

---

# Contents

---

Preface, xxv

Acknowledgments, xxvii

Author, xxix

## SECTION I **Background Materials**

CHAPTER 1 ■ Introduction	3
1.1 WHAT IS A DISTRIBUTED SYSTEM?	3
1.2 WHY DISTRIBUTED SYSTEMS	4
1.3 EXAMPLES OF DISTRIBUTED SYSTEMS	5
1.4 IMPORTANT ISSUES IN DISTRIBUTED SYSTEMS	8
1.5 COMMON SUBPROBLEMS	10
1.6 IMPLEMENTING A DISTRIBUTED SYSTEM	11
1.7 PARALLEL VERSUS DISTRIBUTED SYSTEMS	12
1.8 BIBLIOGRAPHIC NOTES	13
EXERCISES	13
CHAPTER 2 ■ Interprocess Communication: An Overview	15
2.1 INTRODUCTION	15
2.1.1 Processes and Threads	15
2.1.2 Client–Server Model	16
2.1.3 Middleware	16
2.2 NETWORK PROTOCOLS	17
2.2.1 Ethernet	17
2.2.2 Wireless Networks	18
2.2.3 OSI Model	20
2.2.4 IP	22
2.2.5 Transport Layer Protocols	23
2.2.6 Interprocess Communication Using Sockets	24

2.3	NAMING	26
2.3.1	Domain Name Service	28
2.3.2	Naming Service for Mobile Clients	29
2.4	REMOTE PROCEDURE CALL	29
2.4.1	Implementing RPC	30
2.4.2	Sun ONC/RPC	31
2.5	REMOTE METHOD INVOCATION	32
2.6	MESSAGES	33
2.6.1	Transient and Persistent Messages	33
2.6.2	Streams	34
2.7	WEB SERVICES	34
2.8	EVENT NOTIFICATION	35
2.9	VIRTUALIZATION: CLOUD COMPUTING	35
2.9.1	Classification of Cloud Services	36
2.9.2	MapReduce	37
2.9.3	Hadoop	39
2.10	MOBILE AGENTS	40
2.11	BASIC GROUP COMMUNICATION SERVICES	40
2.12	CONCLUDING REMARKS	41
2.13	BIBLIOGRAPHIC NOTES	41
	EXERCISES	42

## SECTION II **Foundational Topics**

CHAPTER 3	■ Models for Communication	45
3.1	NEED FOR A MODEL	45
3.2	MESSAGE-PASSING MODEL FOR INTERPROCESS COMMUNICATION	45
3.2.1	Process Actions	45
3.2.2	Channels	46
3.2.3	Synchronous versus Asynchronous Systems	48
3.2.4	Real-Time Systems	49
3.3	SHARED VARIABLES	50
3.3.1	Linda	51
3.4	MODELING MOBILE AGENTS	52
3.5	RELATIONSHIP AMONG MODELS	53

3.5.1	Strong and Weak Models	53
3.5.2	Implementing a FIFO Channel Using a Non-FIFO Channel	54
3.5.3	Implementing Message Passing Using Shared Memory	56
3.5.4	Implementing Shared Memory Using Message Passing	56
3.5.5	Impossibility Result with Channels	58
3.6	CLASSIFICATION BASED ON SPECIAL PROPERTIES	58
3.6.1	Reactive versus Transformational Systems	58
3.6.2	Named versus Anonymous Systems	59
3.7	COMPLEXITY MEASURES	59
3.8	CONCLUDING REMARKS	63
3.9	BIBLIOGRAPHIC NOTES	63
	EXERCISES	64
<b>CHAPTER 4 ■ Representing Distributed Algorithms: Syntax and Semantics</b>		<b>67</b>
4.1	INTRODUCTION	67
4.2	GUARDED ACTIONS	67
4.3	NONDETERMINISM	70
4.4	ATOMIC OPERATIONS	71
4.5	FAIRNESS	73
4.6	CENTRAL VERSUS DISTRIBUTED SCHEDULERS	75
4.7	CONCLUDING REMARKS	78
4.8	BIBLIOGRAPHIC NOTES	78
	EXERCISES	79
<b>CHAPTER 5 ■ Program Correctness</b>		<b>83</b>
5.1	INTRODUCTION	83
5.2	CORRECTNESS CRITERIA	84
5.2.1	Safety Properties	84
5.2.2	Liveness Properties	86
5.3	CORRECTNESS PROOFS	89
5.3.1	Quick Review of Propositional Logic	90
5.3.2	Brief Overview of Predicate Logic	91
5.4	ASSERTIONAL REASONING: PROVING SAFETY PROPERTIES	92
5.5	PROVING LIVENESS PROPERTIES USING WELL-FOUNDED SETS	93
5.6	PROGRAMMING LOGIC	96
5.7	PREDICATE TRANSFORMERS	99

5.8	CONCLUDING REMARKS	101
5.9	BIBLIOGRAPHIC NOTES	103
	EXERCISES	103
<b>CHAPTER 6 ■ Time in a Distributed System</b>		<b>107</b>
6.1	INTRODUCTION	107
6.1.1	Physical Time	107
6.1.2	Sequential and Concurrent Events	108
6.2	LOGICAL CLOCKS	109
6.3	VECTOR CLOCKS	112
6.4	PHYSICAL CLOCK SYNCHRONIZATION	113
6.4.1	Preliminary Definitions	113
6.4.2	Clock Reading Error	116
6.4.3	Algorithms for Internal Synchronization	116
6.4.4	Algorithms for External Synchronization	118
6.5	CONCLUDING REMARKS	122
6.6	BIBLIOGRAPHIC NOTES	122
	EXERCISES	122
<b>SECTION III Important Paradigms</b>		
<b>CHAPTER 7 ■ Mutual Exclusion</b>		<b>129</b>
7.1	INTRODUCTION	129
7.2	SOLUTIONS ON MESSAGE-PASSING SYSTEMS	129
7.2.1	Lamport's Solution	130
7.2.2	Ricart-Agrawala's Solution	132
7.2.3	Maekawa's Solution	133
7.3	TOKEN-PASSING ALGORITHMS	137
7.3.1	Suzuki-Kasami Algorithm	137
7.3.2	Raymond's Algorithm	138
7.4	SOLUTIONS ON THE SHARED-MEMORY MODEL	139
7.4.1	Peterson's Algorithm	139
7.5	MUTUAL EXCLUSION USING SPECIAL INSTRUCTIONS	142
7.5.1	Solution Using Test-and-Set	142
7.5.2	Solution Using Load-Linked and Store-Conditional	143
7.6	GROUP MUTUAL EXCLUSION	144

7.7	CONCLUDING REMARKS	146
7.8	BIBLIOGRAPHIC NOTES	147
	EXERCISES	148
<b>CHAPTER 8 ■ Distributed Snapshot</b>		<b>153</b>
8.1	INTRODUCTION	153
8.2	PROPERTIES OF CONSISTENT SNAPSHOTS	155
	8.2.1 Cuts and Consistent Cuts	155
8.3	CHANDY–LAMPORT ALGORITHM	156
	8.3.1 Two Examples	159
	8.3.1.1 <i>Example 1: Counting of Tokens</i>	159
	8.3.1.2 <i>Example 2: Communicating State Machines</i>	160
8.4	LAI–YANG ALGORITHM	162
8.5	DISTRIBUTED DEBUGGING	163
	8.5.1 Constructing the State Lattice	164
	8.5.2 Evaluating Predicates	164
8.6	CONCLUDING REMARKS	165
8.7	BIBLIOGRAPHIC NOTES	165
	EXERCISES	165
<b>CHAPTER 9 ■ Global State Collection</b>		<b>169</b>
9.1	INTRODUCTION	169
9.2	ELEMENTARY ALGORITHM FOR ALL-TO-ALL BROADCASTING	170
9.3	TERMINATION-DETECTION ALGORITHMS	172
	9.3.1 Dijkstra–Scholten Algorithm	173
	9.3.2 Termination Detection on a Unidirectional Ring	176
	9.3.3 Credit-Recovery Algorithm for Termination Detection	179
9.4	WAVE ALGORITHMS	180
	9.4.1 Propagation of Information with Feedback	181
9.5	DISTRIBUTED DEADLOCK DETECTION	182
	9.5.1 Resource Deadlock and Communication Deadlock	182
	9.5.2 Detection of Resource Deadlock	184
	9.5.3 Detection of Communication Deadlock	185
9.6	CONCLUDING REMARKS	187
9.7	BIBLIOGRAPHIC NOTES	187
	EXERCISES	188

CHAPTER 10 ■ Graph Algorithms	189
10.1 INTRODUCTION	189
10.2 ROUTING ALGORITHMS	190
10.2.1 Computation of Shortest Path	190
10.2.1.1 Complexity Analysis	192
10.2.1.2 Chandy–Misra Modification of the Shortest Path Algorithm	193
10.2.2 Distance-Vector Routing	194
10.2.3 Link-State Routing	195
10.2.4 Interval Routing	197
10.2.4.1 Interval Routing Rule	198
10.2.5 Prefix Routing	200
10.3 GRAPH TRAVERSAL	201
10.3.1 Spanning Tree Construction	202
10.3.2 Tarry’s Graph Traversal Algorithm	203
10.3.3 Minimum Spanning Tree Construction	204
10.3.3.1 Overall Strategy	206
10.3.3.2 Detecting the Least Weight Outgoing Edge	207
10.3.3.3 Message Complexity	210
10.4 GRAPH COLORING	210
10.4.1 $(D + 1)$ -Coloring Algorithm	211
10.4.2 6-Coloring of Planar Graphs	212
10.5 COLE–VISHKIN REDUCTION ALGORITHM FOR TREE COLORING	214
10.6 MAXIMAL INDEPENDENT SET: LUBY’S ALGORITHM	218
10.7 CONCLUDING REMARKS	222
10.8 BIBLIOGRAPHIC NOTES	223
EXERCISES	223
CHAPTER 11 ■ Coordination Algorithms	227
11.1 INTRODUCTION	227
11.2 LEADER ELECTION	227
11.2.1 Bully Algorithm	228
11.2.2 Maxima Finding on a Ring	230
11.2.2.1 Chang–Roberts Algorithm	230
11.2.2.2 Franklin’s Algorithm	231
11.2.2.3 Peterson’s Algorithm	232

11.2.3 Election in Arbitrary Networks	234
11.2.4 Election in Anonymous Networks	235
11.3 SYNCHRONIZERS	236
11.3.1 ABD Synchronizer	236
11.3.2 Awerbuch's Synchronizers	237
11.3.2.1 $\alpha$ -Synchronizer	237
11.3.2.2 $\beta$ -Synchronizer	239
11.3.2.3 $\gamma$ -Synchronizer	240
11.3.2.4 Performance of Synchronizer-Based Algorithms	242
11.4 CONCLUDING REMARKS	243
11.5 BIBLIOGRAPHIC NOTES	243
EXERCISES	244

## SECTION IV Faults and Fault-Tolerant Systems

CHAPTER 12 ■ Fault-Tolerant Systems	249
12.1 INTRODUCTION	249
12.2 CLASSIFICATION OF FAULTS	250
12.3 SPECIFICATION OF FAULTS	253
12.4 FAULT-TOLERANT SYSTEMS	255
12.4.1 Masking Tolerance	255
12.4.2 Nonmasking Tolerance	255
12.4.3 Fail-Safe Tolerance	256
12.4.4 Graceful Degradation	257
12.4.5 Detection of Failures in Synchronous Systems	257
12.5 TOLERATING CRASH FAILURES	258
12.5.1 Double and Triple Modular Redundancy	258
12.6 TOLERATING OMISSION FAILURES	259
12.6.1 Stenning's Protocol	260
12.6.2 Sliding Window Protocol	261
12.6.3 Alternating Bit Protocol	263
12.6.4 How TCP Works	264
12.7 CONCLUDING REMARKS	265
12.8 BIBLIOGRAPHIC NOTES	266
EXERCISES	267

16.2	ARCHITECTURE OF REPLICATED DATA MANAGEMENT	340
16.2.1	Passive versus Active Replication	341
16.2.2	Fault-Tolerant State Machines	343
16.3	DATA-CENTRIC CONSISTENCY MODELS	344
16.3.1	Strict Consistency	345
16.3.2	Linearizability	346
16.3.3	Sequential Consistency	346
16.3.4	Causal Consistency	347
16.3.5	FIFO Consistency	348
16.4	CLIENT-CENTRIC CONSISTENCY PROTOCOLS	349
16.4.1	Eventual Consistency	349
16.4.2	Consistency Models for Mobile Clients	350
16.4.2.1	<i>Read-After-Read Consistency</i>	350
16.4.2.2	<i>Write-After-Write Consistency</i>	350
16.4.2.3	<i>Read-After-Write Consistency</i>	351
16.4.2.4	<i>Write-After-Read Consistency</i>	351
16.5	IMPLEMENTATION OF DATA-CENTRIC CONSISTENCY MODELS	351
16.6	QUORUM-BASED PROTOCOLS	352
16.7	REPLICA PLACEMENT	354
16.8	BREWER'S CAP THEOREM	355
16.9	CASE STUDIES	356
16.9.1	Replication Management in Coda	356
16.9.2	Replication Management in Bayou	357
16.9.3	Amazon Dynamo	359
16.10	CONCLUDING REMARKS	361
16.11	BIBLIOGRAPHIC NOTES	361
	EXERCISES	362
CHAPTER 17 ■ Self-Stabilizing Systems		365
17.1	INTRODUCTION	365
17.2	THEORETICAL FOUNDATIONS	367
17.3	STABILIZING MUTUAL EXCLUSION	368
17.3.1	Mutual Exclusion on a Unidirectional Ring	368
17.3.2	Mutual Exclusion on a Bidirectional Array	370
17.4	STABILIZING GRAPH COLORING	373

17.5	STABILIZING SPANNING TREE PROTOCOL	375
17.6	STABILIZING MAXIMAL MATCHING	377
17.7	DISTRIBUTED RESET	379
17.8	STABILIZING CLOCK PHASE SYNCHRONIZATION	382
17.9	CONCLUDING REMARKS	383
17.10	BIBLIOGRAPHIC NOTES	384
	EXERCISES	385
SECTION V <b>Real-World Issues</b>		
CHAPTER 18	■ Distributed Discrete-Event Simulation	391
<hr/>		
18.1	INTRODUCTION	391
	18.1.1 Event-Driven Simulation	391
18.2	DISTRIBUTED SIMULATION	393
	18.2.1 Challenges	393
	18.2.2 Correctness Issues	396
18.3	CONSERVATIVE SIMULATION	396
18.4	OPTIMISTIC SIMULATION AND TIME WARP	397
	18.4.1 Global Virtual Time	398
18.5	CONCLUDING REMARKS	399
18.6	BIBLIOGRAPHIC NOTES	399
	EXERCISES	400
CHAPTER 19	■ Security in Distributed Systems	403
<hr/>		
19.1	INTRODUCTION	403
19.2	SECURITY MECHANISMS	404
19.3	COMMON SECURITY ATTACKS	404
	19.3.1 Eavesdropping	404
	19.3.2 Denial of Service	405
	19.3.3 Data Tampering	405
	19.3.4 Masquerading	405
	19.3.5 Man in the Middle	405
	19.3.6 Malicious Software	405
	19.3.6.1 Virus	406
	19.3.6.2 Worms	406
	19.3.6.3 Spyware	406

19.4	ENCRYPTION	406
19.5	SECRET KEY CRYPTOSYSTEM	408
19.5.1	Confusion and Diffusion	408
19.5.2	DES	409
19.5.3	3DES	411
19.5.4	AES	411
19.5.5	One-Time Pad	411
19.5.6	Stream Ciphers	412
19.5.7	Steganography	413
19.6	PUBLIC KEY CRYPTOSYSTEMS	414
19.6.1	Rivest–Shamir–Adleman Cryptosystem	414
19.6.2	ElGamal Cryptosystem	417
19.7	DIGITAL SIGNATURES	418
19.7.1	Signatures in Secret-Key Cryptosystems	418
19.7.2	Signatures in Public-Key Cryptosystems	418
19.8	HASHING ALGORITHMS	419
19.8.1	Birthday Attack	419
19.9	ELLIPTIC CURVE CRYPTOGRAPHY	420
19.10	AUTHENTICATION SERVER	421
19.10.1	Authentication Service for Secret-Key Cryptosystems	422
19.10.2	Authentication Server for Public-Key Systems	422
19.11	DIGITAL CERTIFICATES	423
19.12	CASE STUDIES	424
19.12.1	Kerberos	424
19.12.2	Pretty Good Privacy	425
19.12.3	Secure Socket Layer	426
19.13	VIRTUAL PRIVATE NETWORKS AND FIREWALLS	428
19.13.1	Virtual Private Network	428
19.13.2	Firewall	429
19.14	SHARING A SECRET	429
19.15	CONCLUDING REMARKS	430
19.16	BIBLIOGRAPHIC NOTES	430
	EXERCISES	431

CHAPTER 20 ■ Sensor Networks	435
20.1 VISION	435
20.2 ARCHITECTURE OF SENSOR NODES	436
20.2.1 MICA Mote	436
20.2.2 ZigBee-Enabled Sensor Nodes	437
20.2.3 TinyOS® Operating System	439
20.3 CHALLENGES IN WIRELESS SENSOR NETWORKS	442
20.3.1 Energy Conservation	442
20.3.2 Fault Tolerance	443
20.3.3 Routing	444
20.3.4 Time Synchronization	444
20.3.5 Location Management	444
20.3.6 Middleware Design	444
20.3.7 Security	445
20.4 ROUTING ALGORITHMS	445
20.4.1 Directed Diffusion	445
20.4.2 Cluster-Based Routing	446
20.4.2.1 LEACH	446
20.4.2.2 PEGASIS	447
20.4.3 Metadata-Based Routing: SPIN	448
20.5 TIME SYNCHRONIZATION USING REFERENCE BROADCAST	448
20.5.1 Reference Broadcast	449
20.6 LOCALIZATION ALGORITHMS	451
20.6.1 RSSI-Based Ranging	451
20.6.2 Ranging Using Time Difference of Arrival	451
20.6.3 Anchor-Based Ranging	451
20.7 SECURITY IN SENSOR NETWORKS	452
20.7.1 SPIN for Data Security	453
20.7.1.1 Overview of SNEP	453
20.7.1.2 Overview of $\mu$ TESLA	454
20.7.2 Attacks on Routing	454
20.7.2.1 Hello Flood	455

20.8	APPLICATIONS	455
20.8.1	Health-Care Applications	455
20.8.2	Environment Monitoring and Control	456
20.8.3	Citizen Sensing	456
20.8.4	Pursuer–Evader Game	456
20.9	CONCLUDING REMARKS	459
20.10	BIBLIOGRAPHIC NOTES	459
	EXERCISES	460
CHAPTER 21 ■ Social and Peer-to-Peer Networks		465
21.1	INTRODUCTION TO SOCIAL NETWORKS	465
21.1.1	Milgram’s Experiment	465
21.2	METRICS OF SOCIAL NETWORKS	467
21.2.1	Clustering Coefficient	467
21.2.2	Diameter	468
21.3	MODELING SOCIAL NETWORKS	468
21.3.1	Erdős–Rényi Model	468
21.3.2	Small-World Model	469
21.3.3	Power-Law Graphs	470
21.4	CENTRALITY MEASURES IN SOCIAL NETWORKS	473
21.4.1	Degree Centrality	473
21.4.2	Closeness Centrality	473
21.4.3	Betweenness Centrality	474
21.5	COMMUNITY DETECTION	475
21.5.1	Girvan–Newman Algorithm	475
21.6	INTRODUCTION TO PEER-TO-PEER NETWORKS	476
21.7	FIRST-GENERATION P2P SYSTEMS	477
21.7.1	Napster	477
21.7.2	Gnutella	478
21.8	SECOND-GENERATION P2P SYSTEMS	479
21.8.1	KaZaA	481
21.8.2	Chord	481
21.8.3	Content-Addressable Network	484
21.8.4	Pastry	485
21.9	KOORDE AND DE BRUIJN GRAPH	487
21.10	SKIP GRAPH	488

21.11 REPLICATION MANAGEMENT	491
21.12 BITTORRENT AND FREE RIDING	492
21.13 CENSORSHIP RESISTANCE, ANONYMITY	494
21.14 CONCLUDING REMARKS	494
21.15 BIBLIOGRAPHIC NOTES	495
EXERCISES	496
REFERENCES, 501	
INDEX, 513	