Token Based Packet Loss Control Mechanism for Networks

Pathan Uddandu[1], Sayeed Yasin[2]

[1]*M. Tech, Nimra College of Engineering & Technology, Vijayawada, A.P., India.*
[2]*Asst. Professor, Dept. of CSE, Nimra College of Engineering & Technology, Vijayawada, A.P., India.*

ABSTRACT**:** *Modern IP network services provide for the simultaneous digital transmission of data, voice, and video. These services require congestion control algorithms and protocols which can solve the packet loss parameter can be kept under control. Congestion control is therefore, the cornerstone of the packet switching networks. It should prevent congestion collapse, provide fairness to competing flows and optimize transport performance indexes such as throughput, loss and delay. In this paper we propose a congestion control mechanism with application to Packet Loss in networks with P2P traffic is proposed. In this new method the edge and the core routers will write a measure of the quality of service guaranteed by the router by writing a digital number in the Option Field of the datagram of the network packet. This is called as "token". The token is read by the path routers and then interpreted as its value will give a measure of the congestion especially at the edge routers. Based on the token number, the edge router at the source's edge point will shape the traffic generated by the source, thus reducing the congestion on the path.*

KEYWORDS: *Congestion, CSFQ, STLCC, Token.*

I. INTRODUCTION

There are a number of very good reasons to avoid loss in today's computer networks. Many of these stem from the fact that the loss is often a symptom of overflowing router buffers in the network, which can also lead to high latency, jitter, and poor fairness. In the last few years considerable effort has been expended on the design and implementation of the packet switching networks [1] [2]. A principle reason for developing such packet networks has been to facilitate the sharing of computer resources. In this paper, we study whether the benefits of a network architecture that embraces rather than avoids widespread packet loss outweigh the potential loss in efficiency of the network. We propose an alternative approach to Internet congestion control called as decongestion control.

In a departure from conventional approaches, end hosts strive to transmit packets faster than the network can deliver them, leveraging end-to-end erasure coding and in-network fairness enforcement. In this paper we present a protocol design and philosophy that supports the sharing of resources that exist in various packet switching networks. After a brief introduction to inter network protocol issues, we describe the function of a gateway as an interface between the network and discuss its role in the protocol [3][4]. We then consider the various details of the proposed work, including addressing, formatting, buffering, sequencing, flow control, error control, and so forth.

A typical packet switching network is composed of a set of computer resources called as hosts, a set of one or more packet switches, and a collection of communication media that interconnect the packet switches. The ensemble of packet switches and communication media is called as packet switching subnet as shown in Figure 1. In a typical packet switching subnet, data of a fixed maximum size are accepted from a source node, together with a formatted destination address which is used to route the data in a store and forward fashion.

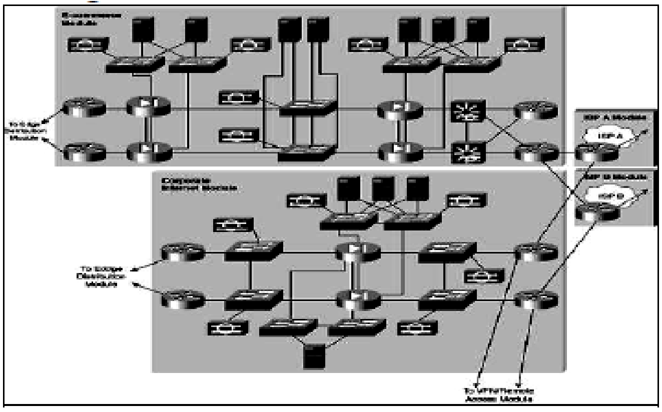
ISP A Module
ISP A
ISP B Module
ISP B
Corporate Internet Module
To Edge Distribution Module
To VPN/Remote Access Module

Figure 1: packet switching communications at network Edge

II. RELATED WORK

The basic idea of peer- to- peer network is to have peers participate in an application level overlay network and operate as both A number of approaches for queue management at Internet gateways have been studied previously. Droptail gateways are used almost universally in the current Internet because of their simplicity. A droptail gateway drops an incoming packet only when the buffer becomes full, thus providing congestion notification to protocols like TCP. While simple to implement, it distributes losses among the flows arbitrarily[5]. This often results in the bursty losses from a single TCP connection, thereby reducing its window sharply. Thus, the flow rate and consequently the throughput for that flow drops. Tail dropping also results in multiple connections simultaneously suffering from losses leading to global synchronization [6]. Random early detection(RED) addresses some of the drawbacks of droptail gateways. The RED gateway drops incoming packets with a dynamically computed probability when the exponential weighted moving average queue size avg q exceeds a threshold.

In [6], the author does per-flow accounting maintaining only a single queue. It suggests changes to the RED algorithm to ensure fairness and to penalize the misbehaving flows. It puts a maximum limit on the number of packets a flow can have in the queue. Besides it also maintains the per flow queue occupancy. Drop or accept decision for an incoming packet is then based on the average queue length and the state of that flow. It also keeps track of the flows which consistently violate the limit requirement by maintaining a per-flow variable called as strike and penalizes those flows which have a high value for strike. It is intended that this variable will become high for non- adaptive flows and so they will be penalized aggressively. It has been shown through simulations [7] that FRED fails to ensure the fairness in many cases. CHOKe [8] is an extension to RED protocol. It does not maintain any per flow state and works on the good heuristic that a flow sending at a high rate is likely to have more packets in the queue during the time of the congestion. It decides to drop a packet during congestion if in a random toss, it finds another packet of the same flow. In [9], the authors establish how rate guarantees can be provided by simply using buffer management. They show that the buffer management approach is indeed capable of providing reasonably accurate rate guarantees and the fair distribution of excess resources.

III. PROPOSED WORK

In the proposed work, a model called the Terminal Dependent Congestion Control case which is a best-effort service in the Internet that was originally designed for a cooperative environment which is the congestion control but still it is mainly dependent on the TCP congestion control algorithm at terminals, supplemented with load shedding at congestion links is shown in Figure 2. In high speed networks Core Stateless Fair Queuing (CSFQ) is enhanced to fairness

set up an open- loop control system at the network layer, which inserts the label of the flow arrival rate onto the packet header at edge routers and drops the packet at core routers based on the rate label if congestion happens. At the core routers CSFQ is the first to achieve approximate fair bandwidth allocation among flows with O (1) complexity.

CSFQ can provide fairness to competing flows in the networks with P2P traffic, but unfortunately it is not what end-users really want. By an end user Token Based Congestion Control (TBCC) restricts the total token resource consumed. It cannot obtain extra bandwidth resources when TBCC is used so no matter how many connections the end user has set up. The Self Verifying CSFQ tries to expand the CSFQ across the domain border. It randomly selects a flow, and then re-estimates the flow’s rate, and checks whether the re-estimated rate is consistent with the label on the flow’s packet. Consequently Self-Verifying CSFQ will put a heavy load on the border router and makes the weighted CSFQ null as well as void.

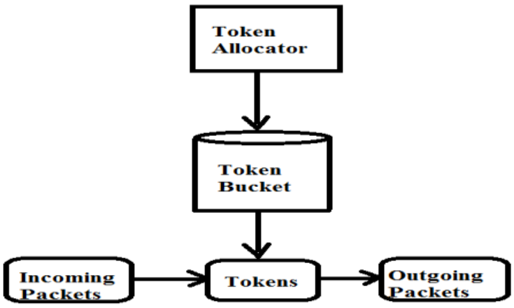

Figure 2: Packet Loss Control

The congestion control architecture re- feedback, which aims to provide the fixed cost to end-users and bulk inter-domain congestion charging to network operators. Re- feedback not only demands very high level complexity to identify the malignant end user, but also is difficult to provide the fixed congestion charging to the inter domain interconnection with low complexity. There are three types of inter domain interconnection polices: the Internet Exchange Points, the private peering and the transit. In the private peering polices, the Sender Keep All (SKA) peering arrangements are those in which the traffic is exchanged between two domains without mutual charge. As Re-feedback is based on the congestion charges to the peer domain, it is difficult for re- feedback to support the requirements of SKA.

The modules of the proposed work are:

- NETWORK CONGESTION
- STABLE TOKEN LIMIT CONGESTION CONTROL (STLCC)
- TOKEN
- CORE ROUTER
- EDGE ROUTER

Network Congestion: Congestion occurs when the number of packets being transmitted through the network crosses the packet handling capacity of the network. Congestion control aims to keep number of packets below the level at which performance falls off dramatically.

Stable Token Limit Congestion Control (STLCC): STLCC is able to shape output and input traffic at the inter domain link with O(1) complexity. STLCC produces a congestion index, pushes the packet loss to the network edge and improves the overall network performance. To solve the oscillation problem, the Stable Token-Limited Congestion Control (STLCC) is also introduced. It integrates the algorithms of XCP and TLCC [10] altogether. In STLCC, the output rate of the sender is controlled using the algorithm of XCP, so there is almost no packet lost at the congested link. At the same time, the edge router allocates all the access token resources to the incoming flows equally. When congestion happens, the incoming token rate increases at the core router, and then the congestion level of the congested link will also increased as well. Thus STLCC can measure the congestion level analytically, and then allocate network resources according to the access link, and further keep the congestion control system stable.

Token: A new and better mechanism for the congestion control with application to Packet Loss in networks with P2P traffic is proposed. In this new method the edge and the core routers will write a measure of the quality of service guaranteed by the router by writing the digital number in the Option Field of the datagram of the packet. This is called as token. The token is read by the path routers and then interpreted as its value will give a measure of the congestion especially at the edge routers. Based on the token number, the edge router at the source, thus reducing the congestion on the path.

Core Router: A core router is a router designed to operate in the Internet Backbone (or core). To fulfill this role, a router must be able to support multiple telecommunications interfaces of the highest speed in use in the core Internet and must be able to forward the IP packets at full speed on all of them. It must also support the routing protocols being used in the backbone. A core router is distinct from the edge routers.

Edge Router: Edge routers sit at the edge of a backbone network and connect to the core routers. The token is read by the path routers and then interpret as its value will give a measure of the congestion especially at the edge routers. Based on the token number of the edge router at the source, thus reducing the congestion on the path.

IV. CONCLUSION

In this paper the architecture of Token based Congestion Control (TBCC) provides fair bandwidth allocation to end users in the same domain is introduced. The two congestion control algorithms CSFQ and TBCC are elevated in this proposed work. STLCC is accessible and the simulation is designed to demonstrate its validity. The Unified Congestion Control Model which is the abstract model of the CSFQ, Re-feedback and STLCC. The simple version of the STLCC is introduced and can be deployed on the current Internet. The inter domain router is added to the TBCC system as the two TBCC domains are inter-connected.

REFERENCES

[1]. G. Appenzeller, N. McKeown, J. Sommers, and P. Barford, “Recent Results on Sizing Router Buffers,” in Proceedings of the Network Systems Design Conference, Oct. 2004.

[2]. M. Enachescu, Y. Ganjali, A. Goel, N. McKeown, and T. Roughgarden, “Part III: Routers with very small buffers,” ACM/SIGCOMM Computer Communication Review, vol. 35, pp. 83- 90, July 2005.

[3]. L. Zhang, S. Shenker, and D. Clark, “Observations on the dynamics of a congestion control algorithm: The effects of two-way traffic,” in Proceeedings of ACM SIGCOMM, pp. 133–147, Sept. 1991. [3] F. R. E. Dell, “Features of a proposed synchronous data network,” in Proc. 2nd Symp. Problems in the Optimization of Data Communications Systems, 1971, pp. 50—57.

[4]. R. A. Scantlebury and P. T. Wilkinson, “The design of a switching system to allow remote access to computer services by other computers and terminal

[5]. devices,” in Proc. 2nd Symp. Problems in the Optimization of Data Communications Systems, 1971, pp. 160-167.

[6]. F. Dabek, M. F. Kaashoek, D. Karger, R. Morris, and I. Stoica. Widearea

[7]. cooperative storage with CFS. In Proceedings of the 18th ACM Symposium on Operating Systems Principles (SOSP ’01), Chateau Lake Louise, Banff, Canada, October 2001.

[8]. Das and R. Srikant. Diffusion approximations for a single node accessed by congestion-controlled sources. IEEE Transactions on Automatic Control, 45(10):1783–1799, October 1998. [11] G. de Veciana and X. Yang. Fairness, incentives and

[9]. Sally Floyd, Van Jacobson, Link-sharing and Resource Management Models for Packet Networks, IEEE\ACM Transactions on Networking, Vol.3, No.4, 1995.

[10]. John Nagle, RFC896 congestion collapse, January 1984.

[11]. S. H. Low and D. E. Lapsey, “Optimization flow control—I: Basic algorithms and convergence,” IEEE/ACM Trans. Networking, vol. 7, pp. 861–874, Dec. 1999.

[12]. Dina Katabi, Mark Handley, and Charles Rohrs, "Internet Congestion Control for Future High Bandwidth-Delay Product Environments." ACM Sigcomm 2002, August 2002.