On the Impact of Aggregation on The Performance of Traffic-Aware Routing

A. Sridharan[1], S.Bhattacharyya[2], C. Diot[2],
R. Guerin[1], J. Jetcheva[2], N. Taft[2]

[1] University of Pennsylvania

[2] Sprint ATL

Background and Motivations

- Efficient operation of IP networks calls for "*matching*" offered traffic to resources
  - ➡ A routing problem
- New usages of IP networks (VPN, streaming apps, etc.) require ensuring resource availability
  - ➡ A routing problem
- New technologies provide greater flexibility in assigning traffic to routes (MPLS, Diff-Serv,...)
  - ➡ A routing problem
- But there is a cost associated with all this!

Routing and Traffic Granularity

- Cost of routing increases with
  - Ingress classification
    - Assigning traffic to routes/paths
  - Number of routes
    - Number of forwarding entries and lookup complexity
  - Frequency of route updates (computation & setup)
    - To adjust to traffic fluctuations or new demands
- Performance of routing increases with
  - Number of routes
    - Better match between demand and resources
  - Frequency of route updates
    - Better match between demand and resources

$\Rightarrow$ What is the trade-off between the two?

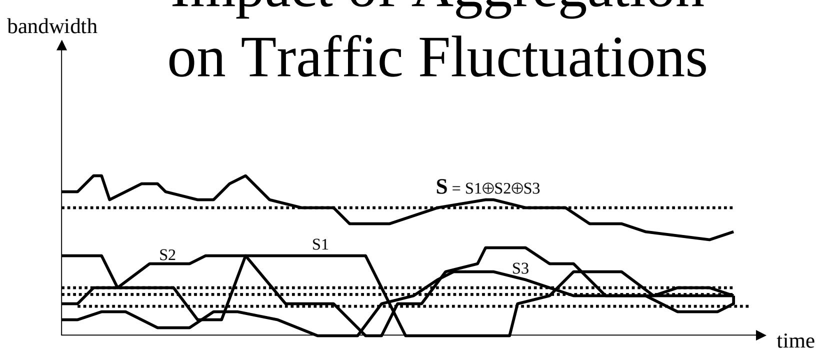
Problem Constraints and Issues

- External constraints
  - Traffic may be inherently unsplittable
    - Forces certain amount of traffic on the *same* path
    - Limits load-balancing ability of routing
- Internal constraints
  - Upper bound on the number of routes that can originate from or traverse a given router
    - Minimize setup cost and forwarding state
- Traffic aggregation trade-off
  - Fine granularity $\Rightarrow$ greater flexibility in matching demand to paths
  - Coarse granularity $\Rightarrow$ potentially smaller traffic fluctuations at small time scales

Traffic Aggregation

- What criteria and what level of granularity?
  - Sample choices
    - Ingress and Egress Routers
    - Type of Service, protocol (TCP, UDP)
    - Source and Destination Pairs ⊕ mask (size?)
  - Goal is to minimize impact on ingress classification while generating schemes that can facilitate load balancing
- Impact of aggregation criteria on “stream” characteristics
  - Number of independent traffic streams
  - Bandwidth distribution across streams
  - Variability of stream traffic

Impact of Aggregation on Traffic Fluctuations



- Aggregate streams may have smoother traffic
- Fine granularity streams can exhibit substantially greater variability
  - This can impact performance when routes are computed based on long term averages

Objectives

- Understand the impact of traffic granularity on network performance from two perspectives
  1 Long term (average) performance
    - Effect of distribution and number of streams on long term average routing performance
      - Quantify evolution of performance improvements
  2 Short term performance
    - Effect of distribution and number of streams on short term fluctuations of network performance
      - Are gains in average performance lost to greater fluctuations in short term performance?

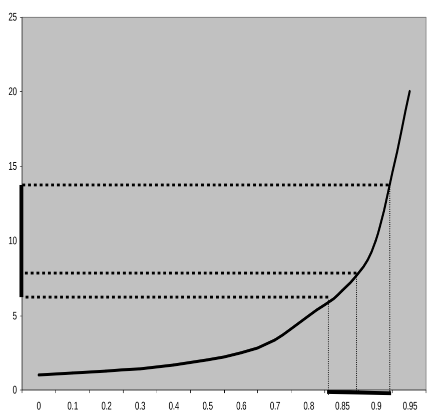
Long Term Performance

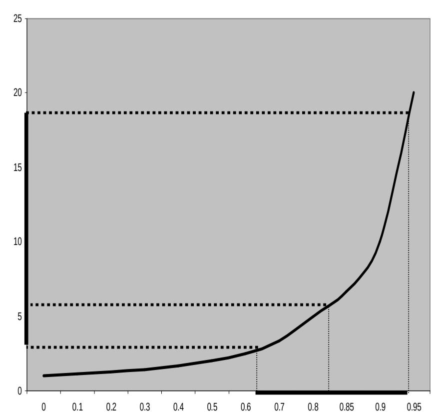
- "Long" term measurement, i.e., 800mins, to characterize offered loads
- Splitting of traffic in multiple streams improves routing's ability to do load balancing
  - Lower *average* link loads (over 800mins interval)
- Investigate improvement in overall *average delay* as number of streams increases
  - How does average performance improve as load balancing ability of routing increases?
  - Focus is on improvement of network performance on a 800mins time-scale

Short Term Performance

- Long term (800mins) measurements are split in 80 short term (10min) measurements
  - Generate 80 average 10min load samples from traffic traces
- Question
  - We know that we improve overall average (800mins) delay by splitting traffic for better load balancing, but what happens when looking at average delays over 10min intervals?
  - Does greater variability of fine granularity traffic over 1min intervals translate into more variable short term *link* loads?
- Evaluation based on the average (over 80 samples) of 10min average network delays
  - Do we also see improvements on the 10min time scale?

A Picture Is Worth 10^x Words





Lower average load ⇒ Lower average delay
Lower average load $\overset{?}{\Rightarrow}$ Greater load fluctuations ⇒ Even greater delay variations

Approach

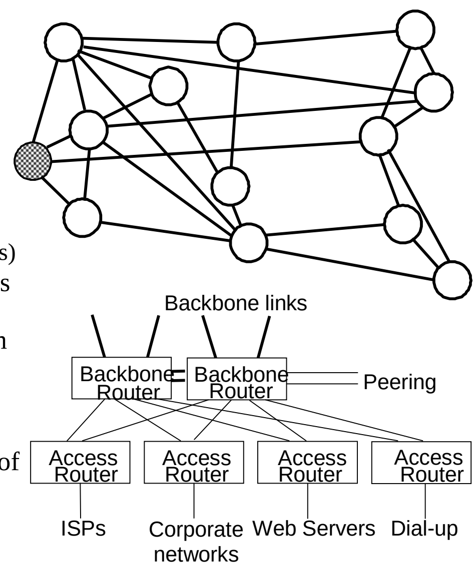
- Traffic characterization
  - Gather detailed traffic traces to study traffic distribution and fluctuations
- Traffic aggregation rules
  - Rely on DA/mask combination
- Heuristic routing algorithm(s)
  - Emulate “optimal” routing but incorporating granularity constraints
- Performance evaluation
  - Combine traffic traces, aggregation rules, and routing heuristic to study evolution of long term and short term performance

Approach

- Traffic characterization
  - Gather detailed traffic traces to study traffic distribution and fluctuations
- Traffic aggregation rules
  - Rely on DA/mask combination
- Heuristic routing algorithm
  - Emulate “optimal” routing but incorporating granularity constraints
- Performance evaluation
  - Combine traffic traces, aggregation rules, and routing heuristic to study evolution of long term and short term performance

Traffic Characterization

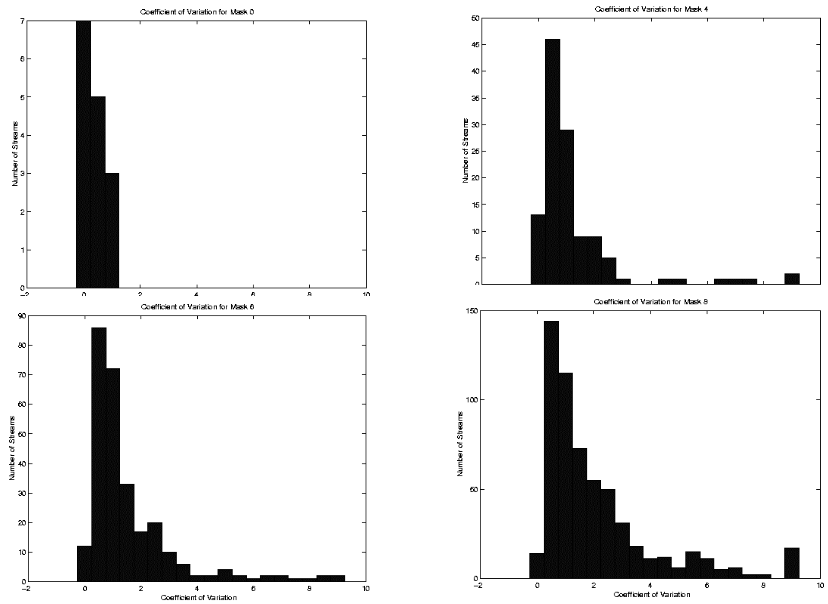
- Experimental setup
  - Traffic monitoring on Sprint backbone network
  - Monitoring probes installed at (initially) one POP
    - Gather first 40 bytes of packets
    - GSM clock time-stamping
    - 800mins traces (80x10min traces)
  - Downloading BGP routing tables and SNMP data
  - Construct full traffic matrix from measurements and SNMP based extrapolation
- Traffic aggregation rules
  - Destination address with masks of size 0, 4, 6, and 8



Traffic Aggregation Results

Granularity Level	Number of Streams	Bandwidth range (Mbps)
Mask 0 : p0	1	[1-14]
Mask 4: p4	[5-10]	[0-8]
Mask 6: p6	[10-25]	[0-4]
Mask 8: p8	[25-64]	[0-4]

Aggregation vs Traffic Variability



Approach

- Traffic characterization
  - Gather detailed traffic traces to study traffic distribution and fluctuations
- Traffic aggregation rules
  - Rely on DA/mask combination
- Heuristic routing algorithm
  - Emulate "optimal" routing but incorporating granularity constraints
- Performance evaluation
  - Combine traffic traces, aggregation rules, and routing heuristic to study evolution of long term and short term performance

Network and Traffic Models

- The network is modeled as a directed graph with *N* vertices (routers) and *E* edges(links)
- For each level *k* of traffic aggregation there is an *N*x*N* traffic matrix T^k with gives average traffic "estimates" for each pair of ingress-egress routers
  - Entries in T^k are of the form

$$T^k_{(i,j)} = [s_1(t_1), s_1(t_2), \ldots, s_1(t_M)], \ldots, [s_L(t_1), s_L(t_2), \ldots, s_L(t_M)]$$

where $s_l(t_m)$ in $T^k_{(i,j)}$ corresponds to the traffic associated with stream *l* between node-pair (i,j) and averaged over the *m*-th measurement interval

Problem Statement

- Algorithm goals and constraints
  - Compute paths and link loads together with assignments of streams to paths so as to optimize some network objective/cost function
    - Stream traffic intensities are based on averages over all *M* measurement intervals

$$\bar{s}_l = \frac{\sum_{m=1}^{M} s_l(t_m)}{M}$$

    - One-to-one assignment of streams to paths (**no splitting**)
- Typical objective/cost functions minimize
  - Average delay, maximum delay, maximum load, etc.
  - Focus will be on minimizing *average network delay*

Average Delay Cost Function

- Notation
  - λ is rate of packets into the network (in bits/sec)
  - C_l is capacity (in bits/sec) of link l
  - B_l is allocated bandwidth (in bits/sec) on link l
  - S is average packet size (in bits)
- Network links are modeled as M/M/1 queues
- Network wide average delay (cost function) is

$$T = \frac{S}{\lambda} \sum_{i=1}^{E} \frac{B_i}{C_i - B_i}$$

- Delay of path P is sum of its link delays

$$T_P = \sum_{l:l \in P} \frac{S}{C_l - B_l}$$

Why A Heuristic?

- Optimal routing of unsplittable flows is unfortunately known to be NP-Complete for all such instances

$\Rightarrow$ Explore and evaluate heuristics

- Trade-off
  - Complexity vs performance

  $\Rightarrow$ Investigate two heuristics
    - Simple greedy allocation of streams
    - Allocation based on optimal unconstrained solution
- Focus is, however, on identifying *trends* in the impact of traffic granularity on performance

A Greedy Heuristic

- Approach
  1. Order streams in some fashion
  2. Route them one-by-one on a minimum cost (delay) path
- Three ordering schemes were tested
  - √ Decreasing order (larger bandwidth first)
  - ◆ Increasing order (smaller bandwidth first)
  - ◆ Random order
- Simple algorithm, but ignores information available from global traffic matrix
  - ⇒ Direction for possible improvement

A Traffic Aware Heuristic

- Incorporates knowledge of traffic matrix
  - **Phase 1:**
    - Obtain optimal solution to problem by ignoring granularity constraints and solving a standard multi-commodity flow problem
    - For each sd-pair, route as many streams as possible on its " optimal network '' while exceeding any link's "optimal load'' by at most Δ
  - **Phase 2:**
    - Route remaining streams using the previous "Greedy Heuristic" on the topology with residual capacities

Review of Multi-Commodity Flow Problem Formulation

$$\min_{X_1, X_2, \ldots, X_K} f(X_{1,} X_2, \ldots, X_K)$$

subject to

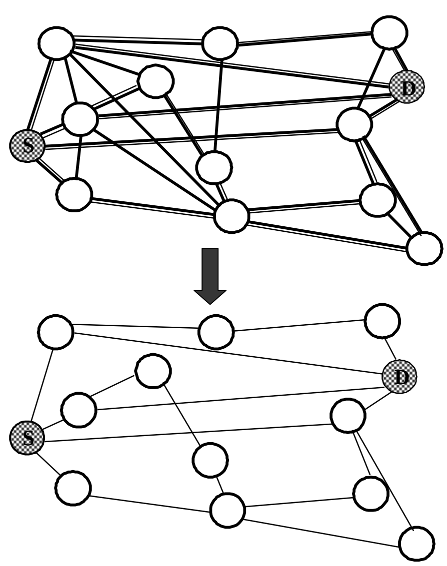
$$AX = R; \text{ where } X = [X_1 \quad X_2 \quad \ldots \quad X_K]$$

$$\sum_{k=1}^{K} X_k \leq C$$

- Where
  - X_k is the Ex1 flow array of each sd-pair
  - A is the NxE arc-node incidence matrix
  - R is the NxK node-sd-pair matrix
  - C is the Ex1 capacity vector of the network
- The output of the MCFP is a flow vector X_k for each sd-pair which specifies the traffic of the sd-pair on each link of the network

Phase 1 of Traffic Aware Heuristic

- Each of the *K* flow vectors produced by MCFP forms an independent "*network*" with link "*capacities*" set to the elements of the flow vector
- Streams between S and D are routed using minimum cost paths on the *network* produced by the MCFP
  - Streams are ordered as in the Greedy heuristic
  - Streams are routed unless "*link capacity*" is exceeded by ≥ Δ



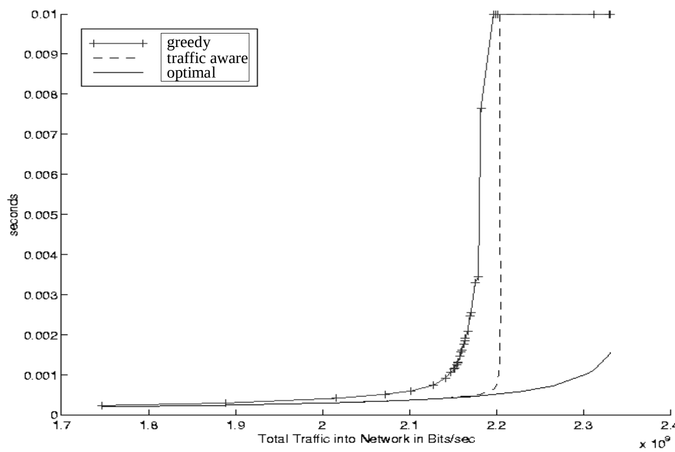
Approach

- Traffic characterization
  - Gather detailed traffic traces to study traffic distribution and fluctuations
- Traffic aggregation rules
  - Rely on DA/mask combination
- Heuristic routing algorithm
  - Emulate “optimal” routing but incorporating granularity constraints
- Performance evaluation
  - Combine traffic traces, aggregation rules, and routing heuristic to study evolution of long term and short term performance

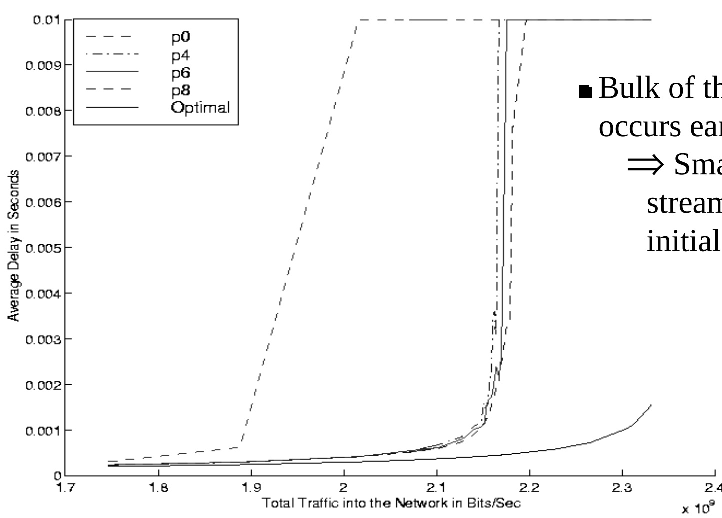
I. Impact of traffic granularity on *average* network performance

Greedy vs Traffic Aware Heuristics

- Comparison for fine granularity streams (p8)

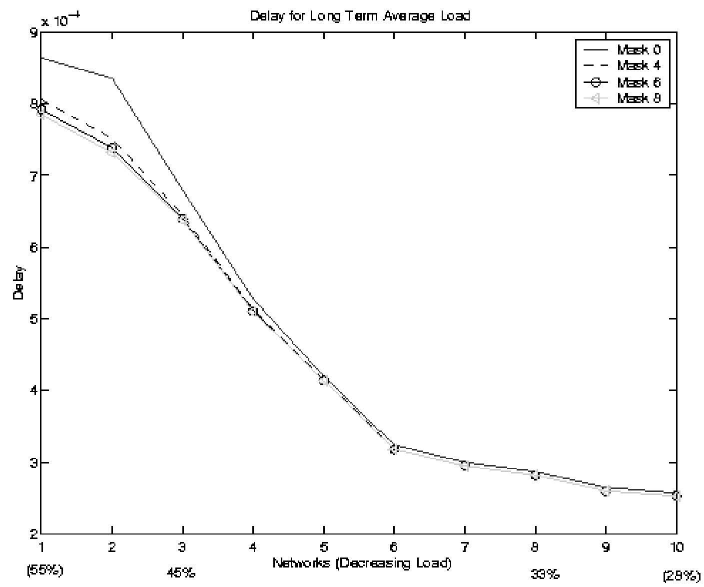


Routing Performance as a Function of Traffic Granularity

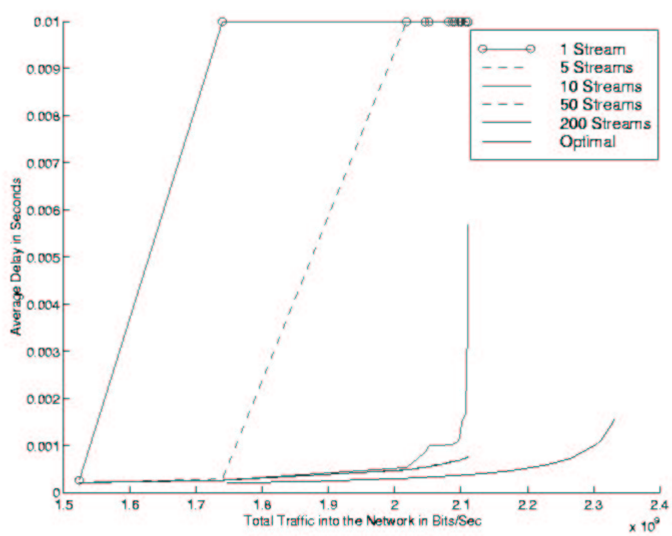


- Bulk of the increase occurs early on
  - ⇒ Small number of streams yield large initial improvement

A Different Look at the Same



A Systematic Look at the Impact of Number of Streams



Summary of Observations

- The larger the number of streams, the better performance should be (closer to optimal), *but*
  - Gains taper off rapidly as the number of streams grows
    - Slope is a function of network size and connectivity
  - The discrete nature of the streams can lead to a *decrease* in performance with increasing fineness of the splitting
    - Impact of packing of flows on network links
- Routing *big* streams *first* consistently yields better results than routing *small* streams first or using a *random* ordering
- **Traffic aware** heuristic typically outperforms **Greedy** heuristic

Another Basic Question

- How many distinct paths are actually needed?
  - Affects cost of forwarding state in the network
  - Affects potential for short term load fluctuations

Granularity	Mean no. of distinct paths	Max no. of distinct paths
Mask 0 : p0	1	1
Mask 4: p4	2	7
Mask 6: p6	2.3	12
Mask 8: p8	2.5	16

II. Long term improvements vs short term impact

Basic Issue

- Routing of streams is done based on their *average* load, but the short term traffic intensities can be drastically different.
- Aggregating traffic into few large streams can potentially minimize differences between short and long term
  - Does this yield a “less variable ’’ network performance?

$\Rightarrow$ Study the temporal behavior of the network cost function as aggregation level varies

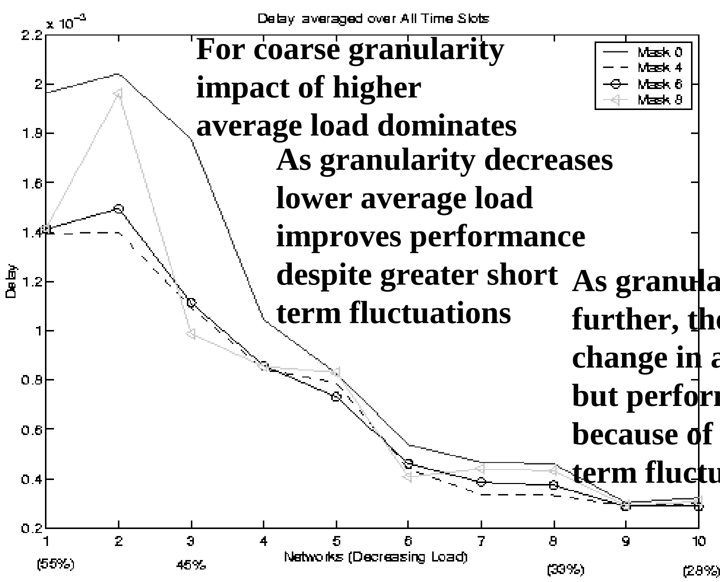
Evaluating Short Term Fluctuations

- Recall format of traffic matrix

$$T_{(i,j)}^{k} = [s_1(t_1), s_1(t_2), \ldots, s_1(t_M)], \ldots, [s_L(t_1), s_L(t_2), \ldots, s_L(t_M)]$$

- Experimental traffic data
  - Total measurement period of 793mins
  - Total of M=80 ten minute measurement intervals
- Compute "average" network performance for each small measurement interval
- Compare average performance for different aggregation levels across all 80 ten minute intervals

Delay vs Traffic Granularity



For coarse granularity impact of higher average load dominates

As granularity decreases lower average load improves performance despite greater short term fluctuations

As granularity decreases further, there is little change in average load, but performance degrades because of greater short term fluctuations

Observations

- Most of the benefits of finer granularity are achieved in the early stage
  - Number of streams and number of paths
- As expected, at low loads traffic granularity has little effect
- As load increases
  - Impact of coarse granularity becomes larger
  - Greater variability of fine granularity can impacts performance
    - Caused by fact that traffic assignments are based on “long term” averages
    - This happens despite the fact that streams are routed over a small set of paths

Conclusions

- Benefits of “traffic-aware” routing need to be examined in light of their impact on short term performance
  - A trade-off exists
- In practice, most of the benefits may be achievable with a small cost increase
- Additional work is obviously needed to better understand the exact relation between traffic granularity and load variability
  - Need additional measurements
  - Variability aware splitting of traffic