A ns2-based simulation framework for performance evaluation of overlay networks

Michele Amoretti
Francesco Zanichelli
Università di Parma

Mauro Andreolini
Riccardo Lancellotti
Università di Modena e Reggio Emilia
Peer-to-peer systems

• Increasing number of P2P-based applications
  - Streaming
  - Filesystems
  - ...

• Need for tools to evaluate performance of overlay networks and routing algorithms
  - Prototypes → too many nodes required
  - Analytical models → may be too complex
  - Simulation

• Need for simulators of overlay networks
  - Already validated models and tools
ns-2 Network simulator

- **Standard “de facto” in network simulation**
  - Widely adopted in scientific community
- **Little support for overlay (P2P) networks**
  - Some P2P feature in code for ad-hoc networks (specific for message routing in wireless links)
  - Gnutella sim
    (support **only** for Gnutella v0.4, requires patches and is not maintained)

**Need of ns-2 extensions for overlay networks**
Proposal

• Simulation framework for overlay networks
  - Based on widely-accepted ns-2 simulator
  - Focus on performance evaluation

• Elements of the framework
  - Overlay routing algorithms (C++ classes)
  - Topology generators (TCL scripts)
  - Data analysis tools
Overlay routing algorithms

- Flexible base class hierarchy
- Object-oriented design: key classes
  - Support classes
  - Transport-level data
  - Logging facility
  - P2P routing
- Easy to extend
  - New algorithm
    → New class (+ TCL binding)
Available algorithms

- **Unstructured networks**
  - Flood-based
  - Probabilistic flood
  - Random Walker
- **Semi-structured networks**
  - Support for supernodes
  - JXTA
- **DHT**
  - Pastry
- **Novel algorithms**
  - Halo
  - Fuzzy-DHT
Logging facility

• Separate logfile for P2P network operations
• Reports routing events:
  – Hits
  – Miss
  – Query drop
• For each event logs additional information such as:
  – Simulation time of the event
  – Query TTL
  – Query key
  – Additional information (algorithm-supplied)
• Log file easy to parse (regexp-friendly)
TCL scripts

- Library of TCL scripts for the simulation
  - Definition of Physical topology
  - Construction of overlay network
    - Logging of generated topologies
  - Population of overlay network (resources)
  - Setup of log files
  - Generation of queries
- Multiple available functions for each phase
- TCL support for algorithms parameters
- TCL support for command-line options
  - Sensitivity analysis can be carried out without changing C++ and TCL code
Physical network topologies

- Star topology
- Real Network model (UniPR)
- Hierarchical topologies (Brite generated)
Overlay network topologies

- **Neighborhood relationships**
- **Available topologies**
  - Poisson networks
  - Scale Free networks
  - Seeded networks
  - Locally preferential networks
  - JXTA growth (JXTANetMap)
TCL support for resources and queries

- **Trace-driven simulation**
  - Repeatable simulation
  - Support for multiple RNG seeds

- **Traces for resources**
  - Distribution of resources over nodes

- **Traces for queries**
  - Sequence of keys to search in the network
  - Support for “wildcard searches”
  - Queries can have different “selectivity” $\sigma$
  - Allows to evaluate overlay routing algorithms performance as a function of $\sigma$
Data collection and analysis tools

• Automated analysis of logfile
  – Perl-based log parser
  – P2P logs
  – Standard ns-2 logs
  – Topology logs
• Automatic statistic extraction
  – Histograms
  – Cumulative distribution
• Automatic plotting
  – Template gnuplot files
Available analyses

- **Hit rate** (number of hits/theoretical max)
  - Exact searches
  - Wildcard-based searches
- **TTL of hits**
- **Time of hits**
- **Overhead** (number of pkts per query)
- **Topology of overlay network**
Experimental testbed

- **CASPUR Cluster**
  - 40 Nodes, dual AMD 64 CPU, 8GB RAM, SUSE Linux
  - OpenPBS batch scheduler

- **WebLAB Cluster**
  - 20 Nodes, Intel Xeon CPU, 1GB RAM, Debian Linux

- **UniPR Cluster**
  - 7 Nodes, Intel Pentium IV, 1GB RAM, Ubuntu Linux

- **Simulator compiles and runs correctly in every environment (including 64 bit)**
- **Simulation of up to 4000 Nodes**
Publications

- M. Andreolini, R. Lancellotti P. S. Yu, “A flexible and efficient lookup algorithm for Peer-to-Peer systems”, Submitted for publication, 2006
- M. Andreolini, R. Lancellotti P. S. Yu, “Analysis of peer-to-peer systems: workload characterization and effects on traffic cacheability” MASCOTS 2004
A ns2-based simulation framework for performance evaluation of overlay networks

Michele Amoretti
Francesco Zanichelli

Università di Parma

Mauro Andreolini
Riccardo Lancellotti

Università di Modena e Reggio Emilia