Introduction

Communications systems continue to evolve rapidly. Users continue to demand more high-performance networking capabilities. Service providers respond to this demand by rapid expansion of their network infrastructure. Network researchers continue to develop revolutionary new communications techniques and architectures to provide new capabilities commensurate with evolving demands. Equipment vendors continue to release new devices with ever-increasing capability and complexity. Technology developers rapidly develop next-generation replacements to existing capabilities to keep up with demand. These rapid developments in the network industry lead to a large, complex landscape.

The network designer and developer wants (and needs) to satisfy the demands of the users. This is difficult, as it is often complicated for the typical network engineer to fully understand this rapidly evolving communications landscape. This challenge is exacerbated by the nature of emerging technologies and techniques that are often extremely complex compared with their legacy counterparts. This leaves the typical network engineer with more questions than answers. The network engineer tasked with maintaining an operational network might ask the following: What is the right approach to solving my problem? Do I buy the latest device from company X that claims to solve all my problems? Do I replace the underlying technology of my system with the latest generation? How do I know whether a technology is mature enough to survive the rigors of my application? How do I know how my already existing network system will respond if I add this device? The network engineer researching next-generation networking techniques might ask: How do I know how this new approach will interact with already-existing protocols? or How do I build confidence in the utility of this approach without producing and deploying the technology? The network engineer developing a particular product might ask: How do I ensure that this design will satisfy requirements

An Introduction to Network Modeling and Simulation for the Practicing Engineer, First Edition. Jack Burbank, William Kasch, Jon Ward.

^{© 2011} Institute of Electrical and Electronics Engineers. Published 2011 by John Wiley & Sons, Inc.

before I go to production? or How can I assess the utility of a design choice compared to its envisioned cost? This book aims to help answer these questions.

There are many tools available to the network engineer that can assist in answering these questions, including analysis, prototype implementation and empirical testing, trial field deployments, and modeling and simulation (M&S). It should be stated now that no one tool is typically sufficient in understanding the performance of a network; unfortunately, there is no "silver bullet" answer to all our questions. The complex nature of emerging systems also introduces significant complexity into the effective evaluation of these systems and how these various tools can be employed. Evaluation is often conducted through the coordinated usage of analysis, M&S, and trial deployments in closely monitored environments. Due to the costs and complexities of deployments, analysis and M&S are often used to determine the most sensitive performance areas that are then the focus of trial deployments. This limits the scope of the trial deployment to a realistic level while focusing on the important cases to consider.

Because of the increasingly interconnected nature of communications systems, and the resulting interdependencies of individual subsystems to operate as a whole, it will often be the case that individual subsystems cannot be tested in isolation. Rather, multiple systems must be evaluated in concert to verify system-level performance requirements. This increases the required scale of trial deployments and adds significant complexity as now several different types of measurements will often be required in several different locations simultaneously. This increases the required support for a deployment in terms of required resources, including personnel and measurement equipment, further limiting the realistic amount of trial deployments. Thus, this will place a premium on analysis and M&S to perform requirements verification and to form the basis of any performance evaluation. In many cases, M&S may provide the only viable method for providing insight into the behavior of the eventual system prior to full-scale deployment.

Once the importance of M&S is established, many additional questions still arise: How does the network engineer properly employ M&S? What are the most appropriate M&S tools to employ? While networking technologies continue to evolve rapidly, so too do M&S tools intended to evaluate their performance. The M&S landscape is indeed a complicated space with a multitude of tools with a variety of capabilities and pitfalls. Furthermore, there is often a poor understanding of the proper role and application of M&S and how it should fit within the overall evaluation strategy. There is even confusion surrounding the term M&S itself. Before we continue, let us provide some basic definitions that will be used throughout the book.

Modeling and simulation (M&S) are often combined as a single term. However, a model is quite different than a simulation. This book defines these two entities as:

- *Model*: A logical representation of a complex entity, system, phenomena, or process. Within the context of communications and networking, a model is often an analytical representation of some phenomena (e.g., a mathematical representation for the output of a system component) or a state machine representation. This analytical representation can either be in a closed form or an approximation obtained through assumptions.
- *Simulation*: An imitation of a complex entity, system, phenomena, or process meant to reproduce a behavior. Within the context of a communications network, a simulation is most often computer software that to some degree of accuracy functionally reproduces the behavior of the real entity or process, often through the employment of one or more models over time.
- *Emulation*: An imitation of a real-world, complex entity or process meant to perfectly reproduce a behavior or process. Emulation can be thought of as perfect simulation of something such that it is equivalent to the original entity.

To illustrate the difference between a model and a simulation, consider a simple signal detection circuit. A simulation of this device would imperfectly mimic the various actions of the detection circuit to determine a likely outcome for a given input. A model of this same device would generally take the form of a mathematical algorithm that would produce (either perfectly or imperfectly) an output for a given input.

Unfortunately, the terms *model* and *simulation* are often incorrectly used interchangeably. Generally speaking, the term simulation has wider scope than the term model, where a simulation is typically a compilation of models and algorithms of smaller components of the larger overall entity or process. This book generally uses the combined term M&S to generically refer to the employment of models, simulations, and emulators to approximate the behavior of an entity or process.

There are numerous types of computer models and simulations. A computer model or simulation can generally be classified according to several key characteristics:

• Stochastic vs. Deterministic: Deterministic models are those that have no randomness. A given input will always produce the same output given the same internal state. Deterministic models can be defined as a state machine. Deterministic models are the most common type of computer model. A stochastic model does not have a unique input-to-output mapping and is generally not widely employed, as it leads to unpredictability in execution. A simulation can be made to act in a pseudo-random manner through the employment of random number generators to represent random events. However, the particular models governing the behavior of each component within the simulation are generally deterministic.



FIGURE 1-1. A block diagram of a wireless communications system simulation.

- Steady-state vs. Dynamic: Steady-state models attempt to find the input-to-output relationship of a system or entity once that system is in steady-state equilibrium. A dynamic simulation represents changes to the system in response to changing inputs. Steady-state approaches are often used to provide a simplified model prior to dynamic simulation development.
- Continuous vs. Discrete: A discrete model considers only discrete moments in time that correspond to significant events that impact the output or internal state of the system. This is also referred to as a discrete-event (DE) model or DE simulation. This requires the simulation to maintain a clock so that the current simulation time can be monitored. Jumps between discrete points in time are instantaneous; nothing happens between discrete points in time corresponding to interesting events. Continuous simulations consider all points in time to the resolution of the host's hardware limitations (all computer simulations are discrete to some extent because of the fact that it is running on a digital platform with a finite speed clock). DE methods are the most commonly used for network M&S.
- Local or Distributed: A distributed simulation is such that multiple computer platforms that are interconnected through a computer network work together, interacting with one another, to conduct the simulation. A local simulation resides on a single host platform. Historically, local simulations have been the most common. But the increasing complexity of simulations have increased the importance of distributed simulation approaches.

In general, a simulation can be thought of as a piece of software residing on a computer platform that implements a set of algorithms and routines and takes a set of inputs to produce a set of outputs that represent the behavior of the system of interest. This is depicted in Figure 1-1.

The typical inputs that are important to consider when simulating a wireless network are summarized in Table 1-1. The typical outputs that are often of interest are summarized in Table 1-2.

Parameter	Explanation
Signal power	This will influence the received power level and consequently the Bit Error Rate (BER) and Packet Error Rate (PER) performance of the wireless link.
Waveform type	This will influence the BER and PER performance of the wireless link in a given channel.
Forward error control coding (FEC) method	This will influence the BER and PER performance of the wireless link in a given channel.
Retransmission protocol	This will affect the throughput and delay performance of the wireless link.
Contention method	This will influence BER, PER, throughput, and delay performance of the wireless link in a given channel.
Channel model	This will determine the performance of a given wireless link in terms of received power level, BER, and PER.
Mobility model	This will impact the performance of the MAC layer protocol and of the higher layers (e.g., IP routing).
Traffic model	This will impact the performance of the MAC layer protocol and of the higher layers (e.g., IP routing).
Network topology	This will impact the performance of the MAC layer protocol and of the higher layers (e.g., IP routing).

TABLE 1-1. Typical Inputs to a Wireless Network Simulation

Parameter	Explanation
BER	The fundamental performance metric of a digital communications link.
PER	Often considered the most important performance metric in a packet-switched network.
Throughput	The data rate supportable by the wireless network.
Goodput	The useful data rate supported by the wireless network (i.e., data rate as available by the application).
Latency	The end-to-end delay that an application or user will experience across the wireless network.

TABLE 1-2. Typical Outputs from a Wireless Network Simulation

6 INTRODUCTION

1.1 ADVANTAGES AND DISADVANTAGES OF MODELING AND SIMULATION

As is the case with any tool, M&S has both advantages and disadvantages. This section provides a tradeoff framework for the designer or developer to consider when choosing to employ M&S. In the following section, M&S is often compared with empirical testing. For the purposes of this book, empirical testing refers to real-world testing of equipment (e.g., physical hardware devices) deployed in a physical environment.

1.1.1 Breadth of Operational Scenario

First and foremost, M&S provides the ability to exercise a wide range of operational scenarios. Empirical testing will exercise a much smaller portion of the possible scenario space than will M&S. This includes the ability to evaluate greatly increased network scale (e.g., number of network nodes), not easily achieved in empirical activities, and more dynamic choice of environmental conditions (e.g., wireless environment). Because of the ability to exercise a wide variety of scenarios, M&S has a clear advantage in this aspect.

1.1.2 Cost

Generally, another advantage of M&S is reduced cost compared with empirical testing and trial deployments. Extensive empirical testing carries a high cost, to the point where extensive empirical-only approaches are largely impossible in the modern wireless networking landscape; however, this advantage is dependent on the scope placed on the M&S development effort.

1.1.3 Confidence in Result

A less obvious advantage of M&S is the amount of precision and control that can be exerted over the scenario in question. In the empirical scenario, measurements are taken and then those measurements are analyzed and understood for their ramifications. However, due to the uncontrolled nature of empirical testing, there are often many variables that affect the measurement. And often the number of uncertain variables is so great that it is impossible to isolate the source of any behavior or to correlate a measurement to its source (i.e., map the effect to the cause). This limits the scientific utility of such measurements, and makes it difficult to associate a high degree of confidence to the measurement. The "the data is what it is" philosophy is rarely justified if the phenomena under observation are not understood. Note, this is much more the case for over-the-air (OTA) empirical activities. Other empirical activities are much more highly controllable (e.g., direct radiofrequency (RF) chain testing). The primary, and most obvious, disadvantage of M&S is that it is not real. It is a representation of the system, rather than the system itself. There are several assumptions that will be built into any M&S tool. Some of these assumptions will be necessitated by real-world complexities that are not easily represented. Others are necessitated by a lack of information available about the system in question. This will naturally lead to inaccuracies. Consequently, this leads to a decreased confidence in results. This confidence decrease is manageable, however, through verification and validation activities, often in conjunction with empirical activities to improve confidence in such models.

A higher degree of confidence is almost always associated with empirical methods, regardless of the methodology or practices employed during those empirical activities. Unfortunately, this confidence can be ill placed. The common belief is that M&S-based methods are more subject to error because software-based "bugs" could introduce unforeseen inaccuracies. And while that is definitely true, the same applies to the empirical-based approach. Any empirical measurement will have error associated with it (e.g., imperfections in hardware employed to make a measurement, misconfiguration of test equipment). Also, human interpretation must at some point be applied to understand an empirical measurement. This human interpretation can be influenced by assumptions, biases, and preconceived opinions.

Another issue is that of statistical significance. Even if measurement error has been minimized, there are several factors that can influence the significance of that measurement. Take, for example, the measurement of an antenna pattern, which is a key characteristic that will impact wireless network performance. This antenna pattern will vary across antenna population due to manufacturing variation, differences in platform, and differences in age and condition. Furthermore, the RF propagation environment characteristics will be temporal in nature. Thus, a particular measurement is somewhat insignificant in the overall sense. In fact, to make empirical activities truly significant from a statistical standpoint is often cost prohibitive.

With all these factors considered, an empirical approach is still considered to have an advantage, especially if issues such as measurement error and uncertainty are built into empirical activities. However, the proper application of verification and validation practices can help minimize this difference.

1.1.4 Perception

Even if a model is highly accurate, and from a scientific perspective is highly regarded, there is the issue of perception. Many individuals will still remain skeptical of the results from a computer model. This is due to sociological and psychological phenomena that are well beyond the scope or timeframe of any particular M&S activity. Rather, this reality must be accepted and factored into the overall evaluation approach. An empirical-based evaluation method has the overwhelming advantage in this area. In fact, this advantage is so

8 INTRODUCTION

strong that some degree of empirical testing is likely required to give credibility to the findings of the overall M&S activity.

1.1.5 The Need for Verification and Validation

While not considered a disadvantage, certainly a burden associated with M&S is the need to conduct verification and validation (V&V) activities. Such activities are generally required to both verify the accuracy and consistency of model output and validate output relative to other models, empirical tests, and theory. While V&V activities are mandated by good software engineering principles and must be adhered to, the formality of a V&V process can levy significant resource requirements on a project. This partially negates the cost advantage of M&S over empirical testing.

In some sense, M&S is disadvantaged in this regard compared with other tools available to the network engineer. As mentioned previously, there is typically less scrutiny placed on empirical measurements and, consequently, there is typically a greater "burden of proof" placed on an M&S developer as compared with the empirical tester.

1.2 COMPARISON OF "HOMEBREW" MODELS AND SIMULATION TOOLS

Custom simulations, or "homebrew" solutions, are those in which the implementer does not rely on any existing tools but rather develops the simulation in its entirety. The advantages of homebrew simulations include:

- The implementer knows exactly what has been implemented.
- Homebrew solutions can have significant performance benefits.

The disadvantages of homebrew simulations include:

- They can be costly to develop.
- They can be difficult to upgrade.
- There is a real risk of these custom simulations not being widely adopted, even within your organization (resulting in perpetual "homebrew" solutions).

Other than small-scale efforts that are supporting analysis, homebrew approaches are generally discouraged. With the ever-increasing complexity of wireless networking systems, the feasibility of a meaningful homebrew solution is dwindling. Even for cases where there are no existing implementations of a particular networking technology and code development is inevitable, it is recommended that this new custom simulation be developed within existing tools/environments so that it can be integrated with and leverage existing simulation libraries.

1.3 COMMON PITFALLS OF MODELING AND SIMULATION AND RULES OF THUMB

There are many potential pitfalls that face those who embark on a network simulation development effort. This section discusses some of those most commonly seen.

1.3.1 Model Only What You Understand

It can be said that the utility of a given model is only as good as the degree to which it represents the actual system being modeled. Indeed, a system whether a wireless network or otherwise—can only be modeled once it is sufficiently understood. While this is a simple tenant, it is one that is certainly not adhered to universally by M&S designers. One may ask why M&S designers develop invalid models. There are many reasons, the first of which is that high-fidelity model development requires a significant investment of time and effort. This statement is not meant to offend developers or to imply carelessness on their part. The fact is that many designers are under time constraints to deliver results. Consequently, a careful understanding of the underlying system being modeled and rigorous validation of the model is not always an option.

While understandable, this is at the same time unacceptable. It is highly unlikely that a simulation developer can provide a meaningful result when they did not understand the system they were intending to model. While the timeline might have been met, the result was likely meaningless. Worse yet, the result was likely wrong and might have adversely affected larger design or business decisions. *Model only what you understand!* If you don't have a fundamental understanding of a technology, there is no way you can effectively model or simulate that technology. This step cannot be skipped in a successful M&S effort. If this step cannot be completed, it is better to not proceed down the path of M&S development.

1.3.2 Understand Your Model

It is quite common for the network engineer to utilize off-the-shelf tools, either commercial or open source. This approach typically lends itself to a faster M&S development cycle; however, it is imperative that the network engineer has a full understanding of the tools being used. Most simulations are likely to have errors—even commercial tools. New simulation implementations almost always contain errors. Simulation implementations can make assumptions that may not accurately reflect the exact performance metric of interest.

9

If the simulation developer utilizes existing simulation implementations, it is imperative to allocate the proper amount of time to closely examine that implementation to fully understand what that code is doing and what it is not doing. There is no better way to lose credibility than to not be able to answer questions about one's own results. *Understand what you have modeled!* There are resources available to help with this, including technical support for commercial tools, online newgroups and user forums for open source tools, and in some cases the simulation designer can contact the author directly (e.g., a contributed simulation to an open source project).

1.3.3 Make Your Results Independently Repeatable

Many academic papers such as [1–3] have discussed the lack of independent repeatability in wireless network simulation results due to improper documentation of the simulator being utilized, model assumptions, and inputs and outputs. There are subtle parameters and assumptions embedded in simulators such as NS-2 and GloMoSim that certainly can impact all results. Often default simulator parameters are chosen that may not capture the intended network conditions for a given scenario [2]. Perhaps the larger problem is that simulation results are often presented as ground truth and not as a relative ranking of a new idea compared to existing ideas. That is, the literature survey component must always be present in wireless network research and simulation results should be compared to existing results to demonstrate advantages and disadvantages of new ideas. Moreover, new simulation results must be compared with results in existing literature using the same simulator, underlying assumptions, and parameter conditions.

1.3.4 Carefully Define M&S Requirements

This is an activity that is too often ignored or given superficial treatment. The authors would argue that network engineers all too often rush into an M&S effort without a clear idea of what they are hoping to accomplish. This is a surefire recipe for failure.

The first step is to clearly understand the metrics of interest that would be generated by a simulation. Is overall network throughput the metric of interest? Is BER the metric of interest? End-to-end delay? Not all simulation tools necessarily lend themselves to the same types of output metrics, so it is important to define these metrics so that tool selection is an informed process.

The next step is to clearly define the required performance of the simulation to be developed. This book contends that there are four primary dimensions of performance:

• *Cost:* The overall investment in resources towards the development and maintenance of the M&S activity. This includes not only original platform

costs, but also development time, upgrade and maintenance costs, and troubleshooting.

- *Scalability*: The total complexity of the system to be simulated. There are two factors that must be considered: network size in terms of number of nodes, and network traffic model in terms of number of messages per unit time. These two factors will drive the computational complexity of the simulation and will ultimately be the limiting factors in the size of the network that can be simulated. This is generally governed by software complexity and hardware capability.
- *Execution Speed*: For a given simulation scenario, how quickly can that simulation complete and provide the desired output metrics? This is generally governed by software complexity and hardware capability.
- *Fidelity*: For a given simulation scenario, how accurately does the simulation's output metrics reflect the performance of the real system.

Note that these dimensions of performance are contradictory; not all performance dimensions can be achieved simultaneously. If you desire a highly scalable simulation with fast execution speed, then the fidelity is likely going to be lower. Do you want high fidelity and scalability with reasonable execution speed? Then the cost will likely be very high. In general, you can pick any three of these metrics.

A common pitfall is to begin an M&S effort with unrealistic expectations. Is it really feasible to model the entire Internet down to every platform with bit-level fidelity? Probably not. Is it possible to model the entire Internet down to every platform with many simplifying assumptions? Probably, but it is unlikely to be useful.

When defining requirements and expectations for an M&S effort it is recommended to begin by choosing the required fidelity. How accurate of an output metric is required? A successful effort will always begin with this metric because, without a meaningful degree of fidelity, any M&S activity is meaningless, despite its scalability or execution speed. Once the required fidelity is established, one can then begin placing limitations on simulation capabilities accordingly. Cost is generally bound by an allocation of resources. So given a known cost constraint and a known fidelity requirement, we can then begin building a conceptual model for the simulation. The target fidelity will mandate the inclusion of particular system characteristics with great detail and inputs with particular degrees of accuracy, and also allow for relaxation on other system details and input accuracy. Note that this exercise requires a strong understanding of the system being modeled and on the underlying concepts of wireless networking. Remember, model only what you understand! Once a conceptual model is designed, the hardware platform can be chosen in accordance with cost constraints to maximize scalability and execution speed performance.

1.3.5 Model What You Need and No More

One of the first decisions that the simulation designer must face is to determine what he or she is attempting to demonstrate through simulation and what is the most simplistic model that captures all necessary components. The engineering tradeoff is that increased detail can provide higher-fidelity output from the model, but at the cost of complexity-potentially introducing error and certainly increasing debugging time and execution time. The designer must also realize that a model is always an abstraction from the real world. Wireless networking devices not only have variables within the standards to which their underlying protocols comply, but there is variability introduced into each manufacturer's products. At least a subset of the key variables should be included: transmission power, antenna type and gain, receiver sensitivity, and dynamic range should be considered in the model, but the extent of modeling detail required depends on the particular system and desired output for a given scenario. Regardless of the level of detail included, a simulation will always be an approximation of the real system; an arbitrarily high degree of fidelity is generally not possible. Also, the cost of increased fidelity at some point becomes greater than the marginal utility of the additional fidelity. This is illustrated in Figure 1-2. It is imperative to understand the limitations of M&S techniques and to understand the relationship between cost and fidelity so that an M&S effort does not become an over-engineered effort in futility.

How much detail is sufficient in a simulation to capture the essence of the real-world network being modeled? Unfortunately, the answer to this question is that it depends on the particular simulation scenario. The reader should first decide exactly what is the problem that he or she seeks to address through simulation. What are the inputs and the outputs of the model? Some outputs may be independent of specific details in the model, while others may be correlated and therefore seriously affected if those components are abstracted.



FIGURE 1-2. The cost of simulation fidelity.



FIGURE 1-3. Illustration of the simulation trade space.

Simulation always takes the form of an abstraction of a system to allow the designer to gain some insight from investigating various operating scenarios of the system. In many cases the simulation allows the user access to knobs and switches that may not be available on the actual system. Consider reliability testing for a consumer networking product that must be tested under as many operating conditions as possible, where the prevention of erratic behavior in a consumer product translates to significant savings for a company. Yet in other cases the researcher desires to investigate a system's reaction to a single condition that may be unlikely to occur in real life. Perhaps testing the actual system under this condition could be harmful and simulation is the only way to examine the problem.

The next step is to decide how much of the system must be implemented for the simulation results to be valid. Ultimately, the reader is going to have to decide the level of detail required in his or her simulation, but this book is intended to guide the reader towards formulating a more educated decision. First, the reader must consider the engineering tradeoffs between adding more detail to a model and increased computational time, increased complexity, and increased debugging time. This simulation trade space is illustrated in Figure 1-3. A more complex simulation may attempt to capture an actual system's complete behavior, but at that point the simulation is generally inflexible to scenario modifications, more prone to errors, and more computationally intensive. A more abstract approach that focuses only on the basic behavior of a system is generally very flexible, easier to debug, and has a shorter execution time. But, it may not capture the behavior of interest.

1.3.6 Avoid M&S When M&S Does Not Make Sense

The purpose of this book is not to help the user decide when simulation is the appropriate method to investigate a given problem. There are too many possible networking scenarios to make these types of recommendations. It is therefore assumed that the reader has already decided that simulation is the best method to apply to a given problem. But, this book will offer some basic advice to help the reader avoid the wrong path. Let us assume that you have performed the initial requirements definition and that the fidelity required for your application includes every detail of a technology standard down to every bit, byte, protocol, and state machine. In this case, you are likely on the upper end of the cost vs. fidelity function of Figure 1-2 and it may not make sense to even pursue an M&S activity. In this case, it may make more sense to just implement a prototype of the device/system and test it empirically. If that is not practical because of the cost and size of the final system, then it is important to understand the cost that will be incurred by M&S, or such efforts may have to be scaled back to a lower degree of fidelity to manage cost. It is ultimately up to the reader to decide if M&S is right for their particular effort.

1.3.7 Channel Models

A quick search of open literature will uncover a plethora of highly complex models of wireless networks and proposed protocols/techniques that are evaluated only in Additive White Gaussian Noise (AWGN) environments (none are referenced here to protect the names of the innocent). This is perfectly fine for many cases. With that said, however, do not expect to model an omnidirectional antenna wireless network in AWGN conditions and be able to make any statements regarding how that system will behave in complex urban environments. It is sometimes a daunting task to provide high-fidelity channel models in large simulations. This is well understood. But it is important to understand and clearly communicate the limitations of the model to constrain performance statements, particularly if those performance statements are going to form the basis for design or business choices. Common RF channel models are discussed in detail in Chapter 2.

1.3.8 Mobility Models

There are many papers in open literature that present the types of mobility models to use when simulating wireless networks (e.g., [5]). However, it is important to understand that, while mobility models will have a profound impact on the performance of the network, they are usually arbitrary and hardly ever reflect reality. It is indeed difficult to predict the true mobility patterns of network users, particularly future patterns. It is important for the simulation designer to do his or her homework and construct the best educated guess when formulating mobility models for use in simulations. It is also important to perform sensitivity analysis to understand how the metrics of interest change with different mobility models to understand the M&S limitations for a particular application. Simplistic assumptions combined with the lack of expectation management can (and usually will) haunt you!

1.3.9 Traffic Models

Like the case of mobility models, traffic models usually have a profound impact on the performance of a network. And, unfortunately, like the case of mobility models, traffic models are also usually arbitrary and hardly ever reflect reality. It is generally possible to construct realistic current traffic models based on traffic monitoring and analysis. But in the case of a new network deployment, it is difficult to ascertain the true pattern of usage. And it is also very difficult to predict future usage patterns since applications evolve rapidly. It is important for the simulation designer to do his or her homework and make the best educated guess possible. However, be cognizant that these are still guesses, best case. It is also imperative to perform sensitivity analysis to understand how the metrics of interest can change with changes in traffic patterns to understand the M&S limitations of a particular application. Again, simplistic assumptions combined with the lack of expectation management can (and usually will) haunt you!

1.3.10 Over-reliance on Link Budget Methods for Abstraction

Even in simulation environments, it is common to simplify complex aspects of the system and turn them into static "losses" in link budgets (e.g., signal quality adjustments at a receiver to represent some physical phenomena causing degradation). This is fine for a simple, steady-state analysis. But in the more general dynamic case, beware that losses are typically scenario dependent. In this case, it is important to understand the degradation source and its sensitivity to scenario-dependent variables. Once sensitive variable relationships are understood, then a potential approach would be to pre-compute these losses as a function of sensitive variables and store them for real-time lookup (e.g., tabular lookup). This will increase simulation fidelity with a negligible impact on execution speed.

1.3.11 Overly Simplistic Modeling of Radio Layers

It is a common practice for network simulations to not perform true bit-level simulations of the lower layers of the protocol stack. Rather, these lower layers are often abstracted into "clouds" with a static probability of performance metrics such as errors and delay. This approach is understandable given the challenges in bit-level simulations of large networks; however, this approach can lead to misleading results as it removes many dynamic aspects of system performance. It is important to understand the impact of these "averaging" approaches on simulation outputs and to manage expectations accordingly.

1.3.12 Disjoint M&S and Implementation Efforts

Too often M&S activities are disjoint from implementation efforts. This is unfortunate since a bit-true simulation can be a great interim milestone towards a real-world implementation and has the leave-behind value of a high-fidelity model. These activities should be tightly coupled. This is increasingly true as large companies continue to expand globally and development teams may be located on different continents instead of working side-by-side. While globalization has increased, so too have the tools to allow remote video teleconferences (VTCs) and information sharing. Hardware and software design tools such as LabVIEW Field Programmable Gate Array (FPGA) [134] or the Xilinx System Generator for Digital Signal Processing (DSP) Simulink blockset [133] also facilitate the conversion of a software model to a hardware implementation.

1.4 AN OVERVIEW OF COMMON M&S TOOLS

There are numerous network M&S tools available either as commercial products or as open source. This section provides a brief introduction to many of these tools. Table 1-3 provides a summary of many of the available network M&S tools [1].

Perhaps the four most commonly used network simulation tools in both academia and industry are Network Simulator 2 (NS-2), OPNET, QualNet, and GloMoSim. A short description of each follows.

1.4.1 NS-2

NS-2 is an open source DE simulator targeted at supporting network research. NS-2 is popular in academia because of its low cost (free) and extensibility. NS-2 was originally developed in 1989 as a variant of the REAL network simulator and, according to the NS-2 home project URL (see Table 1-3), "provides substantial support for simulation of TCP, routing, and multicast protocols over wired and wireless (local and satellite) networks."

NS-2 was built in the C++ programming language and provides a simulation interface through OTcl, an object-oriented extension of the scripting language Tcl. NS-2 will run on several forms of Unix (FreeBSD, Linux, SunOS, Solaris) and has been extended to Microsoft Windows (9x/2000/XP) using Cygwin (http://www.cygwin.com), which provides a Linux-like environment under Windows.

Network Simulation Tool	URL
BRITE	http://www.cs.bu.edu/brite
Cnet	http://www.csse.uwa.edu.au/cnet/
GloMoSim	http://pcl.cs.ucla.edu/projects/glomosim
J-Sim	http://www.j-sim.org/
Matlab*	http://www.mathworks.com/products/matlab/
NS-2	http://www.isi.edu/nsnam/ns/
OMNeT++	http://www.omnetpp.org/
OPNET*	http://www.opnet.com/
PacketStorm-Network	http://www.packetstorm.com/4xg.php
Emulator*	
QualNet*	http://www.scalable-networks.com
Simulink*	http://www.mathworks.com/products/simulink/
SSFNet	http://www.ssfnet.org/homePage.html
x-sim	http://www.cs.arizona.edu/projects/xkernel/
NetSim*	http://www.tetcos.com/software.html
GTNetS	http://www.ece.gatech.edu/research/labs/maniacs/gtnets/ index.html

TABLE 1-3. Available Network Simulation Tools

*Denotes a commercial product.

NS-2 is currently licensed for use under version 2 of the GNU General Public License. Documentation has historically been poor for NS-2, with users left to rely on online user forums and newsgroups; however, there have been additional information sources emerging recently that may help someone new to NS-2, such as [6, 13].

1.4.2 **OPNET**

OPNET Technologies was founded in 1986, becoming a public company in 2000. The company provides a suite of software tools for network designers and administrators. But its flagship product is OPNET Modeler, which is a software tool for network M&S that was originally developed by the company's founder as a graduate project while at the Massachussetts Institute of Technology (MIT). OPNET Modeler is designed to either evaluate changes to existing networks or to design proprietary protocols. Furthermore, OPNET contains detailed models of specific network equipment. OPNET Modeler provides integrated analysis tools and a rich Graphical User Interface (GUI) as well as animation capabilities for data visualization. User development is in C/C++ and XML languages.

OPNET is slightly less common in academia as compared with NS-2, but is widely used in a variety of commercial and military organizations.

1.4.3 GloMoSim

The Global Mobile Information System Simulator (GloMoSim) is a DE simulator developed by the Parallel Computing Laboratory at UCLA in the C programming language and based on the parallel programming language Parsec. GloMoSim currently supports wireless protocols, which limits its utility in wired or hybrid networks. However, according to the GloMoSim project page (see URL in Table 1-3), there is currently development underway for a future revision that supports wired protocols. GloMoSim is available only to academic users; in fact, only users from an .edu domain are allowed to access the download page.

1.4.4 QualNet

QualNet is the commercial spin-off of the GloMoSim simulator offered by Scalable Network Technologies. QualNet is based on the C++ programming language and provides either command line or GUI interface to the user. QualNet provides a wide range of wired and wireless protocol support. Its key selling point is its high degree of scalability, which can supposedly "support simulation of thousands of network nodes" with high fidelity [16].

1.5 AN OVERVIEW OF THE REST OF THIS BOOK

This book takes a bottom-up approach to describing wireless network M&S, following the TCP/IP modified OSI stack model shown in Figure 1-4, recreated from [1].



FIGURE 1-4. Wireless network simulation example demonstrating the interaction between various components [1].

This book first decomposes the wireless network M&S problem into a set of smaller scopes as depicted in Figure 1-4: 1) radio frequency (RF) propagation M&S (Chapter 2), 2) PHY M&S (Chapter 3), 3) MAC M&S (Chapter 4) and 4) higher layer M&S (Chapter 5). After considering each of these smaller scopes somewhat independently, the book then revisits the overall problem of how to conduct M&S of a wireless networking system in its entirety.