C O M P L E X S Y S T E M S

An ITL Program

ACTIVITIES REPORT

FISCAL YEAR 2009

complex
systems

Information Technology Laboratory

NISTIR - 7688

NIST

**National Institute of
Standards and Technology**

U.S. Department of Commerce

Complex Systems Program Activities Report Fiscal Year 2009

Sandy Ressler, Editor
Information Technology Laboratory
Program Manager

May 2010

Table of Contents

Abstract

This report summarizes the work of the Complex Systems Program within NIST's Information Technology Laboratory. The Overview presents some background material and a high-level overview of the Program's activities including significant technical accomplishments. The Projects chapter provides short descriptions of each of the projects which comprise the program. Programmatic Activities describes the activities taking place to manage the Program and enable inter-project interactions. Finally, the Milestones & Plans chapter presents strategic goals and objectives of the program.

For further information contact Sandy Ressler, NIST, 100 Bureau Drive, STOP 8940, Gaithersburg, MD 20899; phone (301) 975-3549, email: sressler@nist.gov or see the Program web site at: http://www.itl.nist.gov/ITLPrograms/ComplexSystems/.

$$(2^{32})^{1000} \longrightarrow O(10^{9633}) \quad [10^{80} = \text{atoms in visible universe}]$$

Discard parameters not germane to study – reduce by 944 parameters

$$(2^{32})^{56} \longrightarrow O(10^{539})$$

Group related remaining parameters – reduce by 36 parameters

$$(2^{32})^{20} \longrightarrow O(10^{192})$$

Select only 2 values for each parameter

Model Reduction

Level Reduction

Experiment Design Theory

Sensitivity Analysis

$$2^{20} \longrightarrow O(10^6)$$

Use experiment design theory to reduce parameter combinations to 256

$$2^{20-12} \longrightarrow 256$$

Use sensitivity analysis to identify six most significant parameters

$$2^{6-1} \longrightarrow 32$$

Use experiment design theory again to reduce parameter combinations to 32

Reduction in parameter space for the experiments conducted, the Measurement Science for Complex Information Systems project.

Overview

"If we knew what it was we were doing, it would not be called research, would it?" - Albert Einstein

Complex Systems are composed of large, interacting entities which taken together, exhibit a macroscopic behavior which is not predictable by examination of the individual entities. The Complex Systems Program seeks to understand the fundamental science of these systems and develop rigorous descriptions (analytic, statistical, or semantic) that enable prediction and control of their behavior. Initially focused on the Internet and Grid Computing, this Program will facilitate predictability and reliability in these areas. The forthcoming report "Study of Proposed Internet Congestion Control Mechanisms" represents a state-of-the-art experiment examining Internet Congestion Control. Equally as important, the methodology used in the experiment provides the clearest example of experimental design applied to this domain and is a major contribution to the field. In the future, we hope to apply these complex systems capabilities to other domains such as biotechnology, nanotechnology, semiconductors, and engineering in support of the other laboratories at NIST.

The Complex Systems Program is one of several programs in the Information Technology Laboratory. These programs represent the strategic direction for research and activities of the lab. In brief, the other programs (taken from the ITL overview web pages at http://www.nist.gov/itl/ are:

Cyber and Network Security

Cyber and network security is focused on ensuring three security objectives of information technology systems: confidentiality, integrity, and availability. The Cyber and Network Security Program addresses NIST's statutory responsibilities in the domain and the near- and long-term scientific issues in some of the building blocks of IT and network security - cryptography, security testing and evaluation, access control, internetworking services and protocols (Domain Name System, Border Gateway Protocol, IPv6, Wi-Max, etc.), security metrics, vulnerability analysis, security automation, and security properties. These efforts will provide a more scientific foundation for cybersecurity, while maintaining a focus on near-term security issues in emerging technologies.

Enabling Scientific Discovery

Modern scientific research has become more and more dependent on mathematical, statistical, and computational tools for enabling discovery. The Enabling Scientific Discovery Program promotes the use of these tools to dramatically advance our ability to predict the behavior of a broad range of complex scientific and engineering systems and enhance our ability to explore fundamental scientific processes. This Program focuses on interdisciplinary scientific projects that involve novel computational statistics and the development of simulation methods and software. These efforts will have a foundational impact on scientific discovery throughout U.S. industry, government, and academia.

Identity Management

Electronic identities are routinely used to access logical and physical resources, and have become a ubiquitous part of our national infrastructure. Identity management systems are responsible for the creation, use, and termination of electronic identities. However, Identity Management as a formal discipline is a fairly new concept. In conjunction with other federal agencies, academia, and industry partners, the NIST Identity Management Systems Program is pursuing the development of common models and metrics for identity management, critical standards, and interoperability of electronic identities. These efforts will improve the quality, usability, and consistency of identity management systems, protect privacy, and assure that U.S. interests are represented in the international arena.

Information Discovery, Use, and Sharing

In the past, data was a precious and rare commodity: expensive to produce and difficult to exchange. Today, we are awash in it – with 90,000 terabytes of web-based information, an annual increment of 400,000 terabytes of e-mail, many companies increasing their web presence, large collections of scientific and sensor-based data, and an increased need for automated newswire, speech, and video analysis and translation. Novel computational approaches to data collection and analysis must be combined with improved interoperability techniques to effectively extract needed information from the wealth of available data. For key areas of national importance, the IDUS program will foster innovation throughout the information life cycle by developing the measurement infrastructure to enable scientific/knowledge discovery, widespread, seamless, and secure information exchange, and the usability of vast stores of information.

Pervasive Information Technologies

Pervasive information technology is the trend towards increasingly ubiquitous connected computing sensors, devices, and networks that monitor and respond transparently to human needs. The Pervasive Information Technologies Program facilitates the creation of standards for sensor communication, networking interoperability, and sensor information security. The Program enables the use of pervasive information technologies to enhance personal and professional productivity and quality of life.

Trustworthy Information Systems

Ensuring that our nation's information systems are trustworthy is becoming increasingly important as we become more dependent on them for reliable, secure, and safe operation in nearly all sectors of our economy, national defense, homeland security, healthcare, and personal life. As systems grow in size and complexity, and become increasingly interconnected through networks and communication links, their vulnerability to attack from hostile elements, or failures due to inherent defects or exploited vulnerabilities, increase their risk of failure or compromise with significant impacts to businesses, services, equipment or users depending on them.

Virtual Measurement Systems

A virtual measurement is a quantitative result and its uncertainty, obtained primarily by computer simulation or computer-assisted measurements. Examples of virtual measurements include computational models of physical systems and visualizations of the results. The Virtual Measurement Systems Program introduces metrology constructs - standard reference computations, uncertainty quantification, and traceability - into scientific computation and computer-assisted measurement technologies. As with physical measurement systems, development of a virtual metrology infrastructure will result in predictive computing with quantified reliability. In turn, this will enable improved decision making contingent on virtual measurement systems.

Context and National Priorities

One important question we must answer is who cares about this "complex systems" work? In the context of IT networks there is a nearly 30-year history of unexpected failures:

- ARPAnet congestion collapse, 1980
- Internet congestion collapse, Oct 1986
- Cascading failure of AT&T long-distance network, Jan 1990
- Collapse of AT&T frame-relay network, April 1998
- Slammer Worm near collapse of several countries Internet access (Korea), Jan 2003
- Mass use of backup routes, 4000 Cisco routers cause 7hr downtime in NE Japan, May 2007

As society becomes increasingly dependent on IT networks these types of failures become increasingly expensive to recover from, and prediction is increasingly valuable. No one understands how to measure, predict or control macroscopic behavior in complex information systems

"[Despite] society's profound dependence on networks, fundamental knowledge about them is primitive. [G]lobal communication … networks have quite advanced technological implementations but their behavior under stress still cannot be predicted reliably."(pp. 1)

"There is no science today that offers the fundamental knowledge necessary to design large, complex networks in such a way that their behaviors can be predicted prior to building them." (pp. 3)

— Network Science, National Research Council, 2006
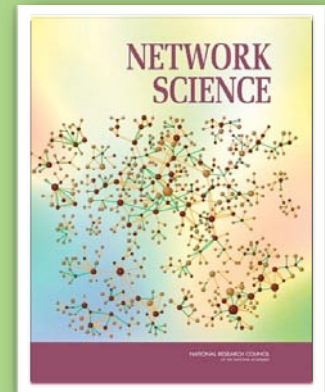
Network problems remain such that cascading failures of the power grid, Internet, phone and cable systems continue to occur. While the Internet was formed organically we really don't understand its topology, therefore management is problematic and the impact of new services and protocols is unpredictable. There is no measurement science for these systems.

A more timely example of a potential complex systems issue is the recent example in which Google Applications lost functionality for approximately 2 hours. Note that such an open accounting of this event is rare, extremely useful and surely helps other organizations with similarly complex data centers address their problems.

The underlying cause of the outage was a power failure in our primary datacenter. While the Google App Engine infrastructure is designed to quickly recover from these sort of failures, this type of rare problem, combined with internal procedural issues extended the time required to restore the service.

What did we do wrong?

Though the team had planned for this sort of failure, our response had a few important issues:

- Although we had procedures ready for this sort of outage, the oncall staff was unfamiliar with them and had not trained sufficiently with the specific recovery procedure for this type of failure.

- Recent work to migrate the datastore for better multihoming changed and improved the procedure for handling these failures significantly. However, some documentation detailing the procedure to support the datastore during failover incorrectly referred to the old configuration. This led to confusion during the event.

- The production team had not agreed on a policy that clearly indicates when, and in what situations, our oncall staff should take aggressive user-facing actions, such as an unscheduled failover. This led to a bad call of returning to a partially working datacenter.

- We failed to plan for the case of a power outage that might affect some, but not all, of our machines in a datacenter (in this case, about 25%). In particular, this led to incorrect analysis of the serving state of the failed datacenter and when it might recover.

https://groups.google.com/group/google-appengine/browse_thread/thread/a7640a2743922dcf
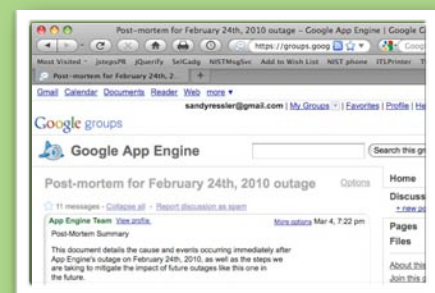
The "Northeast Blackout of 2003" at the time was the "second most widespread electrical blackout in history". Failures were the result both of insufficient maintenance of trees which can cause powerlines to short out, and more interestingly for complex systems issues, a software race condition in the energy management system of the power plant operators systems. The software bug caused a cascade of errors resulting in much slower response times to address the crisis by power plant personnel.

A software bug known as a race condition existed in General Electric Energy's Unix-based XA/21 energy management system. Once triggered, the bug stalled FirstEnergy's control room alarm system for over an hour. System operators were unaware of the malfunction; the failure deprived them of both audio and visual alerts for important changes in system state. After the alarm system failure, unprocessed events queued up and the primary server failed within 30 minutes. Then all applications (including the stalled alarm system) were automatically transferred to the backup server, which itself failed at 14:54. The server failures slowed the screen refresh rate of the operators' computer consoles from 1–3 seconds to 59 seconds per screen. The lack of alarms led operators to dismiss a call from American Electric Power about the tripping and reclosure of a 345 kV shared line in northeast Ohio. Technical support informed control room personnel of the alarm system failure at 15:42.

http://en.wikipedia.org/wiki/Northeast_blackout_of_2003

The Recent crashes of the Washington DC Metro have increased some scrutiny of highly complex interrelated system. Often these systems designed with the best of intent fail to take into account rare circumstances. For another design fault example:

Raja Parasuraman, a psychologist at George Mason University who studies how humans interact with automated systems, said something similar once happened in aviation: Planes coming in to land were running into trouble because pilots occasionally activated reverse-thrust braking systems before the wheels touched down, causing the aircraft to stall and crash.
Designers thought they could solve the problem by eliminating the pilot's judgment from the equation -- they installed weight-sensitive sensors that activated once the wheels touched down. Until the sensors activated, pilots had no control over the reverse-thrust system. On a rainy night in Warsaw, Parasuraman said, a plane touched down but started hydroplaning on the film of water on the runway. With the aircraft skimming over the surface, the weight-sensitive sensors did not trigger -- preventing the pilots from activating the thrust reversers and causing the plane to overshoot the runway. Washington Post, "Metrorail Crash May Exemplify Automation Paradox" Shankar Vedantam

http://www.washingtonpost.com/wp-dyn/content/article/2009/06/28/AR2009062802481.html

Portfolio of Projects

The projects which comprise the Complex Systems Program interact and enhance each other in many meaningful ways. The projects are meant to address a gap in research at NIST or an enhancement to a pre-existing project. Clearly, the IMS Measurement Science for Complex Information Systems forms the core of the program. (IMS, Innovations in Measurement Science projects are internally funded basic research projects.) In addition, the need to develop new competencies is being addressed by three other small projects. Some of the projects have made striking progress towards the development of metrics providing significant insight into the mathematical nature of large scale networks.

Project	Gaps	
Measurement Science for Complex Information Systems	fundamentals, scale	IMS or closely related apply core statistical competence
Visualizing Networks	infoviz	
Discrete Experimental Design Statistics	fundamentals	
Methods for Characterizing Complex Networks	attributes/scale	Building core competence
Complex Sys Fundamentals	fundamentals	
Testbed	testing metrics	
Complex Systems Study Group	fundamentals	Building research community

Projects

*"Adding manpower to a late software project makes it later." - Fred Brooks*

ITLDivision Participation in Complex Systems Program

- 890 Program Office
- 891 Mathematical and Computational Sciences
- 892 Advanced Network Technologies
- 894 Information Access
- 897 Software Diagnostics and Conformance Testing
- 898 Statistical Engineering
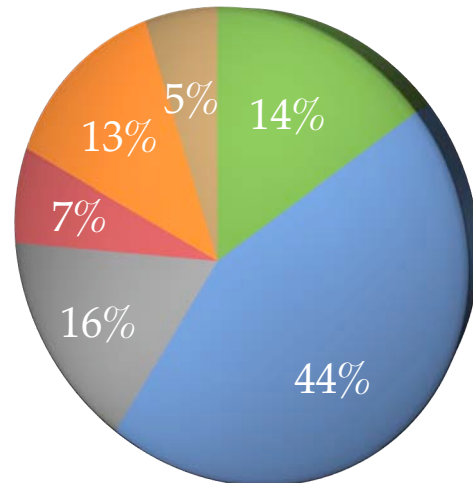
Projects as Percentage of Program

- Program Activities
- IMS Measurement Science
- CxS Fundamentals
- Methods for Characterizing Complex Networks
- Viz of Complex Networks
- CxS Testbed
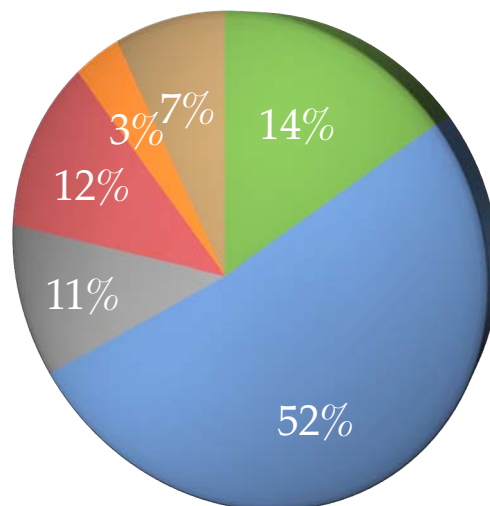
Complex Systems Fundamentals

Roldan Pozo

Standard Reference Data for Complex Network Research

Roldan Pozo

http://math.nist.gov/~RPozo/complex_datasets.html

The study of complex or self-organizing networks is a relatively young, yet active research area. Much of the focus is to determine how graphs from real phenomena (such as protein interactions, social networks, structure of the World Wide Web, etc.) differ from synthetic or engineered graphs that are constructed by a central design. Clearly they are different, but how? Fundamental metrics, such as clustering coefficients, graph diameter, and degree distributions, are commonly cited in the literature, and these have served as encouraging first steps, but further research is needed to yield more meaningful distinctions and truly understand the nature of complex networks.

One important question to ask is: what are the specific real graphs that are being used to drive these theories? Clearly, the conclusions reached can be only as meaningful as the quality and availability of the original data. Many papers provide insufficient information, making comparisons and verification of different approaches difficult. For example, a commonly studied collaboration network in the literature is the Movie Actor graph, where two actors are connected if they have worked together on the same movie. While, this information is usually culled from the Internet Movie Database, there is no single Movie Actor graph because this database is a changing snapshot of the entertainment industry at any given time. Furthermore, what one chooses to include in the search (e.g., foreign films, documentaries, TV movies, specific epochs or genres) yield different graphs. In fact, there are several versions of the Movie Actor graph cited in the literature, so a citation is not meaningful unless it contains a time-stamp and a record of the specific queries used to generate that particular instance. A similar argument

can be made for other association networks, such as co-author or citation graphs.

Other examples include networks based on the structure of the World Wide Web. In this case, we have a directed graph, where nodes represent web pages and two nodes are connected if there a hypertext link from one to the other. Such graphs are typically generated by web crawlers and can vary greatly depending on how they crawl and their policy decisions regarding what constitutes an independent link. For example: Do different anchors to the same page count as separate links? Are aliases counted as separate links? How are non-static pages handled? What file types are processed? How robust is it with broken links and sloppy HTML pages? This list can be quite long, and each decision path will yield a different graph. Aside from this, the Web is, of course, not a static entity. Many of the Web graphs in the literature provide little of these details and usually present only the topological structure (i.e., URLs removed) so it is difficult to confirm or validate them independently.

To address these issues, we have developed and deployed our own set of web crawlers that report not only topological structure, but also record the node URLs for verification and validation, as well as generate a unique checksum to ensure that pages which contain identical material (e.g., aliased, or mirrored pages) are uniquely identified. Furthermore, these web crawls can be fined-tuned to record and characterize data along a precise decision path, thus allowing us to see what differences are generated when we alter the crawling policy.

Similar concerns exist for graphs taken from other application domains. Often the data is incomplete, poorly documented, or is missing metadata crucial to its characterization. There are ongoing debates about whether particular networks are really scale-free or not, or whether the conclusions of some papers are actually

valid. Usually these concerns lie at the heart of how data was collected and analyzed.

To help improve this situation we are also developing standard reference data sets for complex networks. This collection of public and documented real-world graphs will serve as a test bed of verifiable data which researchers can utilize to verify and compare the algorithms and analysis of various approaches in the literature. This ensures that everyone uses the same reference data in their studies, and provides a convenient location where researchers can browse and search for network graphs. Such a focal point allows the research community to contribute and exchange network graphs from various application domains, and utilize a common testbed for development of software analysis tools (e.g., viewers, partitioners, clustering algorithms) that will aid in the further development and research of complex networks.

Furthermore, we have developed a set of software tools for working with mathematical graphs (Ngraph++) and processing and transforming data collections into various formats (e.g., Matrix Market, GraphViz, Matlab) for output and analysis. Ngraph++, a small and portable C++ library, provides a fundamental graph toolkit for computing clustering coefficients, degree distributions for directed/undirected graphs. Although small in size, it presents a straightforward design, and provides intuitive graph operations and complex-network features not commonly found in more complex packages. Together with various Python and scripting tools, and a reference dataset of network graphs, this forms a kernel package for analyzing complex network data in a portable and universal fashion.

Outputs
Software
1. http://math.nist.gov/~RPozo/complex_datasets.html

2. NGraph++: a simplified C++ graph library for analysis of complex networks. Version 1.0.1.

3. Pcrawler: a Python-based web crawler which generates a web graph of hyperlinks, together with time stamps, SHA checksums, and annotated links

so that validation of graph can be independently performed. Version 0.9.

Community Detection in Network Graphs
Roldan Pozo, Clement Rey

Detecting community structure in large graphs is one of the fundamental challenges in network science. This line of research probes at the underlying structure of complex systems and is necessary in understanding how such systems behave over time. A community is loosely defined as a subset of nodes which are highly connected to each other, but less connected to the remaining graph. Identifying such node groups is important in understanding graph evolution, synchronization, and the dynamics of networks. In various application contexts, community detection can identify customers with similar purchasing habits, determine graph layouts for visualization, model the spreading of infectious diseases, and help law-enforcement in identifying crime cells.

Extracting such structure reliably from large-scale networks, is a difficult problem. First, there must be some mathematical metrics for what constitutes a graph community; second, we need an algorithm for finding such communities (related solutions are NP-complete); last, we need to verifying that the computed groups are indeed meaningful.

Various metrics have been proposed and several optimization algorithms introduced in the literature. The fundamental question remains: How well do these approaches work in practice? So far, there has been little work in verifying and validating the results on a wide array of application domains. To aid this study, we are investigating eigenvalue-based methods and fast greedy algorithms which run in $O(n \log^2 n)$ for sparse graphs, while studying their behavior on specific data sets for which we have annotated community information available. Initial results suggest that current state-of-the-art algorithms provide only mixed results and further research is needed to provide truly reliable solutions.

Measurement Science for Complex Information Systems

*Kevin Mills(***proj lead***)*

The Measurement Science for Complex Information Systems project is the core research activity of the Complex Systems Program. The project is pursuing several lines of research via a series of sub-projects. These projects include:

- Simulation Modeling and Analysis Methods for Large Distributed Systems

- Grid Network Simulation using Piece-Wise Homogeneous Markov Chains

- Analysis of a Distributed Protocol for Network Control

- Visualization of Possible Route Probability Distributions

- Fluid-level Modeling of TCP

Simulation Modeling and Analysis Methods for Large Distributed Systems

Kevin Mills Daniel Genin, Christoph Dabrowski, Don Cho Yeon (now at NIH), James Filliben, Vladimir Marbukh, E. Schwartz

As part of the NIST Complex Systems Program, this project investigates and evaluates modeling, experiment design and analysis methods that can be applied to predict and understand macroscopic behavior and variations in user experience that may arise as engineers introduce changes in software components into a large, distributed information system, such as the Internet. The Internet consists of millions (someday billions) of interconnected components that may be changed independently. For example, every time vendors of major operating systems introduce software updates, millions of users download new software modules into computers connected to the Internet. As another example, users may download software to support new functions, such as social networking or distributed gaming. At the current state of the art, system designers lack techniques to predict global behaviors that may arise in the Internet as a result of interactions among existing and altered software components.

Similarly, hardware faults and unexpected usage patterns may occur within the Internet. Engineers have insufficient methods and tools available to forecast global behaviors and resulting effects on individual users. This project aims to improve existing knowledge about a range of methods and tools that could be applied to understand and predict behavior in such complex information systems

We apply techniques often used by scientists at NIST when studying physical systems. First, we propose an
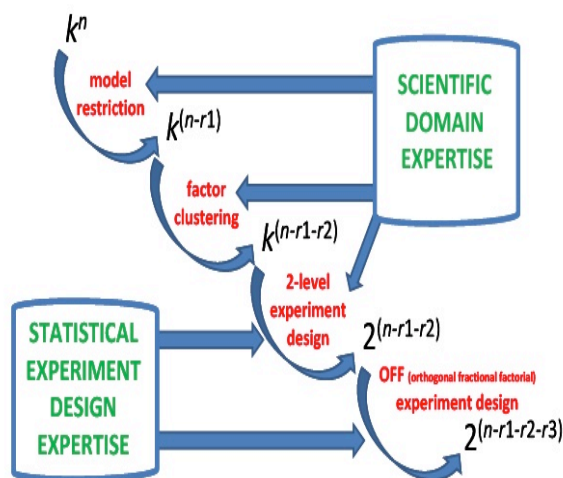
Method to reduce the scale of computational demands from NIST SP500-282

abstract simulation model, representing a system with only a small number of parameters, as compared with the hundreds of parameters typically used in detailed simulators. Second, we adopt 2-level-per-factor experimental designs, which consider each model parameter at only two values, as compared with the billion or so values that each parameter could possibly take on. Third, we leverage orthogonal fractional factorial (OFF) experiment designs that enable us to model a sparse but balanced set of parameter combinations spread widely throughout the space of possible combinations. Reducing the number of parameters, parameter levels and combinations enables feasible simulation of large systems under a wide range of conditions. Third, we use a variety of statistical analysis and visualization techniques designed to explore multidimensional data

sets. Fourth, we use detailed analyses of time series as required to supplement findings from statistical analyses.

To give our project a concrete context, we select challenging problems of current interest and relevance for the information technology community at large. For example, we recently completed a study of the likely consequences for macroscopic behavior and for individual users should any of several proposed mechanisms be introduced to augment or replace congestion control procedures in the standard transmission control protocol (TCP), which is currently deployed to regulate the rate of information transfer among computers connected to the Internet. We demonstrated that our proposed combination of modeling and analysis techniques allows us to predict the influence of seven proposed congestion control mechanisms on macroscopic network behavior and individual user experience. We documented our methods and their application in a NIST Special Publication (SP 500-282): Study of Proposed Internet Congestion Control Mechanisms.

SP 500-282 contributes to current knowledge about modeling and analysis techniques for complex information systems and also contributes to the body of knowledge surrounding proposals for improving congestion control mechanisms considered for deployment in the Internet. Six specific contributions improve current knowledge regarding techniques to understand and predict behavior in complex information systems. First, we summarize the current state of the art in modeling and analysis of communication networks and we identify several hard problems that inhibit the study of large, fast networks. Second, we propose an approach to construct simulation models with a reduced parameter space. Third, we describe and demonstrate how two-level OFF experiment designs can be applied to reduce the number of

parameter combinations that must be considered, while yielding maximum information from available simulation resources. Fourth, we describe and apply a variety of analysis and visualization techniques for interpreting multidimensional data. Fifth, we evaluate our proposed modeling and analysis techniques, discussing the strengths and weaknesses of various methods and identifying those methods that proved most effective for our study. Sixth, we outline future research needed to improve upon the methods we evaluated. Our six contributions enhance understanding of methods and tools available to designers of complex systems.

SP 500-282 also adds four contributions to the body of knowledge surrounding proposals for improving Internet congestion control. First, we characterize likely macroscopic behavior and user performance for seven proposed alternatives to TCP congestion control procedures. Second, we identify key behavioral characteristics to be considered when comparing congestion control mechanisms. Previous studies of congestion control mechanisms did not reflect these key factors. Third, we identify and compare the main differences among the congestion control mechanisms we studied. Fourth, we suggest some future research directions related to Internet congestion control. Our four contributions should help researchers to better understand the problem space surrounding congestion control in the Internet.

To publicize our contributions, we organized a series of speaking engagements throughout the year. We described our study and results to appropriate academic, government and industrial concerns. In particular, we presented our work to the Internet Congestion Control Research Group (ICCRG) of the Internet Research Task Force. The ICCRG is tasked with evaluating proposed congestion control algorithms for the Internet. We first briefed the ICCRG on our work in May 2009. As a result of that briefing, the ICCRG invited us back to brief the group on our findings during the March 2010 meeting of the Internet Engineering Task Force (IETF).

This year we intend to finalize our study of congestion control mechanisms by revising SP 500-282 based on

public comments received. Further, we intend to prepare three or four conference papers focused on particular aspects of our methods and findings. Further, while some of the methods we applied appear quite effective in the context of Internet congestion control, we also need to demonstrate effectiveness in other applications. For this reason, we plan in the upcoming two years to apply our approach and methods to compare resource allocation algorithms proposed for use in cloud computing systems.

Outputs

Draft Report

1. K. Mills, J. Filliben, D. Cho, E. Schwartz and D. Genin. Study of Proposed Internet Congestion Control Mechanisms, NIST Special Publication 500-282, final version scheduled for publication in July 2010.

Papers

1. V. Marbukh, "From Network Microeconomics to Network Infrastructure Emergence", Proceedings of the 1st IEEE International Workshop on Network Science for Communication Networks (NetSciCom 2009), held in conjunction with IEEE Infocom 2009, April 24, 2009 - Rio de Janeiro, Brazil.

2. D. Genin and V. Marbukh, "Toward Understanding of Metastability in Cellular CDMA Networks: Emergence and Implications for Performance." GLOBECOM 2008, New Orleans, Nov. 31 - Dec. 4.

Software

1. DATAPLOT analysis and visualization scripts for Internet congestion control study, contact kmills@nist.gov for availability.

Presentations

1. K. Mills, "Study of Proposed Internet Congestion Control Algorithms", invited presentation to the Internet Congestion Control Research Group (ICCRG) at the 77th meeting of the Internet Engineering Task Force (IETF), March 21-26, 2010.

2. K. Mills and J. Filliben, "Efficient Sensitivity Analysis Method for Mesoscopic Network Models", Complex Systems Study Group, NIST, February 2, 2010.

3. K. Mills, "Study of Proposed Internet Congestion Control Algorithms", seminar sponsored by the Computer Science Department and the C4I Center at George Mason University, Fairfax, Virginia, January 29, 2010.

4. K. Mills, "How to model a TCP/IP network using on 20 parameters", Complex Systems Study Group, NIST, November 17, 2009.

5. K. Mills, "Measurement Science for Complex Information Systems", invited presentation to the Internet Congestion-Control Research Group (ICCRG) of the Internet Research Task Force (IRTF) at Tokyo, Japan, May 20, 2009.

6. K. Mills, "Measurement Science for Complex Information Systems", seminar sponsored by the Computer Science Department and the C4I Center at George Mason University, Fairfax, Virginia, March 27, 2009.

7. K. Mills, "Measurement Science for Complex Information Systems", AOL Network Architecture Group, Dulles, Virginia, March 18, 2009.

8. K. Mills, "Measurement Science for Complex Information Systems", NITRD Large-Scale Networking Working Group, Ballston, Virginia, March 10, 2009.

9. Mills, "Progress Report on Measurement Science for Complex Information Systems", Complex Systems Lecture Series, NIST Information Technology Laboratory, Gaithersburg, Maryland, January 27, 2009.

Grid Network Simulation using Piece-Wise Homogeneous Markov Chains

Christopher Dabrowski and Fern Hunt

The growing use of services provided by cloud and grid computing networks underlines the need for analytical and computational tools for understanding and

predicting the behavior of such complex systems. Our work centers on the analysis of grid networks. Here the challenge is that under decentralized control, the interactions of many service providers and customers could result in emergent behaviors resulting in unpredictable and undesirable outcomes. A straightforward approach is to perform a discrete event simulation to model customers and service providers entering into service level agreements for a variety of computer services. Simulations that seek to accurately reproduce system structure and component behavior are computationally expensive however. Even simulating an entire day takes many hours of CPU time (see [4]). As analysts seek to identify system execution paths and scenarios that lead to system breakdown and other anomalous behavior, there is a critical need for models that can provide rapid scalable simulation of large scale grid systems. One should also be able to quickly perturb these models in a search for such scenarios.

In the previous year, [2,3], we developed a piecewise homogeneous Markov chain model of a grid network system that compared very favorably with discrete event simulations (i.e., large-scale simulation) including several failure scenarios arising from the perturbation of transfer rates in the grid system. Moreover, the Markov chain simulation produced these results in hours while the large-scale simulation required days of processing.

This year, we took another step in our goal. First, we introduced an analytical approximation of the Markov chain that eliminated the need for simulation when computing the task completion rate. Second, we introduced a method for finding failure scenarios in the Markov chain model. The method is based on identifying minimal cut sets in the graph of a Markov chain, thus reducing the need for simulation to a minimum. Both approaches are alternatives to the limited brute force perturbation method used in [2,3] and are up to two orders of magnitude faster than the Markov chain simulation and up to four orders of magnitude faster than large-scale simulation. In a forthcoming paper, (to be available at: http://www.nist.gov/itl/cxs/) we demonstrate that these approaches can be used to quickly identify portions of a Markov chain that are most sensitive to perturbation. In the case of the second method, we also made progress toward applying this method to larger Markov chain problems.

Outputs
Papers

1. C. Dabrowski, K. Morrison, and F. Hunt, "Improving Efficiency of Markov Chain Analysis of Complex Distributed Systems," in preparation.

2. C. Dabrowski and F. Hunt, "Improving Speed and Scalability of Markov Chain Analysis of Complex Distributed Systems using the Spectral Theorem" in preparation.

3. C. Dabrowski and F. Hunt, " Improving Speed and Scalability of Markov Chain Analysis of Complex Distributed Systems using Minimal Cut Set Identification" in preparation.

4. C. Dabrowski and F. Hunt, "Markov Chain Analysis for Large-Scale Grid Systems," NIST Interagency Report 7566.

5. C. Dabrowski and F. Hunt, "Using Markov Chain Analysis to Study Dynamic Behavior in Large-Scale Grid Systems," in Proceedings Seventh Australasian Symposium on Grid Computing and e-Research (Ausgrid2009) , Wellington, New Zealand, CRPIT,99, P. Roe and W. Kelly, Eds. ACS, pp.29-40.

6. K. Mills and C. Dabrowski, "Investigating Global Behavior in Computing Grids," Lecture Notes in Computer Science 4124, pp. 120-136, Springer Verlag.

Analysis of a Distributed Protocol for Network Control

Fern Y. Hunt
Vladimir Marbukh

This year we made progress in studying the properties of a discrete dynamical system that models TCP congestion protocol with multipath routing. Here, traffic is split over several routes according to a probability dis-

tribution that is constrained by an entropy parameter. This parameter measures the robustness or path diversity of the splitting scheme. F. Kelly and others have observed that many protocols currently in use can be viewed as distributed algorithms for solving an optimization problem. In two sample network topologies we show that the corresponding discrete system does tend to a stable equilibrium that is the solution of a network utility optimization problem when the entropy of the route distribution is greater than a critical value. This value is a function of the network link capacities and network topology. For our model we show that there is no stable equilibrium for entropy values less than the critical value. A NIST Internal Report describing this research has been released [1].

The existence of an explicit parameter measuring path diversity allows us to evaluate the trade-off between path robustness and network utility. We looked again at the two sample networks which in both cases have a single source and destination. We computed the the long term average utility as a function of entropy. Starting with the critical value, as the entropy increases, the network utility (which we show depends on the minimum average or mean cost) decreases. Thus, increased robustness leads to decreased utility. This work is also described in the internal report [1] and in a later paper that focuses on the tradeoff between utility and stability [2].

Outputs
Papers
1. F. Hunt and V. Marbukh, "A Model of Joint Congestion Control and Routing Through Random Assignment of Paths," NIST Interagency Report 7590, Feb. 16, 2010.

2. F. Hunt and V. Marbukh, "Measuring the Utility/ Path Diversity Trade-Off in Multipath Protocols," Proceedings of the Fourth International ICST Conference on Performance Evaluation Methodologies and Tools, ValueTools09, Pisa, Italy.

Visualization of Possible Route Probability Distributions

Fern Y. Hunt
Terence Griffin

In order to control congestion and maximize network utility, a variety of routing protocols can be used to allocate network traffic. When multiple paths linking a source and destination are available, but only a single path, e.g., a shortest path or path of minimum cost, is selected, we have the Open Shortest Path First (OSPF) protocol. On the other hand, traffic could be allocated uniformly when the paths have the same cost. A spectrum of path specification strategies lies in between these extremes. We visualize these possibilities in the case of a single source destination pair with 3 possible connecting paths. If β_1, β_2, and β_3 are the fractions (probabilities) of traffic assigned to paths 1, 2, and 3 respectively, then each allocation can be associated with a point in or on the equilateral triangle with an altitude height of 1. Here, β_1, β_2, and β_3 are the lengths of perpendiculars from points on edges of the triangle. The entropy associated with an allocation is a function of β_1, β_2, and β_3, and provides an important quantitative description of its degree of randomness. A point in the triangle is colored according to the value of the entropy of the allocation represented by that point. The surface above the triangle was created by plotting a point above the data point in the triangle where the height is equal to the Entropy. OSPF is associated with $H = 0$, and traffic is allocated according to one of the vertices of the triangle.

This is the least robust allocation. Equal cost multipath allocation, assigned to the center of the triangle has the maximum value of $H = \log(3)$ is the most robust but it has reduced network utility. (See the magenta dot in the center of the triangle). Bands of points with the same value of H (up to accuracy .001) are given the same color. As H decreases, an oval emerges from the center and grows until it touches the sides of the triangle at the maximum entropy for a network with 2 paths (depicted by the magenta oval in the triangle and band in the surface above it). Theoretical studies predict a change in network dynamics for H smaller than this critical value. Our visualization runs immersively and on the desktop. Entropy is displayed as the third dimension and uses an RGB color ramp on entropy. The visualization allows selection of entropy values, and groups of related points. The structure can also be examined with clipping planes of various types, and also by selection of two dimensional or three dimensional representations. The data can be displayed in three ways: points with fatpoint shaders with adjustable point size; polygonal surface; and wireframe surface.
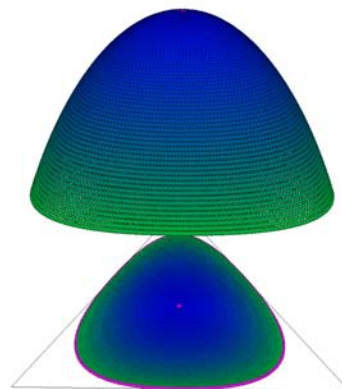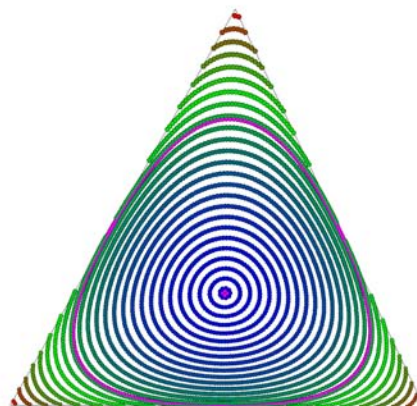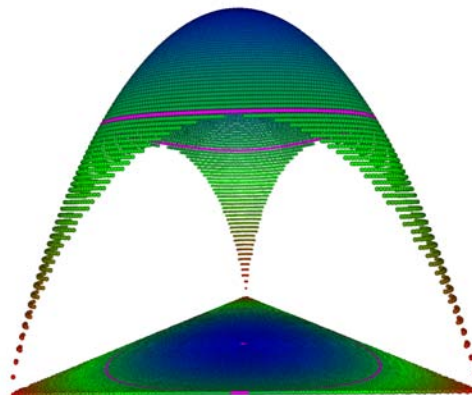
The surface over the triangle defined by the equation: $H(a^1,a^2,a^3) = -a^1*\log(a^1)-a^2\log(a^2)-a^3\log(a^3)$, where the height of a point on the surface above the triangle is the value of H for the point in the triangle representing (a^1,a^2,a^3). H is the entropy function. The top of the surface is the maximum value of the entropy, i.e. log 3 which corresponds to the point (1/3,1/3,1/3) whose image in the triangle is the pink dot in the center. The surface illustrates the changes in the level curves of H (i.e. points in the triangle with a constant entropy value) as H decreases from log 3 to 0. Colors indicate different values of entropy. On the left is the fatpoint shader representation. On the right (top) are two dimensional and polygonal (bottom) representations.

Fluid-level Modeling of TCP

Vladimir Marbukh
Daniel Genin

While modeling of Transmission Control Protocol (TCP) at the packet level yields accurate results, its computational complexity makes packet-level modeling infeasible even for moderate size networks. Flow-level models allow for significant reduction in computational and simulation complexity at the cost of loss in accuracy. Conventional flow-level models [1], which assume that at equilibrium each flow has fixed rate, do not take into account TCP flow burstiness. This simplifying assumption results in underestimation of the packet loss probability and thus overestimation of the TCP throughput which is inversely related to the packet losses. Quantification and mitigation of this inaccuracy is an open issue of significant practical and theoretical importance due to the central role played by TCP in the Internet.

In [3] we assess that conventional fluid models [1] result in unacceptably high underestimation of the numbers of flows in progress and file transfer times in TCP-controlled network under arriving/departing flows. To alleviate this deficiency [3] proposes modeling TCP at flow-level by an on-off fluid process [2]. Model [2], often named the "AMS model" after authors D. Anick, D. Mitra and M. M. Sondhi, accounts for the flow burstiness while retaining the computational tractability. Our extensive ns2 simulations [3] suggest that the AMS flow-level model of TCP may have substantially better accuracy than the conventional fluid model with fixed rate [1], potentially leading to faster yet accurate Internet simulation. More efforts are needed to improve accuracy of the proposed AMS model under high load by accounting for the TCP slow start phase.

References:

1. R. Srikant, Mathematics of Internet Congestion Control, Willey, 2001.

2. M.M. Sondhi D. Anick, D. Mitra. Stochastic theory of a data handling system with multiple sources. ICC'80; International Conference on Communications, Seattle, USA, 1980.

Outputs
Papers

1. D. Genin and V. Marbukh, "Bursty Fluid Approximation of TCP for Modeling Internet Congestion at the Flow Level," Forty Seventh Annual Allerton Conference on Communication, Control, and Computing, Monticello, Illinois, USA, 2009.

2. D. Genin and V. Marbukh, "Do Current Fluid Approximation Models Capture TCP Instability?", submitted to 4th International Conference on Performance Evaluation Methodologies and Tools, Pisa, Italy, Oct. 20-22, 2009.

Methods for Characterizing Complex Networks

*Isabel Beichl (**proj lead**) Brian Cloteaux, David Harris (NSA), Francis Sullivan (IDA Center for Computing Sci.)*

Over the past decade, it has been realized that many of the networks inherent to natural and engineered systems share similar characteristics. These complex networks have been shown in systems such as the power grid, the Internet, communications and transportation networks, and software components. Since these networks are at the heart of our commercial infrastructure, it is important to develop a measurement science for such objects based on sound mathematical methods. Our project looks at both the modeling and measurement of complex networks of these types.

The development of predictive models of network characteristics is an important prerequisite. There have been a number of recent papers proposing possible mechanisms for explaining how complex networks could evolve. Unfortunately, the resulting models have not been able to realistically capture many of the structural characteristics that are observed in real life networks. Thus, we are currently unable to accurately model the time progression of complex networks. As an alternative, we have proposed a method for building random models through a completely structural approach [2].

This structural approach does not try to mimic how the network evolves, but rather it generates random networks with a series of characteristics that we choose as being important to our model. In order for us to produce dynamic models of these complex networks we have the additional constraint that our models must contain a core that reflects the network from an earlier time period. In other words, our method produces models that simulate the evolution of a network over time while maintaining certain critical characteristics, but without having to specify the exact generating mechanism. This approach has shown considerable promise in tests to predict the evolution of the autonomous system topology of the Internet.

A second research direction has been in the measuring of characteristics of these networks. One result has been in the area of computing the reliability polynomial of a network [3]. The reliability polynomial computes the probability that a network stays connected given that each edge in the network has an identical and independent chance of failure. Since it has been shown that this problem is computationally difficult, we have produced a new algorithm for approximating this polynomial. When compared to the best known published algorithm, our method has shown a significant speedup in its computation time.

We have also proposed an enhanced algorithm for computing reliability using a new importance function. A new technique for reducing variance is also being investigated. This technique involves adding a term to the estimator whose expected value is zero.

Outputs
Papers

1. I. Beichl and B. Cloteaux, "Measuring The Effectiveness of the s-Metric to Produce Better Network Models," in Proceedings of the Winter Simulation Conference (2008), Miami, FL, December 7-10, 2008, pp. 1020-1028.

2. I. Beichl and B. Cloteaux, "A Structural Approach to the Temporal Modeling of Networks," in Proceedings of the 2009 Winter Simulation Conference.

3. I. Beichl and B. Cloteaux, "An Approximation Algorithm for the Coefficients of the Reliability Polynomial," Congressus Numerantium, accepted.

Visualization of Large Complex Networks

Roldan Pozo (proj lead), Clement Rey, Terence Griffin

One of the fundamental challenges in complex systems is to understand the nature and structure of large-scale networks. These can be information networks (such as the World Wide Web) biological networks (like protein-protein interactions, food webs) social networks (e.g., online social sites, personal interactions, criminal rings, terrorist cells) and engineered networks. Such networks typically exhibit thousands to millions of nodes and about 10 times as many edges. Given the rapid growth of electronic data, networks with billions of nodes soon will be readily available for analysis, providing greater insight into the behavior of large-scale systems.

To the aid the understanding of such systems, we have developed visualization techniques that utilize the latest immersive and 3D environments, together with state-of-the-art graph layout algorithms, so that researchers can walk around and examine these structures interactively. The visualizations can also be run on independent desktops.
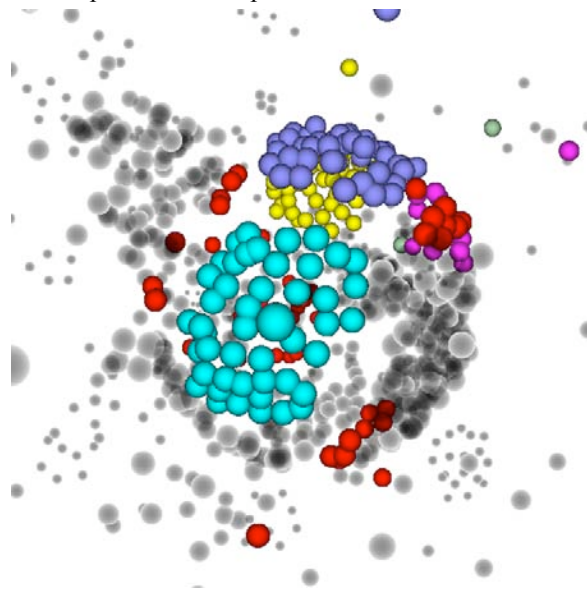
For example, the figure above is a visualization of a web graph, where nodes are individual web pages of George Washington University Computer Science Department (www.cs.gwu.edu). In this instance, we have highlighted the community structure by using a fat-point shader (with an interactive size control) to render the nodes of the graph. The largest groups (7 in this case) are colored individually. The set of small groups (179 in this case) is colored a transparent grey. An individual node can be selected using a cross-hair glyph and its URL displayed in a box tethered to the node with lines. In addition, the web page that the node represents can be brought up, so researchers can investigate its content.

This interaction will include the ability to develop other visualizations on the fly and the ability to conduct varying analyses on the data and display it along with the models. The visualization of the network model will be created, where possible, from a user specified set of descriptions of the model, outputs, and analyses. The visualizations will be created automatically from this file or set of files. This will provide a way to create visualizations from an arbitrary run of the models.

Outputs
Software
1. WebGraphViz (v. 1.0) available from SAVG system at ~rey/webGraphViz_v1.tar.gz

Complex System Testbed

Vincent Stanford(**proj lead**), *Lukas Diduch, Imad Hamchi, Mathieu Hoarau, Antoine Fillinger*

Large Graphs can be searched using an agent-based technique called Ant Colony Optimization (ACO). Our approach based on the NIST Data Flow System II (NDFS-II) uses multiple ant agent populations distributed across processors and clustered computers to solve large-scale problems.

Biological ant colonies have been shown to undergo a phase transition from purely random to social and optimal search patterns. The size of the ant forager populations relative to the size of the search space determines if the ants will collaborate to find efficient solutions (Figure below left) or explore the search space independently and stochastically (Figure below right). Our parallel-distributed processing middleware (NDFS-II) allows us to employ multicore processors, and distributed systems to explore much larger prob-

which periodically send results of local searches to a master node that combines them. Each worker node periodically sends the master node a copy of its local pheromone matrix and the worker nodes do not communicate directly with each other. The mechanism is iterated until the search has converged. The NDFS-II distributed middleware layer manages the worker node creation and dispatches them dynamically to the available processors. The ACO simulation starts with the parameter distribution during which the master node creates a flow and sends the problem graph, evaporation rate, local ant population, and synchronization frequency to the worker nodes. The second phase of the simulation proceeds with the iterative data exchange. After the worker nodes enqueue the local pheromone matrices in the data flows to the master. The master node gathers this information, sums it and
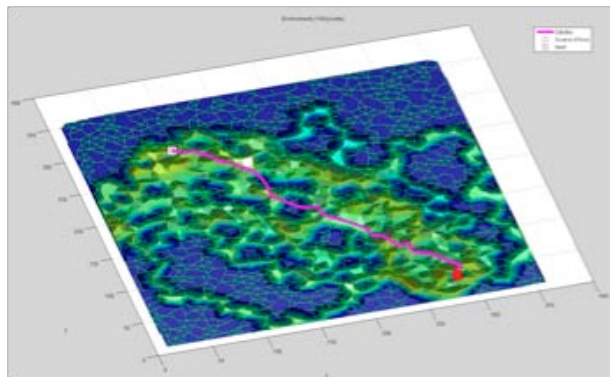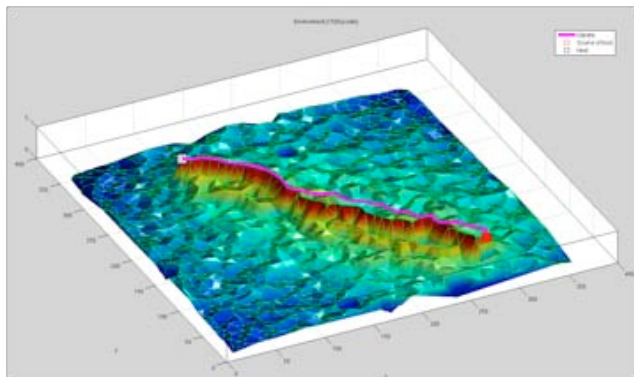
lem spaces using agent-based algorithms like ACO than is feasible with single processor implementations. The parallelization method uses the stygmergic (Stigmergy is a mechanism of indirect coordination between agents or actions.) markings from multiple agent populations running on many processors and host nodes with a periodic rendezvous/update cycle mediated by a master node.

The distributed ACO architecture we developed is based on a model in which multiple copies of the problem graph and associated agent populations are distributed across many hosts and cores on worker processes,

returns it to the worker nodes to update their local search spaces. Thus the distributed ant populations communicate through the stygmergic markings maintained by the master node, and collaborate effectively, even when each subpopulation is below the coherence threshold.

Note, that depending on the problem size, the synchronization frequency must be chosen carefully in order to get an optimal solution in a reasonable time. Master, worker and display nodes are independent Java programs and have a common JNI (Java Native Interface) library to wrap the NDFS C++ API.

To study the phase transition in ACO, we are also using the NDFS II to parallelize our experiments in order to simulate large search spaces that require large population of agents. There are two types of phase transitions: when the ants are able to find a new path and to sustain it, and when there are not enough ants to sustain a path which already exists. We are studying the second one.

For that purpose, we use a Markov model to simulate the behavior of an ant on the graph based on the behavior of real ants (Beekman, Madeline; D. Sumpter and F. Ratnieks. Phase transition between disordered and ordered foraging in Pharaoh's ants. PNAS, volume 98, number 17, pages 9703-9706, August 14, 2001).

All the ant states are modeled: when it leaves the nest, when it gets lost and gets back on the trail, when it reaches the food source and probably the most important feature, when it gets bored from searching and is brought back to the nest.

All the graph characteristics are used in that model: number of points, branching factor, average arc length, average Dijkstra path length.

The characteristic of the search is also used: speed of the ant, quantity of pheromone dropped by an ant, minimum amount of pheromone on each arc (so every path can be chosen by an ant). For our study, we are using a beta skeleton type of graph that allows us to customize the branching factor via that beta parameter. For beta=2, the branching factor is 2.6 and for beta=0.8, the branching factor is 9. We also used a wide range of graph size from 1000 to 50000 nodes. The simulation results versus to model predictions are shown in fig 2.

We find linearity for each beta, which shows consistency in the model and we are working on an expression for ant return rates when lost. We are using a prior probability for that event, which is constant for each beta. We are working on parameter estimation for of the Generalized Extreme Value distribution for return lifetimes. Success will enable another paper in addition to the Parallel Distributed Computing Systems Conference paper we published.
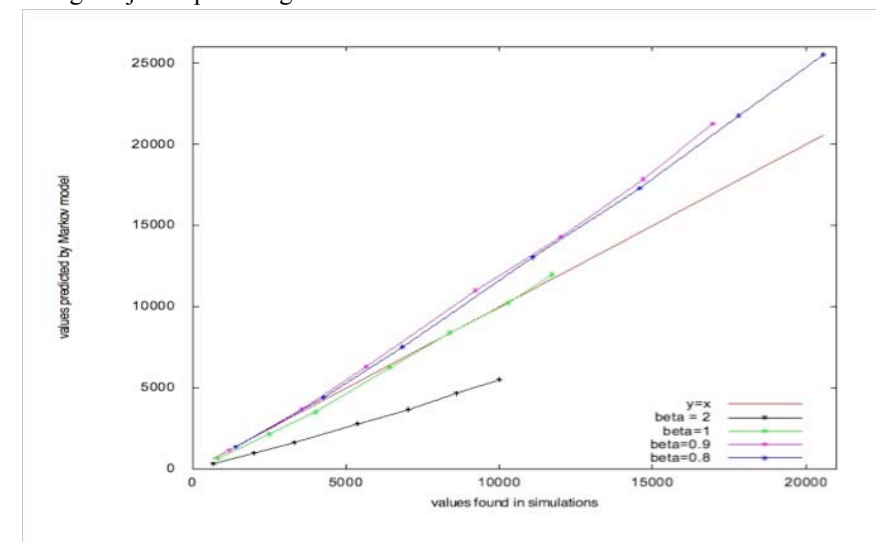
Comparison between the number of ants predicted by the Markov model for the phase transition to occur (prediction) and the number of ants necessary for the phase transition to occur in simulation. For each beta, there are 7 simulations (1000, 5000, 10000, 20000, 30000, 40000 and 50000 points). The perfect fit is y=x (the red line).

Outputs
Papers
I. Hamchi, M. Hoarau ,A. Fillinger, L. Diduch, and V. Stanford. "A Data Flow Implementation of Parallel a Agent-Based Graph Search Algorithm" International Conference on Parallel and Distributed Computing and Systems, Cambridge Mass. November 2-4, 2009.

Future Projects (already underway)

Cloud Computing Measurement
Peter Mell, Chris Dabrowski, Kevin Mills

We have been looking at the problem of deploying customer workloads onto a large resource pooled cloud infrastructure (e.g., Azure) and then migrating those workloads as necessary to optimize both service provider and customer utility in the context of service-level agreements, as demand changes and/or components fail. This appears to us to be an NP-complete problem and so all solutions for large systems would have to be heuristic-based and non-optimal. Thus, we think this might be a ripe area for measurement science research.

We would like to model and simulate large cloud systems and then measure the characteristics of various workload deployment heuristics. This research would include characterizing and evaluating existing approaches and new heuristics that might be suggested by other researchers. We plan to exploit modeling and analysis techniques that we developed recently and applied to study proposed resource allocation mechanisms for the Internet and for computational grids.

Programmatic Activities

"An idea that is developed and put into action is more important than an idea that exists only as an idea."

- Edward de Bono

This chapter describes the activities of the Complex Systems Program office. Primarily these are meant to be a "value add" to the technical conducted by the individual projects. In addition these activities are meant to foster and stimulate a complex systems research community within NIST. All information is available at http://www.itl.nist.gov/ITLPrograms/ComplexSystems/

Major Activities:

• Complex Systems Image of the Month

• Complex Systems Study Group

• Complex Systems Web Site

Complex Systems Image of the Month

complex systems July
IMAGE OF THE MONTH

Imad Hamchi
Mathieu Hoarau
Vincent Stanford

Phase Transition in Cooperative Agent-based Digital Ant Colony Optimization

The figures show a graph search problem and distribution of ant foragers for two populations. One is large enough for the phase transition to collaborative search to occur, and one is too small.

The size of ant forager populations relative to the size of the search space

determines if the ant foragers will collaborate to find efficient solutions (image A), or explore the search space independently and stochastically (image B). Note the high degree of concentration of pheromones on a path near the best path in image A where there is a sufficient forager population for

the phase transition to collaborative foraging. Note also the more diffuse pheromone concentration in image B.

This is an instance of Complex System Phase transitions, or emergent behavior in a system of many simple interacting elements.

More information available at: http://www.itl.nist.gov/ITLPrograms/ComplexSystems/

complex systems May
IMAGE OF THE MONTH

James Filliben
Kevin Mills

Not so FAST

Each of the seven graphs on the left depicts a time series for congestion-window (CW) size on a simulated long-lived flow traversing the length of a highly congested network extending across the United States. Each graph depicts the evolution of the CW when one of seven congestion-control algorithms (CCA) (identified in red) is used within the network. The graphs show the behavior of each algorithm under the same time period and conditions. The x-axes cover 100 simulated seconds, showing 500 measurements taken at intervals of 200 milliseconds. The y-axes show the size of the CW at each measurement. Notice that the CW under the FAST algorithm oscillates more frequently over a larger range of amplitudes than the other CCA. These graphs reveal the cause related to some general findings about the behavior of FAST under spatiotemporal congestion

More information available at: http://www.itl.nist.gov/ITLPrograms/ComplexSystems/

FAST is one of several proposed replacements for the standard TCP congestion-avoidance algorithm. The general aim of such replacement algorithms is to increase the ability of TCP to exploit higher transmission speeds becoming available within the Internet. Using simulation, under congested conditions FAST was found to exhibit a higher retransmission rate (including connection-request

packets) that reduced user throughput on flows transiting congested areas and that increased the number of flows pending in the connecting state. As demonstrated above, this behavior appears to arise from rapid oscillations in congestion-window size when FAST flows transit routers with insufficient buffers to accommodate the flow volume. Practical implications of this behavior include: (1) flows take

longer to connect; (2) flows take longer to complete; (3) user throughput lower for flows transiting congested areas; (4) fewer flows complete. For example, over a 25-minute period FAST completed from 10^5 to 10^7 fewer flows (depending on conditions) than other congestion-control algorithms.

complex systems January
IMAGE OF THE MONTH

Christopher Dabrowski
Fern Hunt

Using Markov Chain Analysis to Study Dynamic Behavior in Large-Scale Grid Systems

State model of grid compute economy from large-scale simulation.

A piece-wise homogeneous Discrete Time Markov chain can be used to model performance of a large-scale computing grid and predict how performance changes in alternative scenarios. Markov chain simulation provides a rapid, scalable representation of system dynamics and can be generated at substantially lower computing cost than detailed simulation models.

Increasing the probability for this transition results in fewer tasks completing, they spend more time negotiating (as illustrated below).

Graph below, illustrates that Markov chain based simulation is virtually identical in accuracy to the large-scale simulation. It runs however, two orders of magnitude faster.

More information available at: http://www.itl.nist.gov/ITLPrograms/ComplexSystems/Papers.html

A Markov chain model can be perturbed by altering selected transition probabilities to predict the effect of changed behaviors, failures, and other events on overall system performance. An example of perturbation analysis is shown in second graph above.

The variations illustrate individual simulation runs testing different probability transitions of network states. The variations illustrate results of individual Markov chain simulation runs that test different values for probabilities of transition between states of grid system tasks. Testing this large number of variations

is impractical using the large-scale network simulation (shown via the red triangle portion of the graph).

complex systems November
IMAGE OF THE MONTH

Brian Cloteaux
Isabel Beichl

A Structural Approach to the Temporal Modeling of Networks

This figure shows a temporal model of a subset of the autonomous system (AS) topology of the Internet. The red edges denote the original "core" graph from a November 2007 instance of the AS topology. The black edges are the additional connections added to the network model to represent a future time period.

More information available at: http://www.itl.nist.gov/ITLPrograms/ComplexSystems/

Simulation of many real world systems such as the Internet and social networks requires developing dynamic models for the underlying networks in these systems. Currently, there is a large body of work

devoted towards determining the underlying mechanisms that create these networks, but the resulting models have not realistically captured many of the important structural characteristics when compared with real

world examples. Towards creating more realistic dynamic models, we propose a method of structurally constructing models of an evolving network.

We continue to produce monthly images which highlight the work of the program. Each month the program office produces a one page flyer to highlight the work of one particular project. The illustrations and text are intended to convey technical information to the layman and are not meant to be overly technical. To date we have produced over 25 which are available at: **http://www.nist.gov/itl/cxs/cxsiom.cfm**

Complex Systems Study Group

An approximately bi-weekly series of meetings, open to all, covering topics of interest and relevance to the Complex Systems research community.

Web Site
A standard system for all ITL programs has been implemented and the figure below is representative of the home

page for the Complex Systems Program.

Milestones and Plans

"Plans are nothing; planning is everything." -Dwight D. Eisenhower

Roadmap

Designed as a five year program we are following a development roadmap.

Five Year Plan/Roadmap

	YR1	YR2	YR3	YR4	YR5
infrastructure	Initiate Program background R&D / infrastructure, study group; TWIKI; mailing list	Establish a CxS Testbed for Net/IT/SOA apps			
fundamentals metrics		Approach to establishing metrics	Metrics from Application(s)	Publish CxS baseline Metrics	
applications	Select initial exploratory projects (small)	Medium scale project (IMS sized) ...will require additional 1M funding		Integration with new application areas (Bio/Nano); national priorities w/core ITL	
outreach	web / podcast	lec series / art show			
eval/impact					Evaluate program impact

NOW

Deliverables Data

"Data is not information, information is not knowledge, knowledge is not understanding, understanding is not wisdom." - Clifford Stoll

Papers

2010

1. K. Mills, J. Filliben, D. Cho, E. Schwartz and D. Genin. Study of Proposed Internet Congestion Control Mechanisms, NIST Special Publication 500-282, DRAFT REPORT final version scheduled for publication in July 2010.

2. C. Dabrowski, K. Morrison, and F. Hunt, "Improving Efficiency of Markov Chain Analysis of Complex Distributed Systems," in preparation.

3. C. Dabrowski and F. Hunt, "Improving Speed and Scalability of Markov Chain Analysis of Complex Distributed Systems using the Spectral Theorem" in preparation.

4. C. Dabrowski and F. Hunt, " Improving Speed and Scalability of Markov Chain Analysis of Complex Distributed Systems using Minimal Cut Set Identification" in preparation.

2009

1. I. Beichl and B. Cloteaux. A Structural Approach To The Temporal Modeling Of Networks. Winter Simulation Conference, Austin, Texas, 2009, 13-16 December.

2. B. Cloteaux and N. Serova. Matching Observed Alpha Helix Lengths to Predicted Secondary Structure. Computational Structural Bioinformatics Workshop, Washington D.C., 2009, 1-4 November.

3. S. Ressler, "Complex Systems Program Activities Summary, Fiscal Years 2007-2008", NISTIR 7569, 4/07/09.

4. F. Hunt, V. Marbukh, "A Model of Joint Congestion Control and Routing Through Random Assignment of Paths", NISTIR 7590, 2009.

5. C. Dabrowski, Reliability in Grid Computing Systems, Concurrency and Computation: Practice and Experience. Volume 21, Issue 8, pp. 927-959.

6. V. Marbukh, "From Network Microeconomics to Network Infrastructure Emergence", First IEEE International Workshop on Network Science for Communication Networks (NetSciCom'09 April 24, 2009 - Rio de Janeiro, Brazil..

7. Antoine Fillinger, Imad Hamchi, Stéphane Degré, Lukas Diduch, Travis Rose, Jonathan Fiscus and Vincent Stanford. Middleware and Metrology for the Pervasive Future. IEEE Pervasive Computing Mobile and Ubiquitous Systems. Vol. 8, num. 3, page 74-83, July-September 2009.

8. Antoine Fillinger, Lukas Diduch, Imad Hamchi, Mathieu Hoarau, Nicolas Crouzier and Vincent Stanford. Pervasive control of distributed sensor systems - The NIST Data Flow System II. Accepted for publication at IEEE International Conference on Pervasive Computing and Communications (PerCom), Galveston, Texas, 2009. Forced withdrawal due to denial of travel.

9. I.Hamchi, A.Fillinger, M.Hoarau, L.Diduch, and Vincent Stanford, A Data Flow Implementation of Agent-Based Distributed Graph Search, IASTED on Parallel and Distributed Computing and Systems (PDCS Nov, 2009)

10. C. Dabrowski and F. Hunt, Markov Chain Analysis for Large-Scale Grid Systems. NISTIR 7566, April 2009.

11. C. Dabrowski and F. Hunt, Using Markov Chain Analysis to Study Dynamic Behavior in Large-Scale Grid Systems, in 7th Australasian Symposium on Grid Computing and e-Research, Wellington, New Zealand, January 21, 2009.

12. B. Cloteaux "Approximating the number of bases for almost all matroids", submitted to Information Processing Letters.

13. F. Hunt & V. Marbukh "Measuring the Utility/Path Diversity Tradeoff in Multipath Protocols", in proceedings of the 4th International Conference on Performance Evaluation Methodologies and Tools, ValueTools09, Pisa Italy Oct. 20-22. 2009

14. D. Genin and V. Marbukh, "Bursty Fluid Approximation of TCP for Modeling Internet Congestion at the Flow Level", D. Genin and V. Marbukh 47th Annual Allerton Conference on Communication, Control, and Computing. University of Illinois at Urbana Sep. 29-Oct. 2, 2009

15. D. Genin and V. Marbukh, "Do Current Fluid Approximation Models Capture TCP Instability?", submitted to 4th International Conference on Performance Evaluation Methodologies and Tools, Pisa, Italy, Oct. 20-22, 2009.

2008

1. F. Hunt and V. Marbukh, "Dynamic Routing and Congestion Control Through Random Assignment of Routes", Proceedings of the 5th International Conference on Cybernetics and Information Technologies, Systems and Applications: CITSA 2008, Orlando FL, July 2008. (BEST PAPER)

2. I. Beichl and B. Cloteaux, Generating Network Models Using the S-Metric, in Proceedings of the International Conference on Modeling, Simulation and Visualization Methods, Las Vegas, NV, July 2008, pp. 159-164.

3. I. Beichl and B. Cloteaux, Measuring the Effectiveness of the s-Metric to Produce Better Network Models, in Proceedings of the Winter Simulation Conference, Miami FL, Dec 2008, pp. 1020-1028.

4. K. Mills and C. Dabrowski, Can Economics-based Resource Allocation Prove Effective in a Computation Marketplace? in Journal of Grid Computing Special Issue on Grid Economics and Business Models. DOI (10.1007/s10723-007-9094-4)

5. D. Genin and V. Marbukh, Toward Understanding of Metastability in Large-Scale Loss Networks with Mobile Users: Emergence and Implications for Performance, Proceedings of 12th World Multiconference on Systems, Orlando FL, June 2008.

6. V. Marbukh and K. Mills, Demand Pricing & Resource Allocation in Marketbased Compute Grids: A Model and Initial Results in Proceedings of ICN 2008, April 2008.

7. C. Dabrowski, Reliability in Grid Computing Systems. submitted to the journal Concurrency and Computation: Practice and Experience as part of a special issue on OGF work.

8. V. Marbukh, Can TCP Metastability Explain Cascading Failures and Justify Flow Admission Control in the Internet? in Proceedings of the 15th International Conference on Telecommunications (ICT'08).

9. I. Fillinger, S. Hamchi, L. Degré, L. Diduch, T. Rose, J. Fiscus and V. Stanford. Engineering the Pervasive Future: Common Middleware, Research Corpora, and Metrology. IEEE Pervasive Computing Mobile and Ubiquitous Systems. Accepted for Publication 2008.

10. L. Diduch, M. Hoarau, A. Fillinger, V. Stanford. Synchronization of Data Streams in Distributed Realtime Multimodal Signal Processing Environments on Commodity Hardware. Proceedings of the IEEE International Conference on Multimedia & Expo (ICME), Hannover, Germany, June 2008.

11. A. Fillinger, L. Diduch, I. Hamchi, M. Hoarau, S. Degré, V. Stanford. The NIST Data Flow System II: A Standardized Interface for Distributed Multimedia Applications. IEEE International Symposium on a World of Wireless, Mobile and MultiMedia Networks (WoWMoM), Newport Beach, California, June 2008 - **Awarded best demonstration**.

2007

1. V. Marbukh, Towards Understanding of Complex Communication Networks: Performance, Phase Transitions & Control to appear in a special issue of Sigmetrics "Performance Evaluation Review" DRAFT version MAMA07, June 13.

2. V. Marbukh and K. Mills, On Maximizing Provider Revenue in Market-based Compute Grids in Proceedings of ICNS07, June 2007.

3. K. Mills and C. Dabrowski, Investigating Global Behavior in Computing Grids, Self-Organizing Systems, Lecture Notes in Computer Science, Volume 4124 ISBN 978-3-540-37658-3, pp. 120-136, Oct 2007.

4. K. Mills, A Brief Survey of Self-Organization in Wireless Sensor Networks, Wireless Communications and Mobile Computing, Wiley Interscience, Vol. 7, No. 7, (pages 823-834) October 2007.

5. V. Marbukh, S. Klink, Decentralized control of large-scale networks as a game with local interactions: cross-layer TCP/IP optimization, ValueTools, 2nd International Conference on Performance Evaluation Methodologies and Tools, Nance France, Oct 2007.

6. V. Marbukh, Utility Maximization for Resolving Throughput/Reliability Trade-offs in an Unreliable Network with Multipath Routing, ValueTools, 2nd International Conference on Performance Evaluation Methodologies and Tools, Nance France, Oct 2007.

7. V. Marbukh, Metastability of fair bandwidth sharing under fluctuating demand and necessity of flow admission control in Electronics Letters, 13th September 2007, Vol 43, No 19.

8. V. Marbukh, Fair bandwidth sharing under flow arrivals/departures:effect of retransmissions on stability and performance, in ACM SIGMETRICS Performance Evaluation Review, Volume 35 Issue 2, September 2007.

9. C. Dabrowski, Investigating Resource Allocation in a Standards-Based Grid Compute Economy. National Institute of Standards and Technology: Gaithersburg, MD. Interagency Report 7463, November 2007.

10. L. Diduch R. Mueller, G. Rigoll. A framework for modular signal processing systems with high-performance requirements. Proceedings of the IEEE International Conference on Multimedia & Expo (ICME), Beijing, China, July 2007.

11. A. Fillinger, L. Diduch, I. Hamchi, S. Degré and V. Stanford. NIST smart data flow system II: speaker localization. Proceedings of the 6th international conference on Information processing in sensor networks (IPSN), Cambridge, Massachusetts, USA, 2007.

Presentations
1. K. Mills, "Study of Proposed Internet Congestion Control Algorithms", invited presentation to the Internet Congestion Control Research Group (ICCRG) at the 77th meeting of the Internet Engineering Task Force (IETF), March 21-26, 2010.

2. K. Mills & J. Filliben, "Efficient Sensitivity Analysis Method for Mesoscopic Network Models", Complex Systems Study Group, NIST, February 2, 2010.

3. K. Mills, "Study of Proposed Internet Congestion Control Algorithms", seminar sponsored by the Computer Science Department and the C4I Center at George Mason University, Fairfax, Virginia, January 29, 2010.

4. K. Mills, "How to model a TCP/IP network using on 20 parameters", Complex Systems Study Group, NIST, November 17, 2009.

5. F. Hunt, "The Dynamics of a Model of a Computer Protocol," Mathematical Problems in Industry Workshop 2009, University of Delaware, Newark, DE, June 15-19, 2009.

6. V. Marbukh, "From Network Microeconomics to Network Infrastructure Emergence," First IEEE International Workshop on Network Science for Communication Networks (NetSciCom'09), Rio de Janeiro, Brazil, April 24, 2009.

7. B. Cloteaux, "Measurements on Networks," Computer Science Colloquium, New Mexico State University, Las Cruces, NM, April 15, 2009.

8. B. Cloteaux, "An Approximation Algorithm For The Coefficients Of The Reliability Polynomial," Southeastern International Conference on Combinatorics, Graph Theory, and Computing, Boca Raton, FL, March 5, 2009.

9. B. Cloteaux, "Measuring The Effectiveness of the s-Metric to Produce Better Network Models," Winter Simulation Conference, Miami, FL, December 9, 2008.

10. R. Pozo, review of paper Statistical Mechanics of Complex Networks by Albert & Barabasi to Complex Systems Study Group, Feb 9, 2007 -

11. V. Stanford, Time Series Prediction Forecasting the Future and Understanding the past. Santa Fe Institute Proceedings on the Studies in the Sciences of Complexity. NIST Complex System Program, Perspectives on Standard Benchmark Data, In Quantifying Complex Systems, internal presentation to CxS Study Group. August 31, 2007.

12. V. Marbukh, Towards Price Based Network Management and Provisioning , May 6, 2008 to Comples Systems Study Group

13. J. Filliben, Sensitivity Analysis Methodology for a Complex System Computational Model, 39th Symposium on the Interface: Computing Science and Statistics, Philadelphia, PA, May 26, 2007

14. D. Genin, Percolation: Theory and Applications, internal presentation to Complex Systems Study Group, Oct 17, 2007.

15. F. Hunt, Mathematical Modeling of TCP/IP protocols , May 17, 2008 to Complex Systems Study Group.

16. D. Genin, Metastability and networks , Feb 26, 2008 to Complex Systems Study Group

Software

1. Flexi-Cluster: A Simulator for a Single Computer Cluster, V. Marbukh and K. Mills.

2. MesoNet: A Mesoscopic Simulation Model of a Router-Level Internet-like Network, K. Mills

3. MesoNetHS adds six congestion-control algorithms for networks with high bandwidth-delay products; algorithms include: binary-increase control (BIC), compound TCP (CTCP), fast AQM scalable (FAST) TCP, H-TCP, high-speed TCP (HS TCP) and Scalable TCP (STCP), K. Mills, E. Schwartz and J. Yuan.

4. EconoGrid: A detailed Simulation Model of a Standards-based Grid Compute Economy, C. Dabrowski and K. Mills.

5. MesoGrid: a mesoscopic scale simulation model of a computational grid economy, C. Dabrowski and K. Mills.

6. NGraph++: a simplified graph library for the analysis of complex networks based on ANSI/ISO C++. Includes basic graph operations (construction, intersections, subgraphs) and algorithms for computing degree distribution and clustering coefficients. R. Pozo

7. Pcrawler: a Python-based web crawler which generates a web graph of hyperlinks, together with time stamps, SHA checksums, and annotated links so that validation of graph can be independently performed. Version 0.9 – R. Pozo. 1.0.1. – R. Pozo.

8. DiVisa: a multi-dimensional visualization tool. Can read any kind of data (simulation, statistics, text or numeric, etc.), and converters have been implemented to read several data formats without need for reformatting.

9. The NIST Data Flow System II, a generic cross-platform middleware for parallel process and distributed computing.

10. Distributed processing for multiple instances of Octave or Matlab computational engines for parallel matrix algebra needed in complex system simulations. Provided with the NDFS-II.

11. Distributed simulation architectures for the study of emergent behaviors in large populations of simple agents that form complex systems. The distributed processing was shown to support very large agent population capable of solving large combinatorial optimization problems. Provided with the NDFS-II.

12. The ant colony Engine that simulates the behavior that biological ant colonies show in nature when foraging for food and finding efficient paths from the nest to the food sources. Provided with the NDFS-II.

13. B. Cloteaux , A C++ implementation of a sequential importance sampling algorithm, created by us, for estimating the number of spanning trees in a graph, Sept. 2007.

14. B. Cloteaux , A C++ implementation of various network metrics including minimum vertex cover (MC). Uses several of the kernelization techniques of Langston et al. The kernelization method involves taking a graph G and producing a smaller graph G', the kernel, with a value k such that MC(G) = MC(G')+k. For the smaller G' we solve for minimum vertex cover exactly, Feb. 2008.

15. B. Cloteaux , A program for generating random graphs with a given degree sequence. Took code written in R by Joseph Blitzstein at Harvard, converting it to C++, and then optimizing it with the use of novel datastructures. These datastructures allowed the C++ program to run much larger cases than the original program. In particular, it could then handle the Autonomous Systems data from the UCLA database. Software has been released to J. Blitzstein (Harvard), C. Priebe (Johns Hopkins), J. Devinney (IDA Center for Computing Sciences), March 2008.

16. B. Cloteaux , Software to generate graphs with specified s values. These used the Monte Carlo technique of threshold acceptance, May 2008.

17. B. Cloteaux , Software to calculate the maximum s value for a set of graphs. This includes a new approximation technique we developed, namely a deterministic version of Blitzstein's algorithm to produce the upper bound, May 2008.

18. B. Cloteaux , Software to enumerate all graphs with a given degree sequence.Done for illustration purposes for small graphs, May 2008.

19. B. Cloteaux , C++ implementations of Wilson's algorithm to uniformly sample spanning trees in graphs and Welsh's Monte Carlo algorithm for counting bases of a frequent matroid, Aug. 2008.

20. B. Cloteaux , Software implementing our algorithm, based on sequential importance sampling, to approximate the coefficients of the reliability polynomial of a graph, Sept 2008.

21. B. Cloteaux , A C++ implementation of Colbourn's algorithm to estimate the coefficients of the reliability poly-nomial, Sept 2008.

22. B. Cloteaux , Software to estimate specific values of the reliability polynomial. Uses Chernoff bounds to deter-mine convergence. It produces a graph of connected probability vs edge probability, Sept 2008.

Visualizations
available at: http://www.itl.nist.gov/ITLPrograms/ComplexSystems/Visual.html
1. Animation of simulation of Abilene network. C. Houard, J. Hagedorn

2. Animations of firefly and pacemaker experiments on NDFS-II testbed L. Diduch

3. Distributed Ant Colony Optimization Simulation is a new method of parallelization based on the NDFS-II mid-dleware. I. Hamchi