

Lecture Notes in Networks and Systems

Volume 61

Series Editor

Janusz Kacprzyk, Polish Academy of Sciences, Systems Research Institute,
Warsaw, Poland

Advisory Editors

Fernando Gomide, Department of Computer Engineering and Automation—DCA,
School of Electrical and Computer Engineering—FEEC, University of Campinas—
UNICAMP, São Paulo, Brazil

Okyay Kaynak, Department of Electrical and Electronic Engineering, Bogazici
University, Istanbul, Turkey

Derong Liu, Department of Electrical and Computer Engineering, University of
Illinois at Chicago, Chicago, USA, Institute of Automation, Chinese Academy of
Sciences, Beijing, China

Witold Pedrycz, Department of Electrical and Computer Engineering, University of
Alberta, Alberta, Canada, Systems Research Institute, Polish Academy of Sciences,
Warsaw, Poland

Marios M. Polycarpou, KIOS Research Center for Intelligent Systems and
Networks, Department of Electrical and Computer Engineering, University of
Cyprus, Nicosia, Cyprus

Imre J. Rudas, Óbuda University, Budapest, Hungary

Jun Wang, Department of Computer Science, City University of Hong Kong,
Kowloon, Hong Kong

The series “Lecture Notes in Networks and Systems” publishes the latest developments in Networks and Systems—quickly, informally and with high quality. Original research reported in proceedings and post-proceedings represents the core of LNNS.

Volumes published in LNNS embrace all aspects and subfields of, as well as new challenges in, Networks and Systems.

The series contains proceedings and edited volumes in systems and networks, spanning the areas of Cyber-Physical Systems, Autonomous Systems, Sensor Networks, Control Systems, Energy Systems, Automotive Systems, Biological Systems, Vehicular Networking and Connected Vehicles, Aerospace Systems, Automation, Manufacturing, Smart Grids, Nonlinear Systems, Power Systems, Robotics, Social Systems, Economic Systems and other. Of particular value to both the contributors and the readership are the short publication timeframe and the world-wide distribution and exposure which enable both a wide and rapid dissemination of research output.

The series covers the theory, applications, and perspectives on the state of the art and future developments relevant to systems and networks, decision making, control, complex processes and related areas, as embedded in the fields of interdisciplinary and applied sciences, engineering, computer science, physics, economics, social, and life sciences, as well as the paradigms and methodologies behind them.

**** Indexing: The books of this series are submitted to ISI Proceedings, SCOPUS, Google Scholar and Springerlink ****

More information about this series at <http://www.springer.com/series/15179>

Paweł Gburzyński

Modeling Communication Networks and Protocols

Implementation via the SMURPH System

 Springer

Paweł Gburzyński
Computer Engineering Department
Vistula University
Warsaw, Poland

ISSN 2367-3370 ISSN 2367-3389 (electronic)
Lecture Notes in Networks and Systems
ISBN 978-3-030-15390-8 ISBN 978-3-030-15391-5 (eBook)
<https://doi.org/10.1007/978-3-030-15391-5>

Library of Congress Control Number: 2019933844

© Springer Nature Switzerland AG 2019

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Contents

1	Introduction	1
1.1	A Historical Note	2
1.2	Event-Driven Simulation: A Crash Course	5
1.2.1	Events	6
1.2.2	The Car Wash Model	7
1.2.3	Discrete Time	12
1.2.4	Nondeterminism and the Time Grain	16
1.3	An Overview of SMURPH	19
1.3.1	Reactive Systems: Execution Modes	20
1.3.2	Organization of the Package	22
	References	24
2	Examples	27
2.1	The Car Wash Redux	27
2.1.1	The Washer Process	28
2.1.2	The Entrance Process	33
2.1.3	Completing the Program	35
2.1.4	About Time	37
2.1.5	Presenting the Results	39
2.1.6	Running the Model	39
2.1.7	In Real Time	41
2.2	The Alternating-Bit Protocol	44
2.2.1	The Protocol	45
2.2.2	Stations and Packet Buffers	47
2.2.3	The Sender's Protocol	49
2.2.4	The Recipient's Protocol	55
2.2.5	Wrapping It Up	58
2.2.6	Running the Model	64

2.3	A PID Controller	69
2.3.1	PID Control in a Nutshell	70
2.3.2	Building the Controller	73
2.3.3	Playing with the Controller	82
2.4	A Wireless Network Model	88
2.4.1	The ALOHA Network	89
2.4.2	The Protocol Specification	90
2.4.3	The Hub Station	94
2.4.4	The Terminals	102
2.4.5	The Channel Model	105
2.4.6	Wrapping It Up	116
2.4.7	Experimenting with the Model	120
	References	132
3	Basic Operations and Types	135
3.1	Numbers	135
3.1.1	Simple Integer Types	135
3.1.2	Time in SMURPH	137
3.1.3	Type BIG and Its Range	141
3.1.4	Arithmetic Operations	142
3.1.5	Constants	143
3.1.6	Other Non-standard Numerical Types	144
3.2	Auxiliary Operation and Functions	146
3.2.1	Random Number Generators	146
3.2.2	Input/Output	150
3.2.3	The XML Parser	153
3.2.4	Operations on Flags	157
3.2.5	Type Boolean	158
3.2.6	Pools	158
3.2.7	Error Handling	159
3.2.8	Identifying the Experiment	160
3.2.9	Telling the Time and Date	160
3.2.10	Decibels	161
3.3	SMURPH Types	162
3.3.1	Type Hierarchy	162
3.3.2	Object Naming	163
3.3.3	Type Derivation	166
3.3.4	Multiple Inheritance	168
3.3.5	Abstract Types	170
3.3.6	Announcing Types	170
3.3.7	Subtypes with Empty Local Attribute Lists	171
3.3.8	Object Creation and Destruction	171
	References	173

- 4 The Virtual Hardware** 175
 - 4.1 Stations 176
 - 4.1.1 Declaring Station Types 176
 - 4.1.2 Creating Station Objects 177
 - 4.1.3 The Current Station 179
 - 4.2 Links 179
 - 4.2.1 Propagation of Signals in Links 179
 - 4.2.2 Creating Links 180
 - 4.3 Ports 182
 - 4.3.1 Creating Ports 182
 - 4.3.2 Connecting Ports to Links 184
 - 4.3.3 Setting Distance Between Ports 185
 - 4.4 Radio Channels 187
 - 4.4.1 Signals, Attenuation, Interference 189
 - 4.4.2 Creating Radio Channels 192
 - 4.4.3 The List of Assessment Methods 193
 - 4.5 Transceivers 198
 - 4.5.1 Creating Transceivers 198
 - 4.5.2 Interfacing and Configuring Transceivers 201
 - Reference 206
- 5 The Execution Framework** 207
 - 5.1 Processes 207
 - 5.1.1 Activity Interpreters: The Concept 207
 - 5.1.2 Declaring Process Types 208
 - 5.1.3 Creating and Terminating Processes 210
 - 5.1.4 Process Operation 213
 - 5.1.5 The Code Method 215
 - 5.1.6 The Process Environment 217
 - 5.1.7 Process as an Activity Interpreter 219
 - 5.1.8 The Root Process 220
 - 5.2 Signal Passing 223
 - 5.2.1 Regular Signals 223
 - 5.2.2 Priority Signals 225
 - 5.3 The Timer AI 226
 - 5.3.1 Wait Requests 227
 - 5.3.2 Operations 229
 - 5.3.3 Clock Tolerance 230
 - 5.3.4 The Visualization Mode 232
 - 5.4 The Monitor AI 233
 - 5.4.1 Wait Requests 234
 - 5.4.2 Signaling Monitor Events 234

6	The Client	237
6.1	General Concepts	237
6.2	Message and Packet Types	239
6.3	Packet Buffers	242
6.4	Station Groups	244
6.5	Communication Groups	246
6.6	Traffic Patterns	248
6.6.1	Defining Traffic Types	248
6.6.2	Creating Traffic Patterns	249
6.6.3	Modifying the Standard Behavior of Traffic Patterns	255
6.6.4	Intercepting Packet and Message Deallocation	259
6.6.5	Message Queues	261
6.7	Inquiries	262
6.7.1	Acquiring Packets for Transmission	263
6.7.2	Testing for Packet Availability	266
6.8	Wait Requests	266
7	Links and Ports	269
7.1	Activities	269
7.1.1	Activity Processing in Links	270
7.1.2	Starting and Terminating Activities	271
7.1.3	Wait Requests and Events	274
7.1.4	Collisions	278
7.1.5	Event Priorities	278
7.1.6	Receiving Packets	280
7.2	Port Inquiries	282
7.2.1	Inquiries About the Present	282
7.2.2	Inquiries About the Past	284
7.2.3	Inquiries About the Future	285
7.3	Faulty Links	287
7.4	Cleaning After Packets	289
	References	291
8	Radio Channels and Transceivers	293
8.1	Interpreting Activities in Radio Channels	293
8.1.1	The Stages of Packet Transmission and Perception	294
8.1.2	Criteria of Event Assessment	295
8.1.3	Neighborhoods	296
8.1.4	Event Assessment	297
8.1.5	Interference Histograms	299
8.1.6	Event Reassessment	301

- 8.1.7 The Context of Assessment Methods 302
- 8.1.8 Starting and Terminating Packet Transmission 303
- 8.2 Packet Perception and Reception 304
 - 8.2.1 Wait Requests and Events 305
 - 8.2.2 Event Priorities 310
 - 8.2.3 Receiving Packets 310
 - 8.2.4 Hooks for Handling Bit Errors 311
 - 8.2.5 Transceiver Inquiries 316
- 8.3 Cleaning After Packets 321
- Reference 321
- 9 Mailboxes 323**
 - 9.1 General Concepts 323
 - 9.2 Fifo Mailboxes 324
 - 9.2.1 Declaring and Creating Fifo Mailboxes 324
 - 9.2.2 Wait Requests 327
 - 9.2.3 Operations on Fifo Mailboxes 330
 - 9.2.4 The Priority Put Operation 335
 - 9.3 Barrier Mailboxes 337
 - 9.3.1 Declaring and Creating Barrier Mailboxes 337
 - 9.3.2 Wait Requests 338
 - 9.3.3 Operations on Barrier Mailboxes 339
 - 9.4 Bound Mailboxes 339
 - 9.4.1 Binding Mailboxes 340
 - 9.4.2 Device Mailboxes 342
 - 9.4.3 Client Mailboxes 344
 - 9.4.4 Server Mailboxes 345
 - 9.4.5 Unbinding and Determining the Bound Status 348
 - 9.4.6 Wait Requests 349
 - 9.4.7 Operations on Bound Mailboxes 350
 - 9.5 Journaling 355
 - 9.5.1 Declaring Mailboxes to Be Journaled 355
 - 9.5.2 Driving Mailboxes from Journal Files 356
 - 9.5.3 Journaling Client and Server Mailboxes 357
- 10 Measuring Performance 359**
 - 10.1 Type RVariable 359
 - 10.1.1 Creating and Destroying Random Variables 360
 - 10.1.2 Operations on Random Variables 360
 - 10.2 Client Performance Measures 362
 - 10.2.1 Random Variables 362
 - 10.2.2 Counters 363
 - 10.2.3 Virtual Methods 364
 - 10.2.4 Resetting Performance Measures 365

10.3	Link Performance Measures	367
10.4	RFChannel Performance Measures	368
10.5	Terminating Execution	369
10.5.1	Maximum Number of Received Messages	371
10.5.2	Virtual Time Limit	371
10.5.3	CPU Time Limit	371
10.5.4	The Exit Code	372
	Reference	372
11	Tools for Testing and Debugging	373
11.1	User-Level Tracing	373
11.2	Simulator-Level Tracing	376
11.3	Observers	377
	References	381
12	Exposing Objects	383
12.1	General Concepts	383
12.2	Making Objects Exposable	384
12.3	Programming Exposures	387
12.4	Invoking Exposures	393
12.5	Standard Exposures	395
12.5.1	Timer Exposure	398
12.5.2	Mailbox Exposure	399
12.5.3	RVariable Exposure	400
12.5.4	Client and Traffic Exposure	401
12.5.5	Port Exposure	404
12.5.6	Link Exposure	405
12.5.7	Transceiver Exposure	407
12.5.8	RFChannel Exposure	408
12.5.9	Process Exposure	410
12.5.10	Kernel Exposure	411
12.5.11	Station Exposure	413
12.5.12	System Exposure	416
12.5.13	Observer Exposure	417
	Appendix A: Library Models of Wireless Channels	419
	Appendix B: SMURPH Under Linux and Windows	441
	Appendix C: DSD: The Dynamic Status Display Program	457
	Bibliography	479
	Index	481