

Workshop Report on

Computational Modeling of Big Networks (COMBINE)

U.S. Department of Energy
Office of Science
Office of Advanced Scientific Computing Research (ASCR)

December 2012

Editors

Constantine Dovrolis, David M. Nicol, George Riley

Contributors

Jay Aikat, Prasad Calyam, David Clark, Ian Foster, Richard Fujimoto, Dan Gunter, John Heidemann, Raj Kettimuthu, Ezra Kissel, Kevin Mills, Jelena Mirkovic, Inder Monga, Kalyan Perumalla, George Polyzos, Taghrid Samak, Martin Swany, Venkatram Vishwanath, Wenji Wu, Zhi-Li Zhang

Workshop organizers

Richard Carlson, K. Claffy, David Clark, Constantine Dovrolis, Richard Fujimoto, John Heidemann, Srinivasan Keshav, Don Towsley, Zhi-Li Zhang

Additional workshop participants

Peter Barnes, John Bodnar, Bryan Lyles, Raju Namburu, Ken Renard, Jun Yi, Dantong Yu

Workshop dates, location and URL

September 11-12, 2012
American Geophysical Union,
Washington DC.
<http://indico.fnal.gov/event/combine>

Executive Summary

The open science communities supported by DOE/SC (Department of Energy – Office of Science) operate and use a large number of unique scientific instruments and a wide range of compute and storage resources. The DOE/SC ASCR (Advanced Scientific Computing Research) facilities division manages a collection of high-performance computing resources and a national high-performance network to link these compute resources and other scientific instruments to the national and international science communities located at DOE laboratories, universities, and other research centers attached to other Research and Education Networks. The ASCR program office research division funds research in several critical areas including high-performance networks and middleware services needed to ensure effective use of this complex infrastructure.

The Energy Science network (ESnet) is engineered to provide extremely high data transfer capacity. There are two major concerns however which are often voiced by DOE user communities.. The first concern is that end-to-end performance is *not predictable* due to a variety of factors. Even when some performance forecasts or predictions can be made, they often cannot *explain* the reasons why some predictions fail. While deploying measurement tools to capture and display time series throughput and/or delay data is a good first step, it is not sufficient to adequately determine if the observed behavior was correct, or how to predict what future behavior should be.

The second concern is that these research networks are *difficult to re-configure and adapt* to specific application and user needs. Different scientific communities, or even different projects and scientific research tasks, have highly heterogeneous end-to-end requirements and workflow patterns. For these networks to be a fully integrated component of the computing infrastructure, it is not enough that they simply have very high overall capacity. Their operation must be better integrated into the overall ecosystem of computing and storage so that their performance can be sufficiently predictable. In addition, they must support a more flexible and adaptive re-configuration mechanism to deal with changes in demand at short and long time scales.

Large-scale computational modeling is an indispensable tool in the process of conceptualizing, designing, testing, refining, and optimizing predictable and adaptable systems. For example, input processes feeding into a complex network of queues are extremely hard to analyze without computational modeling. Currently it is almost impossible to obtain closed-form analytical models of output processes except for the simplest cases. At the highest-level, the network itself acts as an extremely complex composition operator on millions of data processes.

Understanding how this system works is the essential first step in designing and operating future computing ecosystems.

The DOE/SC ASCR program office was instrumental in elevating computational modeling and simulation to a peer relationship with traditional experimental and theoretical elements of the science discovery process. Numerous science communities including high energy physics, combustion, genomics, material science and climate prediction have all benefited from high fidelity computational models and simulations capabilities. These models allow scientists to explore chemical and physical interactions on both extremely short (nanoseconds) and extremely long (century) time scales. Scientists can zoom in and examine subatomic details of a proton-proton collision or zoom out to explore the details of a type 1-A supernova. By using computational modeling mechanisms scientists gain a better understanding of the physical processes taking place, can assist in the development of experimental facilities by determining the important design criteria, and can quickly explore the feasibility of a scientific theory.

A robust and comprehensive research program that focuses on the analytical modeling and simulation aspects of networks is required to address these concerns. This program should be part of a larger ASCR modeling and simulation program that will address the entire end-to-end nature of DOE's scientific applications.

Findings and recommendations

Finding 1: DOE science communities require predictable and explainable end-to-end performance over networks with adaptable operational characteristics and which respond quickly to fast-changing usage patterns. The network research and operations communities must develop the tools, models, algorithms and services needed to build and operate this dynamic infrastructure. A computational science based approach, is required to advance network design, evaluation, and operation. Combining analytical modeling and simulations activities with physical measurement and monitoring activities will significantly improve the user experience and operational scalability of the network.

Recommendation: The ASCR research division should develop a robust research program to establish the foundational elements needed to build and operate predictable and adaptable networks.

Finding 2: Networks used by DOE science communities carry a wide variety of data flows, with each flow tailored to a specific application's behavior. The net aggregate effects of these flows must be captured analytically to understand the complete system.

Recommendation: New basic research is required for the development of rigorous models that can capture the complexity of such "composition operators" in large scale networks with

sufficient accuracy to solve (or simulate) computationally difficult problems. These models must run sufficiently fast to meet the needs of diverse user communities.

Finding 3: The network, computer science, and application research communities must collaborate with each other to collect and disseminate measurement data. Without accurate and detailed data as input, it will not be possible to populate large-scale networks with accurate traffic loads. Without data for validation, it will not be possible to test whether models, or any of our artificial network topologies, capture the essential aspects of real computer networks.

Recommendation: A coordinated research program will require that all funded researchers systematically exchange data, meta-data, methods and results.

Finding 4: The main challenges for large-scale network simulation are dealing with four considerations: 1) Simulation scale; 2) Simulation fidelity; 3) Simulation speed; and 4) Emulation capabilities.

Recommendation: The network research community needs a small suite of versatile, flexible, and scalable network simulation/emulation tools that can be used to tackle these challenges. This suite should include tools that can simplify complex infrastructures for fast execution times, tools that can deal with hundreds to millions of input parameters for high-fidelity models, and tools that can generate complex topologies that accurately represent physical infrastructures of interest.

Finding 5: Rigorously designed simulation experiments, which accurately describe the large-scale networks used by the DOE science community, require the processing of multiple input parameters and response variables.

Recommendation: Research is required to investigate extensions of the current methods, or adoption of alternative methods, for more complex simulators that process orders of magnitude more input parameters and response variables than today's simulation tools. Parallel efforts that create abstract models which accurately represent complex environments with small numbers of input parameters and response variables should also be explored.

Finding 6: Network simulators must be informed by real-world measurements and be able to reproduce empirical behavior measured in operational networks if the simulator's aim to answer questions about specific, real-world networks constructed with commercial, high-speed routers, optical switches and other components found in DOE science networks.

Recommendation: Research is needed regarding methods to characterize uncertainty associated with network simulators when integrating measurements with simulator parameterizations and comparing results against empirical measurements.

1. Grand Challenge: *Develop predictable and adaptable networks to meet the needs of numerous DOE science communities.*

The open science communities supported by DOE/SC operate and use a large number of unique scientific instruments and a wide range of compute/storage resources. The DOE/SC ASCR facilities division operates a number of high-performance computing facilities and a national high-performance network to link these scientific instruments to the national and international science communities located at DOE laboratories and at universities and other research centers attached to other Research and Education Networks. Operational data from ESnet clearly shows that use of this network has continued to grow at a steady rate (a 10x increase in traffic carried every 47 months) since it was established in the early 1990's [1, 2]. The ASCR program office research division funds research in several areas including high-performance networks and middleware services to ensure effective use of this complex infrastructure.

Recommendation: The ASCR research division should develop a robust research program to establish the foundational elements needed to build and operate predictable and adaptable networks.

ESnet is engineered to provide extremely high data transfer capacity. There are two major concerns however which are often voiced by DOE user communities. The first concern is that end-to-end performance is *not predictable* due to a variety of factors [3, 4]. Even when some performance forecasts or predictions can be made, they often cannot *explain* the reasons why some predictions fail. While deploying measurement tools to capture and display times series throughput and/or delay data is a good first step, it is not sufficient to adequately determine if the observed behavior was correct, or how to predict what future behavior should be.

The second concern is that these research networks are *difficult to re-configure and adapt* to specific application and user needs. Different scientific communities, or even different projects and scientific research tasks, have highly heterogeneous end-to-end requirements and workflow patterns. For these networks to be a fully integrated component of the computing infrastructure, it is *not* enough that they simply have very high overall capacity. Their operation must be better integrated into the overall ecosystem of computing and storage so that their performance can be sufficiently predictable, and they must allow a more flexible and adaptive re-configuration mechanism [5] to deal with changes in demand at short and long time scales.

Without these predictable and explainable attributes, the network is a simple black box that carries traffic between source and destination end points. It is currently extremely difficult for a scientist to effectively use modern networks for the scientific discovery process. Transferring data can be an arduous task requiring tens of man hours to move a few terabytes of data between an experimental facility and the user's home institution. This task is complicated by tools that don't report errors or faults, don't collect fine-grained data to allow real-time analysis, or don't allow for post-processing of data to assist in explaining what factors determined the applications

observed behavior. Computational modeling and simulation tools have been successfully used by several domain scientist communities to increase their understanding of complex science problems [6]. The same mechanisms can be leveraged by the network research community to develop a better understanding of today's networks and to continue enhancing this infrastructure to meet the needs of future scientists.

It may be argued that the Internet protocol already offers a highly adaptive network service (especially compared to connection-oriented network protocols). Indeed, the technology of the Internet has proven to be remarkably resilient to changes in use. Internet protocols have also dealt effectively with over seven orders of magnitude growth in transmission capacities (10's of Kilobits to 100's of Gigabits). The basic packet switching paradigm of the Internet allows for highly dynamic multiplexing of traffic flows, and allows for the capacity of the network to be allocated among flows as they come and go. But this sort of dynamics is *not* the only problem the DOE science networks must solve.

Changing research workflow patterns can change the paths along which high-volume flows need to move. Current routing and traffic engineering protocols, however, do not provide adaptive mechanisms to compute and use different paths, especially when these paths should not be limited by the same bottleneck [7, 8]. This problem is much more severe when these routing paths span different autonomous systems.

High-volume data flows can use significant portions of the end-to-end network capacity for significant periods of time. DOE science communities regularly deal with 10's to 100's of Terabytes of data per experiment. Long running experiments can easily generate Petabytes of data each year [9]. ESnet currently carries over a Petabyte of aggregate data each month. High-volume and/or long lived science data flows do not mix well with low-volume or short lived flows [10]. High-volume flows suffer tremendous performance hits when congestion loss is encountered (a 10^{-4} loss rate can lower throughput by 1-2 orders of magnitude) [11, 12, 13]. Low-volume flows suffer from greater delay and jitter as router queues build up and drain. The current Transmission Control Protocol congestion control algorithms do not effectively deal with the large dynamic range of speeds and performance today's science applications demand [14, 15, 16].

The Internet's TCP (Transmission Control Protocol) was designed to hide application complexity from the network service and to hide network complexity for applications. This makes it difficult to develop a deep understanding of how applications behave over Internet infrastructure. Improving this level of understanding, through the use of advanced modeling and simulation tools and services is an essential element in being able to design, build, and operate networks that provide predictable and explainable behaviors.

Recommendation: New basic research is required for the development of rigorous models that can capture the complexity of such “composition operators” in large scale networks with sufficient accuracy to solve (or simulate) computationally difficult problems. These models must run sufficiently fast to meet the needs of diverse user communities.

Advanced computational modeling and simulation tools can be used to evaluate alternative approaches to building and operating adaptable networks. Detailed simulations at scale would advance our understanding of how different factors in real networks influence their performance and to what degree. These factors span low-level network hardware, network and transport protocols, network dynamics, application design and implementation, workloads that these applications handle, as well as the behavior of human users that interact with them.

2. The need for computational modeling

Large-scale computational modeling is an indispensable tool in the process of conceptualizing, designing, testing, refining, and optimizing predictable and adaptable networks. The biggest challenge in this context is to analytically capture the dynamic behaviors of traffic flowing over realistic network simulations. For example, input processes feeding into a complex network of queues are extremely hard to analyze without computational modeling. Similarly, closed-form analytical models of output processes are almost impossible to obtain except for the simplest cases. At the highest-level, the network itself acts as an extremely complex composition operator on millions of data processes.

Recommendation: New basic research is required for the development of rigorous models that can capture the complexity of such “composition operators” in large scale networks with sufficient accuracy, and at the same time, without being prohibitively slow to solve (or simulate) computationally.

Using the power of computational modeling, the design, evaluation, and operation of adaptable networks can be virtually instantiated and subjected to rigorous exploration of the parameter space, which is impossible to achieve by any other methodology [17]. Emergent behaviors, which are known to result from compositions of even the simplest processes, can be discovered with computational modeling. Such unforeseen aggregate effects can be uncovered by intent rather than by accident, fixed ahead of actual design and deployment. While the critical network effects of software elements are infeasible to incorporate into paper-and-pencil design methodologies, they can be incorporated in a concerted computational modeling approach, accurately exercising their workload characteristics. Further, the most challenging factor to

achieve predictable performance, namely the interacting feedback loops of different protocols and network elements, can also be accurately captured by computational modeling.

Computational models of complex networks can help to predict anomalous behaviors, anticipate major workload variations and capacity demands, quantify the end-to-end performance of applications under hypothetical scenarios, and expose operational vulnerabilities to attacks and failures.

We emphasize that the goal should not be to just model network infrastructures in isolation. Capturing end-to-end behavior will be of paramount importance. This includes end-system performance, middleware services, application characteristics and observed user behaviors. The envisioned computational models will also need to explicitly capture the interactions between users, applications and networks. The application data flows are not confined to individual bulk transfers anymore. Parallel “m-to-n flows” are becoming increasingly common [18, 19, 20]. Bulk data movement over wide-area networks is moving beyond file transfers to memory-to-memory, memory-to-disk and disk-to-memory transfers. Real-time analysis of experimental and simulation data is emerging as a critical requirement for many science areas fueled by advancements in equipment (FastCCD, etc) [21] and advancements in computational power. . User actions, such as aborting transfers or re-attempting them from different data storage sites when the network appears to be slow or unpredictable, may also significantly impact behavior.

The development and execution of large-scale network computational models will be very challenging. The time required to create a simulation model of even a basic protocol is nearly that of developing the actual protocol implementation. Once all needed protocols have simulation models, the computational requirements for simulating a large-scale network with hundreds of routers and thousands of links can be prohibitive if done naively. Today’s simple back-of-the-envelope calculations show that only modestly sized networks can be simulated uniformly at packet-level detail at anywhere near real-time, even using super-computers.

Fortunately, we can learn a lot from the physical sciences and approach modeling with an eye towards *hierarchical levels of abstraction*. An abstract model that retains key features of the underlying detailed model offers the opportunity for simulating network behaviors at rates that are several orders of magnitude faster than a naive approach. Flow approximations effectively [22] abstract away individual packets, while protocol specification and behavior is usually packet oriented. The key research challenges for flow approximations are centered in 1) developing methods for quickly capturing the impact of interacting protocols on flows, and 2) developing high-performance algorithms for evaluating those models. Of course the use of flow approximations is only one possible solution; the research community will need to develop additional computational modeling approaches [23] that can simultaneously capture network complexity, large scale and simulation efficiency.

3. The necessity for accurate data

The availability of accurate, timely, and fully detailed network data is essential to large-scale computational modeling. The data is required for a number of purposes including initial development of network models (based on the observed behaviors in the measured data) and to verify the validity of those models once they are created. Without accurate and detailed data as input we cannot populate large-scale networks with accurate traffic loads. Without data for validation, we cannot test if any of our models or any of our artificial network topologies capture the essential aspects of real computer networks.

Collecting data about real networks faces several challenges: useful data is often proprietary and kept secret due to business or user-privacy concerns [24, 25]. Fully detailed network packet information is typically voluminous and it is difficult for researchers to identify the information that is most meaningful to their experiments. Finally, networks evolve rapidly and so data as recent as a few months old may not provide insight into current and future networks..

These challenges pose several goals for medium-term research in network data collection, data storage and large-scale analytics:

1. Can we develop frameworks that allow sharing of semi-sensitive network data to protect commercial interests or user privacy while still allowing research on realistic topologies and traffic?
2. Can we develop new anonymization or abstraction approaches to capture essential elements of topology and traffic?
3. Can we correlate log file data (from end or intermediate nodes) and active measurement data with experimental and/or simulation results given the data may be in unique or application specific formats and timestamps may be inconsistent or missing?
4. Do models built from existing data allow exploration of "what if" scenarios, to understand network behavior on different possible topologies and traffic mixes of tomorrow?
5. How can we design an adaptable set of "meta-data" to describe the actual data being captured, used as simulation inputs or created as simulation outputs?
6. How can we reduce the volume of data created by packet header capturing methods on high-speed core routers and links?

These challenges can only be addressed through a coordinated research program that will require all funded researchers to systematically exchange data, meta-data, methods and results. Additionally, such a coordinated research program would provide the incentive for researchers to catalog and share their approach to modeling, simulation, and data gathering. This also makes it necessary to agree on representations for model inputs and outputs so that cross-validation studies can be easily performed. The availability of a unified data representation across layers (as

well as end-to-end) will also be necessary for accurate and efficient model verification. Efficient ways of data transformations from different representations across layers should be investigated. Such transformations will enable efficient inference of high-level abstractions from low-level measurements data, and efficient use of aggregations and summarizations. Transformations in the reverse direction, from high-level models to low-level data that are generated by those models and that can drive actual networks and systems will also be valuable as a model verification tool.

To date, DOE has funded a few research projects on network performance monitoring and analysis, such as PerfSONAR [26] and Network Weather Service [27]. These research projects provide infrastructure and services to collect real network data. These projects are good starting points for further research to address the above challenges.

Recommendation: A coordinated research program will require all funded researchers systematically exchange data, meta-data, methods and results.

4. New simulation methods and tools

Existing discrete-event simulation platforms such as NS2 [28, 29, 30] and NS3 [31, 32] cannot presently scale to large end-to-end infrastructures of comparable size to existing deployed networks. The main challenges for large-scale network simulation are dealing simultaneously with four considerations:

1. **Simulation scale.** Simulations of modern distributed science communities require scaling in several dimensions including: size - the number of nodes connected to the network, distance – the geographical distances between nodes, speeds – multiple links at different speeds, and application diversity – many different applications with different transmission characteristics.
2. **Simulation fidelity.** Simulations must provide sufficiently realistic models of network components, protocols and applications.
3. **Simulation speed (execution time).** The simulation must complete in a reasonable amount of time, depending on the simulation use (real-time prediction to future infrastructure planning).
4. **Integration and interoperability with real networks.** Simulation results must be comparable with measurements taken from real networks to validate the simulations. Measured data may also be used as input to emulation tools to gain a better understanding of how an infrastructure change would impact current network users.

Several existing simulators focus on two or three of these issues [33] but none can meet all of them at the same time. One path forward might be in multi-scale simulation. Here the term

“multi-scale simulation” [23] refers to modeling some events and components of interest at a fine-grain level and modeling the rest with coarse-grain models. One specific challenge faced by this approach is defining how models at different levels of detail interact, such as what happens when a fluid-model flow meets a packet-model flow. Further, any new and promising approaches must provide the ability to simulate a system of systems, where complex components and segments of networks together with protocols and applications can be modeled at a higher-level, achieving simulation speed and scale, while providing sufficient fidelity. Assuming this multi-resolution simulation is possible, a related challenge is how to educate users about the best way to use such a multi-scale simulator. For instance, it is often hard to identify phenomena that require detailed simulation versus those that can be simulated at coarse grain with less detail.

Recommendation: The network research community needs a small suite of versatile, flexible, and scalable network simulation/emulation tools that can be used to tackle these challenges. This suite should include tools that can simplify complex infrastructures for fast execution times, tools that can deal with hundreds to millions of input parameters for high-fidelity models, and tools that can generate complex topologies that accurately represent physical infrastructures of interest.

It is possible that one or more of the existing network simulation tools such as NS3 [29] can evolve in that direction. The constancy of the simulation platform would enable easy comparison between solutions, recreation of prior simulation results by others, and easy adoption of models and ideas among research groups.

When using flow models and other techniques, the temporal component presents a challenge for implementation of the model, requiring some coarse time-stepping on top of the "pure" model of continuous flow. This is conceptually similar to many large-scale scientific simulations where forces are computed on particles in one time step, then the particles are moved and the computation repeats. From that domain, we may be able to borrow "adaptive" computational techniques [34] that allow the granularity of the computation to vary with the amount of activity in a given section of the flow network.

Modeling large-scale static configurations using model-checking approaches [35] has enabled accurate verification of large networks, as well as answering what-if configuration questions. Leveraging such formal models to enable dynamic network simulation can open a wide range of opportunities for large-scale network analysis. For example, static optimization can enable efficient model verification before the actual start of the simulation.

Another significant problem with all existing network simulation tools is excessive execution time for any non-trivial network with high-speed links and heavy traffic loads. The current approach used by nearly all tools is a “discrete event simulation” whereby every single packet on

every single link results in one or more “events”, each of which takes some non-zero amount of processing time in the simulation tool. Modern networks of moderate size can easily generate trillions of events or more, which results in overall execution time of weeks [36] or longer to model any reasonable amount of simulation time.

However, as the systems being modeled and simulated become more extensive, encompassing larger systems with more components, the questions raised most often relate to different parts of the system simultaneously and are posed at different time scales or modeling granularities. This leads to the need for developing multi-layer, multi-resolution, multi-technology models and simulators or other types of performance evaluation software. An example here could be a simulator where a specific process or event is not realized at the same level of detail as the remainder of the system, but rather its impact on other system metrics (such as delays, bit errors, and other information loss) is projected. This projection could be done by numerically computing the impact based on analytic models with parameters obtained at simulation time. Thus models and simulation tools will need to continuously balance the trade-off between fidelity and execution speed to produce the expected results.

In addition, there are opportunities and needs for having simulators interact with testbeds and real operational networks. For example, in some cases it may be useful and efficient to create partial workloads for simulators from actual software implementations of applications executed on other parts of the system, rather than attempt to model analytically or simulate those parts. In contrast, other network components or behaviors could be difficult to implement in a testbed, but easy to model in the simulator. This simulator-testbed interaction must be done in real-time so that the former can be used to realize the right “transfer function” for that sub-system.

While the primary challenge is developing mechanisms to provide reliable and predictable network performance, the modeling and simulation environment envisioned here could have other uses. For example, a desirable feature for any newly developed network would be the ability to control the networks “energy-efficiency” state which indicates the overall amount of power consumed by a network normalized to the number of bits transmitted by that network. It could be imagined that a researcher may not care about per-packet performance when trying to measure network energy for a year of operation. When dealing with the problem of network energy-efficiency, instead of optimizing a certain layer or function of the network, we need to look at the network as a “whole system” and try to optimize the amount of energy it consumes to deliver the functional capabilities needed by the end-devices and/or users. In order to design new networks and protocols that are energy efficient, we need to develop an end-to-end multi-layer simulation platform to simulate the overall energy consumption of a network. The energy consumption would be calculated with dynamic traffic and flow behavior in addition to dynamically configured networks. In current practice, no such tool exists and the energy of the

networks can only be computed after the network is deployed. This makes energy-efficiency research and the impact on actual energy consumption extremely difficult.

5. Effective methods and best practices

Network simulators enable experiments to study the behavior of models within a computer, rather than the behavior of real objects within a laboratory. The meaningfulness of simulation results depends on the quality of the model with respect to the system it represents (verification and validation) and the utility of the simulation results depends on how the experiment has been conducted (experimental design and analysis). Current tools and models were mainly developed to address the needs of the commercial Internet. Updating tools and models to accurately model and simulate networks of interest to DOE, which support end-to-end operations at orders of magnitude greater rates only exacerbate these issues. Network simulation researchers and modelers cannot work in isolation from these questions, but they should collaborate with statisticians and applied mathematicians, who bring a century of experience in experimentation, modeling, and analysis within many engineering disciplines and physical sciences communities.

Experiment Design: Prior to using a network simulator, an experimenter must understand the main input factors that drive the simulator, and the specific response dimensions that the simulator exhibits. The current state-of-the-art in this area uses orthogonal fractional experimental design methodologies [36] and either principal components analysis or correlation analysis and clustering [37] to conduct sensitivity analyses to determine what input parameters drive a simulator, and along what response dimensions. Using these methods can increase confidence that a simulator is correct, can provide more accurate estimates of main effects (and two-term and even higher-order parameter interactions) and can help determine the parameter combinations and responses that must be considered when a simulator is applied to answer specific “what-if” questions. To date, current methods have proven effective for network [38] and distributed system [39] simulators with up to 20 input parameters and 45 response variables. The state of the art is significantly far from what is needed to rigorously design experiments for large-scale networks.

Recommendation: Research is required to investigate extensions of the current methods, or adoption of alternative methods, for more complex simulators that process orders of magnitude more input parameters and response variables than today’s simulation tools. Parallel efforts that create abstract models which accurately represent complex environments with small numbers of input parameters and response variables should also be explored.

Verification and Validation: To gain confidence that a network simulator is correct and valid for an intended purpose, experimenters must take careful steps to assess model correctness [40].

The current-state-of-the-art in this area compares simulator results to empirical measurements from small-scale topologies that can be created in a laboratory [41]. Once critical behaviors in the simulator can reproduce empirically measured behaviors at small scale, an experimenter will typically investigate those behaviors at large scale by simulating topologies that are very difficult to reproduce empirically in a controlled setting. At present, results from sensitivity analysis of a simulator running at large scale can be discussed with network service providers that design and operate such networks. These discussions can provide confidence that a simulator qualitatively matches the behaviors salient to network operators. Research into empirical test-beds, such as GENI [42, 43, 44], and ESnet [45] might increase the feasibility of producing empirical results that can be compared with quantitative results from a simulator with the same, large topology. On the other hand, a network simulator may require abstractions and simplifications that are difficult to implement in a test-bed, where various details are included even though they might be omitted from abstract models. One technique is to embed implemented protocols in the model from Linux or FreeBSD. In this way a protocol model can be verified and validated against an operational prototype, if the implemented code for the protocol is itself incorporated in the model [46]. Of course, importing such implementation code might increase the computational requirements of a model, thus there can be a significant tradeoff between model performance and fidelity, as discussed previously in Sec. 2.

Establishing that network simulators reproduce empirical behavior measured in test-bed networks may prove insufficient if the simulator aims to answer questions about specific, real-world networks constructed with commercial, high-speed routers. Detailed end-to-end measurements are required for providing model input and verifying and validating the output. Consistent, efficient, access to these measurements as dynamic attributes of the end-to-end topology is needed to enable this analysis. If successful, this program will develop simulations with the ability to accurately model configurations and protocols that have not yet been deployed on real systems.

Recommendation: Research is needed regarding methods to characterize uncertainty associated with network simulators when integrating measurements with simulator parameterizations and comparing results against empirical measurements.

Best Practices: To produce well-understood, rigorous and reproducible scientific network simulation studies, researchers should adopt an agreed upon set of methods and best practices. The current state-of-the-art in this area is that each network simulation study is conducted with unique methods particular to individual experiments and experimenters. Further, network simulation studies are seldom reproduced, either with the same or different simulators. Failure to reproduce network simulation studies leaves the networking research and engineering community with little in the way of scientific underpinnings that can be confidently accepted as a basis for future work. Networking simulation researchers should work together with statisticians

and mathematicians to establish accepted methods and best practices for conducting and reporting simulation experiments with detailed network simulators [47], with abstract network simulators [22] and with hybrid network simulators [23].

7. Linkage to other ASCR programs

In 2012 ASCR began evaluating the current status of modeling and simulation environments for the emerging Exascale computing initiative (ECI). This work focuses on application behavior on current and next generation supercomputers. ASCR should evaluate how to create a coherent modeling and simulation program that encompasses all elements of the end-to-end computing ecosystem.

8. Conclusions

The DOE ASCR program office pioneered the development of computational science as a major activity in the scientific discovery process. A robust and comprehensive analytical modeling and simulation program that addresses the end-to-end performance issues facing DOE scientists will continue this process and make it easier for scientists and engineers to use DOE's unique facilities. This network based program should be part of a larger ASCR program that deals with application modeling on supercomputers and in distributed computing environments. Only by bringing all communities to the table can the end-to-end issues be identified and resolved.

Appendix: Original description of the COMBINE workshop

Most networking research is driven by the creation of artifacts such as protocols, applications, middleware, routers and other network elements. The Internet itself is one amazingly successful outcome of this approach. Despite this success, we have come to realize that we only partly understand how the Internet works. This is brought into sharp focus when an underlying component of the Internet malfunctions or is under attack. If we were to look at the Internet in the same way that a biologist looks at a living organism or a geophysicist looks at the climate system, what would be the Internet research agenda? How would we try to understand, monitor, and control the Internet, considering the full extent of its complexity? Would our approach change? Should it?

Vertical understanding:

Networking research typically focuses on individual components or layers of the overall Internet architecture. For example, if the focus of a research project is on congestion control, there is typically little or no consideration about how applications actually use the network or how users react to congestion events. This is a direct outcome of a reductionist stance in conducting a scientific investigation. In reality, however, users, applications, and transport protocols are all interdependent and it is precisely their interactions that create much of the complexity in determining end-to-end performance. By "vertical understanding" we refer to a research agenda that aims to understand networks in a holistic manner, starting from the users and applications, and socio-economic structures at the top all the way down to effects that occur at the physical layer. Through a careful analysis of this vertical path, we may be able to discover complex interactions and important effects in network behavior and performance that we can only suspect at this point.

Horizontal understanding:

A typical end-to-end Internet path today is highly heterogeneous in terms of the infrastructure as well as the policy boundaries that it traverses (think of the path between a mobile user carrying a smart phone and downloading media-rich content from a set of different sites served by different CDNs and data centers). Most models for Internet performance are still based on simplistic models of individual queues, small-scale simulation topologies, and interdomain routing topologies that collapse an entire Autonomous System to a single node. By "horizontal understanding" we refer to a research agenda that aims to capture the diverse nature of end-to-end Internet paths, considering both the technological heterogeneity along a path as well as the policy and economic boundaries that are crossed by those paths.

Large-scale computational modeling and analysis:

The two objectives of vertical and horizontal understanding will most likely demand new research methods and tools. We suspect that existing analytical tools or experimental approaches (such as test-beds) will not be sufficient in capturing the vertical and horizontal complexity of the Internet. Instead, we believe that large-scale computational modeling, a powerful research tool that has been largely unexplored by networking researchers, may be the right approach to study these objectives. This is motivated by other disciplines, such as climate science or physics, that have been using supercomputers and large-scale computational modeling for a long time with many successful results. What if we could construct large-scale computational models that

can capture what happens to a computer network all the way from the transmission of bits to the complex and multifaceted latest Internet applications? What would be the major challenges and objectives of that research agenda? This, then, is the focus of this workshop.

References:

- [1] <https://stats.es.net/>
- [2] <http://esnetupdates.wordpress.com/2010/12/10/why-this-spiking-network-traffic/>
- [3] G. Liang and I. Matta, "The war between mice and elephants," *Network Protocols*, 2001.
- [4] K-C Lan and J. Heidemann, "A measurement study of correlations of Internet flow characteristics", *Computer Networks* Volume 50, Issue 1, 16 January 2006, Pages 46–62.
- [5] W. Johnson, J. Metzger, M. O'Connor, M. Collins, J. Buresica, E. Dart, J. Gagliardi, C. Guok and K. Oberman, *Network Communication as a Service-Oriented Capability, High Performance Computing and Grids in Action*, Volume 16 *Advances in Parallel Computing*, Editor: L. Grandinetti, March 2008, IOS Press.
- [6] <http://www.scidac.gov/>
- [7] Quoitin, Bruno, et al. "Interdomain traffic engineering with BGP." *Communications Magazine*, IEEE 41.5 (2003): 122-128.
- [8] Caesar, Matthew, and Jennifer Rexford. "BGP routing policies in ISP networks." *Network*, IEEE 19.6 (2005): 5-11.
- [9] W. Wu et al, *An Analysis of Bulk Data Movement Patterns in Large-scale Scientific Collaborations*, J. Phys.: Conf. Ser. 331 012008
- [10] X. Fang and M. Veeraraghavan, "A Hybrid Network Architecture for File Transfers," *IEEE Transactions of Parallel and Distributed Systems*, vol. 20, no. 12, pp. 1714-1725, 2009.
- [11] Lakshman, T. V., and Upamanyu Madhow. "The performance of TCP/IP for networks with high bandwidth-delay products and random loss." *Networking*, IEEE/ACM Transactions on 5.3 (1997): 336-350.
- [12] Kumar, Anurag. "Comparative performance analysis of versions of TCP in a local network with a lossy link." *IEEE/ACM Transactions on Networking (ToN)* 6.4 (1998): 485-498.
- [13] Li et al. 2007. Experimental Evaluation of TCP Protocols for High-Speed Networks. *g p Transactions on Networking*. 15:5, 1109-1122.
- [14] C. Jin et al., "FAST TCP: From Theory to Experiments," *IEEE Network*, vol. 19, no. 1, pp. 4-11, January/February 2005.
- [15] D. J. Leith, et al., "H-TCP: A framework for congestion control in high-speed and long-distance networks," Hamilton Institute Technical Report, August 2005.
- [16] I. Rhee L. Xu, "CUBIC: A New TCP-Friendly High-Speed TCP Variant," *PFLDnet'05*, 2005.
- [17] K. Mills, J. Filliben, D. Cho, E. Schwartz and D. Genin, Chapters 6-9 in *Study of Proposed Internet Congestion Control Mechanisms*, NIST Special Publication 500-282, May 2010, 534 pages.
- [18] V. Vishwanath, M. Hereld, V. Morozov and M. Papka, "Topology-Aware Data Movement and Staging for I/O Acceleration on Leadership Computing Systems", *In IEEE/ACM Supercomputing 2011*
- [19] J. Yi, R. Kettimuthu and V. Vishwanath, "Accelerating Data Movement Leveraging Endsistem and Network Parallelism," *IEEE/ACM SC12 Workshop on Network-aware Data Management*, Nov. 2012.
- [20] D. Gunter, R. Kettimuthu, E. Kissel, M. Swamy, J. Yi, J. Zurawski, "Exploiting Network Parallelism for Improving Data Transfer Performance," *IEEE/ACM Annual SuperComputing Conference (SC12) Companion Volume*, Nov. 2012.
- [21] S. Narayanan, T.P. Madden, A.R. Sandy, R. Kettimuthu, M. Link, "GridFTP based real-time data movement architecture for x-ray photon correlation spectroscopy at the Advanced Photon Source," *IEEE eScience 2012 Workshop on eScience Meets the Instrument*, Oct. 2012.
- [22] Y. Yi and S. Shakkottai, "FluNet: A hybrid internet simulator for fast queue regimes", *Computer Networks: The International Journal of Computer and Telecommunications Networking*, 51:18, Elsevier North-Holland, 2007, pp. 4919-4937.

-
- [23] J. Lee, S. Bohacek, J. Hespanha and K. Obraczka, "Modeling Communication Networks with Hybrid Systems", *IEEE/ACM Transactions on Networking*, 15:3, 2007 pp. 630-643.
- [24] M. Fullmer, S. Romig, "The OSU Flow-tools Package and Cisco NetFlow Logs", *LISA XIV*, 2000.
- [25] S. E. Coull, M. P. Collins, C. V. Wright, F. Monrose, M. K. Reiter, "On Web Browsing Privacy in Anonymized NetFlows", *16th USENIX Security Symposium on USENIX Security Symposium*, p. 23, doi=10.1.1.131.3687, 2007.
- [26] PerfSONAR Website, <http://www.perfsonar.net/>
- [27] Network Weather Service Website, <http://nws.cs.ucsb.edu/>
- [28] NS2 Website, <http://www.isi.edu/nsnam/ns/>
- [29] http://nsnam.isi.edu/nsnam/index.php/User_Information
- [30] N. Baldo, F. Maguolo, M. Miozzo, M. Rossi, M. Zori, "ns2-MIRACLE: a modular framework for multi-technology and cross-layer support in network simulator 2", *Proceedings of the 2nd international conference on Performance evaluation methodologies and tools*".
- [31] NS3 Website, <http://www.nsnam.org/>
- [32] T. Henderson, M. Lacage and G. Riley, "Network Simulations with the NS-3 Simulator", *SIGCOMM 2008*, p. 527.
- [33] E. Weingartner, H. vom Lehn, and K. Wehrle, "A Performance Comparison of Recent Network Simulators," *Communications, 2009. ICC '09*.
- [34] M.J. Berger, P. Colella, "Local adaptive mesh refinement for shock hydrodynamics", *Journal of Computational Physics*, Vol 82 Issue 1, May 1989 pp. 64-94
- [35] R. Alur, "Model-checking for real-time systems", *Logic in Computer Science (LICS '90)*, June 1990, pp. 414-425
- [36] K. Mills, J. Filliben, D. Cho, E. Schwartz and D. Genin, Chapter 4 in *Study of Proposed Internet Congestion Control Mechanisms*, NIST Special Publication 500-282, May 2010, 534 pages.
- [37] K. Mills and J. Filliben, "Comparison of Two Dimension-Reduction Methods for Network Simulation Models", *Journal of Research of the National Institute of Standards and Technology*, 116-5, September-October 2011, pp. 771-783.
- [38] K. Mills, E. Schwartz and J. Yuan, "How to Model a TCP/IP Network using only 20 Parameters", *Proceedings of the Winter Simulation Conference*, December, 2010, pp. 849-860.
- [39] K. Mills, J. Filliben and C. Dabrowski, "An Efficient Sensitivity Analysis Method for Large Cloud Simulations", *Proceedings of the 4th International Cloud Computing Conference*, IEEE, Washington, D.C., July 5-9, 2011.
- [40] Kleijnen, Jack PC. "Verification and validation of simulation models." *European Journal of Operational Research* 82.1 (1995): 145-162.
- [41] K. Mills, J. Filliben, D. Cho, E. Schwartz and D. Genin, Chapter 5 in *Study of Proposed Internet Congestion Control Mechanisms*, NIST Special Publication 500-282, May 2010, 534 pages.
- [42] Elliott, Chip, and Aaron Falk. "An update on the geni project." *ACM SIGCOMM Computer Communication Review* 39.3 (2009): 28-34.
- [43] Mahindra, R., et al. "Integration of heterogeneous networking testbeds." *Proceedings of the 4th International Conference on Testbeds and research infrastructures for the development of networks & communities*. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008.
- [44] J. Aikat and K. Jeffay, Tutorial: Introduction to Network Experiments using the GENI CyberInfrastructure, *SIGMETRICS*, June 2012.
- [45] <http://www.es.net/RandD/100g-testbed/>
- [46] Bagrodia, Rajive, and Mineo Takai. "Position paper on validation of network simulation models." *Computer Science Department of University of California, Los Angeles* (1999).
- [47] J. Cowie, H. Liu, J. Liu, D. Nicol and A. Ogielski, "Towards Realistic Million-Node Internet Simulations", *Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications*, June 28- July 1, 1999, Las Vegas, Nevada.