Contents

1. Introduction 1
  1.1 Motivation 1
  1.2 Distributed Programming Abstractions 3
    1.2.1 Inherent Distribution 4
    1.2.2 Distribution as an Artifact 6
  1.3 The End-to-End Argument 7
  1.4 Software Components 8
    1.4.1 Composition Model 8
    1.4.2 Programming Interface 10
    1.4.3 Modules 11
    1.4.4 Classes of Algorithms 13
  1.5 Hands-On 15
    1.5.1 Print Module 16
    1.5.2 BoundedPrint Module 18
    1.5.3 Composing Modules 20

2. Basic Abstractions 25
  2.1 Distributed Computation 26
    2.1.1 Processes and Messages 26
    2.1.2 Automata and Steps 26
    2.1.3 Liveness and Safety 28
  2.2 Abstracting Processes 29
    2.2.1 Process Failures 29
    2.2.2 Arbitrary Faults and Omissions 30
    2.2.3 Crashes 30
    2.2.4 Recoveries 32
  2.3 Abstracting Communication 34
    2.3.1 Link Failures 35
    2.3.2 Fair-Loss Links 36
    2.3.3 Stubborn Links 36
    2.3.4 Perfect Links 38
    2.3.5 Logged Perfect Links 40
    2.3.6 On the Link Abstractions 41
  2.4 Timing Assumptions 43

2.4.1 Asynchronous System 43
2.4.2 Synchronous System 45
2.4.3 Partial Synchrony 46
2.5 Abstracting Time 47
2.5.1 Failure Detection 47
2.5.2 Perfect Failure Detection 48
2.5.3 Leader Election 50
2.5.4 Eventually Perfect Failure Detection 51
2.5.5 Eventual Leader Election 54
2.6 Distributed System Models 58
2.6.1 Combining Abstractions 58
2.6.2 Measuring Performance 59
2.7 Hands-On 60
2.7.1 Sendable Event 60
2.7.2 Message and Extended Message 61
2.7.3 Fair-Loss Point-to-Point Links 62
2.7.4 Perfect Point-to-Point Links 62
2.7.5 Perfect Failure Detector 63
2.8 Exercises 64
2.9 Solutions 65
2.10 Historical Notes 67

3. Reliable Broadcast 69
3.1 Motivation 69
3.1.1 Client-Server Computing 69
3.1.2 Multi-participant Systems 70
3.2 Best-Effort Broadcast 71
3.2.1 Specification 71
3.2.2 Fail-Silent Algorithm: Basic Broadcast 71
3.3 Regular Reliable Broadcast 72
3.3.1 Specification 73
3.3.2 Fail-Stop Algorithm: Lazy Reliable Broadcast 73
3.3.3 Fail-Silent Algorithm: Eager Reliable Broadcast 74
3.4 Uniform Reliable Broadcast 76
3.4.1 Specification 77
3.4.2 Fail-Stop Algorithm: All-Ack Uniform Reliable Broadcast 78
3.4.3 Fail-Silent Algorithm: Majority-Ack Uniform Reliable Broadcast 79
3.5 Stubborn Broadcast 81
3.5.1 Overview 81
3.5.2 Specification 81
3.5.3 Fail-Recovery Algorithm: Basic Stubborn Broadcast 82
3.6 Logged Best-Effort Broadcast 83
3.6.1 Specification 83

3.6.2 Fail-Recovery Algorithm: Logged Basic Broadcast 83
3.7 Logged Uniform Reliable Broadcast 84
3.7.1 Specification 85
3.7.2 Fail-Recovery Algorithm: Logged Majority-Ack URB .. 86
3.8 Randomized Broadcast 86
3.8.1 The Scalability of Reliable Broadcast 87
3.8.2 Epidemic Dissemination 88
3.8.3 Specification 88
3.8.4 Randomized Algorithm: Eager Probabilistic Broadcast 89
3.8.5 Randomized Algorithm: Lazy Probabilistic Broadcast . 91
3.9 Causal Broadcast 94
3.9.1 Overview 94
3.9.2 Specifications 94
3.9.3 Fail-Silent Algorithm: No-Waiting Causal Broadcast .. 96
3.9.4 Fail-Stop Extension: Garbage Collecting the Causal Past 98
3.9.5 Fail-Silent Algorithm: Waiting Causal Broadcast 98
3.10 Hands-On 101
3.10.1 Basic Broadcast 101
3.10.2 Lazy Reliable Broadcast 103
3.10.3 All-Ack Uniform Reliable Broadcast 106
3.10.4 Majority-Ack URB 108
3.10.5 Probabilistic Reliable Broadcast 109
3.10.6 No-Waiting Causal Broadcast 112
3.10.7 No-Waiting Causal Broadcast with Garbage Collection 116
3.10.8 Waiting Causal Broadcast 122
3.11 Exercises 125
3.12 Solutions 127
3.13 Historical Notes 133

4. Shared Memory 135
4.1 Introduction 135
4.1.1 Sharing Information in a Distributed System 135
4.1.2 Register Overview 136
4.1.3 Completeness and Precedence 139
4.2 $(1, N)$ Regular Register 140
4.2.1 Specification 140
4.2.2 Fail-Stop Algorithm: Read-One Write-All Regular Register 140
4.2.3 Fail-Silent Algorithm: Majority Voting Regular Register 143
4.3 $(1, N)$ Atomic Register 146
4.3.1 Specification 146
4.3.2 Transformation: From $(1, N)$ Regular to $(1, N)$ Atomic 149

4.3.3 Fail-Stop Algorithm: Read-Impose Write-All $(1, N)$ Atomic Register 153
4.3.4 Fail-Silent Algorithm: Read-Impose Write-Majority $(1, N)$ Atomic Register .. 155
4.4 (N, N) Atomic Register 157
4.4.1 Multiple Writers 157
4.4.2 Specification 158
4.4.3 Transformation: From $(1, N)$ Atomic to (N, N) Atomic Registers 159
4.4.4 Fail-Stop Algorithm: Read-Impose Write-Consult (N, N) Atomic Register .. 162
4.4.5 Fail-Silent Algorithm: Read-Impose Write-Consult-Majority (N, N) Atomic Register 162
4.5 $(1, N)$ Logged Regular Register 164
4.5.1 Precedence in the Fail-Recovery Model 165
4.5.2 Specification 166
4.5.3 Fail-Recovery Algorithm: Logged-Majority-Voting 167
4.6 Hands-On 171
4.6.1 $(1, N)$ Regular Register 171
4.6.2 $(1, N)$ Atomic Register 174
4.6.3 (N, N) Atomic Register 178
4.7 Exercises 181
4.8 Solutions 182
4.9 Historical Notes 187

5. Consensus 189
5.1 Regular Consensus 189
5.1.1 Specification 189
5.1.2 Fail-Stop Algorithm: Flooding Consensus 190
5.1.3 Fail-Stop Algorithm: Hierarchical Consensus 193
5.2 Uniform Consensus 195
5.2.1 Specification 195
5.2.2 Fail-Stop Algorithm: Flooding Uniform Consensus 196
5.2.3 Fail-Stop Algorithm: Hierarchical Uniform Consensus . 197
5.3 Abortable Consensus 199
5.3.1 Overview 199
5.3.2 Specification 200
5.3.3 Fail-Silent Algorithm: RW Abortable Consensus 201
5.3.4 Fail-Noisy Algorithm: From Abortable Consensus to Consensus 204
5.4 Logged Abortable Consensus and Logged Consensus 206
5.4.1 Fail-Recovery Algorithm: Logged Abortable Consensus 206
5.5 Randomized Consensus 208
5.5.1 Specification 208

    - 5.5.2 Randomized Algorithm: Probabilistic Consensus 209
  - 5.6 Hands-On 212
    - 5.6.1 Flooding Regular Consensus Protocol 212
    - 5.6.2 Hierarchical Regular Consensus Protocol 216
    - 5.6.3 Flooding Uniform Consensus 219
    - 5.6.4 Hierarchical Uniform Consensus 222
  - 5.7 Exercises 225
  - 5.8 Solutions 226
  - 5.9 Historical Notes 232

- **6. Consensus Variants** 233
  - 6.1 Total Order Broadcast 233
    - 6.1.1 Overview 233
    - 6.1.2 Specifications 234
    - 6.1.3 Algorithm: Consensus-Based Total Order Broadcast 236
  - 6.2 Terminating Reliable Broadcast 239
    - 6.2.1 Overview 239
    - 6.2.2 Specification 240
    - 6.2.3 Algorithm: Consensus-Based TRB 240
  - 6.3 Non-blocking Atomic Commit 242
    - 6.3.1 Overview 242
    - 6.3.2 Specification 243
    - 6.3.3 Algorithm: Consensus-Based NBAC 244
  - 6.4 Group Membership 246
    - 6.4.1 Overview 246
    - 6.4.2 Specification 247
    - 6.4.3 Algorithm: Consensus-Based Group Membership 248
  - 6.5 View Synchronous Communication 249
    - 6.5.1 Overview 249
    - 6.5.2 Specification 250
    - 6.5.3 Algorithm: TRB-Based View Synchronous Broadcast 251
    - 6.5.4 Algorithm: Consensus-Based Uniform View Synchronous Broadcast 255
  - 6.6 Hands-On 258
    - 6.6.1 Uniform Total Order Broadcast 258
    - 6.6.2 Consensus-Based Non-blocking Atomic Commit 263
    - 6.6.3 Consensus-Based Group Membership 266
    - 6.6.4 TRB-Based View Synchrony 269
  - 6.7 Exercices 275
  - 6.8 Solutions 276
  - 6.9 Historical Notes 285

- **7. Concluding Remarks** 287
  - 7.1 Further Implementations 287
  - 7.2 Further Readings 289

Bibliography 296

Index 297