Network Modeling and Simulation with Mixed Traffic Data Sources

Lieutenant Colonel John A. Hamilton, Jr., US Army
Joint Forces Program Office (Code 05J1)
Space and Naval Warfare Systems Command
San Diego, CA 92110-3127
drew@drew-hamilton.com
http://www.drew-hamilton.com

Colonel John C. Deal, US Army
Commander
US Army Information Systems Engineering Command
Fort Huachuca, AZ 85613-5300
dealj@hqisec.army.mil

Colonel John R. Grobmeier, US Army
Director
Technical Integration Center, US Army ISEC
Fort Huachuca, AZ 85613-5300
grobmeierj@hqisec.army.mil

KEYWORDS
Network simulation, emulation, OPNET, hybrid models

ABSTRACT
Advancements in the field of mixed resolution simulation as well as better tools to support network simulation make it practical to combine data observed from network systems with traffic models generated from the emulation of computer networks and traffic models generated by probability distribution functions. End-to-end communications system emulation has long been a technical strength of the Army's Technology Integration Center (TIC). These technological innovations are enabling the TIC to scale their modeling and simulation efforts to support the Army on an enterprise-wide basis. This allows the transition of research-developed simulation techniques to be applied operationally to support ISEC's communication systems integration function.

The Army Communications and Electronics Command's (CECOM) TIC is the Army's honest broker for information technology system evaluation. The TIC is part of CECOM's Information Systems Engineering Command (ISEC) based at Fort Huachuca, Arizona. ISEC is the Army's engineer and system integrator for both infrastructure and force projection information systems that support Army and select joint service information requirements.

ISEC's Technology Integration Center (TIC) plays a major role in moving technological innovations from concept to operational capability. The recent addition of a modeling and simulation group provides a significant capability for network planning and stress testing as well as enterprise-wide network architecture development and validation. This paper will describe some of the capabilities, methodologies and successes of the ISEC TIC modeling and simulation group.

INTRODUCTION

We use modeling and simulation with networks because analytical methods are insufficient to cover all aspects of computer networking. Some phenomena are well represented by analytical models, but many aspects of networking are not, particularly network traffic.

In modeling and simulation, the problem domain determines the appropriate simulation strategy. Some domains, such as global thermonuclear war, are not observable. We can summarize these approaches in table 1.

	Observable System	**Non-Observable System**
Subjective Approach	Comparison of data using graphical displays	Comparison to other models
Objective Approach	Comparison of data using statistical tests and procedures	Comparison to other models using statistical tests and procedures

Table 1. Objective and Subjective Approaches to Simulation.

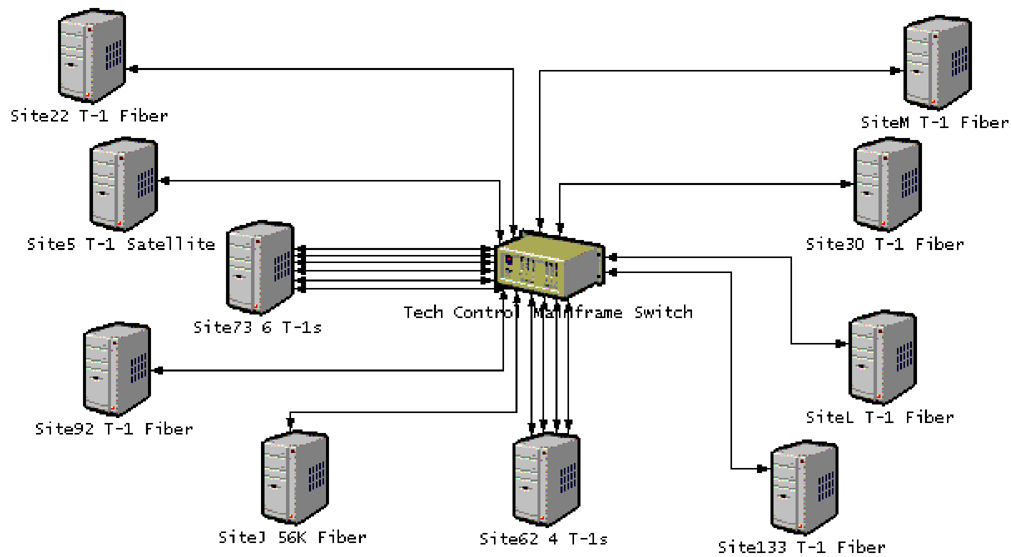

Figure 1. System architecture as a connectivity diagram.

A subjective approach to an observable system can be a reasonable first step. Much has been written on the topic of system architectures. The IEEE defines an architecture as composed of "the structures or components, their relationships, and the principles and guidelines governing their design and evolution over time." A system architecture that graphically lays out the components of a network is a useful first start. Consider the architecture in Figure 1.

The client who commissioned the architecture study believed that they needed additional T-1 lines. Intuitively, one might guess that seventeen T-1 lines feeding into the same switch might indicate a bottleneck at the switch. Low utilization rates on most of the T-1 lines confirmed this diagnosis.

Network loads are an important component in evaluating a network architecture. It is very difficult to evaluate a network design if you do not know what the required utilization and throughput are. We define throughput as the number of packets per second with a packet length measured in bits and utilization as packet throughput multiplied by packet length divided by available bandwidth.

Network monitoring provides the means to observe throughput and measure bandwidth utilization. Applying the data collected to the system architecture allows us to move from a subjective approach to an objective approach. Observed data provides the means statistically validate a network simulation.

VERIFICATION AND VALIDATION

Verification of a simulation is the process of assessing the degree to which the implementation transforms inputs into outputs *as specified by the model*. The ultimate verification test is to model a known system and run the same sets of inputs through the actual system and the simulation. If the results are statistically the same, then you have reasonable assurance that you have implemented the model correctly.

Validation is the process which establishes the extent to which a model does (or does not) acceptably represent the phenomenon of interest. Once we know we have a good model, how do we gauge its predictive power? A good traffic model is necessary for a network simulation to achieve any meaningful predictive power.

The difficulty of validation has made some analysts skeptical about the value of simulation techniques (Hamilton, Nash and Pooch 1997). Simulations that are not validated should be viewed with skepticism. Simulation model validation is well described in (Sargent 1991) and (Sargent 1988). Validation does mean perfect predictive power, merely that the simulation model is worthy of attention.

Even with the advent of improved network monitoring tools, monitoring remains challenging because of the sheer scale involved. An hour's worth of network traffic over a 600-node TCP/IP network can easily generate a gigabyte of raw ASCII data.

Network traffic patterns cannot be relied upon to follow a standard probability distribution (Hamilton 1996). The traffic data shown in Figure 2 is representative of the bursty nature of network traffic. The data is clearly not exponentially distributed. The failure of Poisson processes to model network traffic is outlined in (Paxson and Floyd 1995). For this reason, a hybrid approach to modeling network traffic is indicated.

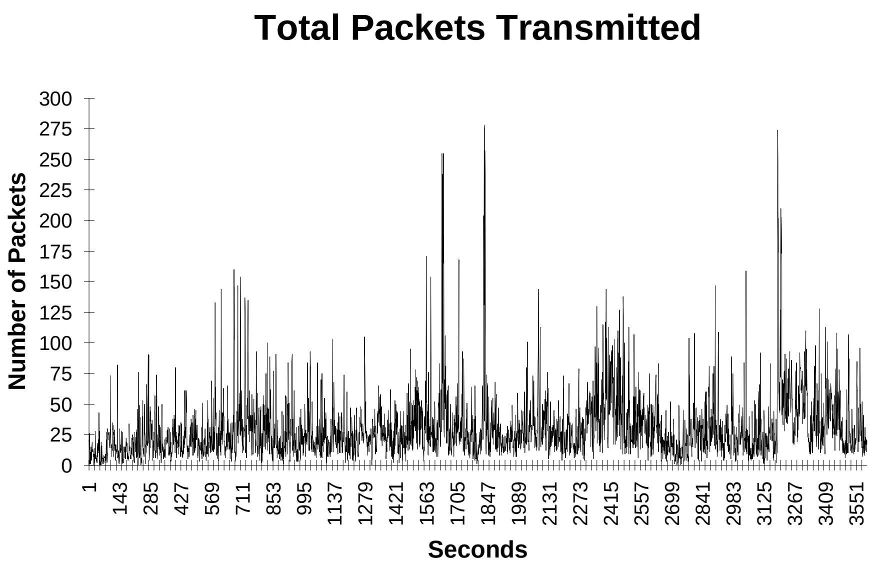

Figure 2. An hour of TCP/IP Traffic.

OPNET AN OPEN ARCHITECTURE NETWORK SIMULATION ENVIRONMENT

In order to use a hybrid approach, an open simulation architecture is required. The U.S. Army has adopted OPNET as a standard under the auspices of the Army Enterprise Strategy developed by the U.S. Army Office of the Director of Information Systems for Command, Control, Communications and Computers. OPNET is widely used in universities as well as many parts of the DOD.

OPNET may be described as a communications-oriented simulation language. The name OPNET is derived from Optimized Network Engineering Tools. The single most significant aspect of OPNET is that it provides direct access to the source code coupled with an easy-to-use front end. This capability allows the introduction of multiple traffic sources, from PDFs, from network emulators and from observed traffic.

OPNET models are composed of three primary model layers: the process layer, the node layer and the network layer. The lowest modeling layer is the process layer.

Network Models	networks and subnetworks
Node Models	individual nodes and stations
Process Models	STD that defines a node

Figure 3. OPNET model hierarchy.

This modeling hierarchy is illustrated in Figure 3. The process model in Figure 4 shows a state transition diagram (STD) for the generation of packets. Process models are built using finite state machines (FSMs) described by STDs. Finite state machines are an effective means of defining discrete-event systems that maintain state information. FSM-based design provides a means to manage complexity. Complex networks can be broken down into individual states and then each state is defined and implemented.

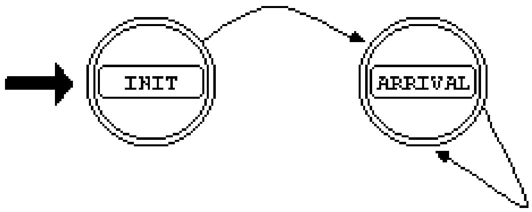

Figure 4. State transition diagram in a process model.

The next level of abstraction up from the process model is the node model. Each element in the node model is either a predefined OPNET artifact or defined by its own STD. Double-clicking on a node model element brings up its underlying process model. Figure 5 is an example of a node model that defines a station on a FDDI network. Packets are generated from the source *llc_src*, processed in the *mac* module and are put on the ring by the *phy_tx* module. Traffic from the ring is received via the *phy_rx* module processed in the *mac* module and finally received and discarded by the *llc_sink* module.

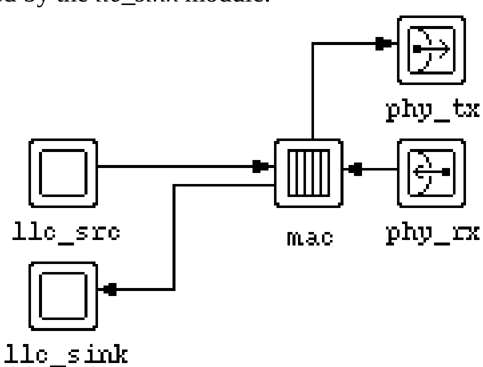

Figure 5. Node model (FDDI node).

The heart of a node model is either a processor module or a queue module. Processor modules are used to perform general processing of data packets as specified in the applicable protocol. Queue modules are supersets of processor modules with additional data collection capabilities built-in. The *mac* module in Figure 5 is an instantiation of a queue module.

The network model is the highest modeling layer in the OPNET model hierarchy. The network model may represent a hierarchy of subnetworks. A network model is shown in Figure 6. A node model such as the one shown in Figure 5 defines each of the stations (nodes) shown in Figure 6. Again, each module in a node model is defined by state transition diagram as shown in Figure 4 thus conforming to the modeling hierarchy shown in Figure 3.

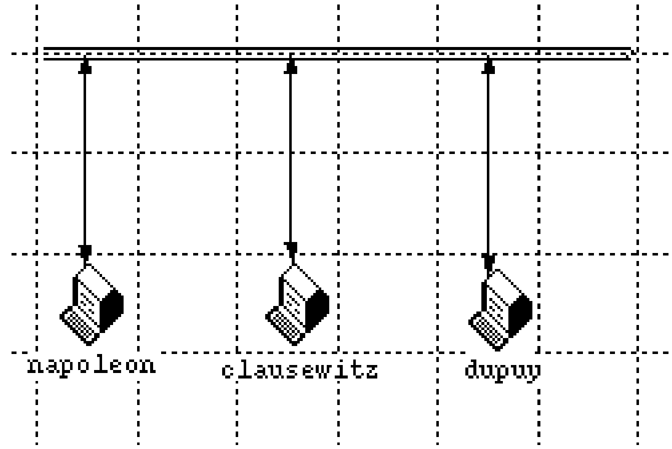

Figure 6. Three node network model.

The network model may be used to model a single network, subnet or segment or a hierarchy of networks, subnetworks or segments. The segment in Figure 6 may be joined with other segments and aggregated into a single subnet icon as shown in Figure 7.

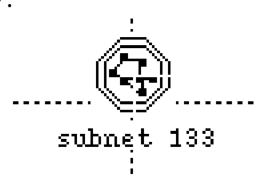

Figure 7. Subnetwork of aggregated segments.

The operation of a single network segment may now be studied. At this point the individual stations on the segment may be customized if a more detailed representation is desired. Individual workstations or types of workstations may be specially modeled. Special characteristics could be implemented by modifying the individual modules of the station of interest or the physical network line connecting the stations. Many modifications can be made via the built-in menus. However, modifications may be made at the source code level should the menu choices not be fully satisfactory.

EMULATOR-BASED SIMULATION

One way to study a network is to replicate the network in a laboratory and use it to emulate the network of interest. This is expensive and time consuming. Live observation of a network, such as one on an Army post, camp or station is also expensive and time consuming. Increasingly, hybrid approaches are being developed which seek to minimize data collection efforts while maintaining model fidelity.

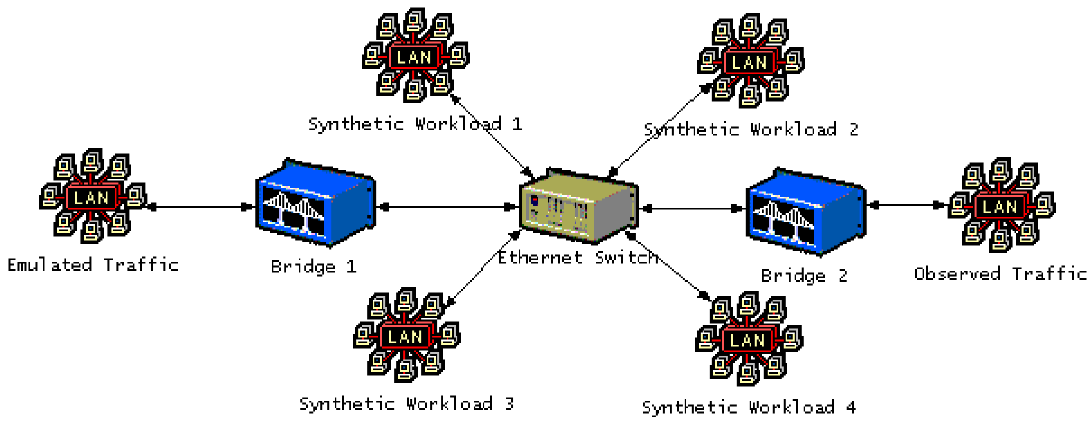

Figure 8. Mixed network simulation traffic sources.

.
With careful planning and close attention paid to the statistical soundness of the simulation model being produced, the amount of data collection can be seriously reduced. Consider the following hypothetical network in Figure 8.

Synthetic workloads may be created by exponentially distributing approximated workloads or better still, emulated workloads. In this example, if the LANs of interest are on the right and left most side of the bridges, then we have a high fidelity representation. The synthetic workloads will provide adequate traffic coming in over the bridges. This will enable statistically validated models of throughput and utilization. Further, this technique can also be applied to a single LAN. Experimentation to this point indicates that at least 75% of the network model can be run with synthetic workloads while still getting a statistically valid result.

CONCLUSION

Research has shown that the "burstiness" of network traffic can be accurately represented in a simulation by using only a relatively small amount of actual traffic. This development is critical for large-scale enterprise architecture efforts. Network simulation is an effective means to verify and validate enterprise architectures. However, practical use of network simulation requires some economies of scale.

Prudent use of network emulators can significantly reduce the amount of data collection required to verify and validate architectural changes to a post, camp or station's information infrastructure. Emulation of network architectural designs is an important evaluation process to insure interoperability among heterogeneous components. In today's COTS environment, this is absolutely necessary to insure the validity of vendor claims and the interoperability of the software and hardware versions of commercial off-the-shelf products.

While network simulation has limitations for general cases, high fidelity simulations are feasible for specific implementations and specified traffic load. Increasingly, network simulation studies are being used to determine the feasibility of network designs before they are implemented. Such studies are becoming a recognized industry best practice.

Network simulation provides both technical and planning data. From a technical standpoint, it is unlikely that a network simulation model can be statistically validated if there are major flaws in the model. For example, given the same traffic pattern, the output from a network simulation should be consistent with the output of the actual network. Tying emulation with simulation can ameliorate many of these drawbacks. Planning data is needed to integrate new technologies into existing networks. For example, what is the impact of full duplex versus half duplex operation in Gigabit Ethernet? Can CSMA/CD continue beyond the 1 gigabyte networks? (Kadambi, Crayford and Kalkunte 1998) In the absence of 10 or 100 gigabyte systems, the best insight can gathered through network simulation.

Network simulation is becoming more practical because research has shown that the need for observed network data can be reduced by using synthetic workloads for parts of the network. With only a limited amount of actual network data, a network simulation can be statistically validated (Hamilton 1996). The integration of traffic data produced by emulation promises to reduce the need for live data collection even further.

REFERENCES

J. A. Hamilton, Jr. 1996. *Multilevel Simulation of Discrete Network Models*, Doctoral Dissertation, Texas A&M University, College Station, Tex.

J. A. Hamilton, Jr., D. A. Nash and U. W. Nash. 1997. *Distributed Simulation*, CRC Press, Boca Raton, Fla.

J. Kadambi, I. Crayford and M. Kalkunte. 1998. *Gigabit Ethernet*, Prentice Hall PTR, Upper Saddle River, N. J.

V. Paxson and S. Floyd. 1995. "Wide Area Traffic: The Failure of Poisson Modeling," IEEE/ACM Transactions on Networking, no. 3 (June) pp 226-244.

R. G. Sargent. 1988. "A Tutorial on Validation and Verification of Simulation Models." *Proceedings of the 1988 Winter Simulation Conference*, Society for Computer Simulation, Abrams, ed. San Diego, Calif.

R. G. Sargent. 1991. "A Tutorial on Validation and Verification of Simulation Models." *Proceedings of the 1991 Winter Simulation Conference*, Society for Computer Simulation, Nelson, Kelton, Gordon, eds. Phoenix, Ariz.

AUTHORS

Lieutenant Colonel John A. (Drew) Hamilton, Jr., US Army, is currently the Director of the Joint Forces Program Office. Previously he served as the Research Director for the Department of Electrical Engineering and Computer Science at the US Military Academy, as Chief of the Ada Joint Program Office, Chief, Officer Training Division at the Computer Science School, Fort Gordon. Lt.Col. Hamilton has a B.A. in Journalism from Texas Tech University, where he was commissioned in Field Artillery; an M.S. in Systems Management from the University of Southern California, an M.S. in Computer Science from Vanderbilt University and a Ph.D. in Computer Science at Texas A&M University. Lt.Col. Hamilton is a graduate of the Naval War College with distinction. CRC Press published his book, Distributed Simulation, written with Major D. A. Nash and Dr. Udo W. Pooch. He is a senior member of the Society for Computer Simulation.

Colonel John C. Deal, US Army, is the Commander of the Information Systems Engineering Command (ISEC). He previously served as the Executive Officer, Office of the Director of Information Systems for Command, Control, Communications and Computers (ODISC4) and has been intimately involved in the development of the Army Technical Architecture, the Army Systems Architecture and the Operational Architecture. Previously Colonel Deal served as a member of the Secretary of Defense Strategic Studies Group. Colonel Deal is a graduate of the University of Alaska, the Naval Command and Staff College, and the Army War College Fellowship Program at the University of Pittsburgh. He has earned a M.S. in Electrical Engineering from the Naval Post Graduate School, a M.A. in National Security Studies from the Naval War College, and a M.A. in International Relations from Salve Regina University.

COL John R. Grobmeier, US Army, is the Director, Technology Integration Center, United States Army Information Systems Engineering Command. He is a senior member of the Army Acquisition Corps (AC). COL Grobmeier's AC assignments include Chief of West Point's Information System Division, DISC4 Staff Action Officer, Deputy Program Manager for the United Kingdom's Royal Artillery Command Information Systems Group (RACISG), and as Product Manager for the Advanced Field Artillery Tactical Data System (AFATDS). COL Grobmeier holds a Master of Science degree in National Resource Strategy and a Master of Science Degree in Business Administration with an emphasis in ORSA. He is a graduate of the Industrial College of the Armed Forces (ICAF), the Department of Defense's Senior Acquisition Course, the Defense Systems Management College, the U. S. Army Command and General Staff College and the Basic and Advanced Field Artillery Courses. In 1992, COL Grobmeier was listed in Who's Who in Business America for his work in the fields of data management and business process reengineering.