Building a Real-Time Internet Traffic Map

Paper # 123

Abstract

This paper presents the design, implementation and evaluation of a near real-time Internet traffic map (ITM) system which any one can query for the available bandwidth of any link or path. The enabling technique and the major difference from existing systems is the massive and continuous, yet passive measurement from end-host applications (e.g., Internet video). We propose a *centralized* architecture to collect passive bandwidth measurement from video players and maintain an ITMbased on this information. Such design rises two challenges. First, the bandwidth measurement from video players is in a much coarser granular both in time (several minutes) and space (path-level) than ITMrequires (e.g., within minute and link-level). We address it by a suit of algorithms that leverage the power of massive coarse measurement to infer finer-grained information. Second, with millions of simultaneous measurement samples, the centralized processing can be a bottleneck to maintain ITMwith the freshest information. To show its feasibility, we present a highly scalable implementation of the algorithms that generate ITM. Our evaluation shows **XXX**

1 Introduction

- Real-time traffic information is increasingly critical for applications to make the best decision for three reasons:
  1. More choices mean more potentials for QoS improvement but also hard to explore locally.
  2. Internet traffic becomes more and more dynamic.
  3. Need for such a real-time Internet traffic is shared by many applications, suggesting the huge impact of a service like this.
- Real-time traffic map for Internet has three requirements
  1. Coverage: both client/server-side and link-level
  2. Overhead
  3. Real-time view
- Existing approaches do not meet all three requirements.
- Our argument is that we can build a real-time Internet traffic map (ITM) that meets all three requirements by using the massive and continous measurement from popular and bandwidth intensive applications (e.g., streaming video) running at the end hosts, together with other publicly available information (e.g., routing information)
- This paper materializes this approach to ITMthrough addressing two challenges:
  1. Link-level traffic inference: we present the techniques to extrapolate link-level traffic statistics by combining different data sources, including static Internet topology information. Note the difference to network tomography.
  2. Scalability: we present a design that is able to handle massive simultaneous measurements from millions of video sessions and maintain a global view of Internet traffic map in near real-time scale.
- We evaluate the performance of ITMsystem and show that **XXX** (from accuracy-wise and scalability-wise)
- Finally, we use **XXX** applications to demonstrate that ITMsystem can significantly improvement the QoS of applications.

2 Motivation

To motivate the need for an ITM, we use real-trace to show significant variability of (link-level and path-level) available bandwidth and bottleneck in both space and time.

2.1 Dataset

- Describe the dataset: number of sessions, client-ASNs, video objects, servers, etc.
- Link-level information dataset: ground-truth over a small set of links that we manually measured using **XXX** .

2.2 Variability in Available Bandwidth

We show that available bandwidth (both of a link and of a path) has significant variability, which suggests that the information must be kept fresh in **XXX** granular in time.

2.3 Bottleneck Variability

Bottleneck location is important since its bandwidth determines the end-to-end available bandwidth and we show that the location of bottleneck also varies significantly, which suggests the importance of wide coverage in space.

3 System Overview

3.1 ITM Architecture

- Give a high-level schematic figure of ITMsystem with input and ouput.
- ITMsystem consists of three parts
  1. Data input: video measurement and routing information
  2. Inference algorithm
  3. Output: ITM
  4. Query interfaces

3.2 Video Measurement Input

- Format of input: information associated to each session and granularity of measurement (per-minute, per-second or per-chunk)
- Client-side measurement collection

3.3 Routing Information Input

- Format of a routing information

3.4 Challenges

Explain and elaborate the two challenges of link-level available bandwidth inference and scalability of backend.

4 Link-level available bandwidth Inference

4.1 Problem Formulation

Formally describe the problem of link-level traffic inference

4.2 Bottleneck Inference Algorithm

Introduce the intuition and mechanism of a basic algorithm that solves the formulated problem.

4.3 Improvement

Improvement techniques (e.g., aggregation, removal of video/player-specific noise, inclusion of incomplete information, etc)

5 Scalable Backend

The implementation of the backend of ITMsystem must be scalable to handle massive simultaneous upates from client-side video sessions and process them scalably in order to maintain a near real-time view of ITM. This section presents a scalable implementation of the algorithms in last section.

5.1 Partition of input

5.2 Run the algorithms as MapReduce

6 Evaluation

6.1 Micro-benchmarks

6.1.1 Bottleneck Inference Algorithms

Use ns-2/3 simulation to show the inference accuracy

6.1.2 Backend Scalability

Use EC2 implementation to test its scalability and process latency.

6.2 Real-trace Evaluation

6.2.1 Coverage

6.2.2 Accuracy

7 Applications

7.1 Path Selection

Compare with static/random selection in terms of bandwidth as well as standard video quality metrics (buffering ratio, join time, etc).

7.2 Edge Server/Cache Selection

Compare with static/random selection in terms of bandwidth as well as standard video quality metrics (buffering ratio, join time, etc)

7.3 Peer Selection

This represents an application beyond video. Compare with static/random selection in terms of bandwidth.

References