Disease Surveillance

- ARAVIND SRINIVASAN
- RITA COLWELL

A generation ago, with smallpox apparently eradicated, many people considered infectious diseases a receding threat, but with the rise of new diseases, drug-resistant strains, and the resurgence and persistence of many old scourges, infectious diseases remain a problem around the world. Researchers are more invested than ever in tracking the spread of viruses and bacteria and in understanding their natural histories.

Aravind Srinivasan (l), Rita Colwell (r)

Aravind Srinivasan, who works on network structures, and Rita Colwell, who has long studied cholera, are two UMIACS researchers whose insights are helping develop smarter ways to detect and combat infectious agents.

Srinivasan studies social networks. He collaborates with researchers in the National Institutes of Health-funded Models of Infectious Disease Agent Study, in which researchers around the country are working together to model the spread of existing and emerging infectious diseases. "My contributions are algorithmic, mathematical, and understanding network structures," Srinivasan says.

Disease Surveillance

- ARAVIND SRINIVASAN
- RITA COLWELL

To contact any researcher in UMIACS, go to **www.umiacs.umd.edu.**

"He is a world-class expert in probabilistic methods," says Madhav Marathe, a network dynamics specialist at the Virginia Bioinformatics Institute at Virginia Tech. "He's a great problem solver." To make models of disease spread more realistic, Srinivasan has worked to factor in the variation in individual susceptibility depending on factors like health, age, and a person's role in a group. Srinivsan has developed mixing models that take variations among people into account.

"There is vast heterogeneity in populations," he says. "For example, in Portland, Oregon, a bus driver will have a very different contact structure than, say, a school teacher." In the face of a potential epidemic, understanding how bus drivers differ from school teachers could be vital.

Along with collaborators, Srinivasan has used Portland, Oregon as a test case. With census data, land use information, and detailed transportation data, they simulated social interactions and the spread of pathogens. Stephen Eubank, also a collaborator from the Virginia Bioinformatics Institute, says Srinivasan helped show that one effective way to monitor disease spread is by focusing on its spread across locations rather than just among people. Insights from such studies can lead to practical guidance, such as how to distribute vaccines to prevent an epidemic or where to put anthrax sensors in a large building.

"The study of large networks is not confined to very specific applications but to any interconnected system," says Srinivasan. Companies such as Microsoft are interested in better understanding social networks in sites such as Facebook or MySpace, in part for advertising. "Our work has epidemiology primarily in mind, but there are other applications, too," Srinivasan notes.

Meanwhile, microbiologist Rita Colwell, who is widely known for serving as director of the National Science Foundation from 1998 to 2004, has worked for three decades on understanding the cholera pathogen. Before her work, cholera bacteria were thought to be spread directly among people. She showed that the pathogen is attached to plankton and transmitted in drinking water. Colwell is very much interested in understanding the interaction between humans and the environment.

"Dr. Colwell is a leader in microbial ecology," says G. Balakrish Nair, the director of the National Institute of Cholera and Enteric Diseases in Calcutta. "Further, her work has contributed to an understanding of the relationship of global climate and human health."

Colwell is collaborating with University of Maryland oceanographers Raghu Murtugudde and Tony Busalacchi to study how warmer temperatures around the world are influencing infectious disease outbreaks and to monitor pathogen populations. With satellite technology, the researchers have shown that increased temperatures promote zooplankton growth and have correlated with cholera epidemics. "We're trying to develop a predictive mechanism," says Colwell, based on satellite data. By monitoring zooplankton populations, one could promote public health measures for cholera when necessary, such as alerting people to boil or filter water. Colwell has shown that simply filtering water through folded sari cloth can effectively remove the cholera pathogen because it is attached to relatively large zooplankton.

Since the early 1980s, Colwell has promoted the use of molecular genetic technology in studying microorganisms. "I'm very excited about a project we have under way to sequence the genomes of 50 *Vibrio* and related strains, including 38 strains of *Vibrio cholerae,"* she says. Microbiologists are looking for differences among the strains that may, among other things, relate to geography.

With her studies on cholera extending from the molecular scale to the global, Colwell says this is "a view of biology that I call biocomplexity—it's a holistic view of health and safety." Geography, seasons and climate all influence the transmission of cholera, and other diseases, such as malaria, flu and dengue fever. Colwell was awarded the 2006 National Medal of Science for her work on the microbial ecology of cholera.

"The biology of this planet has to include the human as but one of the billions of species that comprise the complex web of interactions," says Colwell. "We can better understand health and disease if we understand these interactions."

— Profile written by Karin Jegalian

collaborate : create

Static and Dynamic Analysis of Code

- WILLIAM PUGH
- JEFF HOLLINGSWORTH

Much of modern life relies on software to work as expected—reliably and safely. UMIACS researchers Bill Pugh and Jeff Hollingsworth have developed methods that help programmers detect errors in code. Pugh is the author of a widely used program that searches for common mistakes in Java code. Hollingsworth has developed tools for analyzing software as it runs that are widely used in supercomputers as well as small, handheld devices.

William Pugh (l), Jeff Hollingsworth (r)

Bill Pugh's FindBugs program has been downloaded more than half a million times in the four years since he introduced it. The project started, Pugh says, when he realized that it was easier than expected to find software developers' mistakes. He started to categorize the kinds of errors programmers make in Java code and then developed ways to find the common mistakes. By late 2007, FindBugs could detect 289 different bugs.

A mistake could prevent a program from completing its task, cause a system crash, or allow security breaches. FindBugs has proven popular because it is easy to run, free, and effective in finding mistakes that programmers

Static and Dynamic Analysis of Code

■ WILLIAM PUGH
■ JEFF HOLLINGSWORTH

To contact any researcher in UMIACS, go to **www.umiacs.umd.edu**.

care about. Google now runs FindBugs automatically to help developers write code. "Bill helped us set up our process to take full advantage of FindBugs," says David Morgenthaler of Google. "Some academics don't want to get hands-on, but Bill really goes in depth."

Fortify Software, a company that makes security tools, ships FindBugs as part of its product. "What sets FindBugs apart is that it reports the kind of bugs that programmers want to see—the kinds of problems that will make software misbehave," says Brian Chess, chief scientist at Fortify Software. "Bill does a great service, really, to the Java community."

Pugh says it has been interesting to get feedback from professional programmers on what tools they need to help them design code. Through FindBugs, he gets access to a large pool of the actual errors that beset programmers. In addition to Google and Fortify Software, prominent companies including eBay, Sun Microsystems and financial firms have used FindBugs.

"We've started with real bugs. When you start with real bugs, you see that there are mistakes that nobody had been thinking to look for," says Pugh, "and it turns out to be easy to find them."

"Anyone with an application on a network has to be concerned about what happens when something unexpected happens," he adds. To help develop standardized ways of evaluating the security of software, he is serving on the technical advisory board of a Department of Homeland Security project, run with the National Institute of Standards and Technology, to develop standards for establishing confidence in software.

While FindBugs is a static analysis program that looks at source code written by programmers, Jeff Hollingsworth has developed a dynamic analysis program called Dyninst that examines the binary code that computers actually execute. Dyninst can analyze and gauge the trustworthiness of new software—whether commercial software or malicious software in disguise—to see that it does what it's supposed to do. Hollingsworth calls Dyninst a tool kit for analyzing binary code, simulating how it may work in different systems, and modifying the code when necessary. Code can be edited as it's executed. "Principally, it's a program for tool builders," says Hollingsworth. Its chief strength is in its efficiency.

Hollingsworth compares what Dyninst does to a better way of testing a car for crashworthiness. Instead of destroying a car and then analyzing how it crumpled, what if one could use observations during a crash to change how the collision proceeded? "You can use information and knowledge you've gleaned from an execution to refine your understanding and change what you're measuring during that same execution," says Hollingsworth.

Dyninst can identify performance problems as well as help assess the safety of software. Companies including IBM and SGI have used Dyninst software to manage programs that run on the supercomputers they sell.

Hollingsworth originally started designing Dyninst in the mid-1990s for large parallel processors; to pinpoint problems in such large systems, the program had to be efficient. "Efficiency is important for a performance tool," comments Martin Schulz of the Lawrence Livermore National Laboratory, "You don't want the tool to disrupt the application." He notes that Dyninst can be particularly useful in security applications where a computer never sees source code but only binary code.

"One of the unique things about Dyninst is that it works on a variety of systems and computer architectures," says Hollingsworth. "It has been used on everything from palm pilots to the world's largest parallel processors." Schulz adds, "It's important to have software that is more than just an academic prototype. Jeff and his group really go beyond that and deliver software that we can actually use."

— Profile written by Karin Jegalian

create : collaborate

Mapping the Internet

▪ NEIL SPRING

The Internet is the ultimate black box to many—something that we use all the time with little understanding of how exactly it's put together. Unlike, say, an appliance, the Internet is also vast and seemingly amorphous. As it turns out, not even the experts have a definitive sense of how the whole thing is wired together or how many component parts it has. Neil Spring is trying to fill in this gap and build a map of the Internet.

"We rely on this network everyday. Its operation relies on cooperation both between competing telecom companies and across different countries," Spring notes. Somehow competing companies with little knowledge of each other are able to put together a network.

Spring, who joined UMIACS in 2004, began to work on the problem of mapping the Internet as a graduate student at the University of Washington. "Neil knows what he's doing," says Walter Willinger, a researcher at AT&T who has collaborated with Spring. "Neil's work has been on the forefront in showing there is hope that if you do it right, you can reverse engineer the Internet to a degree that a few years ago a lot of people would not have believed."

Thousands of organizations provide Internet service, including about 20 major providers, such as

Mapping the Internet

■ NEIL SPRING

To contact any researcher in UMIACS, go to **www.umiacs.umd.edu**.

AT&T and Sprint that have built much of the infrastructure, the wires and routers, on which everyone relies. Each company knows how its own network is set up, but because companies keep this information confidential, "reverse engineering" offers an entry point to understanding the larger network.

The raw data for weaving together a map has come from the routes that many desktop computers show listing the IP, or Internet protocol, address between themselves and some network destination. Each IP address identifies a network handoff taking place at a router, a switching device that directs the flow of data through a network. The list of routers between two destinations traces the path that information followed. Piecing together millions of these routes can yield a map.

If the Internet has managed to operate for years without anyone really knowing how information gets from place to place, what is to be gained from mapping the Internet? One big benefit is "traffic engineering," says Spring. With a map in hand, one can figure out shorter routes, less congested routes, and how to avoid dead ends. Also, Spring says, "You can identify problems in the interfaces between organizations. One thing I've been able to study is how cooperative pairs of providers are." Because companies usually know nothing about the network structure in another organization, choices made by one organization can inadvertently cause problems in the wider network, notes Spring.

Spring, his student Rob Sherwood, and their collaborators have relied on data from planet-lab.org, a partnership of hundreds of universities and research labs that provides access to machines around the world for network research. The maps built by Spring's team chart how PlanetLab machines are connected to each other.

Individual routers have many interfaces, each identified by a unique IP address. "One of the challenges was being able to figure out which IP addresses represent the same router," says Spring. To help group IP addresses into the routers they represent, Spring turned to UMIACS director V. S. Subrahmanian and his student Austin Parker to incorporate artificial intelligence techniques into the mapping process.

"To get the most accurate map we can, we try to measure twice," says Spring. His team maps paths from opposite directions. The work remains challenging because the volume of data can be overwhelming, says Spring. Because all variables can't be accounted for at once, solutions in different batches may be inconsistent, so Spring is working on techniques to divide and conquer the problem so as to prevent inconsistencies.

"Neil and his colleagues managed to recover the network topology and routing policy for large-scale Internet service providers for the first time," says Yan Chen, a computer scientist at Northwestern University. "His work has been widely cited."

Having a map of the Internet, or even small parts of it, can reveal the decisions that are made to transmit information in the network and whether any parts of the network are not reliable. Maps can reveal inefficiencies and weaknesses and suggest ways to make the network faster and more reliable. "You can imagine building a network that would make much better decisions and run more efficiently," says Spring. "You want to be able to design a network that's as robust as possible, and you don't want to overestimate or underestimate the redundancy in the network."

Ultimately, a map of the Internet can help quantify how much room for improvement there is in network routing and policies. Maps can also serve as snapshots over time to track how the network continues to grow and evolve.

— Profile written by Karin Jegalian

create : collaborate :

Interfaces for Children

▪ ALLISON DRUIN

▪ BEN BEDERSON

In their research on interfaces for children, Allison Druin and Ben Bederson think of children as their design partners. Building better computer interfaces for kids is one of a range of projects that these two members of the Human-Computer Interaction Lab undertake. "We specialize in solving people's problems," says Druin.

"Kids aren't short adults. They're users with their own needs and behaviors," adds Druin, who leads empirical research on how children work with computers. This research feeds into technical development led by Bederson, who also studies other interface issues, including information visualization, interfaces for mobile devices, and the usability of electronic voting equipment.

"Early on, we recognized that children are very social and very naturally learn collaboratively," Bederson says. Seeing that children tried to use computers together, Bederson and Druin started to think about technology to support "co-present collaboration." Bederson started to modify computers to work with multiple mice. Ideas for co-present collaboration are now widely applied, for example in gaming and in restaurants employing monitors that respond to surface touch.

Allison Druin (l), Ben Bederson (r)

Interfaces for Children

- ALLISON DRUIN
- BEN BEDERSON

To contact any researcher in UMIACS, go to **www.umiacs.umd.edu.**

Druin and Bederson also led the first studies on mouse use by kids. They found that the ideal size of a clickable target decreases with users' maturing motor skills, decreasing from 64 by 64 pixels at age 5 to 16 by 16 pixels at age 9, the same as for adults.

Druin and Bederson strive to help children truly benefit from increasingly widespread access to technology. When working in New Mexico with kids who had little access to books, the researchers realized that exposure to stories inspires children to start telling their own. This idea spurred Druin and Bederson to build the International Children's Digital Library, or ICDL, available at childrenslibrary.org, the largest freely available collection of books for children. Visited by about two million unique visitors from around the world in the five years after its launch in November 2002, the ICDL offers thousands of children's books in more than 40 languages. Outreach for the library is now led by a dedicated non-profit foundation headed by Tim Browne. UMIACS faculty researcher Ann Weeks also helps direct the ICDL, leading efforts to add to the library's collection.

"The singular characteristic of Allison and Ben's work is their deep-rooted respect for the role children can play in the design of educational computing systems," says Browne. "The library is the culmination of their profound exploration of how science can improve the world."

One of the innovative features of ICDL is how it allows children to search. Druin and Bederson noticed that kids tend to classify books by their emotional impact—whether stories are happy, sad, scary or funny. Also, kids often identify a book by the color of its cover or its size. In ICDL, icons around the edge of search pages allow kids to search in ways that feel natural to them. "They can do complex Boolean searches without knowing what complex Boolean searches are," Bederson says.

To make the interface usable for children speaking dozens of languages and for young children who are hardly reading yet, the interface conveys information primarily visually. The library allows for a variety of ways of reading books. Kids may flip through a book or start reading it in the middle. "Depending on who you are, your age and your goals, you're going to want to search and read in different ways," says Druin.

Druin and Bederson work for extended periods with a rotating crew of about half a dozen children aged 6 to 11. They ask these collaborators to perform tasks, like finding a book. They also have these children observe other children, for example at a public library. Children are more honest with each other, the researchers point out, and also interpret the actions of other kids differently than an adult might. "We have to push the boundaries by creating new technology, but we also have to push the boundaries by understanding people," says Druin.

"Ben and Allison do fabulous, cutting-edge work," says Michael Levine of Sesame Workshop, who has worked with them. "Many research scientists know their discipline and do not reach out for broad collaborations. … Ben and Allison are practicing scientists who care deeply about kids' learning and literacy." Druin has also advised LeapFrog, the electronic toy maker. "Her vast experience with kids and technology provided useful insights about how to tailor product design to meet the needs of kids," says Jim Gray of LeapFrog.

Bederson and Druin have traveled around the world to help establish computer resources for children. Bederson builds and deploys systems in places that may have no Internet access or even reliable access to power. He and Druin have also worked with children in Honduras, New Zealand and Germany to start to study how cultural differences influence how children use digital resources.

While Druin and Bederson are proud of the interfaces they've helped develop for children, much of their motivation is to help the wider field design with and for children. "We're not just testing kids. We're working with them as informants," says Druin. "We believe children should have a voice in the design process."

— Profile written by Karin Jegalian

create : collaborate

Scientific Computing

- RAMANI DURAISWAMI
- HOWARD ELMAN
- DIANNE O'LEARY
- G. W. STEWART

Scientific computing is the foundation of computational science and engineering. Anyone desiring to use the computer for understanding particle interactions in stars, modeling the behavior of an airplane wing, or creating a better search engine must develop algorithms that run fast and produce accurate answers. That's where scientific computing comes in.

continues on back >

(l-r) G. W. Stewart, Ramani Duraiswami, Dianne O'Leary, Howard Elman

UMIACS

Scientific Computing

- RAMANI DURAISWAMI
- HOWARD ELMAN
- DIANNE O'LEARY
- G. W. STEWART

To contact any researcher in UMIACS, go to **www.umiacs.umd.edu.**

Ramani Duraiswami, Howard Elman, Dianne O'Leary and Pete Stewart work in numerical analysis and scientific computing, designing algorithms to help scientists and engineers solve their problems. "We are a loosely collaborating group that work in overlapping areas," says Stewart. In various combinations, they have published many joint papers and garnered grants from institutions such as the National Science Foundation and the National Institutes of Health. As O'Leary observes, "We are definitely a resource for one another."

O'Leary and Stewart have worked together for more than 30 years, but they have different perspectives. O'Leary prefers to find her problems in specific areas, where her collaboration with scientists and researchers gives her great satisfaction. Stewart prefers to look for a general problem that has applications in several areas and develop a single algorithm to solve it. "Infrastructure," he calls it. His satisfaction lies in seeing his algorithms percolate through various disciplines.

Elman says he often studies algorithms that researchers use to solve their problems. "I'll try to convince them to use new algorithms that can speed up codes five or 10 times," he says. For example, if the algorithms used in testing a design for an airplane wing can be speeded up to run in 3 hours instead of 24, it can allow more possible designs to be considered. Elman works with researchers at Sandia National Laboratory to develop faster code for studying fluid dynamics and reactive flows. In the past, he has worked with the oil industry on oil extraction problems.

O'Leary develops algorithms for finding useful subject clusters during online Web searches and for solving ill-posed problems. "In ill-posed problems," O'Leary says, "small changes in the data can cause arbitrarily large changes in the results." The algorithms, for example, can recover the original image from a blurry photo or from medical imaging data.

Duraiswami wants to create algorithms that are "much faster, more stable and more accurate." His work on multipole methods finds applications wherever there is a need for super-fast number crunching. He collaborates with physicists at the university to simulate the behavior of plasma and to study how to use light to manipulate nanoparticles. "I like to do concrete, practical things, he says. "What drives me is doing them quickly."

Stewart points out that with their collaborators in the university's mathematics department, "We have one of the strongest groups in scientific computing in the country." The four professors are all the authors of popular books and monographs on their research areas. Through their publications, their computer software, their research papers and their collaborations with other researchers, the scientific computation researchers at UMIACS help answer important scientific and engineering questions.

Stewart, who is a member of the National Academy of Engineering, says he "learned his trade solving practical problems." While he studied mathematics in college, he also worked at the computation center of the Oak Ridge Gaseous Diffusion Plant. "I acquired a lot of pure mathematics, most of which has turned out to be eminently practical," he says. "I can't interest myself in a problem if I don't foresee a practical application." That might well be the motto of the entire scientific computing group.

To learn more, go to the following Web sites:

RAMANI DURAISWAMI
www.umiacs.umd.edu/~ramani/

HOWARD ELMAN
www.cs.umd.edu/users/elman/

DIANNE O'LEARY
www.cs.umd.edu/users/oleary/

G. W. STEWART
www.cs.umd.edu/users/stewart/

create : collaborate

University of Maryland Institute for Advanced Computer Studies
2119 A.V. Williams Building • University of Maryland • College Park, MD 20742

UNIVERSITY OF MARYLAND