CSE - 322
Computer Networks Sessional

NS2 Project Proposal

Name: Abdus Samee
ID: 1805021
Section: A1

DiffQ: Practical Differential Backlog Congestion Control for Wireless Networks

Link to the Paper

DiffQ: Practical Differential Backlog Congestion Control for Wireless Networks | IEEE Conference Publication | IEEE Xplore

Authors: A.Warrier; S. Janakiraman; S. Ha; I. Rhee

Published In: IEEE INFOCOM 2009

PDF

Motivations

- Tackle the flow in the middle problem
  - An initial flow gets hindered at a node due to flows generated later on

- Ensures fairness by proper distribution of the throughput in all of the flows
  - Few flows get disproportionately large amount of bandwidth, many flows are starving

DiffQ Design

Maintain a queue for each destination whose packet a node forwards

A node delivers a packet or inserts it in FIFO order into the queue of destination

A node also keeps track of the sizes of the destination queues of its neighbouring nodes

Queue is updated twice: if a node receives a packet or it overhears a reception

A queue differential is defined as:
$QDi(d) = |Qi(d)| - |Qj (d)|$
Qi(d) is destination queue of d at node i

Set a priority based on this queue differential for the HOL packet of each destination queue

Sender selects the HOL packet of highest priority and transmits it

Pseudocode

Source Rate Control()

```
F = Destination of flow originating at this node
qlen ←|Qi(F)|;
if qlen > QUEUE THRESH
    rate = rate/β;
else
    rate = rate + α;
```

Forwarder Algorithm()

```
Δ ←Number of priority levels supported by MAC;
D←Maximum per-dest queue size;
Flow Scheduling
F ←argmaxd QDi(d);
P ←HOL packet of Qi(F);
P.priority ←MAX( QDi(F ) D Δ, 0); P.qlen ←|Qi(F)|;
Transmit P;
On receiving packet P from local application
Encapsulate P with DiffQ header;
if P is the first packet
    Create flow entry for P’s destination;
F ←Destination of P;
Enqueue P into Qi(F);
On reception of packet P from node j
F ←Destination of P;
if F is this node
    Decapsulate DiffQ Header;
    Send it up to the application;
else
    if No flow entry exists for F
        Create flow entry for F;
    if node j is the routing next-hop for F
        QDi(F) ←|Qi(F)|−|Qj (F)|;
    else
        Enqueue P into Qi(F);
```