distance information to build a virtual distance map of the Internet and estimate the distance between any pair of IP addresses. The goal is to provide distance information in terms of latency and, where possible, bandwidth. In the context of nearest mirror selection for clients, we showed that significant improvement over random selection can be achieved using placement heuristics that do not require a full knowledge of the underlying topology. In addition, we showed that IDMaps overhead can be minimized by grouping Internet addresses into APs to reduce the number of measurements, the number of Tracers required to provide useful distance estimations is rather small, and applying t-spanner to the Tracer-Tracer VLs can result in linear measurement overhead with respect to the number of Tracers in the common case. Overall, this study has provided positive results to demonstrate that a useful Internet distance map service can indeed be built scalably. Through Internet experiments and simulations, it is showed that this approach can indeed provide useful distance information.

In this paper [7], a proactive queue management scheme called GREEN, which regulates TCP flows over the same link to a fair sending rate and hence prevents them from inducing congestion, is proposed. It does so by using the knowledge of TCP's steady state behavior. It exhibits high fairness with flows of widely varying RTT's. However their design suffers from severe under utilization in the presence of short lived or low bandwidth flows. From the detailed study of the previous works, it is clear that the Active Queue Management schemes such as RED, BLUE, SRED, REM do not provide adequate fairness at the cost of higher utilization and for widely varying RTT's. They also suffer from severe under utilization in the presence of short lived or low bandwidth flows.

3. Existing System

Most of the networks use Drop-Tail queue management where packets are dropped on queue overflow which is global synchronization problem. Random Early Detection was proposed as a solution to the 'Global Synchronization' problem and this opened up a new area of research called Active Queue Management. The key aims of AQM are to prevent global synchronization, reduce queuing delays and improve resource utilization.

4. Proposed System

The proposed system is lightweight, proactive and hence is ideally suited for deployment. It reduces Packet loss ratio, Increases transmission efficiency, Computational overhead is negligible. It avoids the congestion by effectively managing the queues.

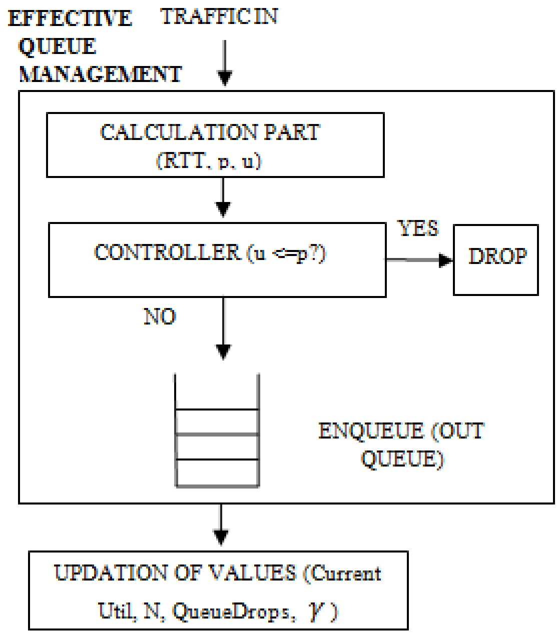

Figure 1: Effective Queue Management Architecture

4.1 System Architecture

When a packet is received at the EQM router, EQM first obtains the packet's RTT. Then it calculates the value of p (packet drop probability) and u (random number selected over a uniformly – distributed interval [0, 1]). If u <= p then the packet is dropped else it is added to the outgoing queue. If the window of time has elapsed for updating the values of current link utilization (currentUtil), the number of active flows (N), and the number of queue drops due to overflow (queueDrops), then these values are updated. Additionally the value of γ is adjusted based on currentUtil and queueDrops.

4.2 Module Description

4.2.1 Traffic IN

It signifies incoming of the packets from the sender into the EQM router.

4.2.2 Calculation Part

Calculation of RTT: When a packet enters, its RTT (Round Trip Time) is calculated. The method which is used for calculation of RTT is Embedded RTT's. It requires TCP senders to embed their current RTT estimates within the TCP header.

Calculation of p: It Signifies Packet Drop Probability. It is calculated using the Formula,

$$p = \left(\frac{N \times MSS \times c}{\gamma(t) \times L \times RTT} \right)^2$$

Where,
N → No of Active Flows
L → Outgoing Link capacity
MSS → Maximum Segment Size
RTT → Round Trip Time
c → A Constant value that depends on the acknowledgement strategy used

$\gamma(t)$ → A Constant at time "t"

Calculation of u: It's a random number selected over a uniformly-distributed interval [0, 1].

4.2.3 Controller

If u<=p then the packet is dropped (i.e., each packet is probabilistically dropped with the calculated probability p) and otherwise it is added to the outgoing queue.

4.2.4 Updation of Values

If the window of time has elapsed for updating the values of current link utilization CurrentUtil, the number of active flows N, and the number of queue drops due to queue overflow QueueDrops, then these values are updated. Additionally, the value of γ is adjusted based on CurrentUtil and QueueDrops.

5. Simulation SETUP

5.1 Pseudo Code for EQM Algorithm

```
Enqueue(Packet pkt)
RTT ← obtainRTT(pkt)
P ← [(N*MSS*c)/(γ*L*RTT)]^2
u ← UniformRand(0,1)
if (u<=p) then
        drop(pkt)
else
        addToQueue(pkt)
endif
if (currentTime() – lastUpdate >= window) then
        update(currentUtil, N, queueDrops)
        lastUpdate ← currentTime()
                if(queueDrops > 0 ) then
                        γ ← 0.95γ
                elseif currentUtil < 0.98 then
                        γ ←
                [(1+currentUtil)/(2*currentUtil)] γ
                end if
end if
```

5.1.1 EQM-Effective Queue Management

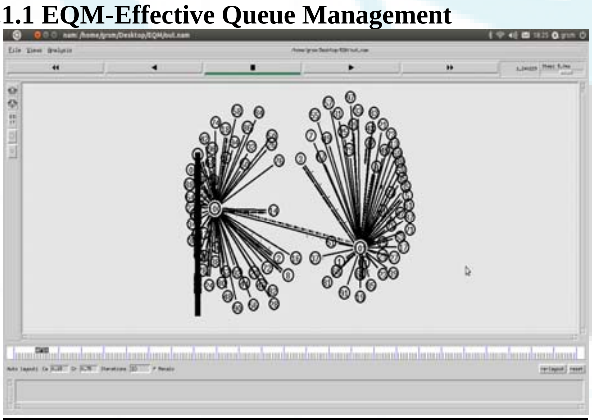

Figure 5.1: EQM Nam Window

5.1.2 End to End Delay - XGraph

End-to-end delay refers to the time taken for a packet to be transmitted across a network from source to destination. The End to end delay for the existing system is plotted using GNU plot.

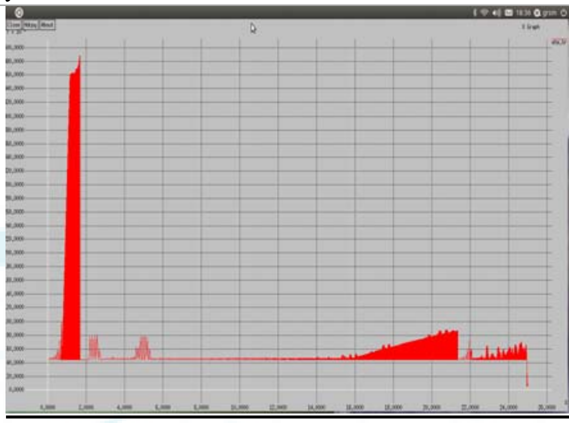

Figure 5.2: EQM End To End Delay

6. Performance Analysis

The necessary inputs are given to the Existing and Proposed systems and the output is received. From those output parameters, using Trace File, AWK commands and gnuplot, XGraphs and Bargraphs are drawn. Then using these graphs the performance is evaluated.

6.1 Average Queue Size

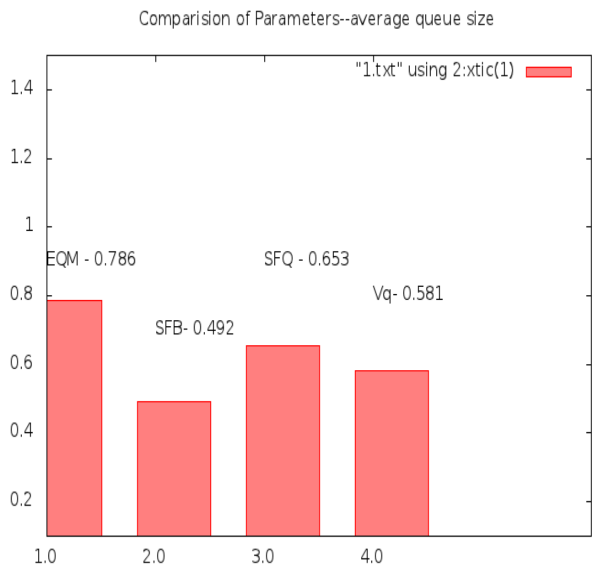

Figure 6.1: Average Queue Size

The Average Queue Size of EQM is higher than the existing schemes such as SFB, SFQ, Virtual queue etc.

International Journal of Scientific Engineering and Research (IJSER)
www.ijser.in
ISSN (Online): 2347-3878
Volume 2 Issue 7, July 2014

6.2 Packet Drop

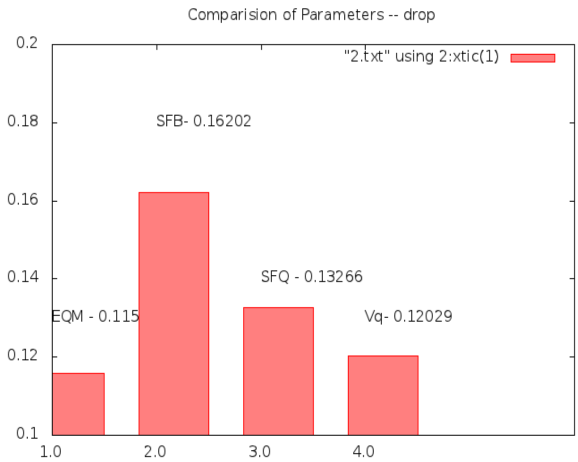

Figure 6.2: Packet Drop

The Packet Drop in EQM is lower than the existing schemes such as SFB, SFQ, Virtual queue etc..

6.3 Fairness

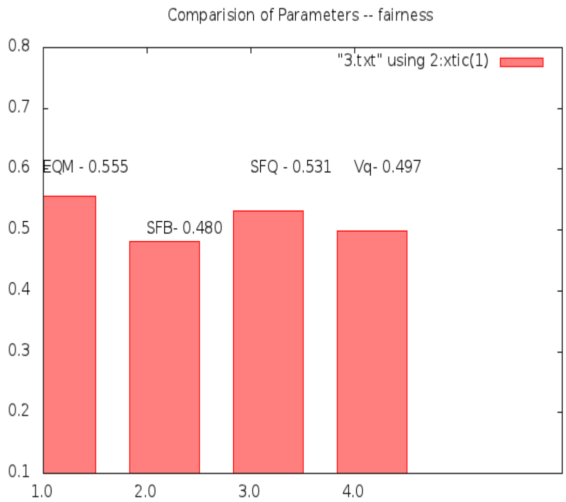

Figure 6.3: Fairness

The Fairness of EQM is higher than the existing schemes such as SFB, SFQ, Virtual queue etc.

6.4 Link Packet Loss

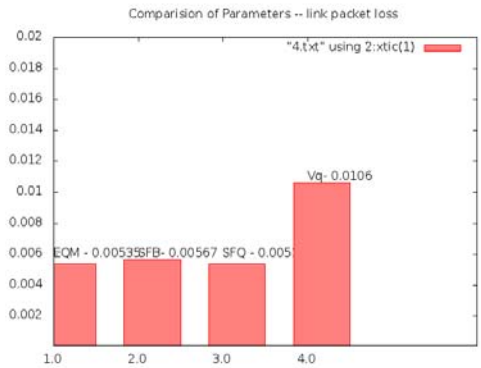

Figure 6.4: Link Packet Loss

The Link Packet Loss of EQM is Lower than the existing schemes such as SFB, SFQ, Virtual queue etc..

Fig 5.14 Throughput

The Throughput of EQM is higher than the existing schemes such as SFB, SFQ, Virtual queue etc.

6.5 Utilization

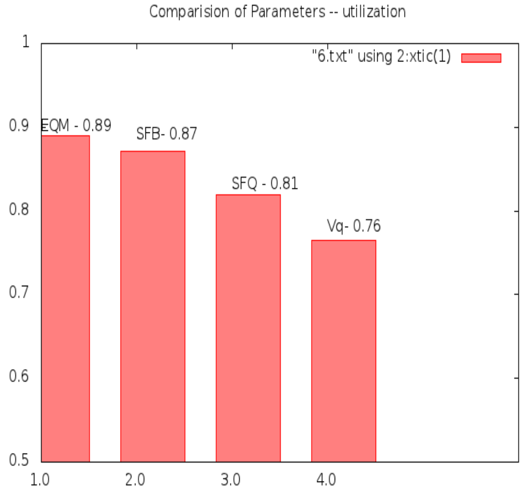

Figure 5.15: Utilization

The Utilization of EQM is higher than the existing schemes such as SFB, SFQ, Virtual queue etc.

7. Conclusion and Future Work

In this paper a new scheme called Effective Queue Management that overcomes the deficiencies in well known queue management algorithms has been proposed. Its effectiveness over the existing algorithms is proved using the comparison graphs. The algorithm is simple, robust, low in computational complexity, easily configured, and self-contained to a single router, making it easy to deploy. Deployment of low delay, low loss algorithm such as EQM will improve Internet performance and enable real-time applications. In future effectiveness of EQM will be analyzed using various parameters and further improvements will be done for various scenarios. It will also be deployed in MANET and its effectiveness will be analyzed.

References

[1] S. Floyd and V. Jacobson "Random early detection gateways for congestion avoidance". IEEE/ACM Transactions on Networking, 1(4):397–413, Aug. 1993.

[2] R. Braden, D. Clark, J. Crowcroft, B. Davie, S. Deering, D. Estrin, S. Floyd, V.J.G. Minshall, C. Patridge, L. Peterson, K. Ramakrishnan, S. Shenker, J. Wroclawski, and L. Zhang. "Rfc- 2309 recommendations on queue management and congestion avoidance in the internet". Technical report, IETF, pp 246–350, 1998.

[3] W. Feng, D. Kandlur, D. Saha, K. Shin, "Blue: A New Class of Active Queue Management Algorithms" U. Michigan CSE-TR-387-99, April 1999.

[4] T.J. Ott, T.V. Lakshman, and L.H. Wong. "SRED: Stabilized RED. In IEEE INFOCOM", volume 3, pages 1346– 355, 1999.

[5] W. Feng, D. Kandlur, D. Saha, K. Shin, “Stochastic Fair Blue: A Queue Management Algorithm for Enforcing Fairness”, in Proc. of INFOCOM 2001, April 2001.
[6] P. Francis, S. Jamin, C. Jin, Y. Jin, D. Raz, Y. Shavitt, and L. Zhang, “IDMaps: A Global Internet Host Distance Estimation Service”, IEEE/ACM Transactions on Networking, 2001 October.
[7] W.Feng, A.Kapadia and S.Thulasidasan. “GREEN: Proactive Queue Management over a Best-Effort Network”. in proc of IEEE Globecom 2002, November 2002.
[8] The Network Simulator - ns-2 homepage: http://www.isi.edu/nsnam/ns/