LIRA: Service Differentiation for Traffic Aggregates With Large Spatial Granularities

Ion Stoica  Hui Zhang

School of Computer Science
Carnegie Mellon University

Outline

- Introduction
- LIRA - Location Independent Resource Accounting
- Simulation experiments
- Conclusions and future work
**Motivation**

- Traditional QoS models
  - per-flow end-to-end

- Appropriate for
  - long duration and steady traffic
  - video/audio traffic, virtual lease line

- Not appropriate for
  - short duration and bursty traffic, e.g., web traffic
  - aggregate traffic over multiple destinations

---

**Service Differentiation for Traffic Aggregates**

- All traffic from Hui’s workstation vs. Ion’s workstation
- All traffic from CMU vs. University of Pittsburgh
- Service can be defined for both a local link and a network
Hierarchical Link Sharing

- QoS for traffic classes with different granularities
- Defined over a single physical link
- Models and algorithms
  - Class-Based Queueing (CBQ)
  - Hierarchical Packet Fair Queueing (H-PFQ)
  - Hierarchical Fair Service Curve (H-FSC)

QoS For Traffic Aggregates Over a Network

- User A pays $1000
- User B pays $500
- User C pays $1000
- User C pays $100
- What should be the right service?
Question

- What is an appropriate QoS or service differentiation model for aggregate traffic with **large spatial granularity**?
  - Traffic aggregate with a large set of destinations

Example: Assured Service

- Proposed by Clark & Wroclawski
- Each user is associated a traffic profile independent of destination
  - usually defined in terms of absolute bandwidth
- Traffic is of two types:
  - marked (in-profile)
  - unmarked (out-of profile)
- In-profile traffic is delivered with high probability
Potential Problem

- Worst-case provisioning needs to assume all marked packets traverse slowest link
- No obvious optimistic provisioning algorithm
- Cannot achieve high assurance and high utilization simultaneously

Conflict

- Profile definition: large spatial and temporal granularity
  - space: defined over a large set of destinations
  - time: defined over periods much larger than flow duration

- Achieving high service assurance requires congestion avoidance for any link, at any time
  - dynamic and local phenomenon
Another Example

- User Share Differentiation (USD) by Zheng Wang
  - each user is allocated a share
  - at each congested link, bandwidth is allocated proportionally to each user according to its share
- Undesirable property

Our Proposal: LIRA

- Define service in relative rather than absolute terms
  - service defined in resource tokens instead of fixed amounts of bandwidth
- Associate to each marked packet a cost as a function of
  - congestion level of path it traverses
  - packet length
- Mark a packet only if user covers its cost
- Key properties:
  - users receive service in proportion to their assigned token rates
  - high assurance - by appropriately choosing the cost function
  - high utilization - by using dynamic feedback and load balancing
Packet Marking Algorithm

Upon packet arrival:
packet_cost = f(path, packet, ...);
if (preferred(packet) && L > packet_cost)
L = L - packet_cost;
mark(packet);

Packet Cost

- Packet cost - product between packet length and path cost
- Path cost - sum of costs of all links on the path
- Link cost - cost to forward a marked bit on that link

packet_cost = packet_length * (c1 + c2 + c3 + c4 + c5)
Link Cost

- Objective
  - no marked packet is ever dropped

- Implication
  - when link utilization approaches unity the cost should exceed the total number of tokens in the system
    - in general, if number of tokens is unbounded, link cost should approach infinity

- Our choice
  \[ c(t) = \frac{a}{1 - u(t)} \]
  - \( a \) - fixed cost
  - \( u(t) \) - link utilization at time \( t \)

  link cost reflects link congestion

Link Cost Computation

- In a real system information is obsolete. This leads to
  - system oscillations
  - inaccuracies in cost computation

- Solution
  - make cost function more robust - use an iterative formula
    \[ c(t_i) = a + c(t_{i-1}) \times u(t_{i-1}, t_i) \]
  - account for “unexpected” variations by using only 85-90 % of link’s capacity for marked traffic
Load Balancing

- Maintain the $k$-th shortest paths
- Select among alternate paths based on their cost
- Potential concerns
  - oscillations
  - packets reordering within the same flow

Avoid Oscillation and Reordering

- Solution
  - bind probabilistically a flow to a path; probability depends on path’s cost
- Binding technique
  - each path is encoded by a label - XOR over (IP) addresses of all hops to destination
  - label is stored in forwarding table entry and packet header
  - forwarding based on match of labels in packet header and forwarding table
  - router updates the label in packet’s header by XOR-ing it with its address
Implementation Issues

- Path cost computation and distribution
  - link cost computation: $O(1)$ space/time complexity
  - leverage existing routing protocols to compute/distribute path cost
    - link-state protocols - make cost part of link state
    - distance-vector protocols - embed link cost in routing messages

- Packet marking and forwarding
  - per-user state at edge; no state inside network
  - $O(1)$ time complexity

- Load balancing
  - extend routing protocols to compute the $k$-th shortest paths
  - limit $k$ to avoid routing/forwarding tables explosion
  - integrate CIDR
Simulation Experiments

- Packet level simulator implementing both DV and SPF protocols
  - extended to compute the \( k \)-th shortest paths
- Traffic generation
  - self-similar -many ON-OFF flows with ON and OFF periods drawn from a Pareto distribution [Willinger et al]
  - shape parameter \( a = 1.2 \)
- Largest simulation
  - 30 nodes
  - 15000 flows
  - over 8 millions packets

Experiments Setting

- Schemes
  - BASE - single path routing
    - model today’s best effort Internet
    - use DV or SPF algorithms
  - STATIC - single path routing + LIRA
  - DYNAMIC-k - \( k \) shortest path routing + LIRA
- Metrics
  - user overall throughput - aggregate rate of user’s traffic delivered to all destinations
  - user in-profile throughput - aggregate rate of user’s in-profile traffic delivered to all destinations
- Simulation parameters
  - simulation time - 200 sec
  - routing update interval - 5 sec
  - link capacity: 10 Mbps, buffer size - 256 KB; threshold - 64 KB
Exp. 1 - Local Fairness and Service Differentiation

Exp. 1: Overall Throughput

Throughput (Mbps)

<table>
<thead>
<tr>
<th></th>
<th>BASE</th>
<th>STATIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>User 1</td>
<td>5</td>
<td>4.5</td>
</tr>
<tr>
<td>User 2</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>User 3</td>
<td>3.5</td>
<td>3</td>
</tr>
</tbody>
</table>

Throughput (Mbps)

0 0.5 1 1.5 2 2.5 3 3.5 4 4.5 5

User 1  User 2  User 3

1997 Hui Zhang
Exp. 1: STATIC

User 1
User 2
User 3

Throughput (Mbps)

- out-of profile
- in-profile

Exp. 1: Service Differentiation

- User 2 receives tokens at twice the rate of other two

User 1
User 2
User 3

Throughput (Mbps)

- out-of profile
- in-profile
Exp. 2 - Global Fairness and Load Balancing

Exp. 2: Overall Throughput
Exp. 2: STATIC

Throughput (Mbps)

user 1 user 2 user 3 user 4

out-of profile
in-profile

Exp. 2: DYNAMIC-2

Throughput (Mbps)

user 1 user 2 user 3 user 4

out-of profile
in-profile
Exp3: Complex Topology

- Similar to T3 NSFNET backbone
- Link capacity: 10 Mbps; User’s sending rate: ~13 Mbps
- Use DYNAMIC-3 scheme

Balanced Load: In-profile Throughput

- Scenario 1: token rate of each user: $0.5 \times 10^8$ tokens/sec
- Scenario 2: token rate of users 1,3,5,7,9,12,15,18 changed to: $10^8$ tokens/sec
**VPN Experiment**

- Token rate allocated per VPN
- Each VPN divides its rate equally among its nodes

**VPN Experiment: VPN In-profile Throughput**

- Scenario 1: token rate of each VPN: $2.4 \times 10^8$ tokens/sec
- Scenario 2: token rate of VPN 1 changed to: $4.8 \times 10^8$ tokens/sec

![Diagram showing VPN Experiment and corresponding throughput data graphs for two scenarios.](image-url)
Other Results

- Service assurance
  - no marked packets dropped in small simulations
  - around 0.1% market packets dropped in large simulations
  - 60% of unmarked packets are dropped

Related Work

- Assured service [Clark & Wroclawski]
- User-Shared Differentiation (USD) [Wang]
  - little correlation between user’s share and its throughput
- Resource allocation, e.g., [Waldspurger & Weihl], [Ferguson et al],
  - do not consider problem of allocating resources for traffic aggregates
- Smart markets [MacKie-Masson & Varian]
  - relation between packet’s priority and user throughputs not clear
  - difficult to achieve high service assurance
- Routing and load balancing
  - LIRA is first work to combine routing, load-balancing and congestion control
  - when all links have the same capacity, path cost is within a constant factor of $\text{shortest-dist}(P, 1)$ cost [Ma & Steenkiste]
Summary

- QoS model for aggregate traffic with large spatial granularity
  - LIRA: Location Independent Resource Accounting
  - a service not supported by current Intserv framework

- Global fairness reference model
  - each packet carries resource tokens
  - each router implements WFQ
  - packet pays token to every router traversed, weight proportion to amount of tokens paid

- Mechanisms that implements the model
  - leveraging on existing routing infrastructure to propagates the cost along a path
  - compatible in spirit with existing proposals (token-bucket based profile at edge, packet marking, semantics of marked packets)

Properties of LIRA
- high service assurance
- high resource utilization

Key ideas: separation of two levels of differentiation
- user level differentiation: destination/path independent
- packet level differentiation: destination/path dependent
- service profile defined in relative terms instead of absolute bandwidth bridges the gap
Future Work

- Extend LIRA to support
  - receiver based payment, multicast
- Utilize path cost at
  - egress nodes for active queue management
  - end systems for better end-to-end congestion management

Exp. 1: STATIC

- costs of links 5 and 6 are increased five times
Exp. 3 - Load Distribution and Load Balancing

Exp. 3 (cnt’d)

- Balanced Load - average user total and in-profile throughputs
Exp. 3 - Load Distribution and Load Balancing

Exp. 3 (cnt’d)

Unbalanced Load - average user total and in-profile throughputs

Throughput (Mbps)

- DEF
- DIF-1
- DIF-2

Unbalanced Load - average user total and in-profile throughputs

Carnegie Mellon

1997 Hui Zhang
Distributed Model

- Sender payment scheme
  - each user distributes its shares to access points
- Receiver payment scheme
  - ISP credits each marked packet received by a user up to a negotiated profile
    - packet should carry its cost (due to route asymmetry it can't be determined by the receiver)

```c
void EdgeRouterAlg()
{
    if (marked(packet))
    {
        if (credited(packet))
        {
            L = length(packet) * cost_per_bit;
        }
        else
        {
            if (L < length(packet) * cost_per_bit)
            {
                unmark(packet);
            }
            else
            {
                L = length(packet) * cost_per_bit;
            }
        }
    }
}
```

Exp. 3: Large Scale Example

- Topology similar to NSFNET backbone
- Each node Si acts both as a sender and as a receiver
Exp. 3: Balanced Load

- Nodes communicate with each other with equal probability

Exp. 3: Unbalanced Load

- Nodes inside island are 10 times more active than the other ones
Exp. 3: Virtually Partitioned Network

- Only nodes within the same island communicate among them

VPN Experiment: VPN Total Throughput

- Scenario 1: token rate of each VPN: $2.4 \times 10^8$ tokens/sec
- Scenario 2: token rate of VPN 1 changed to: $4.8 \times 10^8$ tokens/sec
Balanced Load: Total Throughput

- Scenario 1: token rate of each user: $0.5 \times 10^8$ tokens/sec
- Scenario 2: token rate of users 1, 3, 5, 7, 9, 12, 15, 18 changed to: $10^8$ tokens/sec