Design and Implementation of Efficient Packet Scheduling Algorithm

Anusha N, Reema Sharma

Student, Dept. of ECE, The Oxford College Of Engineering, Bangalore, Karnataka, India

Assistant Professor, Dept. of ECE, The Oxford College Of Engineering, Bangalore, Karnataka, India.

ABSTRACT: This paper presents a new scheduling technique to support expedited forwarding in the differentiated service (diffserv) architecture. The key feature of the new scheduling technique is to change the scheduling weights adaptively. This technique holds weighted round robin as a base scheduling scheme. Here we adaptively adjust the weights according to the dynamics of the average queue size of expedited forwarding. This adaptive packet scheduling is done to absorb the transient burstiness of the expedited forwarding (ef) aggregate which is caused by the traffic distortion inside the network. By doing so, our scheme can achieve low loss rate, low delay and delay jitter for the expedited forwarding.

 KEYWORDS: expedited forwarding, adaptive weights, delay, average, dynamic

I. INTRODUCTION

The Internet is evolving from a network that provided just best-effort transportation to a network capable of providing a wide range of services. The delivery of data related to some services has tougher requirements than best-effort packets. This has to be differentiated at network level. Differentiated Services (DiffServ) enables scalable network designs with multiple classes of service. MPLS traffic engineering(TE)[4] enables the fault-tolerance, resource reservation and optimization of transmission resources.

The basic goal of Differentiated Services architecture[6] is to fulfill the performance requirements of the users. Users request a certain performance level and the network provides it as long as the user traffic has certain characteristics. The performance level provided and the characteristics of the traffic to be injected in the network are defined in a SLA (Service Level Agreement)[2]. The way each router in the path treats a packet is referred to as PHB (Per-Hop Behavior). The default PHB is best-effort but there are more PHBs already defined or being standardized. Examples of PHB are EF (Expedited Forwarding) and AF (Assured Forwarding)[5].Here we introduce an efficient packet scheduling scheme to support delay-sensitive and loss-sensitive traffic network.

In this paper, we propose an adaptive-weighted packet scheduling scheme which can apply to weighted round robin and weighted fair queueing to support delay-sensitive and loss-sensitive traffic in the DiffServ architecture. Although the deployment of bandwidth broker could make dynamic resource provision a possibility, and the traffic
conditioning at edge routers shapes the incoming traffic as their traffic specification, there are still many factors that could cause traffic distortion inside the network:
* the transient effect caused by the dynamic flow aggregation;
* inaccurate traffic shaping at edge routers, and no traffic conditioning at core routers;
* packet clustering caused by cascaded queueing effects; and
* the path changes caused by route flip.

It is therefore important to make the packet scheduler at a core router adaptive to absorb the traffic distortion inside the network. The performance of the proposed scheme is evaluated by simulation. The simulation results have shown the proposed scheme to reduce the loss rate significantly without degrading the delay and delay jitter.

II. RELATED WORK

We will briefly discuss timestamp based and round-robin packet scheduling schemes since both relate to FRR. Some timestamp based schedulers, such as Weighted Fair Queuing (WFQ) [8] and Worst-case Fair Weighted Fair

Queuing (WF2Q) closely approximate the Generalized Processor Sharing (GPS). These schedulers compute a timestamp for each packet by emulating the progress of a reference GPS server and transmit packets in the increasing order of their timestamps. Other timestamp based approaches, such as Self-Clocked Fair Queuing (SCFQ) and Virtual Clock, compute timestamps without referring to a reference GPS server. These methods still need to sort packets according to their timestamps and still have an O(log N) per packet processing complexity. The Leap Forward Virtual Clock reduces the sorting complexity by coarsening timestamp values and has an O(log N) complexity. This scheme requires complex data structures and is not suitable for hardware implementation. Deficit Round Robin (DRR) is one of the round-robin algorithms that enjoy a good proportional fairness property. A number of methods have recently been proposed to improve delay and burstiness properties of DRR. The Smoothed Round Robin (SRR) scheme improves the delay and burstiness properties by spreading the data of a flow to be transmitted in a round over the entire round using a weight spread sequence. Aliquem allows the quantum of a flow to be scaled down, which results in better delay and burstiness properties. The Stratified Round Robin (STRR) scheme bundles flows with similar rate requirements, scheduling the bundles through a sorted-priority mechanism, and using a round robin strategy to select flows in each bundle. STRR guarantees that all flows get their fair share of slots.

III. PROPOSED ALGORITHM

A. *Key Features:*

Here we propose a packet scheduling scheme with adaptive weights to deal with the traffic congestion in the current networks. This can be applied to weighted round robin or fair queueing. The key features of our algorithm include:

- The buffer space for expedited forwarding aggregate is slightly increased to achieve low delay and low delay jitter.
- Exponential weight moving average (EWMA) is employed to estimate the average queue size of expedited forwarding.
- The weight of expedited forwarding is adaptively adjusted, according to the dynamics of average queue size. However, there is an upper limit by which the weight of premium service should be bounded and by maintaining a very small average queue size, low queueing delay is achieved.
- Also, a low-pass filter is used to reduce the fluctuation of instantaneous queue size, achieving low delay jitter.

B. *Description of the Proposed Algorithm:*

Similar to random early detection (red)[7], we employ the estimated average queue size of expedited forwarding as the index to adaptively adjust the weights. The average queue size of expedited forwarding is calculated by using a low-pass filter with an exponential weighted moving average. Assuming avg is the average queue size, q is the instantaneous queue size and f1 is the low-pass filter, the average queue size of the premium service is estimated using eq. (1):

$$avg = (1 - f1) * avg + f1 * q \qquad \text{eq. (1)}$$

f1 is set to 0.01 in our algorithm to reduce fluctuation in q resulting in better delay jitter. To adaptively calibrate the weight of expedited forwarding, two thresholds (minimum and maximum) are introduced.

The minimum threshold represents the desired queueing delay, and the maximum threshold represents the acceptable queueing delay. By keeping average queue size below the maximum threshold, a low queueing delay is achieved. To accomplish this, the weight of premium service should be proportionally increased once the average queue size exceeds the minimum threshold. However the weight of premium service cannot exceed an upper limit after the average queue size re$A = \pi r^2$aches maxth otherwise, the proposed algorithm would temporarily degrade. In our proposed scheme, there is a linear relationship between the weight of expedited forwarding and the average queue size. Assume the original weight of expedited forwarding is wp, then the weight function of expedited forwarding is given by eq. (2) :

$$f(avg) = \begin{cases} wp & avg \in (0,0.5) \\ \frac{(upper-wp)*(avg-minth)}{maxth-minth} & avg \in (0.5,2) \\ upper & avg \in (2,full) \end{cases} \qquad \text{eq. (2)}$$

where *upper* is the upper limit that the weight of expedited forwarding can reach, and *avg* is the average queue size of expedited forwarding. We suggest the upper limit of expedited forwarding to be set to 0.7, and the rest of weight to be used by assured-forwarding (AF) and best-effort services. Since the total weight for a shared link is fixed, the increase of premium service's weight must cause the same amount of decrease in the best-effort's weight or AF's weight. The rule we applied here is: first shift the weight of best-effort to premium service, and if this is not enough and the weight of premium service has not reached its upper limit, then part of AF's weight will be shifted to premium service. However, once the average queue size of premium service backs down below maxth the weights taken from best-effort or AF will be returned. To meet the requirement of no or a very small queueing delay of the premium service, we set the minimum threshold to 0.5 and the maximum threshold to 2 and measured the packets instead of bytes.

IV. SIMULATION RESULTS

Our simulations are done in Omnet++ version 4.2.2 using INET 2.1.1. The IDE also allows you to configure the build, start the build process, launch simulations, and debug the model without leaving the IDE.

Here we use a simple yet sufficient network topology to suit our requirements. The network is as shown in Fig. 1. Each end host that is both source and destination hosts are connected to its respective edge routers. Edge routers are where traffic conditioning and shaping is done. The edge routers are connected via two core routers R2 and R3. The link capacity and the one-way propagation delay between an end-host and an edge router are 500Mbps and 0.5 ms respectively. However, the bandwidth and the link delay between routers are set to 3 Mbps and 10 ms, respectively.

We now present the results obtained. The goal of our simulation is to evaluate the adaptive weighted round robin in terms of packet loss rate and end to end delay. Fig. 2 shows the run time environment of the simulator. Fig. 3shows the end to end delays of the different destinations with respect to their applications. We can see that the end to end delay [1] of host 5 has the least end to end delay and host 6 has the greatest end to end delay. Fig. 4 shows the queuing time of the three behaviour aggregates. BE has the highest queuing time. Fig. 5 shows the queue lengths of different behaviour aggregates namely AF, EF and BE. We can see that the queue length of BE is highest as we have given least priority and that of EF is shortest as it is given higher priority.
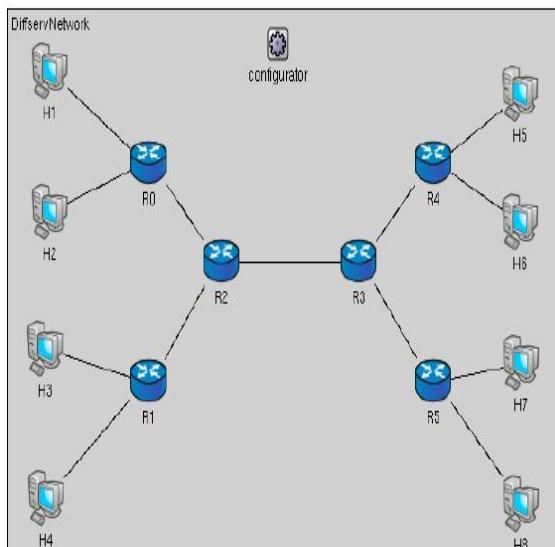
Fig. 1. Network Topology
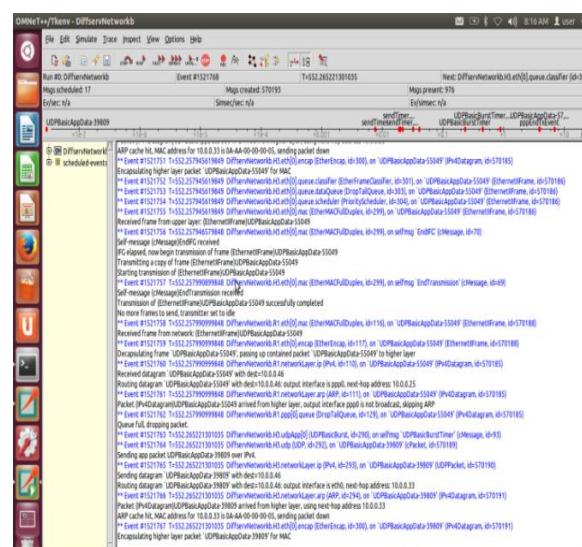
Fig. 2. Run time environment of Simulator
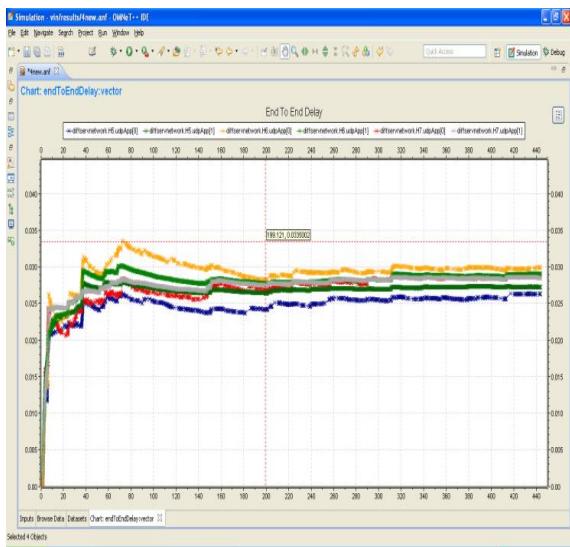
Fig. 3. End To End Delay
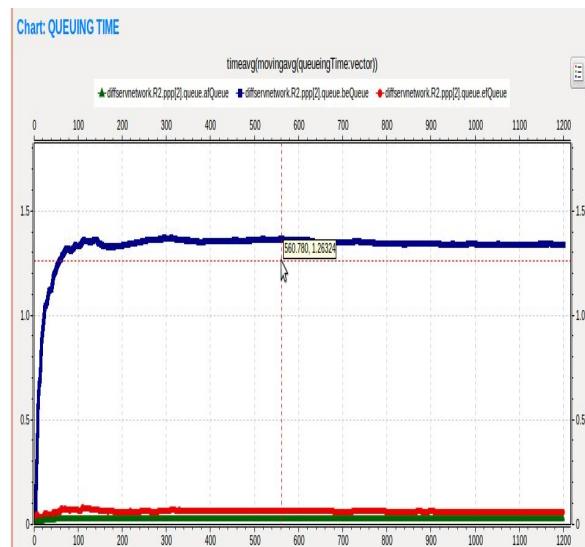
Fig. 4. Queuing Time
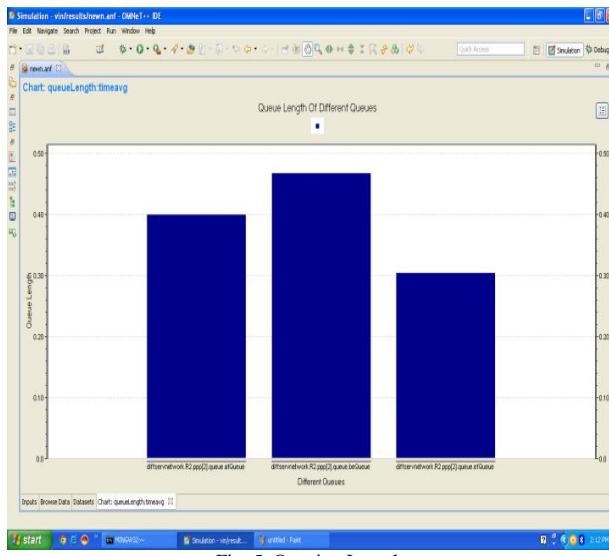
Fig. 5. Queuing Length

V. CONCLUSION AND FUTURE WORK

We have proposed and implemented an efficient packet scheduling algorithm for supporting expedited forwarding. Here the scheduling weights of behaviour aggregates are dynamically adapted to the changes in the average queue size of expedited forwarding that is premium service. The traffic distortion inside the network can be contained without disturbing the delay or delay jitter. We also achieve rigid admission control and accurate traffic conditioning. Finally we have shown that our algorithm achieves low delay, low loss rate and low delay jitter for expedited forwarding. In future, we can try to provide better performance for best effort.

REFERENCES.

1. V. Jacobson, K. Nichols, and K. Poduri, "An Expedited Forwarding PHB",*Internet Draft*, June 1999
2. K. Nichols, V. Jacobson, and L. Zhang, "An Approach to Service Allocationin the Internet", *Internet Draft*, November 1997.
3. Y. Bernet *et al.*, "A Framework for Differentiated Services", *IETF InternetDraft, February, 1999.*
4. L.Zhang. S. Deering. D. Estrin. S.Shenker. and D. Zappala, "RSVP:A New Resource Reservation Protocol", IEEE Network, 7(5):8-18,September.
5. S. Blake et al., "An Architecture for Differentiated Services", RFC 2475, December 1998.
6. Z. Zhang, Z. Duan, L. Gao, and Y. T. Hou, "Decoupling QoS Control from Core Routers: A Novel Bandwidth Broker Architecture for Scalable Support of Guaranteed Services" *Proceedings of ACM SIGCOMM'2000*, September, 2000
7. S. Floyd and V. Jacobson, "Random Early Detection gateways for CongestionAvoidance", *IEEE/ACM Transactions on Networking*, Vol. 1, No.4, August 1993
8. A. Parekh and R. Gallager, .A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks: the Single Node Case,. *IEEE/ACM Trans. on Networking*, 1(3):344-357, June 1993.

BIOGRAPHY

Anusha N is a M.TECH post-graduate student in Electronics and Communication Department, The Oxford College Of Engineering, Bangalore, Karnataka, India.. She received Bachelor Of Engineering (BE) degree in 2012 from YDIT, Bangalore, Karnataka, India. Her research interests are Computer Networks , Algorithms, Multimedia communication etc.

Mrs. Reema Sharma is Pursuing PhD in Multimedia Communication from VTU Belgaum, Karnataka, India. Also is a Assistant Professor in Electronics and Communication Department, The Oxford College Of Engineering, Bangalore, Karnataka, India. She has received Master's Degree in Communication Engineering from National Institute Of Technology Kurukshetra. She is a Member of IEEE Comsoc Society. Also M.Tech Project Coordinator in department of ECE at The Oxford College Of Engineering.