridge CPs) such as dead load distribution, live load distribution, live load actions on a structural component, and complete resistance and specification checks of a structural component. Within each of these smaller subdomains, the system is described in parametric form. Here, the inputs and outputs are formally outlined. Additionally, the assumptions of analysis are presented. This parametric form is then used to generate the inputs and outputs (in a common format) from which results from numerous CPs can be compared. CPs may be performed by hand, using a spreadsheet, or employing a complete software program dedicated to bridge engineering. A schematic of Process 12-50 is provided and discussed in Chapter 2.

For most of the subdomains, automated CPs were used to generate problems and associated results. Wyoming DOT's BRASS-GIRDER (LRFD) and BRASS-PIER<sup>1</sup> (PAPIER) and Pennsylvania DOT's LRFD Steel, LRFD Prestressed Concrete, and Pier programs<sup>2</sup> were used extensively. These independently developed CPs were compared. Where the results differed, the differences were explained. A typical comparison plot is shown in Chapter 3 (Figure 6). Other programs from Alaska DOT<sup>3</sup>, Washington DOT<sup>4</sup>, private developers, and research team spreadsheets were also used.

The following are the key findings from NCHRP Project 12-50:

- The results from different CPs associated with bridge design can be standardized and compared.
- The automation of comparisons provides a basis for checking the differences of the computations.
- An increased level of detail in the results can significantly improve the ability of the engineer to determine the reasons for any differences between CPs. Differences may be attributed to errors, specification interpretation, and/or office practice.
- The automation of comparisons provides a basis for the development and long-term maintenance of a subdomain test suite.
- A suite of results (inputs and outputs) for problems of limited scope provides designers, software developers, and owners a basis for validation of highly complex computations—computations that are difficult to validate by hand.
- Specification writers may use the process to investigate the effect of a change during the development of the specifications. As the AASHTO specifications change, Process 12-50 can be used to update the suite in an automated manner. In short, automation can be used to investigate a large number of bridges and to help formulate the specification language, intent, and application (commentary).
- Although the test suite is comprehensive (see Table 1), it is not exhaustive. The suite is intended to be dynamic and will grow as the specifications change or as new permutations of an existing specification are encountered.
- The suite results may be made available to the bridge engineering community and this would be best accomplished via a web site where not only the data could be shared but also ideas for its usage could be exchanged.

In summary, Process 12-50 is general and supports the formal comparison of CPs derived from a host of sources. The results have numerous applications, one of which is the development of a test suite.

---

<sup>1</sup> Developed by BridgeTech, Inc., in close collaboration with the Wyoming Department of Transportation (WYDOT), FHWA, and Kansas Department of Transportation (KDOT). Many other agencies have helped in testing and validation as well.

<sup>2</sup> Developed for the Pennsylvania Department of Transportation (PennDOT) by Michael Baker Jr., Inc. in close collaboration with Modjeski and Masters, Inc.

<sup>3</sup> Developed by the Alaska Department of Transportation and Public Facilities (ADOT&PF).

<sup>4</sup> Developed by the Washington Department of Transportation (WSDOT).

---

## CHAPTER 1

# INTRODUCTION AND RESEARCH APPROACH

### INTRODUCTION

Bridge engineers are relying increasingly on software to perform calculations critical to their practice. As this software becomes more intricate and includes more calculations, verification of the results is becoming increasingly difficult. With so many different possible paths through the logic of a given software package, testing for all scenarios to ensure correctness is nearly impossible. In order for the engineer to have confidence in the software, a significant amount of resources must be applied to tedious output review and verification. The state of the practice is to compare computed results to independently developed hand computations. Unfortunately, hand computations are expensive and limiting. Typically, only a select number of cases within the problem domain (a set of problems, such as bridge examples, associated with an engineering computational process (CP) that exercises the limits of that process) can be adequately investigated. Moreover, the bridge engineering community and software developers perform this validation in a diverse and independent manner. In the present state of the practice, results, test cases, validation procedures, etc., are proprietary and are usually neither published nor shared. The end user (bridge engineer) must either trust the computations or loosely validate with an independent program, spreadsheets, and/or hand computations. Minor changes to the specifications quickly render hand computations outdated. Although the responsibility for engineering computations resides with the engineer of record, some assistance, procedure, or measure for quality assurance and control is necessary and can be provided by the bridge software community. Such efforts, if systematically pursued and

shared, would benefit both the bridge-software and end-user communities. The resulting products could be published or shared with the community to provide a large set of computations. Software developers and end users could use these computations in many ways.

Throughout this document, terminology specific to this research and to software development is used. To aid the reader, a glossary is provided following Chapter 4.

### RESEARCH OBJECTIVES AND RESULTS

The goal of this research was to develop a process by which engineering computations can be systematically organized into their basic components, efficiently validated for a well-defined portion of the computational domain, and distributed to the bridge community. A domain is defined as a collection of computations and associated data. Research results are threefold:

- Development of a process (named Process 12-50) for systematic validation,
  - Formal definition of a set of domains with well-defined inputs and outputs that can be used to validate computations associated with *the AASHTO LRFD Bridge Design Specifications* (AASHTO 1994 with 1997 interim versions, herein referred to as the *LRFD Specifications*), and
  - Provision of well-defined domains that may be used in a host of applications, including software validation, comparisons for a specific bridge design, and (AASHTO) specification development and research.
-

## CHAPTER 2

# FINDINGS

### INTRODUCTION

Several specific types of findings are documented in this chapter. These specific findings are supplemented and supported by various appendices as described throughout this chapter. A summary of Process 12-50 completes the chapter.

### SURVEY FINDINGS

Key findings from surveys of the DOTs and of the developers in Phases I and II are summarized in the following subsections.

#### DOT Survey

Thirty-one DOTs responded to a survey requesting information. A summary of the key findings follows:

- A complete description of the method of solution is important because the *LRFD Specifications* contain numerous decision areas.
- Several state DOTs indicated that they would be moving to LRFD design within 2 years and are interested in validation, examples, and training prior to or at the time they switch to LRFD (by 2000).

#### Developer Survey (Phase I)

Thirteen bridge software developers responded to a survey during the first phase of this research. The following is a summary of the key findings:

- Current test methods vary widely and are informal.
- Well-defined validation procedures are needed.
- Administrative processes associated with software validation and maintenance of a test suite are a concern.
- Compliance with a test suite should be at the discretion of the engineer of record.
- ASCII results files are the preferred method of data storage.
- Clear definition of assumptions, inputs, and outputs is necessary.

- A test suite should test as many computational paths as possible.
- Automated problem generation and results comparison would be ideal.

#### Developer Survey (Phase II)

During the second phase of the research, a second developer survey was conducted. This survey focused on the use and format of the developed test suites. The comments of the five developers who responded to this survey are summarized below:

- Input/OutputResults should be available in both XML and comma-delimited ASCII.
- Forums for the discussion of results would be useful.
- Downloads for the basic tools would be useful.

### FINDINGS FROM PROCESS DEVELOPMENT

Findings from the process development are summarized below:

- A systematic process (Process 12-50) was developed, documented, and validated.
- Process 12-50 is independent of any specification, CP, or analytical tool.
- Bridge design is a large and complex computational domain that can be divided into smaller, more manageable subdomains.
- Subdomains can be described in terms of well-defined inputs and outputs.
- Inputs and outputs can be independent of specific CPs.
- Computational results from most processes can be organized into a standard format and then compared by using plotting and numerical differencing.
- The computational results can be directly linked to specification references, including different AASHTO specification and program versions.
- Computational results, specification references, and process specifics can be stored and manipulated using relational databases, XML, or comma-delimited ASCII files.

## FINDINGS FROM DEMONSTRATION/TESTING OF PROCESS 12-50

Demonstration/testing findings include the following:

- Process 12-50 has been demonstrated with numerous subdomains.
- Process 12-50 has been demonstrated with 12 independent CPs (e.g., programs, hand computations, and spreadsheets).
- Process 12-50 can be highly automated, so that input and output can be generated and compared for hundreds of problems.
- Process 12-50 can be used to study a limited number of problems when results are generated by hand.
- Process 12-50 can be used to study a limited number of problems for software for which inputs and results cannot be automatically generated.
- Process 12-50 was used to find and resolve errors in most CPs used in testing.
- Process 12-50 was expanded from the initial set of simple-span superstructures to include results for multi-span superstructures. This expansion demonstrated the ability to add new subdomains using the same procedures (i.e., extensibility).
- Process 12-50 was expanded to include piers. An auxiliary ID was added to allow for the deeper data hierarchy necessary for the pier analysis.
- Although the input generation and results comparisons can be automated, engineering judgment is imperative to proper interpretation of the results.

## FINDINGS FROM IMPLEMENTATION DETAILS

Key findings from implementation are as follows:

- Relational data structures are valuable for increasing the speed and ease of results comparison.
- Relational databases permit the development of queries to investigate specific problem areas associated with a particular specification reference or structural characteristic.
- Detailed reports of intermediate computations revealed differences between CPs and facilitated the correction of errors and the resolution of differences.
- Depending on the assumptions made within the CPs, different intermediate and final results can be explained and deemed acceptable.
- Graphical display of results is highly effective in detecting differences/errors.

## FINDINGS FROM DISTRIBUTION OF DATA INVESTIGATION

Key findings from the investigation of the distribution of data are as follows:

- Results are distributed in comma-delimited ASCII format and XML. See Appendix B for details.
- A web site for the distribution of the Process 12-50 data should be developed and maintained.
- The use of Process 12-50 should be encouraged for future AASHTO specification development and research.

## Use Cases

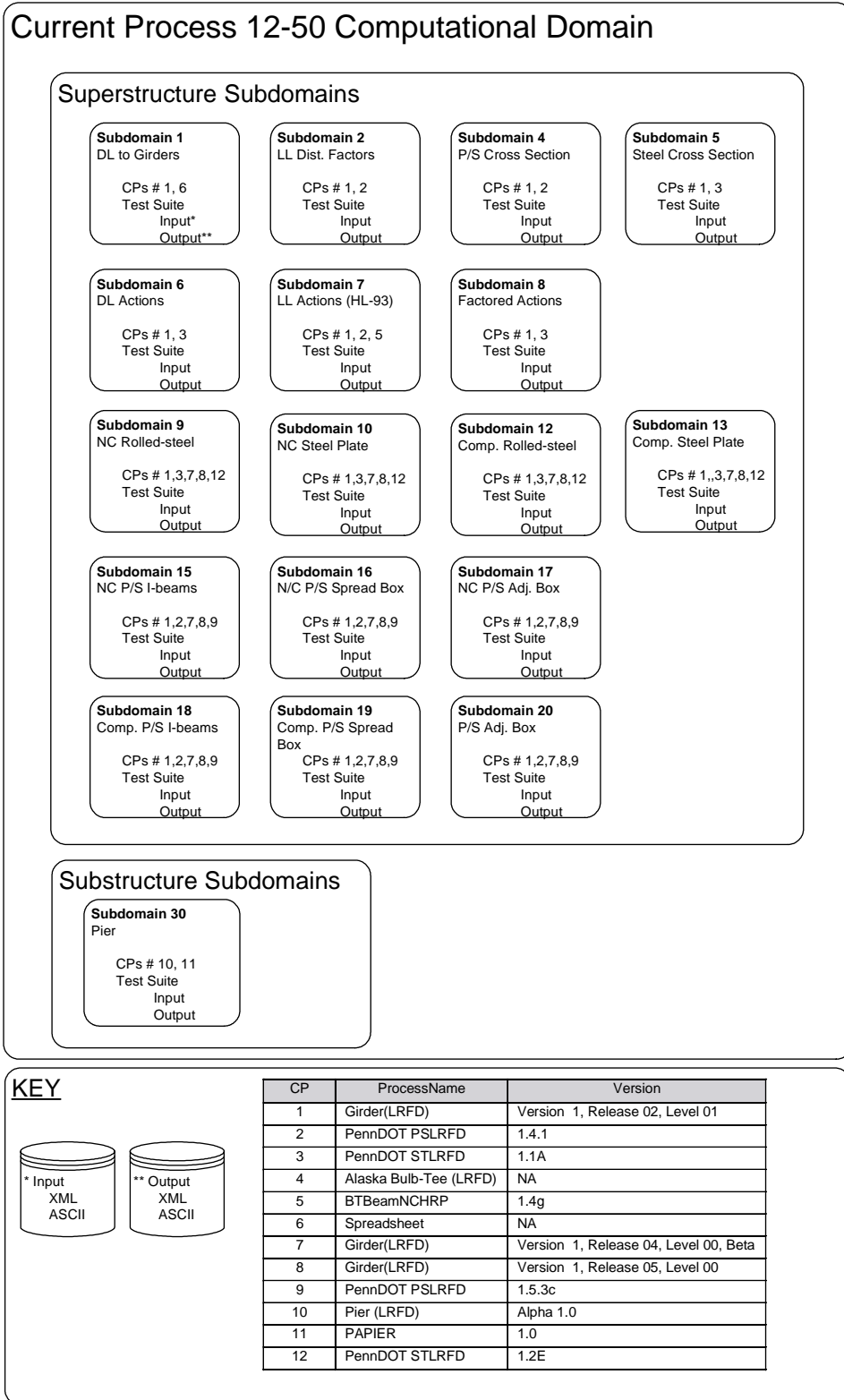
To determine the proper use and distribution of Process 12-50 data, use cases were developed. Those use cases, in order of priority, are summarized as follows:

1. Regression test a single CP (software versions).
2. Test possible changes to AASHTO specifications.
3. Provide examples for engineers.
4. Use the large number of problems for research via data mining.
5. Develop a new test suite.
6. Facilitate software development and validate software for professional use (by design engineers).
7. Perform a design review with multiple processes for a specific structure or subdomain within a structure.
8. Validate in-house software such as spreadsheets and solvers.
9. Add a new subdomain conforming to an existing Process 12-50 data format.
10. Add a new subdomain based on a new data format.
11. Facilitate a discussion forum for Process 12-50.

## PROCESS 12-50 OVERVIEW

Conceptually, Process 12-50 is simple and is outlined as follows in this section.

- **Understand the Problem Domain Associated with an Engineering Computational Process.** To ensure that Process 12-50 is to be effective, the person using the procedure must have a thorough understanding of the problem domain being tested. It is also essential that the user understand the limitations of numerical comparisons, namely that identical results are highly unlikely, if not impossible, in many cases.
- **Divide the Large Problem Domain into Smaller Subdomains as Necessary.** For relatively simple computations, dividing the large problem domain into smaller subdomains may not be required. The bridge engineering computational domain is huge and is divided into smaller subdomains as illustrated herein. Appendix E provides a list of the subdomain divisions of the *LRFD Specifications* used for this research. A schematic of the key components of Process 12-50 is provided in Figure 1.
- **Define Parameters for the Subdomains.** Each subdomain is described in a general sense, outlining the



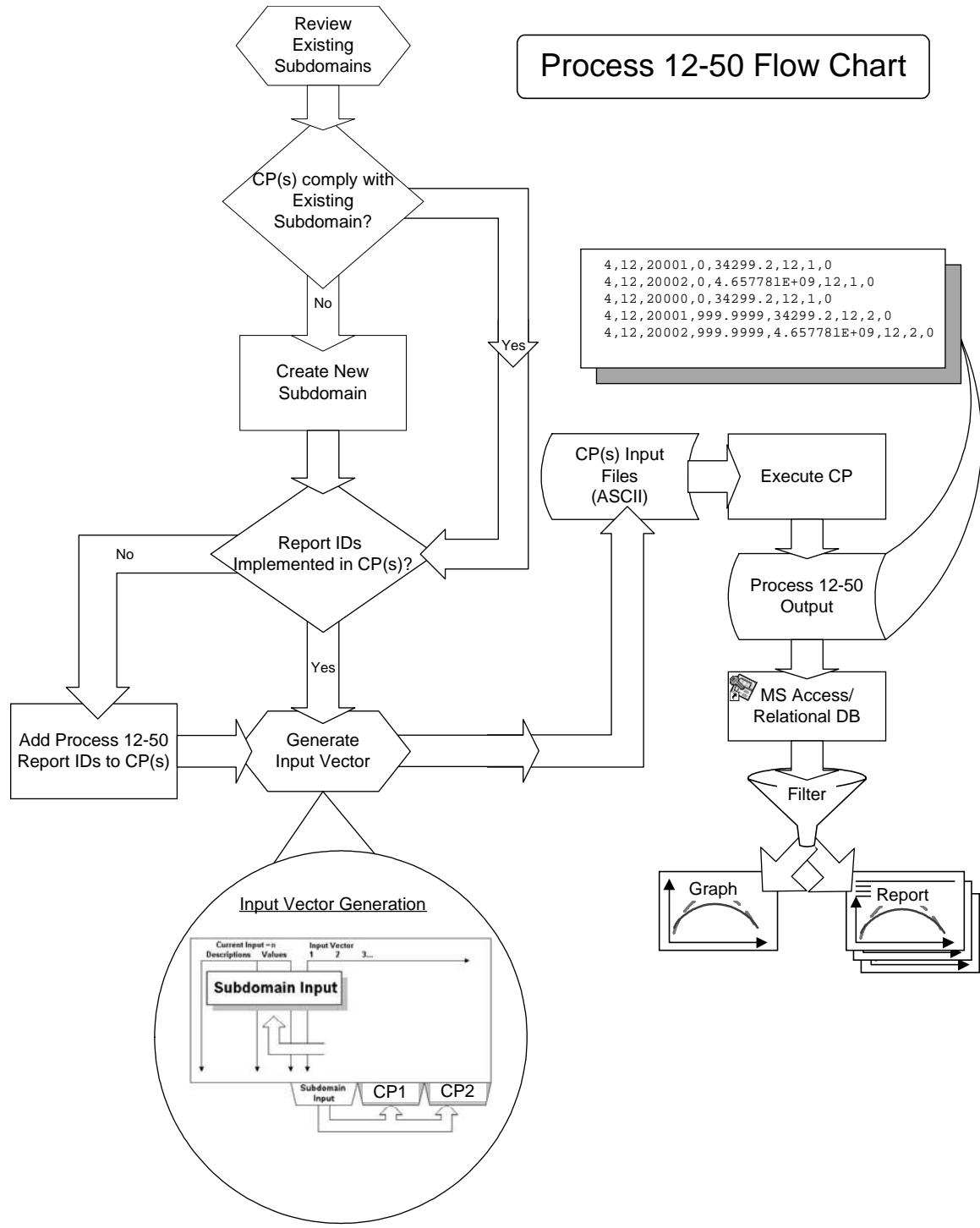


Figure 1. Process 12-50 (continued).

purpose of the computation, the limitations and scope of the computation, the parametric definition of the inputs and outputs, and the CP(s) to be used for testing. The assumptions associated with the CP must be clearly outlined as well. In short, inputs are systematically mapped to outputs. Additional details for each subdomain are provided in Appendix E.

- **Generate Problems Using the Parametric Form.** Each computational subdomain has different functions and characteristics. For example, in the bridge engineering computational domain, many computations associated with structural analysis are founded on the principles of structural mechanics. Other computations are associated with AASHTO specification articles, which may be

based on a combination of empirical results, mechanics, experience, construction considerations and so forth. The characteristics of the subdomain must be understood. The challenging task for the subdomain creator (subdomain designer) is to design input data in the parametric form that addresses the majority of issues (AASHTO checks) associated with a specific subdomain. For example, if a subdomain is to address prestressed concrete design/review for the *LRFD Specifications*, then all the branches of the specifications and all of the practical parametric geometry and material properties need to be addressed. Moreover, the process should provide “proof” that various articles are checked/validated. This proof can be obtained by querying the results data and checking the Subdomain Report ID/Specification Reference Table. Process 12-50 methods used to generate input using the parametric form are outlined in Appendix A. Note that this is one of many possible ways to generate input automatically. Additional details on the generation of problem input for each of the subdomains are provided in Appendix E.

- **Perform the Computation for Problems within the Subdomain.** This task depends on the implementation of the computations within software or hand processes. For example, CPs may use hand calculations, spreadsheets, macros, and traditional programs to perform calculations. The implementation may be “batch oriented” or highly interactive—the approach varies. For all implementations (except hand calculations), the parametric data outlined above must be entered into the program and the program must be executed.
- **Examine the Results from the Computations.** How results are examined depends on the intent and can vary

widely. The examination might include, but would not be limited to, the following:

- Creation of a test suite for program validation;
- Comparison of results from other CPs, including hand computations;
- Development of plots illustrating trends in input versus output parameters;
- Development of plots illustrating trends in output versus location; and
- Development of plots illustrating trends in output versus various methods, assumptions, specification versions, state specifications, and so forth.

Examples of how the results were examined for the development of Process 12-50 are contained in Appendix E. (The actual results and the method used to examine them are contained in the accompanying CD-ROMs.)

- **Automate Comparisons.** The examination of results can be a laborious task and should be automated whenever appropriate. For example, in this research, the focus was to develop a process for checking software computations directed at the LRFD Specifications. For this development, hundreds of problems were parameterized for numerous subdomains. These data could be systematically processed with numerous approaches. Herein, a few methods are outlined. Process 12-50 depends neither on the tools used for input generation nor on the examination of results. This is critical, because tools will change, the results will change, but the underlying parameterization should remain relatively constant. The tools used in this research to demonstrate Process 12-50 can be examined on the accompanying CD-ROMs. An example of Process 12-50 using a non-automated CP is given in Appendix D.
-



## CHAPTER 3

# INTERPRETATION, APPRAISAL, AND APPLICATIONS

### INTRODUCTION

Although the responsibility for engineering computations resides with the engineer of record, some assistance with, procedure for, or measure of quality assurance and control is desirable and can be provided by the bridge-software community. Such efforts, systematically pursued and shared, benefit both the bridge-software and end-user communities. The resulting products can be published/shared with the community to provide a large set of validated computations. Software developers, software and specification developers, and research users can use the resulting information in many ways, some of which are described in the following subsections.

#### Software Developers

Software developers can obtain test suites in their original form so the user can implement other CPs, modify the test problems, and make comparisons as desired. As more CPs are developed and added to the suite, the degree of confidence in the results should increase (assuming that good agreement is achieved among processes). Different versions of the same CP can be given different Process IDs so that changes to existing code can be thoroughly tested.

Programs often reflect experience and engineering judgment. With tools such as those provided on the accompanying CD-ROMs, the consequences of such assumptions can be investigated more readily.

#### Software Users

For software users, completed test suites are available. Input and output data are available for each of the subdomains documented in Appendix E and results are ready to be viewed using the charting features in the database. This form of the subdomain allows a user to gain confidence in the tested CP and others that will be added to the subdomain. The user also can input results from another CP into the database manually, thereby “spot-checking” the results currently stored in the database.

#### Specification Developers and Researchers

Changes to the AASHTO design specifications are often proposed that have been checked against only a limited number of test problems. How a specification change may affect the bridge design as a whole is seldom investigated rigorously. At times, problems resulting from these changes only become evident after the new specifications have been adopted and published. Test suites such as those documented herein can be used to investigate trends that result from changes to the design specification prior to the implementation of such changes. Using Process 12-50, parametric studies can be performed on a much wider range of bridge types and configurations than is currently possible.

### ACCEPTANCE CRITERIA

Because of the many possible uses of the test suites and the potentially different objectives for the application of Process 12-50, universal acceptance criteria are difficult, if not impossible, to establish. After the process has been applied, one can readily see whether two or more programs are in general agreement for key output parameters. However, in the realm of bridge engineering software, no two programs are exactly alike. Subtle variations in assumptions may result in differences that may be minor when applied for one purpose but significant when applied with different goals in mind. For example, an end user (i.e., bridge engineer) may find a difference in results of 5% perfectly acceptable. A developer applying the process to test and compare different versions of the same program, however, would need to explain and justify even the smallest of differences. Process 12-50 usually facilitates such explanations because the computations can be broken down into discrete steps as described in the next section.

### IMPLEMENTATION ILLUSTRATION—AASHTO LRFD SUBDOMAINS

#### Full Automation

For the purposes of developing Process 12-50 and verifying its effectiveness, all steps were automated as much as possible. Automation streamlines development by allowing

changes to be examined quickly. Automation also allows for rapid updating of results that reflect changes to specifications, processes, and input bounds. Although Process 12-50 is independent of automation, its effectiveness is greatly increased by automating the various tasks involved. The effect of automation on the number of examples reviewed is evident in the results shown in Table 1.

### Subdomain Definitions and Parameterization

The *LRFD Specifications* encompass superstructures, substructures, foundations, and other structures as well as associated structural analysis and design-based computations. During the first phase of this research project, the domain was limited to simply-supported steel and prestressed concrete superstructures. Phase II expanded to include testing for multi-span steel and prestressed concrete girder systems and concrete piers.

Each subdomain is parameterized by selecting the input values that are critical to the computations (i.e., input values are selected to rigorously test the specifications associated with the subdomain). An example would be the selection of

input values for the live load distribution factor subdomain (see Appendix E) that exercise the distribution factor applicability limits provided in the *LRFD Specifications*. While the live load distribution factor subdomain clearly can be defined using the specification applicability limits, other subdomain input parameterization methods, such as choosing input to test the limits of the HL-93 live load subdomain, are less obvious (see Appendix E).

Once the parameterization of a subdomain has been defined, a range of indices is assigned and each computational result is assigned a Report ID that can be used in many ways. For a fully automated approach, the Report IDs need to be programmed into the software so the CP results are linked to the appropriate data definition. The parameterization clearly defines the critical input data and computational results.

Ideally, at least one other CP exists that performs nearly the same computations for comparisons. If other software is available, the same Report IDs must be assigned. If no additional software is available (or the tester does not have access to the code to implement the Report IDs), the process can still be applied successfully with simple data entry, such as using the paste command (see Appendix D).

**TABLE 1 Process 12-50 subdomain content**

Subdomain ID	Description	No. of Test Cases in Subdomain Test Suite
1	Dead Load to Girders	33
2	Live Load Distribution Factors—Simple Span	77
4	Prestressed Concrete Cross Sections	180
5	Steel Cross Section	73
6	Dead Load Actions (Superstructure)	40
7	Live Load Actions (HL-93)	726
8	Factored Load Actions	40
9	Non-Composite Rolled Steel Sections	43 Simple span 10 Multi-span
10	Non-Composite Steel Plate Girders	30 Simple span 135 Multi-span
12	Composite Rolled Steel Sections	43 Simple span 10 Multi-span
13	Composite Steel Plate Girders	30 Simple span 135 Multi-span
15	Non-Composite Prestressed I-Sections	81 Simple span 112 Multi-span
16	Non-Composite Prestressed Spread Box Sections	56 Simple span 66 Multi-span
17	Non-Composite Prestressed Adjacent Box Sections	43 Simple span 114 Multi-span
18	Composite Prestressed I-Sections	81 Simple span 112 Multi-span
19	Composite Prestressed Spread Box Sections	56 Simple span 66 Multi-span
20	Composite Prestressed Adjacent Box Sections	43 Simple span 114 Multi-span
30	Piers	

Note: For a list of all subdomains, see Appendix E.

## Problem Generation

After the subdomains have been defined and the Report IDs have been assigned, it is necessary to develop a group of input data sets to test as many of the logic paths as possible for the results being computed. The simplest way to do this is to automate the generation of the input, which permits the tester to modify all of the input with a minimum of effort.

Input generation can be accomplished in several ways (depending on the implementation). For example, data can be created manually (e.g., an input file is created for the CP) and then manually copied and each copy changed slightly in order to test as many paths through the AASHTO specifications as possible. Input data generation can be automated through the development of a spreadsheet. This spreadsheet also provides the user one place in which the input for all test sets can be compared, thus eliminating several cumbersome tasks associated with manual approaches.

For this research, two methods were used to generate files automatically. The first is the loop method, which is only practical for groups of results for which the input is relatively simple (e.g., the HL-93 actions along a girder line). For the HL-93 actions example, all live load distribution factors are assumed to be 1.0, so the only variable that affects magnitudes of the live load actions is the length of the spans of the bridge. For this problem set, Visual Basic for Applications (VBA) code was developed to automatically loop through the number of spans and the length of each span, generating several hundred input sets in a matter of minutes. After each increment of span length, the VBA code automatically created an input file in the format expected by each of the CPs used in the testing. Figure 2 illustrates the loop control parameters for the HL-93 live load actions. The start, stop, and increment values can be changed to control the number of input files generated, as well as the types of problems tested. If the tester suspects that the software may have a problem computing shear for cases with high adjacent span ratios, the values can be adjusted to ensure that this type of problem is

Short Spans			
	start	stop	inc
number of spans	1	5	1
span 1 length	6	32	13
span 2 length	6	32	13
span 3 length	6	32	13
span 4 length	6	32	13
span 5 length	6	32	13
Long Spans			
	start	stop	inc
number of spans	1	5	1
span 1 length	36	96	30
span 2 length	36	96	30
span 3 length	36	96	30
span 4 length	36	96	30
span 5 length	36	96	30

Figure 2. Sample loop control parameters.

generated. As the parameters are incremented, the input files are automatically generated by reading an input template from the spreadsheet and substituting the incremented values. Figure 3 shows a typical input template.

The second method of automatic file generation used was the input vector method, which is more versatile than the loop method and can be used for more complicated problem sets, such as precast, prestressed concrete girderlines. The number of different input parameters that have an effect on the calculation of the desired results makes the incremental looping of the input parameters impractical. Simply put, the user defines all of the input parameters in a spreadsheet, and then VBA code generates the data files for each CP involved. This method requires input for all problem sets to be entered in the spreadsheet, but the different input formats of different CPs are handled automatically. The user can also view the input for all problems at once, which allows easy changes. Figure 4 provides a portion of the input file generator for the simple-span prestressed composite I-section problem set. This section of the generator contains a portion of the input for three separate problems, listed under the *Input Vector* heading. The input is converted to the units required by each CP based on the conversion factor listed. For example, one program may require input in meters, while another requires input in millimeters. The VBA code loops on the CPs and writes each input data set. (Templates similar to that shown in Figure 4 are used to write the input for this method as well.)

One of the main advantages of the input vector method is that it allows the user to customize each problem to test a certain path through the program and specification. Although it takes a significant amount of time to customize each problem, this method increases the efficiency by cutting down the number of data sets necessary to test the desired logic paths. The spreadsheet generator can also be developed to include tools to help the user to calculate intermediate values and thereby determine whether or not a certain problem tests a certain portion of the program/specifications—all without actually running the program. Whichever method is used (i.e., loop or input vector), definition of the input data in one place permits the generation of input for several CPs from one central set of data.

## Performing Computations

After the input files for the different CPs have been generated, the next step is to run the input files through each CP. As the input files are being generated, a batch file that permits all of the input files to execute sequentially is created automatically. The batch file can be modified easily to run a certain set of problems, or the problems can be executed one at a time. With the automated approach, a database-readable output file is generated by each CP; the format is shown in Figure 5. Although the generation of the file requires modifications to CP software, the advantages of having the results

Command	Parameter					
	1	2	3	4	5	
TITLE	1	Generated input for BRASS				
TITLE	1	Modjeski and Masters, Inc.				
ENGINEER	1	Chad M. Clancy				
COMMENT	1	Generator Version: 1.0				
BRIDGE-NAME	2	1 NCHRP12-50 Test Bridge				
COMMENT	1	Vector range description/comment: Varying span length (zero)				
ANALYSIS	4	B	3	REV		
UNITS	1	SI				
POINT-OF-INTEREST	3	T	OFF	OFF		
DIST-LL-APPLICATION	1	AP				
DIST-BEAM-SCHEDULE	6	1	M	0.85	0.85	0
DIST-BEAM-SCHEDULE	6	1	V	0.85	0.85	0
DIST-BEAM-SCHEDULE	6	1	D	0.85	0.85	0
OUTPUT	4	1	ON	ON	OFF	
OUTPUT-STAGE	3	ON	ON	ON		
OUTPUT-DIST-LL	3	NO	NO			
OUTPUT-ECHO-EXPAND	1	YES				
OUTPUT-STIFFENERS	3	OFF	ON	OFF		
OUTPUT-SHEAR-CONN	2	ON	ON			
OUTPUT-LIMIT-STATE	4	ST	1	ON	ON	
OUTPUT-LIMIT-STATE	4	ST	3	ON	ON	
COMMENT OUTPUT-LIMIT-STATE	4	ST	5	ON	ON	
OUTPUT-LIMIT-STATE	4	SE	2	ON	ON	
OUTPUT-LIMIT-STATE	4	FA	1	ON	ON	
FACTORS-LOAD	11	ST	1	0.95	0.95	1.05
FACTORS-LOAD	11	ST	3	0.95	0.95	1.05
OUTPUT-POST-TABLES	3	NO	YES	NO		
STEEL-MATERIALS	3		200000			
COMPOSITE-MATERIALS	6	28	400	8	3	3.33

Figure 3. Sample input template.

from separate CPs in a similar format greatly outweigh the cost. During the development of Process 12-50, emphasis was placed on producing an output format that puts the least burden on the software developer while still yielding exceptional analysis capabilities. The result of this effort is a simple file format that contains eight fields separated by commas. The fields are listed below, along with the notation of whether the field is a primary key (PK) or a foreign key (FK) as applicable. A primary key is a field or a combination of fields that uniquely identifies each record in the table. A foreign key is a field in one table that matches the primary key of another table. The following is a summary of the field definitions:

- **Bridge ID (PK)**—A unique identifier that defines the bridge or problem. The bridge is unique and has a defined set of input parameters (e.g., span length, beam size, and beam spacing). The definition of the bridge is also defined in the database and can be referenced using the Bridge ID. Generally, this field value can be input into the CP and echoed back out on each line of the Process 12-50 results file.
- **Process ID (FK)**—This ID defines the process used to generate the results (or value) contained in this record (see Table 2). The Process ID is a unique number that

can define a CP. Each version of the same CP is given a unique Process ID. This feature allows for regression testing within a particular program. Examples of hand computations can also be entered into the database, which allows for completed examples like those in the *PCI Bridge Design Manual* to be included in the test suite and compared with the results of other *LRFD* CPs. Generally, this ID can be input into the program and echoed back out on each line of the Process 12-50 results file.

- **Report ID (FK)**—This ID identifies the value that is being reported (e.g., a moment, shear, or stress calculation). A unique Report ID is provided for each value reported. The purpose of the Report ID is to provide an association that allows the comparison of values from two CPs. Table 3 provides a sample of Report IDs. For a comprehensive list of Report IDs see Appendix E.
- **Location**—This field defines the physical location along the structure.
- **Value**—This field defines the value of the data.
- **Subdomain ID (FK)**—This field identifies the subdomain. The test suite is divided into several subdomains based on the portion of the specification that is being checked (see Table 4). For example, if a user is interested in examples that tested the *LRFD* distribution factors for

Input Items	UnitID	Units	Current Input	Raw Input	Current	Engine Conversion Factors					Input Vector		
			1	1	Conv. Fact.	Engine Number					BridgeID		
			1SI	1US	1SI	2US	2SI	1	2	3			
Description	UnitID	Units	1	1	1SI	1US	1SI	2US	2SI	1	2	3	
Exterior girder indicator	18	[bool]	FALSE	FALSE						FALSE	FALSE	FALSE	
Single lane ADTT			1200	1200						1200	1200	1200	
Live load vehicle	1	[dim]	HL-93	HL-93						HL-93	HL-93	HL-93	
Girder spacing	2	mm	3	3000	0.001	0.083333333	0.001			3000	3000	3000	
Overhang width	2	mm	1.5	1500	0.001	0.083333333	0.001			1500	1500	1500	
Span length	2	mm	14.3	14300	0.001	0.083333333	0.001			14300	14300	15400	
Type of strand (L/S -> Low-Lax/Stress-Relieved)	1	[dim]	L	L						L	L	L	
P/S transfer time	17	Days	2	2						2	2	2	
Relative humidity in percent	15	%	60	60						60	60	60	
Strand configuration (Draped = DP, Debonded = DB, Straight, bonded = ST)	1	[dim]	DP	DP						DP	DP	DP	
28 Day concrete strength of beam, f'c	7	MPa {N/mm^2}	55	55						55	55	55	
Initial concrete strength of beam, f'ci	7	MPa {N/mm^2}	46.75	46.75						46.75	46.75	46.75	
Slab/Beam Interface (Monolithic = M, Unroughened = U, Intent. Roughened = I)	1	[dim]	I	I						I	I	I	
Beam concrete cure (S = Steam, M = Moist)	1	[dim]	S	S						S	S	S	
P/S tendon ultimate strength, fpu	7	MPa {N/mm^2}	1860	1860						1860	1860	1860	
P/S tendon initial stress, fpj (jacking stress, not inclusive of initial relaxation loss)	7	MPa {N/mm^2}	1395	1395						1395	1395	1395	
P/S tendon yield strength, fpy	7	MPa {N/mm^2}	1674	1674						1674	1674	1674	
Stirrup yield strength, fys	7	MPa {N/mm^2}	420	420						420	420	420	
Beam longitudinal mild steel, fyb	7	MPa {N/mm^2}	420	420						420	420	420	
Beam designation string (See BeamData tab)	1	[dim]	04550760	045507600						04550760	05050760	04550835	
Harp point location (percent of span length) - N/A if straight strands	15	%	33	33						33	33	33	
P/S tendon diameter	2	mm	12.7	12.7						12.7	12.7	12.7	
P/S tendon area (single tendon)	4	mm^2	98.7	98.7						98.7	98.7	98.7	
Known strand pattern (TRUE/FALSE); must be true for debonded!	18	[bool]	TRUE	TRUE						TRUE	TRUE	TRUE	
CGS at end of beam (at CL Brg) - N/A for straight strands	2	mm	310	310						310	330	340	
CGS at midspan -> target CGS for known strand pattern	2	mm	240	240						240	260	260	
Number of strands (total)	1	[dim]	32	32						32	42	34	
Distance to bottom row of strands from bottom of beam	2	mm	50	50						50	50	50	
Vertical spacing of strand rows	2	mm	50	50						50	50	50	

Figure 4. Sample input vector.

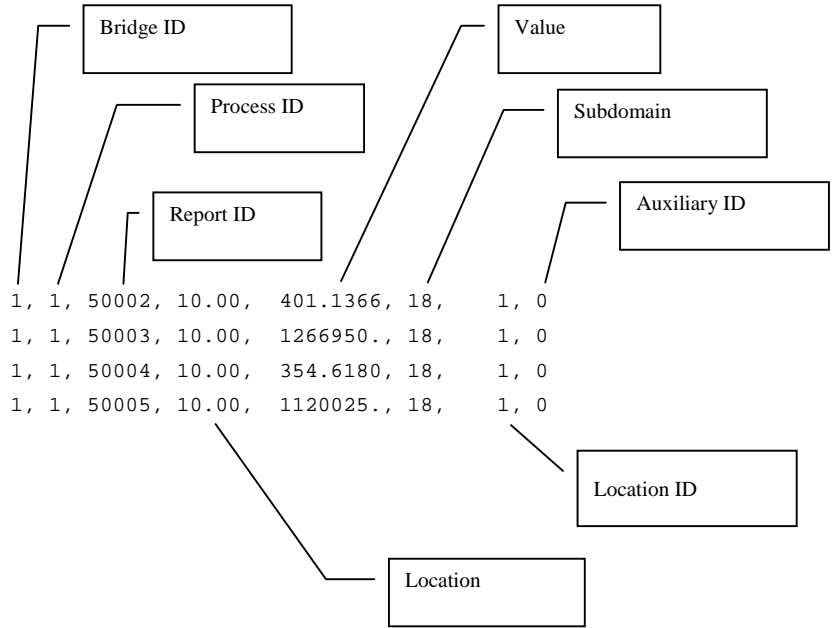


Figure 5. Sample ASCII output.

simple-span girders, a review of Subdomain 2 would be the most beneficial. See Appendix E for a comprehensive list of subdomains.

- **Location ID (FK)**—This field is optional but can be used to define the type of point at “Location” (e.g., a point could be defined as a 20th point, a 10th point, a flange transition, and a draped or harped location). This field permits a detailed-level query for information.
- **Auxiliary ID (FK)**—This field provides an optional “hook” if lower-level tables are necessary. This option is used in the Piers Subdomain (see Appendix E).

A more detailed description of the format is available in Appendix B.

**Examination of Results**

The results are written in a format for import by database management or spreadsheet/graphing software. The database software allows queries and/or sorts. As an example of the process, a macro was developed in the database to import all of the results automatically, after which queries were developed.

**TABLE 2 Process ID table—selected columns**

CP Process ID	Process Name	Version
1	Girder (LRFD)	Version 1, Release 02, Level 01
2	PennDOT PSLRFD	1.4.1
3	PennDOT SLRFD	1.1A
4	Alaska Bulb-Tee (LRFD)	NA
5	BTBeam NCHRP	1.4g
6	Spreadsheet	NCHRP Distribution of Dead Loads
7	Girder (LRFD)	Version 1, Release 04, Level 00, Beta
8	Girder (LRFD)	Version 1, Release 04, Level 01
9	PennDOT PSLRFD	1.5.3c
10	Pier (LRFD)	Alpha 1.0
11	PAPIER	1.0
12	PennDOT SLRFD	1.2E
13	Girder (LRFD)	Version 1, Release 05, Level 00, Alpha
14	Girder (LRFD)	Version 1, Release 05, Level 01, Beta 1

**TABLE 3 Report ID table—selected columns**

Report ID	Description
40022	Dp
40023	Dcp
40024	Dcp
40025	bf/tf
40026	Dc/tw
50002	Initial prestress tendon stress, $f_{pi}$
50003	Initial prestress tendon force, $P_i$
50004	Final prestress tendon stress, $f_{pf}$
50005	Final prestress tendon force, $P_f$
50006	Total Loss, $f_{pT}$

The most basic and powerful query displays the results from one or more software packages for a single Report ID and a single input data set. These results can then be examined manually or graphed for comparison. This graphical comparison is effective, since the differences between software packages, as well as the general trends from one or many programs, then become obvious.

#### *Process for Examination*

Figure 6 is an example of the graphical comparison of the data from two CPs. Note that one CP, PSLRFD, outputs results at 20th points, while the other, BRASS-GIRDER (LRFD) outputs results at the 10th points. Minor differences near both ends of the girderline in the results graph are due to the discretization. The results over the rest of the chart match well, and the results from both CPs follow the same trends. Examining the standard ASCII output from the CPs demonstrates that these differences are not cause for concern. Techniques also have been developed that allow a user to compare two sets of results numerically.

Process 12-50 can be applied even if there is only one CP that produces results for a particular Report ID. See Figure 7

for an example in which results were produced from a single CP. This type of graph can be used to examine output trends for the single program. For example, if the girderline is symmetrical, the results for the stress in the top of the slab should be symmetrical as well. By checking the results from multiple input sets, it can be determined if the results are changing as expected. For example, if the span length increased 5%, but no changes were made in the cross section being examined, the stress in the top of the slab should increase. If this trend is not observed, the reasons for the apparent error can be determined and, if an error exists, corrected.

An additional file is generated for each run containing information about locations for which data are reported. This is the optional point-of-interest (POI) file, which is linked to the results by the POI ID. Appendix B provides a detailed description of the format of this table. This file contains information about each point of interest.

Table 5 describes all of the fields output to the POI file. Once imported into a database, this table can be linked to the Results Table and the user can query for information about a particular point. This can help explain steps in results plots, such as that shown in Figure 7 where steps in the results plot are due to a flange transition in the steel plate girder.

**TABLE 4 Subdomain table**

Subdomain Number	Subdomain Description
1	Dead Load to Girders
2	Live Load Distribution Factors—Simple Span
3	Live Load Distribution Factors—Continuous Span
9	Non-Composite Rolled Steel Sections
10	Non-Composite Steel Plate Girders
11	Non-Composite Built-Up Steel Sections
12	Composite Rolled Steel Sections
13	Composite Steel Plate Girders
14	Composite Built-Up Steel Sections
18	Composite Prestressed I-Sections
19	Composite Prestressed Spread Box Sections
20	Composite Prestressed Adjacent Box Sections

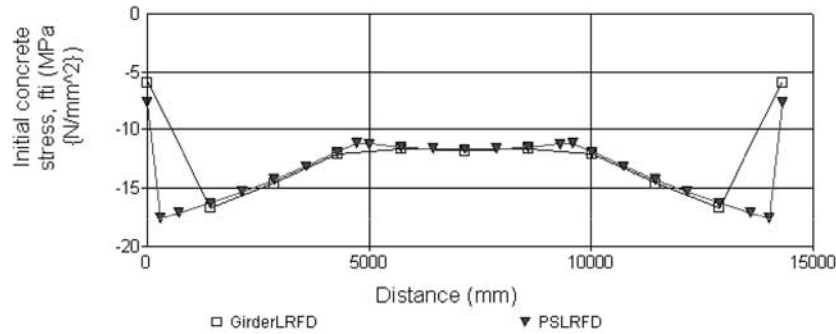


Figure 6. Graphical results comparison.

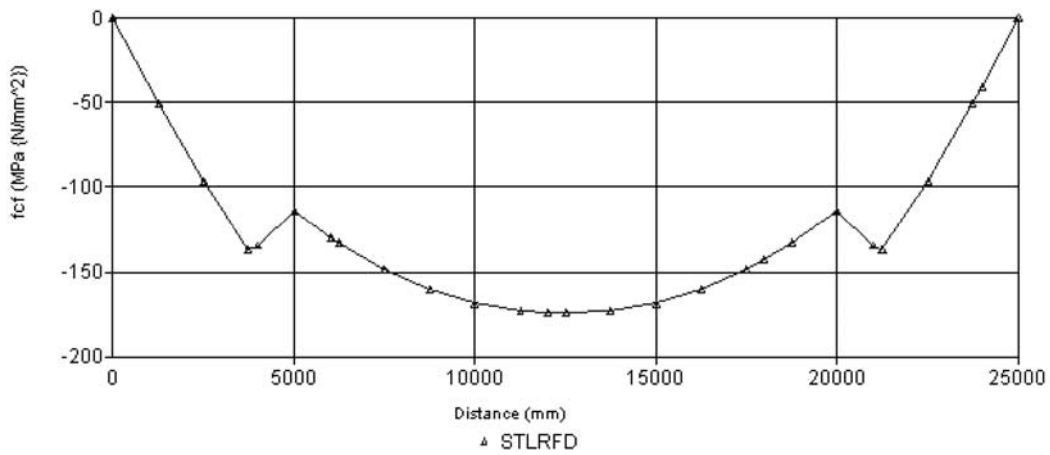


Figure 7. Graphical results for a single process.

TABLE 5 POI table fields

Name	Description	Data Type
LocationID	To match the location ID provided in the results record.	Integer
BridgeID	Same as results record	↓ Boolean ↓
ProcessID	“ “	
SpanNo	Span number for this location	
SpanPercent	Percentage along the span.	
SubDomain	Bridge subdomain	
boolTWENTH	Twentieth point	
boolTENTH	Tenth point	
boolSMPBRG	Simple span bearing location	
boolMIDSPN	Midspan of the simple span	
boolTRANSF	Prestressing transfer length	
boolDRAPE	Drape point location	
boolEXTSUP	Exterior or end support	
boolINTSUP	Interior support	
boolCONCEN	Concentrated load location	
boolTopFlange	Top flange cutoff	
boolBotFlange	Bottom flange cutoff	
boolWeb	Web cutoff	
boolDiaph	Diaphragm location	
boolUser	User defined	



Regression testing is another way to use Process 12-50 for verification when only one program is involved. Regression testing allows a user to compare the results from different versions of the same program. If the results from the latest version do not change as expected, then there is a possible error. Figure 8 shows some erroneous results for the shear resistance of the concrete for a prestressed I-section. The non-symmetric plot for a symmetric girderline indicates that the results are incorrect. The lack of symmetry, in this case, was due to an error in the sign of the shear past midspan. The error was corrected, and the results from the new version of the program are presented in Figure 9. The new version of the program outputs the concrete shear resistance based on the maximum and minimum load factors for two separate live loads (i.e., design truck and design tandem), producing four results for each location. The symmetry of the results from

the corrected program indicates that the problem has been addressed.

Regression testing is also useful when a program has been modified to include a new specification change. The results of the original version of the CP are produced and compared with the results of the new version. The new version of the program is regarded as a separate CP and may be given a different Process ID. By comparing the results, developers can review the results and see the effects of a specification change within the same program. This method can be especially useful for examining new specifications *before* approval or implementation. If the proposed specification is programmed into a CP, parametric studies can be quickly produced that compare the new results with those of the existing specification.

For many subdomains, there are numerous Report IDs that address results that lie beyond the scope of that subdomain.

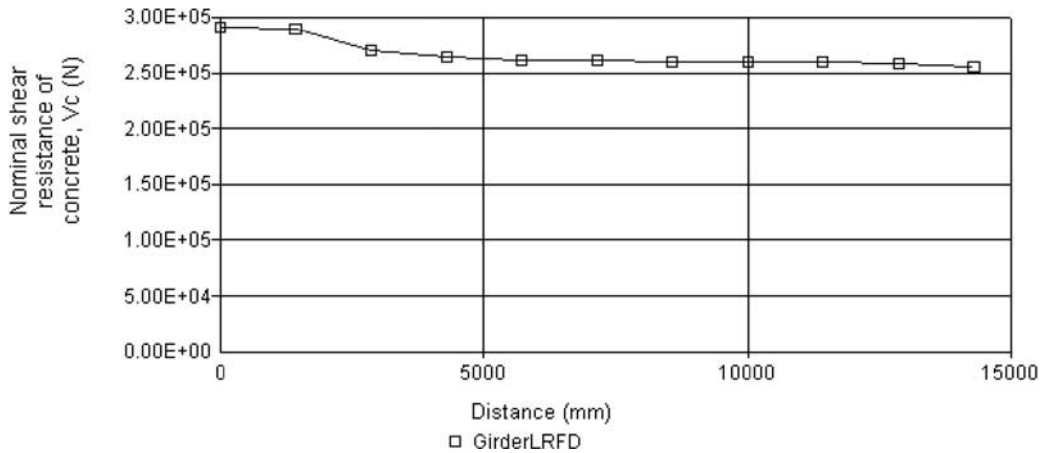


Figure 8. Example of incorrect results seen in a graph.

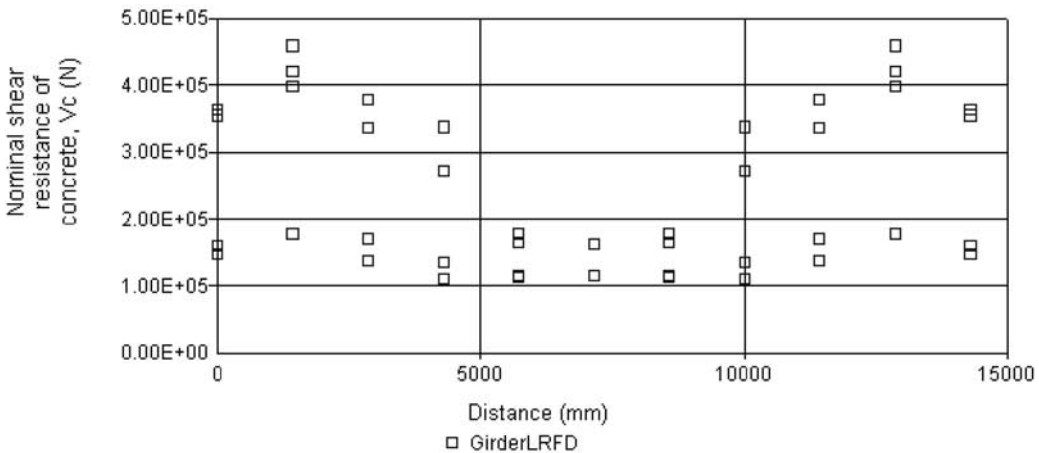


Figure 9. Corrected results.

For instance, for a composite, prestressed I-section girderline, the section properties are reported along with the stress in the tendons, shear resistance, and so forth. In some cases, it may be desirable to remove these “extra” results in order to minimize the amount of data that must be handled. The HL-93 live load actions subdomain is such an example. Many prestress-girderline results are generated along with the live load action, but they do not reflect on the correctness of the CPs involved because input for the programs has not been generated with these Report IDs in mind. Conversely, for more complex subdomains, additional results can be useful in determining the reason for differences between CPs. For example, a difference between the concrete stresses in the prestressed concrete girderline subdomain can be resolved by investigating the applied actions, section properties, or many other results not specifically targeted by the subdomain being studied. The need to remove these additional results is left to the discretion of the implementation engineer.

### Manual Implementation

Figure 10 illustrates the results from the analysis of a prestressed bulb-tee girderline with three independent CPs. This example was developed without the automation features outlined above but using the general process outlined in the overview. Although a bit more cumbersome, the methods outlined were applied successfully.

### Application of the Process—Program Limitations

This section describes examples of program errors or limitations that were exposed during the development of the val-

idation process. Examples of the errors and limitations found are as follows:

- Prestress shear corrections (e.g., concrete, steel, and prestress contributions) (with BRASS);
- Input command corrections (with BRASS);
- PennDOT programs that are not able to input eta factors less than 1.0 (with STLRFD and PSLRFD);
- Differences in tandem vehicle implementation (with BRASS and PennDOT); and
- A difference in handling distribution factor calculations outside the applicability limits (BRASS versus PSLRFD/STLRFD).

### Application of the Process—Confirmation of Program Modifications

This section lists some of the changes made to BRASS verified by researchers during the development of Process 12-50:

- Implementation of tandem train outlined in the LRFD commentary,
- Implementation of a modified/enhanced live load distribution module,
- Implementation of prestressing (P/S) development length, and
- Implementation of initial concrete stresses for P/S.

### Distribution of the Data

Researchers produced a large number of test cases for each test suite during the development of Process 12-50 (see Table 1). A primary task of Phase II of this research was to

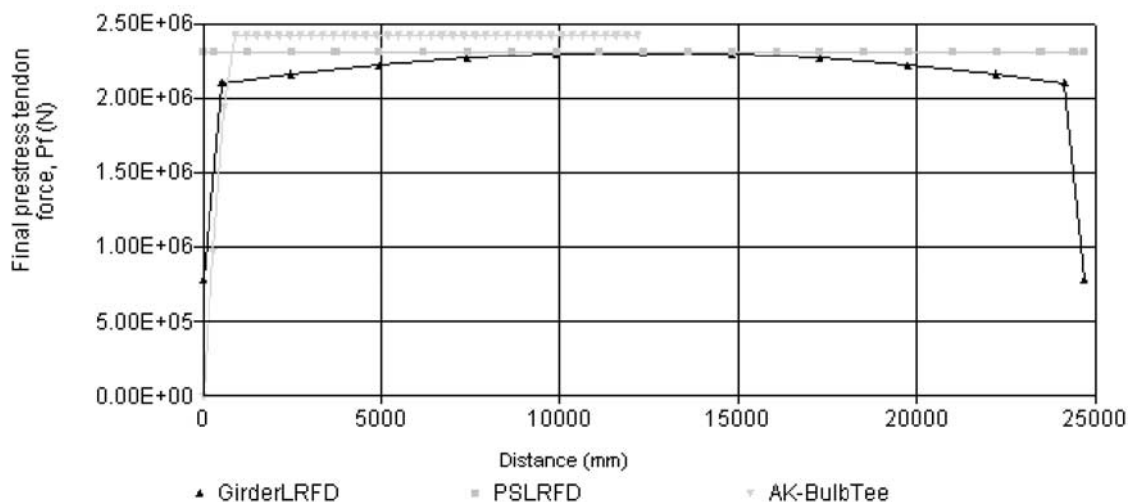


Figure 10. Example of results from manual comparison.

examine the continued use of the test suites and determine the best method for distributing these materials to the bridge community. Process 12-50 final data format and distribution were addressed in terms of target audience and maintenance.

#### *Process 12-50 Data Format*

The project team investigated using extensible markup language (XML) for Process 12-50 input/output format. XML is an ASCII-based format that is being embraced readily by the Internet community for the distribution of a variety of types of data. Using XML for the distribution of Process 12-50 data works well. In addition to the researchers' investigation of XML, a second developer survey and an independent review of the project were also conducted. Suggestions from both the survey and review included the following:

- Use XML for input and output,
- Provide an alternative ASCII format (comma delimited),
- Provide both input and output, and
- Keep the format simple for both input and output.

The result of the investigation is the final XML format used in the accompanying CD-ROMs. Because XML is an ASCII format, the Process 12-50 data files can become rather

large. Tools for compressing XML files are available. A list of XML terminology is provided in the Glossary.

#### *Distribution of Process 12-50 Data*

Although development of a format for the data was important, the distribution and use of Process 12-50 data are critical. The development of Process 12-50, and its implementation for the several subdomains, provides a proof-of-concept for the validation of CPs. The extension of this research is the continued use and maintenance of the developed test suite.

While working with the Project 12-50 panel, the research team developed a plan for the continued maintenance and distribution of Process 12-50 data. Initially, the plan was developed as a series of use cases. The use cases were ultimately expanded to create a functional web-site prototype and a Preliminary Requirements Definition Document for that web site. A list of activities required to distribute the data follows:

- Distribute the data via a web site,
  - Divide the web site into different user categories (e.g., developers, general users, specification writers, and administrators),
  - Facilitate discussion forums, and
  - Distribute tools developed for Process 12-50.
-

## CHAPTER 4

# CONCLUSION AND SUGGESTED RESEARCH

The automation of bridge computations is an integral part of bridge design and rating practice and will continue to evolve. Additionally, migration to the *LRFD Specifications* and their annual revisions cause engineers to modify their CPs and associated computer-based algorithms. The surveys completed throughout the performance of this research project have indicated that most DOTs implemented or planned to implement the *LRFD Specifications* by the Year 2000. Moreover, the 1999 AASHTO vote (by the Subcommittee on Bridges and Structures) to discontinue maintenance of the *Standard Specifications* accelerated this migration. As indicated by the survey, most software developers also support the development of a standardized test suite.

Process 12-50 is simple. The bridge analysis and design process computational domains are divided into smaller subdomains such as dead load distribution, live load distribution, live load actions on a structural component, and complete resistance and specification checks of a structural component. The system within each subdomain is described in parametric form. Here, the inputs and outputs are formally outlined and the analysis assumptions are presented. This parametric form is then used to generate the inputs and outputs (in a common format) from which results from numerous CPs can be compared. CPs may be performed by hand, spreadsheet, or a complete program dedicated to bridge engineering.

In summary, Process 12-50 is general, independent of implementation or software, and supports the formal comparison of CPs derived from a host of sources. The results generated by Process 12-50 have numerous applications, one of which is the development of a test suite. Additionally, Process 12-50 provides the formalization of an output format for bridge engineering, which is especially important to the bridge-design and end-user communities.

### LIMITATIONS

The scope of this project has been limited to computations associated with steel and prestressed concrete girder superstructures and piers. The pier work was limited to a proof of concept with a few pier substructures. Although the project was restricted to these types of structural systems, Process 12-50 is not limited to these systems. Generally, Process 12-50 can be applied to other computations associated with, but not limited to, various types of superstructures, substructures, founda-

tions, abutments, culverts, and bridge ratings. Other applications of Process 12-50 are described in the following section.

### ADDITIONAL TESTING POSSIBILITIES

#### Design Comparisons

Using Process 12-50, a designer can easily compare the results of alternative CPs. The CP results can be imported into a common viewer for comparison so that differences are apparent. In the past, validation typically has consisted of producing a hand calculation that was computed and compared with results of the program. One typical hand calculation could take anywhere from several days to several weeks to complete, with the product being a set of results representing a single bridge structure. Using Process 12-50, dozens of calculations have been generated and compared in the same period of time required to produce only one hand calculation.

In addition to the calculation time saved by using Process 12-50 instead of performing hand calculations, there is also a maintenance savings that is realized. Design examples produced by hand calculations are almost as costly to maintain as they are to produce. Changes in specifications or office practice procedures often lead to major, time-consuming revisions to hand computations. Process 12-50 simply requires that the automated processes be re-executed.

#### Specification Review

Specification writing committees can benefit from the efficiency of Process 12-50 by being able to compare different versions of a large set of bridges to determine whether or not changes accomplish the desired objectives and to prevent problems from arising because of the changes. Similarly, bridge engineers can see how their changes may affect current engineering practices.

#### Software Validation

The cost of rigorous validation of software is often 25% or more of the cost of development for that same software. Validation costs are higher for engineering programs that involve hand-based computations. The cost associated with validating improperly functioning software leading to unser-

viceability or collapse can be much more than 25% of development costs. Process 12-50 may be used by software developers for verification and regression testing (version testing) of their software and to reduce verification times and costs.

## SUGGESTED RESEARCH

Some of the many possibilities for future research are outlined in this section. Implementing these suggestions will broaden the effectiveness of the test suites developed as an illustration of Process 12-50.

### Short-Term Research/Action Items

Suggested action items and research activities follow:

- Maintain the Report ID list provided in Table E-2 and the other common tables (Process ID Table, Specification Version Table, Specification Article Table, Subdomain Report ID/Specification Reference Table, and Unit ID Table). These tables are all provided in a relational database on the accompanying CD-ROMs (see `CommonTables.mdb`).
- Develop and offer short courses that train/educate development engineers in the benefits, philosophy, approach, and implementation of Process 12-50. Learners should leave the course with an initial implementation in their program(s). Next, developers should evaluate, critique, and make recommendations for improving Process 12-50 based on the findings of the pilot project(s).
- Have software developers develop one or more pilot projects using Process 12-50.
- Promote the use and expansion of Process 12-50 by requiring that software testing be “12-50 compliant” for award of software development and maintenance contracts. Compliancy would require using Process 12-50 but would not mean matching existing Process 12-50 results.
- Promote the use of Process 12-50 in the development of AASHTO specifications. This can be accomplished by including Process 12-50 in NCHRP research as often as possible. Process 12-50 is ideal for regression testing,

allowing an existing CP (with the existing specifications) to be compared with a revised CP (with the proposed specifications). Thoroughly testing specifications before release will help prevent problems after release. Moreover, Process 12-50 is also a valuable tool during the research phase prior to developing specification language.

- Continue to promote Process 12-50 for third-party and in-house software development.
- Report the benefits of Process 12-50 at the annual meeting of AASHTO bridge engineers. This report could describe how Process 12-50 is being used in NCHRP research.

### Long-Term Research

Expand Process 12-50 to include test suites in the following areas:

- Structures made of materials other than concrete and steel (e.g., timber and composites),
- Different types of superstructures (e.g., slabs, box culverts, and post-tensioned systems),
- Rating of existing structures (e.g., standard and new LRFR), and
- Additional substructures and foundations.

### *Administration*

The publication/adoption of a test suite should be an AASHTO activity. The Technical Committee of the AASHTO Subcommittee on Bridges is an appropriate group to administer the process. The extent and intent of adoption should be addressed by AASHTO. It is likely that the long-term support and maintenance would be an AASHTO function as well. In this work, the research has focused on the technical issues and has not explicitly addressed the administrative, political, or other business aspects of Process 12-50 use within AASHTO or its member states. As previously shown, Process 12-50 can be used in many ways for the benefit of the bridge-engineering community, software development, and—most importantly—the traveling public.

---

## GLOSSARY

**AASHTOWare**—software that is supported and maintained by AASHTO

**Actor**—someone who uses the test suite or web site

**Administrator**—actor controlling the content and functionality of the web site

**Architectural and Engineering XML**—a consensus-building organization that promotes the development of XML standards for data interchange; see <http://xml.coverpages.org/XML.html>

**Bridge ID**—a unique numerical value used to reference (tag) a specific bridge

**BridgeWare (AASHTO)**—bridge engineering software supported and maintained by AASHTO

**Comma-delimited text file**—a file containing ASCII data in a format similar to a database table as defined below; for the tools developed as a demonstration of Process 12-50, the tables in the databases are usually created by importing a comma-delimited text file in which each field is separated by a comma

**Computational domain**—the set of all values produced by bridge computational processes

**Computational Process (CP)**—a unique method of computation that can be software, hand calculations, or a set of examples from a published source (e.g., specific version of a computer program, a set of hand calculations, a spreadsheet, or a set of known results such as Portland Concrete Institute (PCI) example problems); the software can be divided into multiple CPs to reflect different versions

**Data**—herein, refers to results of Process 12-50; may be used in the context of a set of data, and therefore appears as either singular or plural, depending on context

**Data mining**—the semi-automatic discovery of patterns, changes, associations, and anomalies in large data sets

**Document Object Model (DOM)**—an object model that exposes the contents of an XML document; see W3C's *Document Object Model (DOM) Level 1 Specification* for definitions of what a DOM should expose as properties, methods, and events

**Document Type Definition (DTD)**—the markup declarations that describe the syntax or grammar for an XML document; a set of rules defined for a particular type of XML document; this type of definition seems to be losing favor with the W3C committee in favor of *schema* (see below)

**Element**—XML structural construct that consists of a start tag, an end tag, and the information between the tags, which is often referred to

as the contents; elements used in an XML file are described by a DTD or schema, either of which can provide a description of the structure of the data

**Field**—one column in a database table or in a comma-delimited text file

**Maintenance**—actor to update, maintain, and operate the web site

**NCHRP 12-50 procedure**—referred to as Process 12-50 in order to avoid confusion; includes the method for defining bridge calculation input and output in a standard format for the purposes of comparing it with existing like-bridge data

**Oversight Committee**—a group of engineers and possibly other technical professionals who will provide administrative review of the site operations, maintenance, and policies; is likely associated with the AASHTO Subcommittee on Bridges and Structures

**Problem domain**—a set of problems (bridge examples) associated with an engineering computational process that exercises the limits of that process

**Process 12-50**—the procedure for comparing the results of two computational processes (i.e., the process developed under the NCHRP12-50 research)

**Process ID**—a unique numerical value used to reference a computational process that created a result; unique Process IDs will be provided for different versions of the same computer program

**Report ID**—a unique numerical value used to reference (tag) a single computational result, which may be from analysis, resistance computations, loads, or a specification value, and is a potential output item from one or more processes as defined above

**Schema**—a formal specification of element names that indicates which elements are allowed in an XML document, and in which combinations; functionally equivalent to a DTD but is written in XML; also provides for extended functionality such as data typing, inheritance, and presentation rules

**Subdomain**—a subset of the entire test suite, so subdivided to create a more manageable set of data; the subdomain contains a set of data in the test suite that represents a specific area of the specifications (e.g., distribution factor subdomain, dead load, HL-93 load effects and stresses)

**Table**—this is a database entity used to store data in a set structure with a number of fields, each representing a certain item of interest; the fields in a table are often related to each other but have different meanings and a table and a comma-delimited text file are generally the same in structure

Tag—used to define elements in XML; a start tag, such as <abc> and a corresponding end tag, such as </abc>, are used to mark the beginning and end of data information

Technical—actor using the web site to obtain information

Test suite—a test suite is composed of a documented set of sub-domains and all of the input/output data that they contain

Use case—a narrative document that describes the sequence of events of an actor (an external agent) using a system to complete a process

Valid XML document—XML that is well formed and conforms to the vocabulary specified in a DTD or schema

W3C (World Wide Web Consortium)—international consortium founded in 1994 to develop standards for the Web

Well-formed XML document—XML that meets the requirements listed in W3C's *Recommendation for XML 1.0* because it contains one or more elements, has a single document element with any other elements properly nested under it, contains parsed entities referenced directly or indirectly within the document that are also each well formed; does not necessarily include a DTD

---