Contents

Introduction **1**
What is a Web system? 1
Contributions of the Thesis 2
Part I – Static Web Verification 3
Part II – Dynamic Web Verification 6

Preliminaries **9**
Term Rewriting Systems 10
Rewrite Theories 11

I Static Web Verification **13**

1 Rewriting-based Web Verification **15**
1.1 Web Site Description 15
1.2 Web Specification Language 17
1.3 Homeomorphic Embedding and Partial Rewriting 20
1.4 Error Diagnoses 22

2 Semi-Automatic Repairing of Web Sites **27**
2.1 Repairing Faulty Web Sites 27
2.1.1 Fixing Correctness Errors 28
2.1.2 Fixing Completeness Errors 32
2.2 Automatic Error Repair 37
2.2.1 Incompatibility of Conditions 40
2.3 Related Work 40

3 Optimization Strategies for Repairing Web Sites **43**
3.1 Fixing Web Sites by Using Correction Strategies 43
3.1.1 Correctness Error Dependencies 44
3.1.2 Correction Strategies 48
3.2 Fixing Web Sites by Using Completeness Strategies . . . 55
3.2.1 Completeness Error Dependencies 56

3.2.2 Completion Strategies 59

4 The Web Verification Service WebVerdi-M 67
4.1 Web Site Verification Using Maude 67
4.1.1 Homeomorphic Embedding Implementation . . . 69
4.2 Prototype Implementation 70
4.2.1 The WebVerdiService 71
4.2.2 The XML API 72
4.2.3 The Verdi-M Engine 72
4.2.4 The WebVerdiClient 72
4.2.5 The Database 74
4.3 The API 74
4.3.1 Data Representation 74
4.3.2 Methods Exported 79
4.4 Experimental Evaluation 81

5 An Abstract Generic Framework for Web Verification 83
5.1 Introduction 83
5.1.1 Web Site Description 84
5.2 Abstract Web Site Verification 85
5.2.1 Abstract Web Specification 88
5.2.2 Abstract Web Site 89
5.2.3 Abstract Verification Soundness 92
5.3 Implementation 101
5.4 Related Work 102

II Dynamic Web Verification 103

6 Specification and Verification of Web Applications in RL 105
6.1 A Navigation Model for Web Applications 108
6.1.1 Graphical Navigation Model 108
6.2 Formalizing the Navigation Model as a Rewrite Theory . 110
6.2.1 The Web Scripting Language 110
6.2.2 The Web Application Structure 111
6.2.3 The Communication Protocol 113
6.3 Modeling Multiple Web Interactions and Browser Features 117
6.3.1 The Extended Equational Theory $(\mathbf{\Sigma_{ext}}, \mathbf{E_{ext}})$ . . 117

6.3.2 The Extended Rewrite Rule Set $\mathbf{R_{ext}}$ 119
6.4 Model Checking Web Applications Using LTLR 122
6.4.1 The Linear Temporal Logic of Rewriting 123
6.4.2 LTLR properties for Web Applications 124
6.5 Related Work 128

7 Backward Trace Slicing for Rewriting Logic Theories 129
7.1 Introduction 129
7.2 Rewriting Modulo Equational Theories 131
7.3 Labeling and Tracing in Term Rewrite Systems 133
7.4 Backward Trace Slicing for Elementary Rewrite Theories 135
7.4.1 Labeling Procedure for Rewrite Theories 136
7.4.2 The Backward Trace Slicing Algorithm 139
7.5 Backward Trace Slicing for Extended Rewrite Theories . 146
7.5.1 Dealing with Collapsing and Nonleft-linear Rules 146
7.5.2 Built-in Operators 148
7.5.3 Associative-Commutative Axioms 149
7.5.4 Extended Soundness 151
7.6 Experimental Evaluation 156
7.7 Related Work 158
7.8 Conclusions 160

8 Model-checking Web Applications with Web-TLR 163
8.1 The Web-TLR System 164
8.2 A Case Study in Web Verification 165
8.3 Web-TLR Graphical Web interface 168
8.4 Debugging of Web Applications 174
8.4.1 Filtering Notation 174
8.4.2 A Debugging Session with Web-TLR 176

Conclusions 183
Static Web Verification 183
Dynamic Web Verification 185
Future Work 186

Bibliography 189

A Operational Semantics of the Web Scripting Language 203

B Specification of the Evaluation Protocol Function **207**

C Example: The War of Souls **209**
The Backward Trace Slicing Technique in Action 210
Trace Slice Analysis 214

D Web-TLR$^{\text{mini}}$ **217**

List of Figures

1.1 A Web page and its corresponding encoding as a ground term . . . 15
1.2 An example of a Web site W for a research group 16

3.1 Taxonomy of error dependencies. 46
3.2 The $\mathbb{MNO}$ strategy 54

4.1 Components of WebVerdi-M 71
4.2 WebVerdiClient Snapshot 73
4.3 XML encoding of a Web site 75
4.4 XML representation for a Web specification 77
4.5 XML error representation 78
4.6 Completeness error representation 78
4.7 XML action representation 79

5.1 Web site and Web specification for an on-line auction system. 85
5.2 Abstract embedding 95

6.1 The navigation model of a Webmail application. 109

7.1 A term slice and one possible concretization. 141

8.1 The navigation model of an Electronic Forum 165
8.2 Specification of the electronic forum application in Web-TLR 167
8.3 One Web state of the counter-example trace of Section 8.4.2. 168
8.4 Maude specification of the navigation model 170
8.5 Maude specification of the curPage state predicate 170
8.6 Electronic Forum Application in Web-TLR 172
8.7 Slideshow of the Web-TLR execution 173
8.8 Trace slice $\mathcal{T}^{\bullet}$. 178
8.9 Snapshot of the Web-TLR System. 181
8.10 Snapshot of the Web-TLR System for the case of no counter-examples. 181

C.1 Maude specification of the *WoS* game. 210
C.2 Trace given by the `frew` command of Maude. 211
C.3 Labeling and sequence of relevant position sets. In order to facilitate the understanding, the terms involved in each step are underlined. 213
C.4 The execution trace $\mathcal{T}$ and its corresponding trace slice $\mathcal{T}^{\bullet}$. 214