Implementation of Start-Time Fair Queuing Algorithm in OPNET

CMPT885/ENSC835 Final Project
by
Daryn Mitchell daryn@cs.sfu.ca
Jack Man Shun Yeung yeung@cs.sfu.ca

April 4, 2002.
Road map

- Background and Introduction to Start-Time Fair Queuing (SFQ)
- Project Implementation
- Some Simulation Results
- Conclusion
- References
Queuing in Routers - FIFO

- Buffers in routers allow multiple sources share one outgoing link
- Router schedules order in which packets leave
- Almost all FIFO/FCFS queuing (First-In, First-Out / First-Come First-Serve)
  - Simple, Cheap
  - Yet: Unfair?

FIFO queue
Queuing in Routers – FIFO unfair

- No bandwidth guarantee in the presence of congestion
- Misbehaving sources can steal bandwidth from behaving sources
Fair Queuing (FQ)

- Concept: Everyone gets a fair share of router bandwidth [Nagle RFC970, 1985]
- Assign a ‘weight’ to each data flow or category

<table>
<thead>
<tr>
<th>Method</th>
<th>Cost</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>WFQ [Demers89]</td>
<td>Expensive</td>
<td></td>
</tr>
<tr>
<td>WF$^2$Q [Bennett96]</td>
<td>Expensive</td>
<td></td>
</tr>
<tr>
<td>DRR (Deficit Round Robin) [Shreedhar95]</td>
<td>Cheap</td>
<td>Unfair for VBR, Not guaranteed, Punishes use of idle time [Parekh93]</td>
</tr>
<tr>
<td>Virtual Clock [Zhang90]</td>
<td>Okay</td>
<td></td>
</tr>
</tbody>
</table>
Start-Time Fair Queuing (SFQ)

- Proposed in 1996 by Goyal, Vin and Cheng [Goyal96]
- Like most FQ, assign weights to flows to give minimum bandwidth and maximum delay guarantees.
  - Simple algorithm (unlike WFQ)
  - Fair for VBR (unlike most FQ algorithms)
  - Fair use of idle bandwidth (unlike Virtual Clock)
SFQ Overview

- Basic idea:
  - Choose packet to send by lowest “start tag” among all flows
  - Start Tags in a flow generally increment by: $\frac{\text{Packet Size}}{\text{Flow Service Rate}}$

- If packets arrive as agreed, they are all served with guaranteed maximum and average delay
- If packets arrive faster than agreed, they will be delayed so they don’t affect other flows
Project Proposal

- Implement SFQ in OPNET as a queue module
- Create network incorporating SFQ in a router
- Run simulations to compare performance with other scheduling algorithms (e.g. WFQ and Virtual Clock)
- If possible, test with a real traffic trace
Implementation: Queue Process Model

SFQ Process Model
Was built by modifying the Opnet "acb_fifo" queue model to run according to the Start-Time Fair Queueing algorithm
Implementation: ICI

- Opnet “Interface Control Information”
  - Use to associate data fields with packets

- SFQ fields:
  - Start Tag
  - Finish Tag

- VC fields
  - Virtual clock
  - Auxiliary Virtual Clock
Implementation: Network Model

- Custom Source
  - Rate control at different intervals
  - Read Star Wars MPEG trace
Result 1: FIFO unfair

Incoming traffic per flow

1 Pk/8 sec
10 Pks/sec Misbehaving!
Result 1: FIFO unfair

Throughput per flow

FIFO = blue
SFQ = red

Stolen bandwidth

4/04/02
Start-Time Fair Queuing
Result 2: Verification of Fairness

- Source traffic as in VC paper [Alborz01]
  - src1: CBR 4 pk/s
  - src2: misbehaves
    - Fair queues make it wait
  - src3: VBR, avg. 4 pk/s
- Congestion
Result 2: Verification of Fairness

Throughput per flow

Virtual Clock =  
SFQ =  

4/04/02  Start-Time Fair Queuing
Result 3: Fair Use of Idle Bandwidth

- 2 clients
  - Client 0 (4 pks/sec) has a period of low arrival rate
  - Client 1 (4 pks/sec) has a period of high arrival rate at same time
  - There is congestion (8 pks/sec sharing 6 pks/sec)

Input Traffic per flow
Result 3: Fair Use of Idle Bandwidth

Throughput per flow

Virtual Clock =  
SFQ =  

4/04/02  Start-Time Fair Queuing
Conclusion: Project Proposal - Revisited

- Implemented SFQ in OPNET as a queue module
- Created a network incorporating SFQ in a router
- Ran simulations to compare performance with other scheduling algorithms (WFQ and Virtual Clock)
- Used a real traffic trace (Star Wars MPEG)
- Converted/Implemented Virtual Clock as a queue module in Opnet
References (1)


[Cis95] "Interface Queue Management" Cisco Systems White Paper  


References (2)


http://globecom.net/ietf/rfc/rfc970.html (14.Feb.02)
