Technische Universität München
Lehrstuhl für Netzarchitekturen und Netzdienste
Prof. Dr. Georg Carle

Thesis
(B.Sc./M.Sc./Dipl.)

Distributed traffic engineering with Linux

Motivation

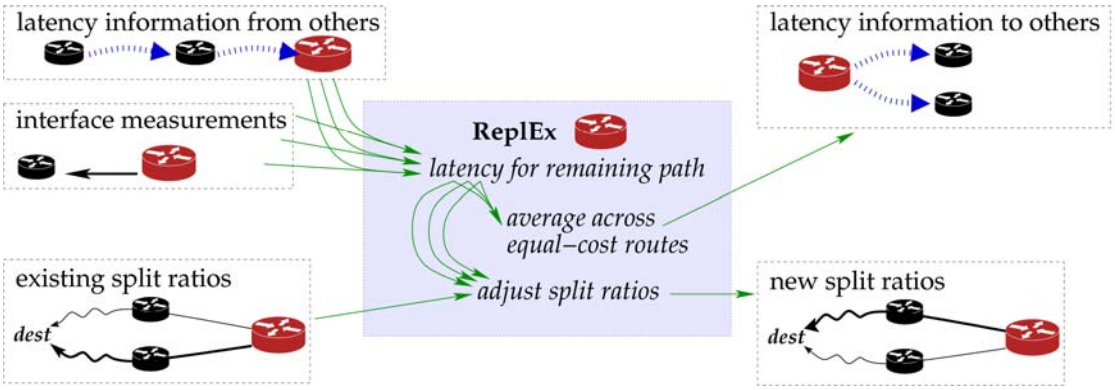
Routing in today's networks only reacts to failures of routers and links, but usually not to overloaded links – congestion control in the Internet usually is done in end hosts. Traffic engineering techniques (i.e., routing optimisation) only work on timescales of hours, but cannot react to sudden traffic shifts.

The ReplEx algorithm, which was developed at TUM, is a distributed traffic engineering algorithm that dynamically reacts to changes in traffic volumes on a very short timescale. It is very simple and universally applicable in a very broad range of contexts, as it can be used together with almost any routing infrastructure and can use almost arbitrary optimisation metrics. However, so far ReplEx has only been evaluated in a simulator environment, but not “in the wild”.

Your Task

Your task will be to implement ReplEx in a Linux environment, and to conduct evaluations and measurements that will be compared to simulation results that have been established already. ReplEx consists of an easy distance vector like protocol, which can be implemented in userspace, and a route adaptation engine which probably can be implemented in userspace as well.

The implementation should work on Linux and preferrably on BSD as well; at least it should be made easily portable. The preferred languages for the implementation are Java, Scala, Perl – but we also can talk about Python or C++.



Requirements

Basic networking knowledge (routing tables, IP prefixes, longest prefix match, distance vector protocols).
Knowledge of the following concepts is *not* required, but it will make your life easier: handling POSIX routing tables, accessing interface statistics, configuring routes and TUN/TAP devices, experiment planning (DOE), statistical tools like GNU R or Gnuplot, using PlanetLab, using tcpreplay or Python Scapy or other packet replayers / generators.

Keywords

Linux, Routing, Traffic engineering