TRN
TRNmag.com
The Latest Technology Research News

Home Offline TRN Newswire Archive Research Directory Events Directory Resources About

Simple search lightens Net load

By Kimberly Patch, Technology Research News

Researchers working on finding better ways to search the Internet are increasingly turning to methods that require individual nodes, or servers, to know a little bit about nearby servers, but don't require servers to look much beyond their own neighborhoods.

This type of co-operation, which is also found in many natural networks such as insect communities, uses local rules in such a way that the system as a whole has predictable global properties.

Researchers from the University of California at Los Angeles have devised a fast search algorithm that uses local rules to find nodes and content in randomly-formed, scale-free networks such as the Internet. Scale-free networks have a few nodes that have many connections to other nodes, and many nodes that have far fewer connections.

"Without global knowledge of the network, and only doing local operations, we can make the cost of searching an entire network grow less than linearly [with] the size of the network, and still have the query be very fast," said Vwani Roychowdhury, a professor of electrical engineering at the University of California at Los Angeles.

The search algorithm could be used to increase Internet efficiency by making it easier to find routes between hosts, said Roychowdhury. It could also be used to reduce traffic in peer-to-peer networks running on the Internet that allow people to exchange files, like Kazaa and Gnutella.

Queries need to be passed along only a few links rather than flooded throughout the network, which keeps search-related traffic low, said Roychowdhury.

"Many networks are known to be scale free... our search algorithm could be applied to all of them," said Roychowdhury. The researchers' simulations have showed that the algorithm could reduce Gnutella traffic by one or two orders of magnitude, he said.

In 2001, researchers at Stanford University and Hewlett-Packard Laboratories developed a simple, light-weight random search algorithm for peer networks. That algorithm forwards queries one node at a time whereas the UCLA researchers' algorithm operates in parallel, said Sarshar.

The researchers' algorithm is based on the bond percolation threshold, or the smallest probability that a message is guaranteed to reach a core sub-network of highly-connected nodes, said Roychowdhury.

As connections randomly percolate through the network at a low rate, only small, isolated islands form. Once the bond percolation threshold is passed, the core of the network becomes connected. The threshold is an abrupt phase transition like the quick transformation of would in that takes place when water boils or freezes.

The algorithm involves three basic steps: content caching, query implantation, and bond percolation, said Roychowdhury.

September 8/15, 2004

Page One

Automatic icons organize files

Simple search lightens Net load

Chip architecture uses nanowires

Polymer serves up single photons

Briefs:
Alumina glass made in bulk
Pure crystal promises hardy chips
Nanoribbons channel light
Photonic crystal throttles light
Nano memory scheme handles defects
Nanotube transistor has power

News:
Research News Roundup
Research Watch blog

Features:
View from the High Ground Q&A
How It Works

RSS Feeds:
News XML | Blog XML | Books XML

What's New Log

Ad links:
Buy an ad link

Content caching happens when a node joins a peer-to-peer network and performs a one-time short random survey, or walk of nearby nodes and adds its content directory to each of these neighboring nodes.

The query implementation step is similar, but happens at the beginning of every query. When a node has a query, it performs a short random walk and passes the query along to each node it encounters.

These random walks are long enough that any given node will almost surely encounter at least one highly-connected node, said Roychowdhury. "So after these two steps one of the high-degree nodes has a copy of a node's directory, and a query is implanted at one of the high-degree nodes."

Once this has been set up, the bond percolation step makes sure that the directory and query connect.

In this last step, all of the initially queried nodes percolate the query throughout the network so that the query is guaranteed to reach a core sub-network of highly-connected nodes. "Since a copy of the query is in one of the nodes in the core network, and since the content list of a node is cached at one of these high-degree nodes, one is guaranteed to find the content as long as at least one node in the network has it," said Roychowdhury.

The traffic generated per query using this method is relatively low, and grows more slowly than the network, said Roychowdhury. In a 100-million node network, for example, the random walks will hit about 100 nodes each, and the traffic generated per query 10,000, he said.

The key step in working out the algorithm was making sure that each query could reach a high-degree node starting from any node in the network. The conceptual breakthrough was the realization that the whole search process could be implemented locally, through messages passed among neighbors only, said Nima Sarshar, a researcher at the University of California at Los Angeles.

The researchers are taking the technology a step further by incorporating it into a peer-to-peer network, dubbed BruNet, that operates by local and simple actions, and that provides both unstructured and structured searches in the same network, said Roychowdhury.

Currently the peer-to-peer networking community is divided into two distinct groups, said Roychowdhury. In structured peer-to-peer networks like Chord, Pastry, and Tapestry, each piece of data has a unique ID, which makes searches efficient. In the unstructured peer-to-peer networks popular with the file-sharing community, users search using keywords and expect to get several documents for each query.

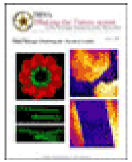
The researchers' peer-to-peer architecture provides both options, said Roychowdhury.

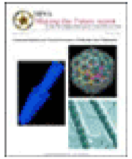
The search algorithm could be used in practical applications, including improving existing peer-to-peer systems like Gnutella, within one to two years, said Roychowdhury. The researchers are working on a software library that will make it easy for developers to include the search algorithm in their systems, he said.

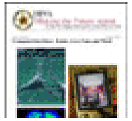
The general approach of looking at distributed engineering systems as dynamic physical systems could eventually enable self-organization and coordination of many types of distributed systems including colonies of mobile sensors or robots, said Roychowdhury. "It could... allow a distributed system to efficiently make decisions about global events, and enable distributed monitoring systems to detect and respond to attacks," he said.

Roychowdhury and Sarshar's research colleague was P. Oscar Boykin. The research was funded by the National Science Foundation (NSF) and the Defense Advanced Research Projects Agency (DARPA).

Timeline: 1-2 years
Funding: Government
TRN Categories: Internet; Databases and Information Retrieval







Story Type: News
Related Elements: Technical paper, "Scalable Percolation Search in Power Law Networks," posted on the Computing Research Repository (CoRR) at http://arxiv.org/abs/cond-mat/0406152

Home Archive Resources Feeds Offline Publications Glossary
TRN Finder Research Dir. Events Dir. Researchers Bookshelf
Contribute Under Development T-shirts etc. Classifieds
Forum Comments Feedback About TRN
TRN Newswire and Headline Feeds for Web sites