AN ADVANCED SIMULATOR FOR EVALUATING PERFORMANCE OF PEER-TO-PEER PROTOCOLS

Gergely Csúcs, Bertalan Forstner, Kálmán Marossy, Imre Kelényi, Dr. Hassan Charaf
Budapest University of Technology and Economics
Department of Automation and Applied Informatics
H-1111 Budapest, HUNGARY
Email: {gergely.csucs, bertalan.forstner, kalman.marossy, imre.kelenyi, hassan.charaf}@aut.bme.hu

ABSTRACT

In the domain of Peer-to-Peer (P2P) networking different protocols are evolving to make sharing of resources or information retrieval more efficient. The goals of advanced protocols are to reduce the number of messages sent and increase the hit rate of finding the requested resource. There are two main approaches to prove the efficiency of a P2P protocol. The first is to estimate the results with mathematical models; the second way could be the actual simulation of the particular P2P network. The latter approach is also suitable for experimenting with protocol modifications. The GXS Peer-to-Peer Simulator introduced in this paper is flexible enough to support nodes implementing different protocols, run defined test cases, and to gather statistics from any properties of the simulated network. GXS – besides of actually being utilized for P2P development – is a great tool for educational purposes, where high-level abstractions ("send the message", "handle incoming message") serve understanding better than looking at and trying to modify an actual, multi-threaded networking code.

KEY WORDS
Modeling and Simulation, Peer-to-Peer, Optimization, Protocols

1. THE GXS SIMULATOR

The benefit of Peer-to-Peer (P2P) networking has been recognized since around the middle of the year 2000. As in P2P networks the required network bandwidth is a great concern, more and more attempts were made to analyze the performance of the various P2P protocols.

With the GXS Simulator a P2P protocol can be implemented with minimal effort and analyzed in several ways: protocols can be validated, improved or fine-tuned, while goodness of fitting to a mathematical model or simply the generated network traffic can also be evaluated.

The requirements for the simulator were as follows:

- Providing an infrastructure for simulation (i.e. a simulation environment)
- Defining a clear interface which captures the key characteristics of general P2P nodes and messages
- Collecting statistics automatically about network traffic required for performance analysis
- Being flexible enough to use with totally different P2P protocols in different network environments

The simulator has been written in Java language, thus GXS itself benefits from reflection API on instantiating nodes, gathering statistics and so on. On the other hand, it is also faster getting a reliably running protocol prototype in Java. The simulator uses discrete time.

2. IMPLEMENTING PROTOCOLS

Simulated nodes and GXS interact via two interfaces: every node implementation has to extend an abstract Node class, and nodes reach the network through an interface called Net.

Extending the abstract Node class also implies the following features:

- Nodes have active and passive state
- Nodes get notifications on connection events (incoming connection attempts and connection drops)
- Every node is able to handle incoming messages

A node can invoke a few methods via the Net interface:

- Initiate a connection: if the target node exists, it can accept or decline the connection, oth erwise the Connect attempt automatically fails
- Drop a connection (target node gets notification)
- Send a Message using reliable/unreliable transport
- Start and cancel timers

Simulation output is the cumulative statistics of type-code of sent/received messages, and the return-code provided by message handler method. These results are available by per node and per system resolution at any time.

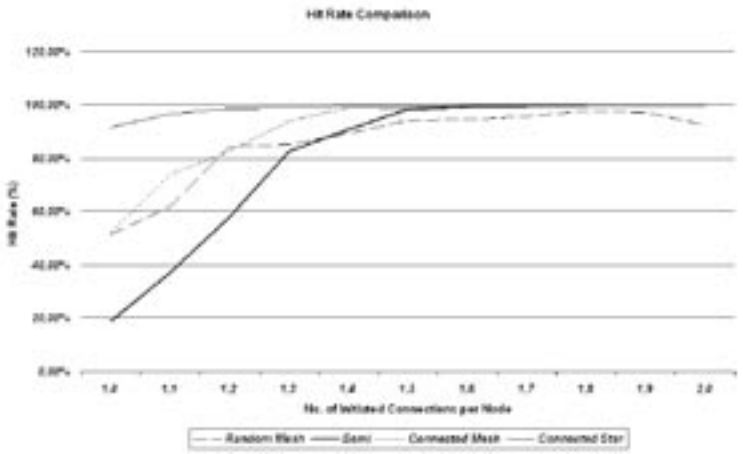

Figure 1

3. RESULTS AND FUTURE WORK

For actual simulations both the node capabilities and command set have to be extended. For validation of GXS, we have successfully reproduced the Gnutella-related results of Theodore Hong [1]. We have also evaluated different network topologies (Figure 1) and constructed Adaptive TTL, a way of achieving significant reduction of network traffic while the hit rate remains preserved [2].

Employing GXS, we have on-going research for extending the Gnutella protocol with a semantic layer and metadata-based routing [3]. Retaining backward-compatibility, it will be a promising improvement for desktop and our mobile [4] Gnutella clients.

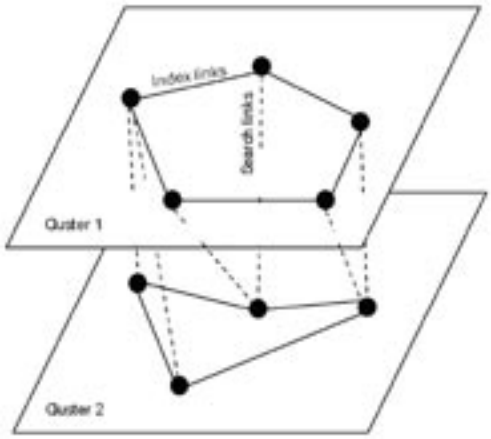

Figure 2

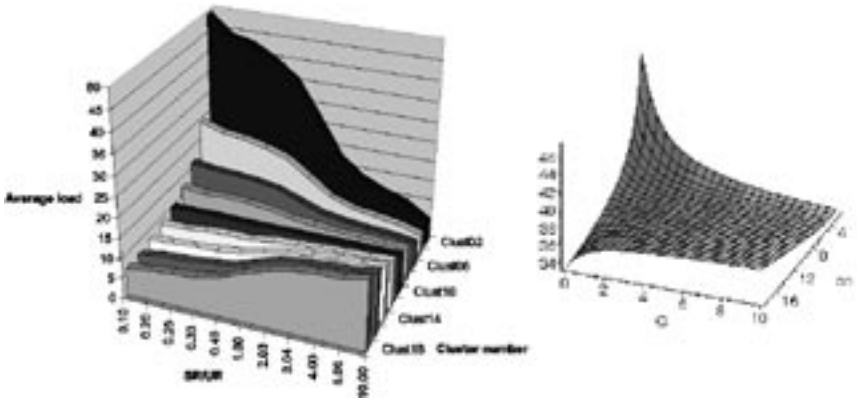

Figure 3

A past application of GXS was the development of PIC (Parallel Indexing Clusters) P2P patent [5] (Figure 2), and the validation of its statistical model approximation (Figure 3).

GXS itself is evolving in two directions: a graphing tool is going to be provided for presenting smaller networks based on the output of the simulator for validation purposes, and a distributed simulator engine is also under development for harnessing multiprocessor environments.

6. REFERENCES

[1] Andy Oram et al.: Harnessing the Benefits of a Disruptive Technology (O'Reilly & Associates, Inc., 2001)

[2] Csúcs, G. - Nurminen, J. K. - Bakos, B. - Farkas, L.: Peer-to-peer Protocol Evaluation in Topologies Resembling Wireless Networks. An Experiment with Gnutella Query Engine, (ICON, 2003)

[3] Bertalan Forstner, Hassan Charaf: Semantic Peer-to-Peer Information retrieval (Microcad 2004 International Scientific Conference, Miskolc, Hungary)

[4] http://symella.aut.bme.hu

[5] Marossy, K. - Csúcs, G. - Nurminen, J. K. - Bakos, B. - Farkas, L.: Peer-to-peer content sharing in wireless networks, (PIMRC, 2004)