OPTIMIZATION OF COMPUTER NETWORKS – MODELING AND ALGORITHMS
OPTIMIZATION OF COMPUTER NETWORKS – MODELING AND ALGORITHMS
A HANDS-ON APPROACH

Pablo Pavón Mariño

WILEY
To my sons, Pablo and Guille, and to my wife Victoria, the smiles of my life.
# Contents

About the Author xv
Preface xvii
Acknowledgments xxi

## 1 Introduction 1

1.1 What is a Communication Network? 1
1.2 Capturing the Random User Behavior 4
1.3 Queueing Theory and Optimization Theory 5
1.4 The Rationale and Organization of this Book 6
   1.4.1 Part I: Modeling 6
   1.4.2 Part II: Algorithms 7
   1.4.3 Basic Optimization Requisites: Appendices I, II, and III 10
   1.4.4 Net2Plan Tool: Appendix IV 11

## Part I MODELING 15

### 2 Definitions and Notation 15

2.1 Notation for Sets, Vectors and Matrices 15
   2.1.1 Norm Basics 15
   2.1.2 Set Basics 16
2.2 Network Topology 17
2.3 Installed Capacities 19
2.4 Traffic Demands 19
   2.4.1 Unicast, Anycast, and Multicast Demands 20
   2.4.2 Elastic versus Inelastic Demands 21
2.5 Traffic Routing 21
References 22
3 **Performance Metrics in Networks**

3.1 Introduction 23
3.2 Delay 23
3.2.1 Link Delay 23
3.2.2 End-to-End Delay 27
3.2.3 Average Network Delay 27
3.2.4 Convexity Properties 27
3.3 Blocking Probability 28
3.3.1 Link Blocking Probability 28
3.3.2 Demand and Network Blocking Probability 30
3.3.3 Other Blocking Estimations 31
3.3.4 Convexity Properties 34
3.4 Average Number of Hops 34
3.5 Network Congestion 36
3.6 Network Cost 36
3.7 Network Resilience Metrics 37
3.7.1 Shared Risk Groups 40
3.7.2 Simplified Availability Calculations 41
3.7.3 General Model 41
3.8 Network Utility and Fairness in Resource Allocation 44
3.8.1 Fairness in Resource Allocation 44
3.8.2 Fairness and Utility Functions 45
3.8.3 Convexity Properties 47
3.9 Notes and Sources 47
3.10 Exercises 49

References 51

4 **Routing Problems**

4.1 Introduction 53
4.2 Flow-Path Formulation 54
4.2.1 Optimality Analysis 55
4.2.2 Candidate Path List Pre-Computation 58
4.2.3 Ranking of Paths Elaboration 58
4.2.4 Candidate Path List Augmentation (CPLA) 59
4.3 Flow-Link Formulation 61
4.3.1 Flow Conservation Constraints 62
4.3.2 Obtaining the Routing from $x_{de}$ Variables 63
4.3.3 Optimality Analysis 64
4.4 Destination-Link Formulation 65
4.4.1 Obtaining the Routing Tables from $x_{te}$ Variables 67
4.4.2 Some Properties of the Routing Table Representation 67
4.4.3 Comparing Flow-Based and Destination-Based Routing 71
4.5 Convexity Properties of Performance Metrics 71
4.6 Problem Variants 72
4.6.1 Anycast Routing 72
4.6.2 Multicast Routing 74
## Contents

4.6.3 Non-Bifurcated Routing 75  
4.6.4 Integral Routing 77  
4.6.5 Destination-Based Shortest Path Routing 77  
4.6.6 SRG-Disjoint 1+1 Dedicated Protection Routing 79  
4.6.7 Shared Restoration Routing 80  
4.6.8 Multi-Hour Routing 81

4.7 Notes and Sources 83  
4.8 Exercises 83  
References 86

5 Capacity Assignment Problems 88  
5.1 Introduction 88  
5.2 Long-Term Capacity Planning Problem Variants 89  
5.2.1 Capacity Planning for Concave Costs 89  
5.2.2 Capacity Planning with Modular Capacities 94  
5.2.3 Multi-Period Capacity Planning 97  
5.3 Fast Capacity Allocation Problem Variants: Wireless Networks 98  
5.3.1 The Wireless Channel 99  
5.3.2 Wireless Networks 100  
5.3.3 Modeling Wireless Networks 101  
5.4 MAC Design in Hard-Interference Scenarios 104  
5.4.1 Optimization in Random Access Networks 105  
5.4.2 Optimization in Carrier-Sense Networks 109  
5.5 Transmission Power Optimization in Soft Interference Scenarios 113  
5.6 Notes and Sources 116  
5.7 Exercises 117  
References 118

6 Congestion Control Problems 120  
6.1 Introduction 120  
6.2 NUM Model 121  
6.2.1 Utility Functions for Elastic and Inelastic Traffic 121  
6.2.2 Fair Congestion Control 122  
6.2.3 Optimality Conditions 123  
6.3 Case Study: TCP 124  
6.3.1 Window-Based Flow Control 125  
6.3.2 TCP Reno 126  
6.3.3 TCP Vegas 131  
6.4 Active Queue Management (AQM) 134  
6.4.1 A Simplified Model of the TCP-AQM Interplay 135  
6.5 Notes and Sources 136  
6.6 Exercises 137  
References 139

7 Topology Design Problems 141  
7.1 Introduction 141
Part II ALGORITHMS

8 Gradient Algorithms in Network Design
8.1 Introduction
8.2 Convergence Rates
8.3 Projected Gradient Methods
  8.3.1 Basic Gradient Projection Algorithm
  8.3.2 Scaled Projected Gradient Method
  8.3.3 Singular and Ill-Conditioned Problems
8.4 Asynchronous and Distributed Algorithm Implementations
8.5 Non-Smooth Functions
8.6 Stochastic Gradient Methods
8.7 Stopping Criteria
8.8 Algorithm Design Hints
  8.8.1 Dimensioning the Step Size
  8.8.2 Discrete Step Length
  8.8.3 Heavy-Ball Methods
8.9 Notes and Sources
8.10 Exercises
References

9 Primal Gradient Algorithms
9.1 Introduction
9.2 Penalty Methods
  9.2.1 Interior Penalty Methods
  9.2.2 Exterior Penalty Methods
9.3 Adaptive Bifurcated Routing
  9.3.1 Removing Equality Constraints
  9.3.2 Optimality and Stability
  9.3.3 Implementation Example
9.4 Congestion Control using Barrier Functions
  9.4.1 Implementation Example
  9.4.2 Exterior Penalty
References
# Contents

9.5 Persistence Probability Adjustment in MAC Protocols 201
   9.5.1 Implementation Example 203

9.6 Transmission Power Assignment in Wireless Networks 205
   9.6.1 Implementation Example 207

9.7 Notes and Sources 210

9.8 Exercises 211

References 213

10 Dual Gradient Algorithms 214

10.1 Introduction 214

10.2 Adaptive Routing in Data Networks 217
   10.2.1 Optimality and Stability 219
   10.2.2 Implementation Example 219

10.3 Backpressure (Center-Free) Routing 221
   10.3.1 Relation between $\gamma$, $\Delta P$, and Average Queue Sizes, $Q_{nd}$ 224
   10.3.2 Implementation Example 225

10.4 Congestion Control 228
   10.4.1 Optimality and Stability Conditions 229
   10.4.2 Implementation Example 230

10.5 Decentralized Optimization of CSMA Window Sizes 231
   10.5.1 Implementation Example 234

10.6 Notes and Sources 236

10.7 Exercises 236

References 238

11 Decomposition Techniques 240

11.1 Introduction 240

11.2 Theoretical Fundamentals 241
   11.2.1 Primal Decomposition 241
   11.2.2 Dual Decomposition 244
   11.2.3 Other Decompositions 246

11.3 Cross-Layer Congestion Control and QoS Capacity Allocation 247
   11.3.1 Implementation Example 249

11.4 Cross-Layer Congestion Control and Backpressure Routing 249
   11.4.1 Implementation Example 252

11.5 Cross-Layer Congestion Control and Power Allocation 253
   11.5.1 Implementation Example 254

11.6 Multidomain Routing 256
   11.6.1 Implementation Example 258

11.7 Dual Decomposition in Non-Convex Problems 259
   11.7.1 Implementation Example 261

11.8 Notes and Sources 261

11.9 Exercises 263

References 265
12 Heuristic Algorithms

12.1 Introduction
   12.1.1 What Complexity Theory Tells Us that We cannot Do
   12.1.2 Our Options
   12.1.3 Organization and Rationale of this Chapter

12.2 Heuristic Design Keys
   12.2.1 Heuristic Types
   12.2.2 Intensification versus Diversification
   12.2.3 How to Assess the Solution Quality
   12.2.4 Stop Conditions
   12.2.5 Defining the Cost or Fitness Function
   12.2.6 Coding the Solution

12.3 Local Search Algorithms
   12.3.1 Design Hints

12.4 Simulated Annealing
   12.4.1 Design hints

12.5 Tabu Search
   12.5.1 Design Hints

12.6 Greedy Algorithms

12.7 GRASP

12.8 Ant Colony Optimization
   12.8.1 Design Hints

12.9 Evolutionary Algorithms
   12.9.1 Design Hints

12.10 Case Study: Greenfield Plan with Recovery Schemes Comparison
   12.10.1 Case Study Description
   12.10.2 Algorithm Description
   12.10.3 Combining Heuristics and ILPs
   12.10.4 Results

12.11 Notes and Sources

12.12 Exercises

References

A Convex Sets. Convex Functions

A.1 Convex Sets

A.2 Convex and Concave Functions
   A.2.1 Convexity in Differentiable Functions
   A.2.2 Strong Convexity/Concavity
   A.2.3 Convexity in Non-Differentiable Functions
   A.2.4 Determining the Curvature of a Function
   A.2.5 Sub-level Sets
   A.2.6 Epigraphs

A.3 Notes and Sources

Reference
B  Mathematical Optimization Basics  313
B.1  Optimization Problems  313
B.2  A Classification of Optimization Problems  315
    B.2.1  Linear Programming  315
    B.2.2  Convex Programs  318
    B.2.3  Nonlinear Programs  320
    B.2.4  Integer Programs  321
B.3  Duality  324
    B.3.1  Dual Function  324
B.4  Optimality Conditions  330
    B.4.1  Optimality Conditions in Problems with Strong Duality  330
    B.4.2  Graphical Interpretation of KKT Conditions  333
    B.4.3  Optimality Conditions in Problems Without Strong Duality  336
B.5  Sensitivity Analysis  337
B.6  Notes and Sources  339
    References  340

C  Complexity Theory  341
C.1  Introduction  341
C.2  Deterministic Machines and Deterministic Algorithms  342
    C.2.1  Complexity of a Deterministic Algorithm  342
    C.2.2  Worst-Case Algorithm Complexity  343
    C.2.3  Asymptotic Algorithm Complexity  343
    C.2.4  Complexity is a Real Barrier  345
C.3  Non-Deterministic Machines and Non-Deterministic Algorithms  346
    C.3.1  Complexity of a Non-Deterministic Algorithm  347
C.4  \( \mathcal{P} \) and \( \mathcal{NP} \) Complexity Classes  347
C.5  Polynomial Reductions  349
    C.5.1  A Polynomial Time Reduction Example  350
C.6  \( \mathcal{NP} \)-Completeness  351
    C.6.1  An Example Proving \( \mathcal{NP} \)-Completeness for a Problem  352
C.7  Optimization Problems and Approximation Schemes  352
    C.7.1  The \( \mathcal{NP}\tilde{O} \) Class  353
    C.7.2  Approximation Algorithms  354
    C.7.3  PTAS Reductions  356
    C.7.4  \( \mathcal{NP}\tilde{O} \)-Complete Problems  356
C.8  Complexity of Network Design Problems  357
C.9  Notes and Sources  357
    References  358

D  Net2Plan  359
D.1  Net2Plan  359
D.2  On the Role of Net2Plan in this Book  360

Index  363
About the Author

Pablo Pavón Mariño is Associate Professor at the Universidad Politécnica de Cartagena (Spain) and Head of GIRTEL research group, MSc and Ph.D in Telecommunications, and MSc in Mathematics, with specialization in operations research. His research interests in the last 15 years are in optimization, planning, and performance evaluation of computer networks. He has more than a decade track as a lecturer in network optimization courses. He is author or co-author of more than 100 research papers in the field, published in top journals and international conferences, as well as several patents. He leads the Net2Plan open-source initiative, which includes the Net2Plan tool and its associated public repository of algorithms and network optimization resources (www.net2plan.com). Pablo Pavón has served as chair in international conferences like IEEE HPSR 2011, ICTON 2013 or ONDM 2016. He is Technical Editor of the Optical Switching and Networking journal, and has participated as Guest Editor in other journals such as Computer Networks, Photonic Network Communications, and IEEE/OSA Journal of Optical Communications and Networking.
Preface

Computer and communication networks have evolved into more and more complex structures of heterogeneous technologies with multiple interactions between different protocols and layers. From a didactic point of view, it is challenging to show newcomers how things are, and more importantly, why they are like that.

This is the task I faced in 2011 when starting the preparation of two basic courses in network theory fundamentals at the Universidad Politécnica de Cartagena in Spain for second and third year students doing telecommunications engineering degrees. My wish list included three musts: keep it simple, provide technology-agnostic fundamentals enriched with application examples, and make it practical.

Keep it Simple

There is added value in simplicity per se. In this case, it was also a constraint given the still incipient mathematical skills of the undergraduate students targeted. In plain words, there was no room to cover the different mathematical disciplines traditionally used in network courses, mainly queuing theory, control theory, and game theory for analysis of network protocols and the network as a distributed dynamic system, and optimization for static provisioning and dimensioning.

Within this less is more philosophy, I got convinced that optimization was the most convenient choice for a sort of didactic theory to rule them all1:

- Optimization is a popular approach in network provisioning and dimensioning problems, and fits well when the network is seen as a static system.
- The work initiated in the 1990s extended the application of optimization to capture the macroscopic dynamic behavior of network protocols. This methodology is complementary to queuing theory or a stochastic network characterization (e.g., using Markovian analysis): these are left for studying the fast timescale interactions, and their main average and system equilibrium results are then introduced and exploited in macroscopic network optimization models. Gradient projection algorithms have a prominent role in this framework for understanding network dynamics. Network protocols and their interactions among different layers are seen as gradient schemes that globally optimize a network problem. Then, stability

1 This is the only quote from The Lord of the Rings, I promise.
in a fair equilibrium solution emerges easily from the convergence properties of gradient iterations under asynchronous distributed executions, subject to delays or losses in the signaling between the nodes, or noisy observations of the network.

Eventually the goal of simplicity led to a careful compilation of a relatively reduced mathematical optimization corpus (summarized in the book’s appendices), and the quest of making the most of it to describe networks.

**Technology-Agnostic Fundamentals**

The book intends to develop a methodology for understanding and optimizing computer networks, applicable to any network technology. With this aim, the material is separated into two parts: problem modeling (Part I) and algorithm design (Part II).

- **Part I** identifies and models as constrained optimization formulations, the essential network design problems appearing in any network technology: routing the traffic, allocating capacities to the links, controlling the source rates, and deciding the network topology. Multiple real-life problem variants are included to illustrate the modeling process. When possible, Karush–Kuhn–Tucker (KKT) optimality conditions are used to give insight as to what the optimum network designs look like.
- **Part II** covers a set of mathematical techniques suitable for computer network problems. We concentrate on gradient-based algorithms for creating distributed schemes and network protocols and heuristics for offline algorithms suitable, for example, for capacity planning. Also, we show how the same technique can be applied to apparently different problems leading to different protocols. For instance, a dual decomposition approach can help to devise a decentralized transmission power allocation scheme in wireless networks or provide a cross-layer algorithm where congestion control and traffic routing cooperate to globally optimize network performance.

The book is full of examples and applications in IP, optical, and wireless networks to illustrate how the theory applies into real algorithms. We hope this prepares the reader to adapt this methodology to other existing technologies, or new technologies appearing.

**Make it Practical**

The hands-on philosophy of the book aims to permit students to perform practical optimization of networks in their homework, and the general reader to see how the ideas and mathematical approaches take real form in algorithms and models. Three practical skills are pursued:

- Formulate and obtain numerical solutions to the network problem instances, interfacing with numerical solvers.
- Implement and fine-tune the parameters of distributed network algorithms, and observe their performances and convergence under realistic scenarios, with asynchronous executions, random delays, or losses in signaling messages and subject to noisy observations of the network.
• Implement heuristic-based offline algorithms for network dimensioning and adjust their parameters to perform an efficient exploration of the solution space.

At the moment of designing the network optimization courses in 2011, no software tool was even close to match these requirements in the form I expected: easy to use, technology-agnostic, and open-source. This was the motivation to start JOM and Net2Plan open-source initiatives, the latter in collaboration up to release 0.3.1 with my Ph.D. student and colleague José Luis Izquierdo Zaragoza. JOM (Java Optimization Modeler, www.net2plan.com/jom) is a library to solve constrained optimization models written inside Java programs in a human-readable syntax, interfacing with several solvers (at this moment, GLPK and CPLEX for mixed integer linear programs, and IPOPT for differentiable programs). Net2Plan (www.net2plan.com/) is a network optimization tool that supports the fast-prototyping in Java of offline and online (dynamic) network algorithms.

Net2Plan and JOM are enabling tools for the reader interested in gaining practical skills in network optimization. All the models and algorithms in the book’s text, examples, and selected exercises are included as Net2Plan algorithms, and are freely available for inspection and reuse in:

www.net2plan.com/ocn-book

The reader is encouraged to access the web page and follow the instructions there to use Net2Plan, and get the most of this book.

Both Net2Plan and JOM are stable software. As a resource for network optimization courses they are used today by several hundreds of students in my university and other institutions. Net2Plan has also become a powerful software tool for research and industry, and is present in a number of ongoing projects.

Reader Requisites and Intended Audience

Reader prerequisites are just the basic skills to handle functions of multiple variables, at the level of a first-year university course in calculus. The book appendices are then all that is needed to follow the results in the book. These appendices include a fair amount of examples and can be the base of introductory lectures.

As a textbook, this book can support courses in different forms. Several examples follow:

• Courses in the mathematical fundamentals of computer networks. In particular, those following the technology-agnostic view, accompanied by examples in different technologies. In this context, I use Part I (modeling) in a second year course for a four-year degree in Telecommunications Engineering.
• Courses in design of distributed network algorithms. Part II of the book can help to illustrate the network dynamics and the design, implementation, and test of network distributed algorithms.
• Courses in heuristic algorithms for network planning, including development of planning algorithms. Chapter 12 and Appendix C support a third year course in my University for a four-year degree in Telecommunications Engineering, specializing in networking.
• Ph.D. courses in network optimization can benefit from advanced material in the book, such as the chapters devoted to decomposition techniques and cross-layer algorithms.
• The book can be a secondary resource for different courses in computer networks and related degrees that focus on a particular technology (e.g., wireless networks, optical networks, IP networks), and rely on this book for network dimensioning, or protocol and algorithm design for these technologies.

The book can be useful for researchers and practitioners in network planning, or protocol design for multiple network technologies. For instance:

• This book, and in particular its hands-on approach, would be quite appealing for network specialists with a limited background in optimization, to address the network problems variants appearing in their technologies of interest, making benefit of a rigorous methodology that leads to successful models and algorithms.

• In addition, practitioners in operations research willing to specialize in computer networks, will appreciate the systematic approach to categorize network optimization problems, and the consistent methodology showed in the book to apply classical optimization results to communication networks.
Acknowledgments

I am heartily grateful to the many people who have contributed in some way in the preparation of this book. First, the many colleagues and students who helped me with their day-to-day discussions. Among them, I would like to give special thanks to José Luis Izquierdo Zaragoza and Victoria Bueno Delgado for their thoughtful feedback when reading the book drafts. I am thankful to the members of the GIRTEL group for the time spent together in this fascinating work. I want to express my gratitude to the Wiley editors for their confidence, help, and cooperation in the process.

Last, but foremost, I owe my family and my friends a huge debt of gratitude. Thank you, Victoria, for your love, for putting up with my mental absences this last year, and your flexibility and support. Big thanks also to Marta, María Jesús, Sonia, and Sara for helping us with the kids. I am grateful to my sisters and my parents for being there any time I needed. Finally, Pablo and Guille. Still too small to read this. You are everything to me.
1

Introduction

1.1 What is a Communication Network?

We are surrounded by communication networks, they are part of our life. If asked, we can easily enumerate examples of them: the fixed or mobile telephone network, the Internet, or someone’s Ethernet home network would probably be the most popular answers. However, difficulties appear when we try to define the concept “communication network” more formally, without mentioning any specific network technology, looking for a definition applicable to any of them. We will start this book addressing precisely this basic question.

There are two basic elements on which networks are constructed: telecommunication systems and switching systems. A telecommunication system or “link” consists of a transmitter and one or more receivers connected through a medium that propagates the involved electromagnetic signals. Applying this definition, two telephones A and B directly connected through a bidirectional cable pair (Fig. 1.1) contain two telecommunication systems: (i) one composed of the transmitter at A, the medium $A \rightarrow B$, and the receptor at B, and (ii) another system formed by the transmitter at B, the medium $B \rightarrow A$ and the receiver at A.

Telecommunication systems are the basis for assembling any communications service. However, no service can be reasonably built by pure aggregation of telecommunication systems. As an example, imagine we want to provide the telephone service to four users A, B, C, and D, using just “links”. To do that, we would need two telephones and one cable dedicated for each different user pair. The result would be something like Fig. 1.2.

The previous example illustrates that, although possible, it is not economically feasible to provide a communication service using just links. Inefficiency in Fig. 1.2 appears because each link is dedicated exclusively to a particular user pair, and is thus idle when corresponding users are not talking each other. Improving the efficiency in our example requires adding new elements to the picture that permits a link to be shared among several communications. And that is precisely where switching systems come into play.

A switching system or “node” is a device that connects telecommunication systems (links) among them, so that the information from one link can be forwarded to other. We can represent a switching system with a node with $N_{in}$ input ports and $N_{out}$ output ports, as shown in
Fig. 1.3a. Each input port represents the receiver side of a link, each output port represents the transmitter side of other. The state of the switching system at a given moment is defined by the particular scheme in which input and output ports are internally connected. For instance, Fig. 1.3b represents a node configuration where information from input port 1 is forwarded to output port 2, while input port 3 is connected to output port 1.

The defining aspect to make a system like the one in Fig. 1.3 a switching system, is that it must be reconfigurable. That is, using mechanical, electronic, optical, or any other physical procedure, it should be possible to rearrange the internal connections between input and output ports, so that an outgoing link can be carrying at different moments the information received from different input links. Reconfigurability is the enabling feature to make output links become a shared resource among the input links.

Figure 1.4 helps us to illustrate how a combination of nodes an links supports a telephone service to seven users, using four nodes and seven (bidirectional) links. Naturally, a network like the one shown, requires extra elements to enable end-to-end communications. First,
telephones become now a more sophisticated device, since a single telephone can be used in Fig. 1.4 to communicate with any other telephone. They are actually the sources and destinations of information. An addressing or numbering scheme is now required to identify each telephone individually, so that each user can decide the destination telephone to call to. Second, a decision on the sequence of links (route) to be traversed by each telephone call should be taken and signaled to the switching nodes that must reconfigure accordingly.

The previous network scheme should be enriched with the concept of multiplexed links. Previous examples have shown links as elements that can either carry one single telephone call, or otherwise be idle. In its turn, a multiplexed link is able to simultaneously carry...
several communications, so that the aggregated traffic can be de-aggregated again at the link receiver side. Many multiplexing technologies exist, according to the physical form in which the aggregation/deaggregation takes place, the most frequent being frequency multiplexing, time multiplexing, code multiplexing, or a combination of them. The particular multiplexing technique has no importance for the abstract network model we are pursuing. What is important, to capture the essence of multiplexing, is that links should be characterized by a link capacity, measured in arbitrary units (e.g., bits-per-second – bps – in digital links). In turn, sources should now be characterized as producers of traffic measured in the same units as the link capacities. Eventually, the link capacities become the shared resource, so that we will be able to compare the capacity of a link with the sum of the traffic of the sources traversing it.

Put together, we have identified four elements in a communication network: (i) information sources and destinations, (ii) links capable of propagating information between their end points, (iii) switching nodes capable of interconnecting links in a reconfigurable form, and (iv) addressing and signaling systems for controlling network operation.

1.2 Capturing the Random User Behavior

A characteristic aspect of communication networks is that they have to deal with the random nature of the user behavior. In Fig. 1.4 this means that we do not know beforehand who each user is willing to talk to at each moment. Random behavior is crucial for the dimensioning of resources such as link capacities in the network.

In Fig. 1.4, capacities are set so that two idle users will always be able to call each other, irrespective of what other users are doing. This is called a worst-case network dimensioning. However, this approach is clearly economically unfeasible for nontrivial scenarios. Just imagine that in Fig. 1.4 10,000 phones were connected to nodes 1 and 2. A worst-case dimensioning would need a capacity of 10,000 calls for links 1-3 and 2-3. These links would be clearly underutilized, since the probability of the two sets of 10,000 users using the phone at the same time is small.

For this reason, communication network design has been historically based on probabilistic models that characterize the random user behavior. A statistical characterization of the performances observed by the users, permits dimensioning the network resources so that the service degradation (or the probability of this to happen) becomes small enough. The statistical metrics of interest can be quite variable. However, two main alternatives appear corresponding to the two main strategies network apply for link capacity sharing: delay systems and loss systems:

- **Delay systems.** Delay systems typically correspond to the form in which packet switching networks operate. In packet switching networks, traffic sources split the information into fragments called packets. Each packet is attached to a header, with sufficient control information to permit the packet reach its destination. Nodes process incoming packets one by one and forward them to their corresponding output link. Link capacities are dimensioned so that the average flow of packets that traverses a link does not exceed its capacity, potentially with some safety margins. However, the random nature of packet arrival times makes that packets could find their output links busy with other packets. For this reason, nodes incorporate memories for storing packets waiting for their turn to be transmitted. The storage time or queue time is an added delay to the end-to-end communication. Moreover,
traffic fluctuations can fill up these memories and force the nodes to discard packets. In these systems, the delay and the packet loss probabilities are the performance metrics of interest.

- **Loss systems.** Loss systems can appear in networks where the traffic takes the form of end-to-end connections. These connections can be circuits, in so-called circuit switching networks (e.g., telephone calls in the telephone network, lightpaths in WDM optical networks), or virtual circuits over packet switching networks (e.g., virtual circuits in MPLS networks). Each connection reserves a given amount of capacity in each of the traversed links. The reserved bandwidth is kept throughout the communication. If a connection request does not find a sequence of links with enough capacity, it is discarded (lost or blocked). The probability that a new connection request is blocked, or blocking probability, is the main performance metric for dimensioning loss systems.

### 1.3 Queueing Theory and Optimization Theory

Queueing Theory has been traditionally the mathematical corpus supporting the design of computer or communication networks, capturing the non-deterministic user behavior and network occupation. This theory focuses on systems where a set of resources (e.g., links in a network) are shared among different users, so that the moment in which each user requests a resource (e.g., a new connection or a packet to transmit), and/or the time in which the resource is to be occupied (connection/packet duration), are known statistically. As such, Queueing Theory is a branch of Applied Probability.

Queueing Theory has been extremely successful and elegant for analyzing network subsystems such as the performance of the traffic traversing a link or a switching node. However, applying this strategy to a network view is extraordinary challenging. Actually, the probability models used in Queueing Theory become intractable when applied to non-trivial communication networks, unless strong simplifying assumptions are made. The essential disadvantage is that the exact statistical characterization of the traffic traversing several links and/or merging with other users traffic, can be mathematically intractable. For these situations, the combination of Queueing Theory and Network Optimization has shown to be a successful approach.

Optimization is a branch of Applied Mathematics, studying the maximization or minimization of functions, subject or not to a set of constraints. Network optimization is just the application of standard general optimization results to network design problems, exploiting any special structure they may have. Decision variables in these problems take the form of, for example, the capacity to assign to each link or the amount of a user traffic to route in each valid path.

The application of network optimization requires getting rid of the unreachable goal of a full statistical characterization of the traffic in the links. In its turn, in most of the cases (as happens in this book), the traffic in network design problems is modeled as a continuous flow, not identifying individual connections or packets in it. Flows are simply characterized by a real number: its **average intensity**, measured in bits-per-second, number of active connections, or any suitable unit. Characterizing a traffic flow just by its average yields a significant simplification: the intensity of the traffic in a link aggregating the contributions of several flows, is just the sum of the intensities of the traversing flows.

Previous simplification, opens the door to introducing into the optimization models expressions coming from Queueing Theory that estimate performance metrics like delay or blocking probability, as a function of **average** flow intensities. In general, these relations are closed
Optimization of Computer Networks – Modeling and Algorithms

formulas that assume as constants other parameters (e.g., variance or Hurst parameter of the traffic flows), and result in simple expressions. Thanks to that, these network performance estimations can be introduced into the optimization model as objective functions (e.g., minimize the average network delay), or design constraints (e.g., blocking probabilities should be below 1%), without significantly augmenting the problem complexity.

1.4 The Rationale and Organization of this Book

The casuistic of the different network design problems that can exist is unlimited, as the multitude of technologies, protocols and heterogeneous network conditions yield to an unnumbered amount of variants. Since it is impossible to study them all, the target of this book is providing a methodology that can be applied whatever network technology we focus on, and eventually results in (i) insights to understand the network behaviors and (ii) design keys to produce algorithms that solve network problems.

The mathematical corpus of our approach is network optimization, or the application of general optimization theory to network problems. In this strategy, we distinguish two main steps, elaborated into the two parts (Part I and Part II) in which this book is divided: problem modeling and algorithm design. This is followed by a small set of appendices covering basic optimization concepts used throughout the book.

1.4.1 Part I: Modeling

In this part, we pursue the modeling of network design problems appearing in communication networks, as optimization problems. This means translating them into the problem of finding the values of a vector \( x \) of decision variables that maximize or minimize an objective function, subject to some constraints. The reasoning of the approach is that, once a problem is modeled, we can benefit from optimization theory to characterize what their optimum solutions look like and how to reach them.

For didactic purposes, four problem types are addressed separately: (i) traffic routing, (ii) link capacity dimensioning, (iii) bandwidth sharing among network users (i.e., congestion control), and (iv) topology design. These problems appear in different forms in all network technologies. In this part of the book, multiple variants of each are described and analyzed, covering most of the aspects appearing in production networks. The source code for Net2Plan tool of the examples and selected exercises is available for the reader. Then, the he/she will be able to find numerical solutions to these problems in any network instance, using the solvers interfacing with Net2Plan.

Part I chapter organization is described below:

- **Chapter 2. Definitions and notation.** Key definitions and the notation that will be used along the book is presented. We elaborate on concepts like network links, nodes, traffic demands (unicast, anycast, multicast, broadcast), and routing (bifurcated, non-bifurcated, integral).
- **Chapter 3. Performance metrics.** This chapter describes different performance metrics that are commonly used within network optimization models: delay and blocking probabilities in packet-switched and circuit-switched networks, respectively, average number of hops, network congestion, network cost, network availability in failure scenarios, and fairness.