Overview of Network Simulation Research (Postprint)

Authors: Hao Zhiyu

Date: 2017-03-10T00:00:00+00:00

Abstract

This paper first introduces the significance of network simulation in the study of network behavior and the current challenges confronting network simulation research. It then focuses on the two most critical research directions in network simulation: routing strategies and traffic model abstraction techniques, elaborating on their respective research contents and the latest advances both domestically and internationally. Finally, it concludes with a summary and outlook on potential future development directions for network simulation research.

Full Text

Preamble

Vol. 9 No. 3 Information Technology Letters Vol.9 No.3
Overview of Network Simulation Research

Abstract

This paper first introduces the significance of network simulation in network behavior research and the current challenges facing network simulation studies. It then focuses on the two most important research directions in network simulation: routing strategies and traffic model abstraction techniques, describing their research content and the latest domestic and international progress. Finally, it summarizes and outlines possible future development directions for network simulation research.

Keywords: network simulation; routing strategy; traffic model abstraction

1 Introduction

As a hallmark of societal informatization, the Internet has penetrated every domain of social life and become a strategically significant infrastructure. Studying large-scale network environments is an essential process for understanding networks and forms the foundation for ensuring stable and robust Internet operation at higher levels. Current methods for Internet research primarily include mathematical model analysis, network measurement, network simulation, network emulation, prototype experimental platforms, and real network testing [1,2,3]. Reference [3] analyzes the advantages and disadvantages of these methods, noting that mathematical model analysis is flexible and low-cost but may produce incorrect results due to oversimplified models; network measurement methods are inherently imperfect and can lead to incorrect measurements; real network testing is rarely adopted due to uncontrollable experimental environments, non-repeatable experiments, and significant risks; prototype experimental platforms can yield more practical results but are costly, inconvenient to use, unsuitable for large-scale network behavior research, and suffer from poor experimental repeatability; network emulation builds models with low abstraction levels, achieving high fidelity but limited network scale, and must operate at real network speeds to interact with actual networks, thus restricting emulation scale.

Reference [3] further points out that network simulation establishes simplified models (including protocol models, traffic models, topology models, etc.) based on abstracting network behavioral characteristics. Its features include: detailed 刻画 of network operation with high fidelity; capability to study networks with tens of thousands of nodes, enabling research on relatively large-scale networks; and convenient simulation tools with low operational costs.

compares six Internet research methods across dimensions of fidelity, scale, and cost. As shown, these methods represent trade-offs among these three aspects [3].

shows the proportion of simulation methods used in top-tier international conferences. Due to its low cost, simple implementation, ease of use, reasonable fidelity, ability to study unimplemented network mechanisms, and guaranteed research scale, network simulation has been widely applied throughout network research—becoming the primary means for testing network protocols, analyzing network behavior, maintaining normal network operation, designing network architectures, and evaluating novel network mechanisms. Statistics from reference [1] on articles in top-tier networking conferences SIGCOMM and INFOCOM (shown in Table 2) indicate that simulation has become a widely used foundational research tool in Internet behavior studies, applied to all aspects of Internet behavior research—protocol design and analysis, network security verification, network behavior prediction, and network planning and construction.

2 Problems in Network Simulation Research

Given the significant application value of network simulation in network behavior research, the study of network simulation technology itself has attracted widespread attention from researchers. Researchers are committed to improving simulation scale and result fidelity while reducing computational and storage overhead during simulation, thereby further decreasing development and operational costs and enhancing application value.

However, the inherent complexity, heterogeneity, dynamism, and enormous scale of the Internet pose tremendous challenges to network simulation research. First, as the core of the entire network architecture, the IP protocol aims to seamlessly interconnect different network technologies and administrative domains. It provides users with consistent virtual connections, shielding them from connectivity details. However, network complexity does not disappear—numerous inconsistent behaviors in the underlying network create difficulties in network analysis and behavioral understanding.

Second, the network changes rapidly. This includes two aspects: First, network heterogeneity and dynamic characteristics, such as topology changes, link diversity, routing dynamics, and protocol complexity, make network modeling and simulation extremely difficult. Second, changes in network applications lead to traffic pattern evolution: from early email and FTP to current WWW services and multimedia streaming. Additionally, network applications exhibit a "Success Disaster" phenomenon, where an application or protocol may become widely deployed before its design and implementation are fully mature (e.g., HTTP), further complicating network simulation.

Finally, modern networks are massive in scale, creating two problems: First, the number of network hosts grows extremely rapidly. In recent years, Internet host counts have maintained high growth rates, while Internet data volume has grown nearly exponentially. Simulating such a large number of network objects is undoubtedly very difficult. Second, network scalability issues arise—some network protocols and mechanisms operate well in small-scale networks but become inefficient or even unusable when scaled up.

These characteristics of the Internet create difficulties for network simulation research in terms of simulation scale, efficiency, and fidelity.

From a performance perspective, network simulators face two major challenges—simulation scale and simulation efficiency. With the rapid development of the Internet, network scale continues to grow, increasing demands for Internet simulation. However, due to hardware resource limitations, especially memory constraints, network simulators can only simulate limited network scales, unable to meet Internet simulation requirements. Moreover, increasing network scale and increasingly complex and diverse network behaviors have made low simulation performance a bottleneck restricting practical application of network simulators. According to estimates in reference [7], for a network with 1×10^8 nodes, traditional packet-level discrete-event simulators (such as NS-2) need to simulate approximately 2.9×10^{11} events. At an event processing rate of 1,000,000 events per second, simulating one second of network operation requires about 4 days of simulator runtime; the memory space needed to store essential information about nodes, links, and packets is approximately 2.9×10^{14} bytes (about 3 TB);

if network behavior simulation processes need to be recorded, storing one second of network behavior requires about 1.4×10^{13} bytes of storage space. To date, according to relevant materials and literature, known large-scale network simulators can only simulate up to 10^6 nodes, and only under the premise of extremely simple topology connections and applications.

From a functional perspective, network simulators should be able to realistically simulate various network behaviors on the target network topology, such as various protocols and security events, to meet application requirements for network behavior analysis and research.

From a simulation fidelity perspective, Internet heterogeneity and dynamism pose enormous challenges to network simulation fidelity. How to capture the core, invariant characteristics of the Internet while ignoring secondary attributes to establish correct Internet simulation models is also key to network simulation technology.

3 Major Research Teams and Projects

Foreign research on network simulation technology began in the 1990s. Currently representative research organizations and projects mainly include:

(1) The VINT (Virtual InterNet Tested) project [8], funded by DARPA and collaborated on by the University of Southern California/Information Science Institute (USC/ISI), Xerox PARC, Lawrence Berkeley National Laboratory (LBNL), and UC Berkeley. The most famous NS-2 [9] in network simulation research originated from this project.

(2) The Security and Systems Modeling (MOUSE) research group led by Professor David Nicol at the University of Illinois at Urbana-Champaign (UIUC), a leading team in network simulation that developed the iSSF simulator [10] and has made outstanding research achievements in network security behavior modeling and simulation, network traffic modeling, and real-time Internet behavior simulation.

(3) The team led by Professors Fujimoto and Riley at Georgia Tech (GaTech), which developed the PDNS [11] and GTNetS [12] parallel network simulators and conducted in-depth research on network simulation routing strategies and simulation parallelization techniques.

(4) The research group led by Jason Liu at Florida International University (FIU), a branch of the aforementioned project (2) but with deep research in traffic model abstraction, now becoming a top research force in this direction.

(5) The network research team led by Young Sun Kim at the Electronics and Telecommunications Research Institute (ETRI) in South Korea, which uses the SSFNet simulator for in-depth Internet research and has published articles in multiple international conferences and journals [13].

(6) Shie-Yuan Wang from National Chiao Tung University in Taiwan, who developed NCTUns [14] in 2003. This simulator uses sampling of actual network traffic as simulation input, achieving extremely high fidelity, and has continuously published papers and demonstrations at top international conferences such as Mobicom and Infocom [15-17].

Domestic research on network simulation started relatively late with a relatively weak foundation. Based on publicly available literature, research strength mainly comes from the Institute of Computing Technology of the Chinese Academy of Sciences, Harbin Institute of Technology, Xi’ an Jiaotong University, and other research institutions. Their work began by introducing foreign classic simulators (NS-2) and simulation techniques, gradually developing their own research interests and directions, and making significant progress in recent years, reaching international advanced levels in routing strategies and parallelization techniques.

4 Current Status of Key Technology Research

There are many technical means to improve network simulation scale and performance, including abstraction techniques for various models (topology models, protocol models, traffic models, application models), parallelization techniques, scheduling mechanisms, etc. Among them, abstraction techniques for various models are key to network simulation technology. In fact, the essence of simulation technology is to reasonably abstract the simulated objects to accurately represent them.

Among the various models required for network simulation technology, routing models and traffic models are the most critical and receive the highest attention from researchers. This is because: (1) Data packet routing technology is the foundation for various applications on the Internet, and the calculation, storage, and lookup methods of routing information have enormous impact on computational and storage overhead of network simulation tools; (2) Network behavior is typically manifested by the continuous transmission of data packets from various applications on the Internet, and it is precisely this data traffic that brings a huge number of simulation events to network simulators, meaning enormous computational and storage overhead. The following sections introduce representative research achievements in these two areas.

4.1.1 Research Content

As demands for network simulation scale continue to expand, routing abstraction technology has become one of the key issues in network simulation research. The Internet consists of over 18,000 Autonomous Systems (AS). Each AS typically contains hundreds of router nodes responsible for inter-AS routing communication and thousands of router nodes responsible for intra-AS routing. As network scale continues to grow, the routing table size on an Internet Service Provider (ISP) core router node has reached the point of exhausting router memory space. One can imagine how severe a test this poses to simulator storage space when needing to save complete routing information for all these router nodes to complete Internet behavior simulation. Moreover, calculating and looking up routing information required by simulation applications in such massive routing state data will also pose a huge challenge to processor computational resources.

Meanwhile, routing simulation fidelity is also a key research focus. If routing simulation differs greatly from routing processes in actual network environments, simulation results such as traffic, congestion, and packet loss generated during network behavior simulation will lose their significance, seriously affecting the credibility of network behavior simulation. Researching high-performance, high-fidelity network simulation routing abstraction technology is a hot topic in network simulation research.

4.1.2 Related Research Progress

Based on whether simulators require dynamic routing, routing abstraction technology can be divided into static routing simulation technology and dynamic routing simulation technology. The former addresses static routing, studying how to complete routing calculation, storage, and lookup strategies on individual simulation nodes and among nodes under the assumption that topology information remains unchanged throughout the simulation. The latter addresses network topology dynamics, studying how to realistically simulate routing information changes caused by topology changes. The following sections introduce related research progress in these two areas.

(1) Static Routing Simulation Technology

Static routing simulation technology is currently the focus of research efforts. For local static routing strategy research, current simulators mostly adopt a routing strategy that centrally calculates and stores global static routing tables (Flat strategy). This method calculates global routing tables based on global topology connectivity during simulator initialization and stores them in memory. During simulation of data packet routing, it looks up static routing tables to route packets. This method requires large storage space to save static routing tables. For a network topology with n nodes, the space complexity is $O(n^2)$; when looking up routes, based on source and destination address information, required routing information can be obtained in $O(1)$ time, representing minimal routing lookup time.

Huang et al. proposed an approximate routing calculation and lookup strategy in reference [18], using a minimum spanning tree as the routing table, assuming all routing information is stored in this tree. This method uses $O(n)$ space for the routing table and $O(\ln(n))$ lookup time complexity, achieving good balance between storage space and lookup time. However, since a single minimum spanning tree can only cover a limited number of shortest paths, this is only an approximate routing strategy.

Hiromori et al. proposed a method in reference [19] that uses a minimum spanning tree combined with a partial static routing table to store routing information (called the STree_Flat method). This method uses a minimum spanning tree to store partial routing information, while storing shortest path routing information not covered by the tree in a static routing table. When looking up routes, it first searches the static routing table; if found, it uses that routing information; otherwise, it searches on the minimum spanning tree.

The time and space complexity of the STree_Flat method depends on the number of shortest path routing information covered by the minimum spanning tree —i.e., the shortest path coverage percentage of the minimum spanning tree. Assuming the shortest path coverage rate of the minimum spanning tree is p, the STree_Flat method requires memory space of $O(pn^2)$; however, when searching the static routing table, it must traverse the entire table, so its worst-case lookup time is $O((1-p)n^2)$. This method saves significant space while ensuring accurate routing query results, but due to the high time complexity of traversing the partial routing table, lookup time increases dramatically, resulting in poor overall performance.

Chen et al. proposed a method in reference [20] that uses multiple minimum spanning trees combined with a static routing table to store routing information (called the MTree_Flat method). Using multiple minimum spanning trees to store routing information improves the shortest path coverage rate, further saving storage space. When looking up routes, it still first searches the static routing table portion; if found, it routes directly; otherwise, it indicates the path is covered by one of the minimum spanning trees, so it searches each tree sequentially and selects the shortest path as the routing information for packet routing.

The time and space complexity of this method are still determined by the coverage rate of the minimum spanning trees. Assuming k minimum spanning trees are used with coverage rate p, its space complexity is $O((1-p)n^2 + kn)$. Since multiple spanning trees are used, the shortest path coverage rate is necessarily much larger than that of a single minimum spanning tree under the same topology conditions, thus saving more storage space. Although it needs to search all k trees to select the shortest path, the high coverage rate means the static routing table is smaller, requiring less time to search. Therefore, the overall performance is better than the STree_Flat method. The worst-case lookup time is $O((1-p)n^2 + k\ln(n))$.

In addition to the above methods that pre-calculate and statically store routing information, there is another routing strategy that stores no routing information but calculates routes in real-time during simulation. The Nix-Vector method proposed by Riley et al. [21,22] is representative of this approach. This method performs no routing information calculation or storage during simulation initialization. Instead, when a data packet is generated, it calculates the route in real-time based on source and destination address information, stores the calculated routing information in the packet header, and at each router node the packet traverses, parses the routing information from the header.

This method requires no additional storage space for routing information, achieving minimal space complexity. However, each time a new data packet transmission is simulated, it must traverse the global topology to calculate the route, resulting in high time complexity. For a topology described using adjacency lists, the lookup time complexity is $O(n + e)$, where e is the number of edges. This method can effectively handle routing changes caused by topology changes and proposes methods for calculating when different nodes perceive topology changes. However, because calculating and verifying Nix-Vector vectors introduces a large number of additional events and computations, and because using the Nix-Vector strategy for routing calculation itself consumes significant processor time, its simulation efficiency is very low, and this inefficiency is further amplified as simulated network scale increases.

Based on the above routing strategy research, we proposed the Multi-Tree-Nix-Vector technology (MTree_Nix) in our previous work [23,24]. The MTree_Nix technology combines multiple minimum spanning trees with Nix-Vector technology, achieving optimal balance between storage overhead and computational overhead in routing simulation technology.

(2) Dynamic Routing Simulation Technology

Internet dynamics is one of the fundamental properties of the Internet. Topology structure changes that may occur at any time on the Internet will cause dynamic updates to routing states. To realistically reflect Internet behavioral properties requires more realistic simulation of the dynamic change process of routing state information. This is the research content of dynamic routing abstraction simulation technology. Research methods for dynamic routing simulation technology mainly include two types: dynamic routing protocol simulation and static routing table reconstruction.

Research on dynamic routing protocol simulation mainly focuses on simulation of dynamic routing protocols such as Border Gateway Protocol (BGP) and Open Shortest Path First (OSPF), with relatively more research work [25-28]. However, the main goal of these works is research on the protocols themselves and they are not suitable for use as basic routing modules in network simulators.

Routing table reconstruction is a simple and effective approach for handling dynamic routing simulation. All static routing strategy algorithms can reconstruct routing tables when topology connectivity changes occur. However, because centrally stored routing tables are used, when topology changes occur, routing table updates are completed simultaneously for all nodes, seriously damaging simulation fidelity. To address this problem, Riley et al. proposed a dynamic routing simulation method that validates Nix vectors stored in cache when topology information changes to determine whether they still satisfy shortest path requirements [29]. This method can effectively handle routing changes caused by topology changes and proposes methods for calculating when different nodes perceive topology changes. However, because calculating and verifying Nix-Vector vectors introduces a large number of additional events and computations, and because using the Nix-Vector strategy for routing calculation itself consumes significant processor time, its simulation efficiency is very low, and this inefficiency is further amplified as simulated network scale increases. Based on the work in reference [29], our research group improved the calculation model for topology change information update time and proposed dynamic MTree_Nix technology based on this model [30], further reducing computational and storage overhead for dynamic routing simulation technology, representing the most optimized dynamic routing simulation technology published to date.

(3) Summary

From the research status of routing abstraction simulation technology, problems mainly exist in dynamic routing simulation technology. On one hand, topology and routing dynamics as an important property of the Internet create urgent demand for dynamic routing simulation. On the other hand, dynamically changing topology and routing information bring sharply increased computational and storage overhead to simulators, making already inefficient network simulators face more severe competition for computational and storage resources, making it even more difficult to meet real-time simulation application requirements. According to the latest experimental results, when using the dynamic MTree_Nix technology from reference [30] in a network simulation system to simulate large-scale network security events like worm propagation, its simulation efficiency is only about 1/20 of that when using static routing simulation technology, let alone meeting real-time simulation requirements. Therefore, researching how to reduce computational and storage overhead of dynamic routing simulation technology to improve simulation performance is an urgent priority in routing abstraction simulation technology research and an important factor affecting network simulation performance.

4.2.1 Research Content

Internet network traffic consists of individual data packets. For discrete-event simulators, to accurately describe network behavior, each behavior of every data packet (such as packet enqueueing and dequeueing on links, processing by routers and end nodes, etc.) must be simulated as an event. This is so-called packet-level simulation. Currently, many network simulators including NS-2 adopt packet-level simulation. Although packet-level simulation can ensure high fidelity, because each behavior of every data packet needs to be described by separate events, this poses an enormous challenge to computational overhead for network simulators. To reduce computational overhead, it is necessary to simplify individual data packets and network traffic models composed of single packets. This is the research content of traffic model abstraction technology.

4.2.2 Related Research Progress

For traffic model abstraction technology, most current research adopts the method of increasing traffic model abstraction granularity, i.e., using so-called “fluid-level” simulation. This method abstracts data packets with certain common attributes (such as source and destination addresses) into a flow, uses piecewise functions of simulation time to represent key flow attributes (such as packet loss rate, queue length, etc.), and updates these attribute values as needed during simulation to describe network characteristics. Through this fluid simulation approach, the number of events the simulator needs to process per unit time can be effectively reduced, thereby improving simulation efficiency. Based on different scheduling mechanisms of fluid simulation models, current research is mainly divided into two categories: flow event-driven simulation methods and flow time-driven simulation methods.

The flow event-driven simulation method generates a discrete event when a data flow size needs to change (such as when multiple data flows converge on the same link and superimpose), triggering changes in relevant attributes of this data flow. Representative research includes references [31-33]. When data flow changes are not dramatic, this method can effectively reduce the number of discrete events needed for simulation and lower computational overhead. However, when data flow changes are relatively dramatic, this method’ s performance suffers severely and is prone to ripple effects, even leading to more events than packet-level simulation [32].

The flow time-driven simulation method advances simulation time periodically during simulation, moving forward by a fixed time interval (time-step) each time. Each time step advance generates a discrete event that completes three operations: (1) generates new data flow size based on traffic models; (2) calculates network state at the beginning of the next period based on newly generated flow size and network state from the previous time period; (3) calculates feedback of network state to traffic models and modifies traffic model parameters. Representative work includes references [34-36]. Compared with flow event-driven simulation, flow time-driven simulation has greater advantages in simulation efficiency but correspondingly suffers some loss in simulation fidelity.

Additionally, considering the respective advantages and disadvantages of packet-level and flow-level simulation, researchers have proposed organically combining the two, using hybrid simulation technology with different precision levels for network behavior simulation according to different application requirements. For example, using flow-based simulation technology to simulate background traffic while using packet-level simulation technology for foreground traffic of higher interest. Representative work includes references [37-40].

From the research status of traffic model abstraction technology, packet-level and flow-level hybrid traffic model simulation is increasingly becoming the development trend due to its efficient combination of simulation efficiency and fidelity. However, current research results still have significant problems. First, the two methods of flow-based simulation each have advantages but also difficult-to-solve problems. Flow event-driven simulation methods have difficulty eliminating the harm caused by ripple effects and have basically been abandoned in latest research; flow time-driven simulation methods have greater advantages in computational performance and represent the development trend of traffic model abstraction technology, but have inherent defects in simulation fidelity and currently lack effective solutions for time step selection strategies. Second, in hybrid simulation technology, the interaction technology between packet-level traffic and flow-level traffic is difficult, and in most current research, flow time-driven simulation methods flatten packet-level traffic and incorporate it into the flow time-driven model, participating in traffic model update calculations at the end of each time step. This makes it difficult to ensure that attributes of individual packets of interest to users can be accurately simulated.

4.3 Our Research Work

In recent years, the Information Security Research Center of the Institute of Computing Technology, Chinese Academy of Sciences, has been dedicated to network simulation technology research and has accumulated deep technical reserves. Key researchers from the center have successively participated as core members in multiple national key projects related to network behavior simulation, mainly including: (1) National 973 Program project "Virtual Computing Environment Testbed and Simulation Platform," (2) National 863-917 Special Project "Distributed Network Simulation Technology," (3) National 863 Program Key Project "High-Performance Large-Scale Network Behavior Simulation System." More than 10 high-level academic papers have been published in this field, with in-depth research conducted on routing abstraction technology, simulation parallelization technology, traffic model abstraction technology, etc., accumulating a large number of characteristic technologies, research methods, and practical tools.

Based on the above work, our research group applied for and received the National Natural Science Foundation project "Research on High-Performance Network Simulation Technology Based on Abstract Traffic Models" in 2010. This project aims at the shortcomings of current traffic model abstraction technology research described in Section 4.2, studying traffic model abstraction technology with higher simulation performance and fidelity. Research on this project is currently in progress.

5 Conclusion

As an effective foundational tool for studying Internet behavioral properties, network simulation is increasingly valued by researchers. Research on network simulation itself has also become a current hot topic. This paper provides an in-depth introduction to two important directions in network simulation research: routing strategies and traffic model abstraction technology, analyzing current research progress, major existing problems, and possible improvement directions.

Looking forward to future network simulation research development, in addition to the routing strategies and traffic model abstraction technology detailed in this paper, network simulation parallelization technology and protocol model abstraction technology will also become hot research directions. Additionally, with increasing performance of computational and storage resources, the combination of network emulation technology based on virtualization technology with network simulation technology to form a network real-time simulation-emulation environment with multi-level abstraction granularity and both high performance and high fidelity will also become a development direction for future network behavior research tools.

References

Ammar M. Why we still don’ t know how to simulate networks. In Proceedings of 13th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems(MASCOTS’05), Atlanta, GA, USA, 2005:179~182

雷擎，王行刚．计算机网络模拟方法与工具．通信学报．2001,22(9)：84~90

王晓锋．提高大规模离散事件网络模拟性能方法的研究．哈尔滨工业大学博士学位论文．2007,10.

Xia H., Dail H., Casanova H., et.al. The MicroGrid: Using Emulation to Predict Application Performance in Diverse Grid Network Environments. In Proceedings of the 13th IEEE International Symposium on High-Performance Distributed Computing, Honolulu, Hawaii, 2004:52~63.

Vahdat A., Yocum K., Walsh K., et.al. Scalability and Accuracy in a Large-scale Network Emulator. In Proceedings of the Fifth Symposium on Operating System Design and Implementation (OSDI ’ 02), Boston, MA, 2002:271~284.

Chun B., Culler D., Roscoe T., et.al. Planetlab: an Overlay Testbed for Broad-coverage Services. ACM Computer Communications Review. 2003,33(3):3~12.

Riley G. and Ammar M. Simulating Large Networks -How Big is Big Enough?. In Proceedings of the First International Conference on Grand Challenges for Modeling and Simulation, San Antonio, TX, 2002:28~31.

The VINT Project, http://www.isi.edu/nsnam/vint/, network simulator.

McCanne R. and Floyd S. The LBNL network simulator. Lawrence Berkeley Laboratory, 1997.

Nicol D., Liu J., Liljenstam M., et.al. Simulation of Large-scale Networks Using SSF. In Proceedings of the 2003 Winter Simulation Conference(WSC’ 03). New Orleans, LA, 2003:650~657.

Riley G., Fujimoto R., and Ammar M. A Generic Framework for Parallelization of Network Simulations. In Proceedings of Seventh International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS' 99), October 1999.

Riley G. The Georgia Tech Network Simulator. In Proceedings of the ACM SIGCOMM workshop on Models, methods and tools for reproducible network research, ACM Press, 2003, 5–12.

Yoon S. and Kim Y. A Design of Network Simulation Environment Using SSFNet. First International Conference on Advances in System Simulation, 2009, 73-78.

NCUTns. http://nsl.csie.nctu.edu.tw/nctuns.html

Wang S. and Chou C. NCTUns Tool for Wireless Vehicular Communication Network Researches. Simulation Modelling Practice and Theory. 2009,17(7):1211-1226.

Wang S. and Kung H. A Simple Methodology for Constructing Extensible and High-Fidelity TCP/IP Network Simulators. IEEE INFOCOM' 99, New York, USA, March 1999

Wang S. and Kung H. A New Methodology for Easily Constructing Extensible and High-Fidelity TCP/IP Network Simulators. Computer Networks. 2002, 40(2): 257-278.

Huang P. and Heidemann J. Minimizing routing state for light-weight network simulation. In Proceedings of the IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems(MASCOTS' 01), August 2001.

Hiromori A., Yamaguchi H., Yasumoto K., et.al. Reducing the Size of Routing Tables for Large-scale Network Simulation. In Proceedings of the IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems(MASCOTS' 03), October 2003.

Chen J., Gupta D., Vishwanath K., et.al. Routing in an Internet-Scale Network Emulator. In Proceedings of the IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems(MASCOTS' 04), October 2004.

Riley G., Ammar M. and Zegura E.. Efficient routing with Nix-Vectors. In Proceedings of IEEE Workshop on High Performance Switching and Routing(HPSR ' 01), 2001.

Riley G., Fujimoto R. and Ammar M.. Stateless routing in network simulations. In Proceedings of the 8th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems(MASCOTS' 00),, May 2000.

Hao Z., Yun X. and Zhang H. An Efficient Routing Mechanism in Network Simulation. In Proceedings of the 20th ACM/IEEE/SCS Workshop on Principles of Advanced and Distributed Simulation (PADS' 06), May 2006.

Hao Z., Yun X. and Zhang H. An Efficient Routing Mechanism in Network Simulation. Simulation: Transactions of the Society for Modeling and Simulation International, 2008, 84(10-11):511-520.

Dimitropoulos X. Measuring and Modeling Internet Routing for Realistic Simulations. Georgia Institute of Technology, Ph. D. dissertation, 2006.

Liljenstam M. and Nicol D.. On-demand computation of policy based routes for large-scale network simulation. In Proceedings of the 2004 Winter Simulation Conference (WSC' 04), 2004.

Hao F. and Koppol P.. An Internet scale simulation setup for BGP. Computer Communication Review, 2003, 33(3):43–57.

Yaun G., Bauer D., Bhutada H., et. al. Large-Scale Network Simulation Techniques: Examples of TCP and OSPF Models. ACM SIGCOMM Computer Communications Review, 2003.

Riley G. and Reddy D.. Simulating Realistic Packet Routing Without Routing Protocols. In Proceedings of the Workshop on Principles of Advanced and Distributed Simulation (PADS' 05), 2005.

郝志宇, 翟健宏, 云晓春, 张宏莉. 动态路由模拟策略研究. 通信学报, 2007, 28(12):19-24.

Kumaran K. and Mitra D.. Performance and Fluid Simulations of a Novel Shared Buffer Management System. ACM Transactions on Modeling and Computer Simulation. 2001,11(1):43~75.

Nicol D. and Yan G.. Discrete event fluid modeling of background TCP traffic. ACM Transactions on Modeling and Computer Simulation, 14:1–39, July 2004.

Cole R., Riley G., Cansever D., et.al. Stochastic process models for packet/analytic-based network simulations. In Proceedings of the Workshop on Principles of Advanced and Distributed Simulation (PADS' 08), 2008.

Liu Y., Presti F., Misra V., et.al. Scalable fluid models and simulations for large-scale IP networks. ACM Transactions on Modeling and Computer Simulation, Aug. 2004.

Nicol D. and Yan G.. Simulation of network traffic at coarse time-scales. In Proceedings of the Workshop on Principles of Advanced and Distributed Simulation (PADS' 05), 2005.

Kumar S., Park S., Iyengar S., et.al. Time-adaptive numerical simulation for high speed networks. In Proceedings of the International Conference on High Performance Computing, Networking and Communication Systems(HPCNCS' 07), 2007.

Liljenstam M., Liu J., Nicol D., et.al. RINSE: the real-time immersive network simulation environment for network security exercises. Simulation: Transactions of the Society for Modeling and Simulation International, 2006.

Liu J. Packet-level integration of fluid tcp models in real-time network simulation. In Proceedings of the 2006 Winter Simulation Conference(WSC' 06), 2006.

Liu J. Parallel simulation of hybrid network traffic models. In Proceedings of the Workshop on Principles of Advanced and Distributed Simulation (PADS' 07), 2007.

Liu J. and Li Y.. On the performance of a hybrid network traffic model. Simulation Modelling Practice and Theory, 2008, 16(4): 656-669.

Author Biography:

郝志宇: Assistant Researcher, Ph.D., Information Security Research Center, Institute of Computing Technology, Chinese Academy of Sciences
haozhiyu@software.ict.ac.cn

Note: Figure translations are in progress. See original paper for figures.

Source: ChinaXiv —Machine translation. Verify with original.