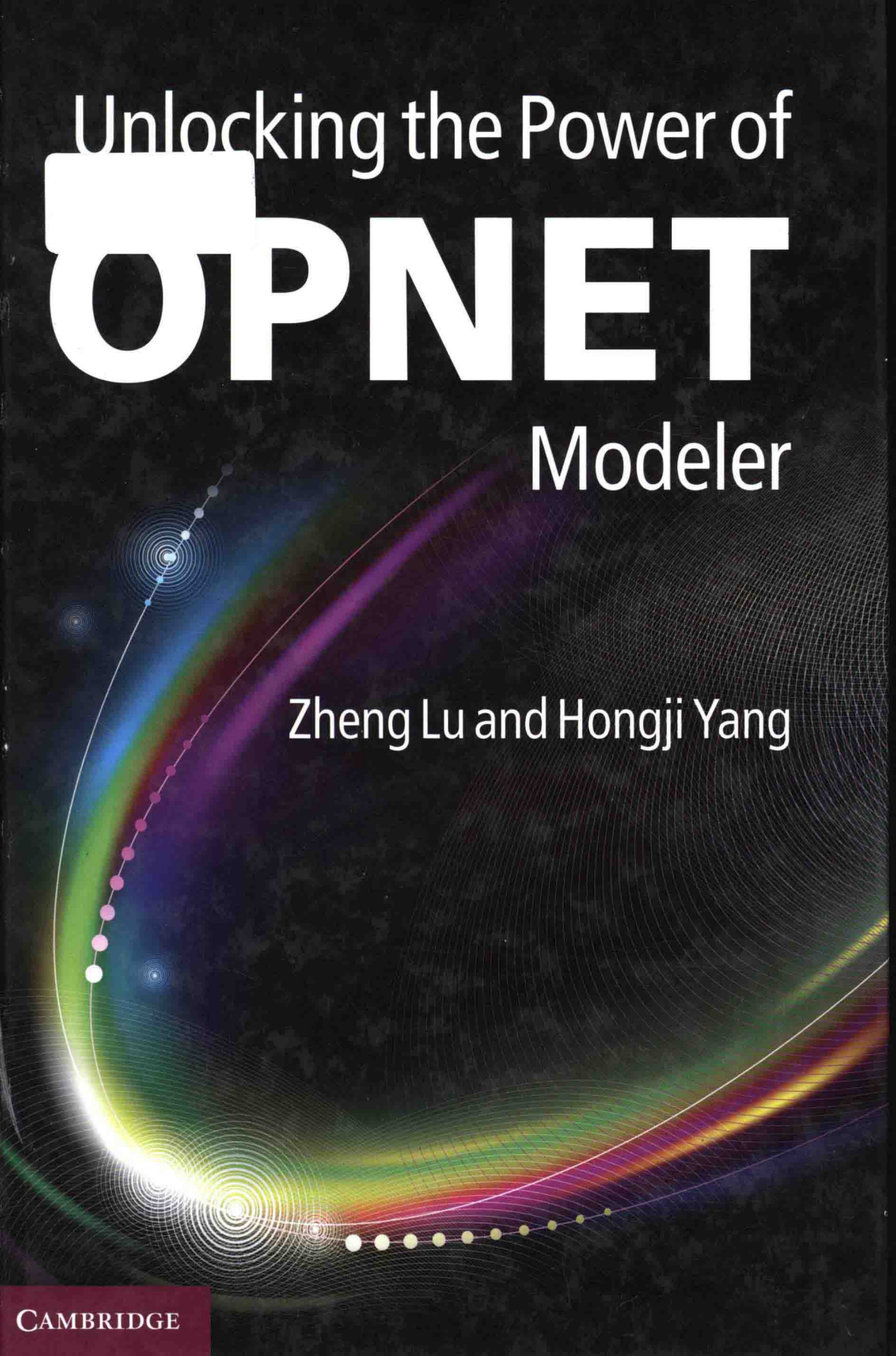
Unlocking the Power of OPNET Modeler

Zheng Lu and Hongji Yang



CAMBRIDGE

Unlocking the Power of OPNET Modeler

ZHENG LU

HONGJI YANG

CAMBRIDGE
UNIVERSITY PRESS

CAMBRIDGE UNIVERSITY PRESS
Cambridge, New York, Melbourne, Madrid, Cape Town,
Singapore, São Paulo, Delhi, Tokyo, Mexico City

Cambridge University Press
The Edinburgh Building, Cambridge CB2 8RU, UK

Published in the United States of America by Cambridge University Press, New York

www.cambridge.org
Information on this title: www.cambridge.org/9780521198745

First published 2012

Printed in the United Kingdom at the University Press, Cambridge

A catalogue record for this publication is available from the British Library

Library of Congress Cataloguing in Publication data
Lu, Zheng.
Unlocking the power of OPNET modeler / Zheng Lu, Hongji Yang.
p. cm.
Includes bibliographical references and index.
ISBN 978-0-521-19874-5 (hardback)
1. Computer networks – Mathematical models. 2. Computer networks – Simulation methods.
3. Computer network protocols. I. Yang, Hongji. II. Title.
TK5105.5.L825 2011
005.7′13–dc23 2011032466

ISBN 978-0-521-19874-5 Hardback

Additional resources for this publication at www.cambridge.org/9780521198745

Unlocking the Power of OPNET Modeler

For fast, easy modeling, this practical guide provides all the essential information you need to know. A wide range of topics is covered, including custom protocols, programming in C++, External Model Access (EMA) modeling, and co-simulation with external systems, giving you the guidance not provided in the OPNET documentation. A set of high-level wrapper APIs is also included to simplify programming custom OPNET models, whether you are a newcomer to OPNET or an experienced user needing to model efficiently. From the basic to the advanced, you'll find topics are easy to follow with theory kept to a minimum, many practical tips and answers to frequently asked questions spread throughout the book, and numerous step-by-step case studies and real-world network scenarios included.

Zheng Lu received his Ph.D. from the University of Essex, after which he stayed on to research optical networks and wireless sensor networks. He is experienced in modeling network protocols and has many years of experience using OPNET Modeler in his research and laboratory demonstrations.

Hongji Yang is currently a Professor at the Software Technology Research Laboratory, De Montfort University. He received his Ph.D. from Durham University in 1994 and was a main contributor to the Distributed Computer Networks project sponsored by the Chinese Ministry of Education, 1982–1986.

Preface

Network simulation is an important methodology in network research fields and OPNET Modeler is a very useful tool for network modeling and simulation. OPNET Modeler is generally used by researchers, protocol designers, university teachers and students in the fields of electronic engineering, computer science, management information systems, and related disciplines. The friendly design of its graphical user interface (GUI) makes it nice and easy to start with. However, the complexity of OPNET Modeler and lack of useful support material make it difficult for many users to fully make use of its benefits. OPNET Modeler has its documentation covering many aspects on using the modeler. However, it covers too many aspects in parallel form rather than a step-forward form, making users unable to decide where to start and causing them to lose focus.

This book is an effort to partially fill this gap and should be useful for courses on network simulation and OPNET modeling for university students, as well as for the researchers on this topic. The book covers a wide range of knowledge from basic topics to advanced topics. All case studies in the book are step-by-step and progressive. Relevant files and sources can be downloaded from the publisher's website. A set of high-level wrapper APIs are provided to help even new users to write complex models, and experienced users to write large, complex models efficiently. Question-and-answer pairs are spread over the chapters to answer the most common questions users may experience in practice.

The book is composed of four parts. Part I: Preparation for OPNET Modeling introduces OPNET and OPNET Modeler. It leads the reader through the required basics on using OPNET Modeler and provides familiarization with OPNET Modeler user interfaces. Part II: Modeling Custom Networks and Protocols first teaches the reader how to create custom models by directly using OPNET API packages. It then introduces a high-level wrapper API package and demonstrates how to model systems easily using these high-level wrapper API packages instead. Part III: Modeling and Modifying Standard Networks and Protocols teaches the reader how to model networks and protocols based on existing standard OPNET modules and how to modify existing standard models in order to extend standard protocols by adding custom features. Part IV: OPNET Modeling Facilities covers content that is used to facilitate OPNET modeling, including debugging, hybrid simulation, External Model Access (EMA), co-simulation, programming OPNET models in C++, etc.

We thank deeply the various people who, during the months over which this endeavor lasted, provided us with useful and helpful assistance. Without their care and consideration, this book would likely not have matured.

First, we thank Dr. David K. Hunter and Dr. Yixuan Qin, who gave us useful suggestions and comments before and during the writing of the book.

Second, we thank the publisher and people who demonstrated interest in publishing this book. The production team at Cambridge University Press has been great. Many thanks go to people who helped us with the book development, including Mrs. Sarah Marsh and Dr. Julie Lancashire.

Dr. Zheng Lu would like to thank his wife Dr. Gui Gui; without her support, he could not have got through that difficult time and thrown himself into finishing the book.

Professor Hongji Yang would like to thank his wife, Xiaodong Zhang, for her full support in finishing the writing of this book.

Zheng Lu
Hongji Yang

Abbreviations

API	application programming interface
BSS	basic service set
CDB	Microsoft Console Debugger
CMO	Categorized Memory
DB	diagnostic block
DES	discrete event simulation
EMA	External Model Access
ESA	External Simulation Access
ESD	External System Definition
Esys	External System
ETS	external tool support
FB	function block
FPP	Fractal Point Process
GDB	GNU Project Debugger
GUI	graphic user interface
HB	header block
ICI	Interface Control Information
IDE	Integrated Development Environment
KP	Kernel Procedure
LAN	local area network
MSVC	Microsoft Visual C++ Debugger
ODB	OPNET Simulation Debugger
ODK	OPNET Development Kit
PDF	probability density function
PMO	Pooled Memory
PPP	Point to Point Protocol
QoS	quality of service
RPG	raw packet generator
SDK	software development kit
STD	state transition diagram
STL	Standard Template Library

SV	state variable
TB	termination block
TV	temporary variable
UI	user interface
WAN	wide area network
WLAN	wireless local area network

Contents

Part I

Preparation for OPNET Modeling

1 Introduction

This chapter introduces network modeling and simulation, and both OPNET and OPNET Modeler. If you already have relevant background, you can quickly read through this chapter and go to Chapter 2.

1.1 Network modeling and simulation

There are several feasible methods for investigating networking protocols and evaluating network performance (Leemis and Park 2006; www.opnet.com):

- Analysis and mathematical modeling
- Simulation – typically time-based simulation or discrete event-based simulation
- Hybrid simulation with both analysis and simulation
- Test-bed emulation

Analysis and mathematical modeling can provide quick insights and answers to the problems being studied. It is generally faster than simulation, but in many cases is inaccurate or inapplicable. Analytical models are not available for many situations. Even so, many of the available models lack accuracy and some are modeled through approximations. Especially for a network of queues, it can either be decomposed via the Kleinrock independence assumption or be solved using a hop-by-hop single system analysis, both of which lose accuracy. The modeling difficulties and loss of accuracy can be greatly exacerbated when the networking protocols become even slightly complex. It is often necessary to resort to approximation by reducing the general model to a typical and representative analytical path in order to reduce the analytical difficulties (Kleinrock, 1976).

Network simulation provides a way to model the network behaviors by calculating the interactions between modeling devices. Discrete event simulation (DES) is the typical method in large-scale simulation studies instead of a simpler time-based method. DES enables modeling in a more accurate and realistic way, and has broad applicability (Leemis and Park 2006). DES creates an extremely detailed, packet-by-packet model for the activities of network to be predicted. However, it often has significant requirements for computing power; in particular, for very large-scale simulation studies, the process can be time-consuming. It can take several hours or even days to complete. However,

simulation can always provide accurate solutions for either a single-node queuing system or a network of queues, from simple algorithm to complex protocol.

One way to work around these issues in mathematical analysis and explicit simulation is combining the methods in the simulation in order to gain access to the advantages of both while overcoming their disadvantages. This combined method is typically called hybrid simulation, i.e., partially modeling in DES for accuracy and partially in mathematical analysis for faster speed and less computational burden (see www.opnet.com).

There are many network simulators like OPNET (see www.opnet.com), NS (www.isi.edu/nsnam/ns), and OMNeT++ (www.omnetpg.org) which are popular and widely used. Among them, OPNET is capable of simulating in both explicit DES and hybrid simulation modes, and supports other simulation features like co-simulation, parallel simulation, high-level architecture, and system-in-the-loop interactive simulations.

Test-bed emulation typically involves implementing the studied algorithms and protocols into real-world hardware but in a much smaller scale or size. Since test-bed emulation considers the aspects of both protocols and real-world situations, it is the best way to provide a benchmark estimating how feasible the algorithms and protocols are and how close they are to the actual situation. Also, it is a useful way demonstrate new networking concepts. The disadvantage is it will also deal with all other real-world difficulties and some unexpected engineering problems which can be completely irrelevant to the studied algorithms and protocols but can be significant in the overall emulation results. Further, the cost of building an emulation test-bed may be significant. Test-beds are not suitable for investigating large systems.

Accordingly, research methodologies for data traffic and networking can be a combination of some or all of these methods. These methods can be used to cross-check each other in order to capture the system in a more accurate, efficient, and cost-effective way.

1.2 Introduction to OPNET

OPNET stands for OPtimized Network Engineering Tools, and was created by OPNET Technologies, Inc., which was founded in 1986. OPNET is a network simulation tool set; its products and solutions address the following aspects of communications networks (see www.opnet.com):

- Application performance management
- Planning
- Engineering
- Operations
- Research and development

This tool set is powerful and can create and test large network environments via software. To address each of these aspects, OPNET provides corresponding product modules throughout its product line.