Congestion Avoidance and Control Mechanism Using Border Gateway Protocol

Akram H. Shaikh[1], **Bharat K. Chaudhary**[2]

[1]ME (Pursuing) Department of Computer Engineering, Dr. Pdm. V. B. Kolte College of Engineering,
Malkapur Dist-Buldhana Maharashtra India

[2]Assistant Professor, Department of Computer Engineering
Dr. Pdm. V. B. Kolte College of Engineering,

Abstract: ***In the Network for the proper functioning, congestion free network communication is more important issue. In the congestion period data packets loss and delay in sending and receiving the data packets from undelivered data packets are experienced. In this view the traffic on one network is controlled as per the requirement on the other network capacity, which uses the feedback in-between these two networks. By incorporating border routers (ingress router, egress router) on each network, which is connected each other via router, the congestion free environment is maintained in the network.***

Keywords: Border Gateway Protocol; Leaky Bucket Mechanism; Feedback control Mechanism; Rate Control Mechanism;

1. Introduction

As per analysis the existing system having the TCP/IP protocols uses the TCP Vegas and TCP Reno to avoid the congestion. As the data packets from the network are send to the other network via router, router having the buffer with some capacity after that packet drop and congestion occurred at router and data loss experienced. To avoid the same in the network here mechanism for congestion control "Network Border Patrol" is proposed to solve the congestion in network. To avoid the congestion in the network the existing heterogeneous networks are connected via router, so to avoid congestion why not the data packets send on the router are send via mechanism in router then in router passes to router which passes to other network where out router is implemented via forward feedback from in router and backward feedback from out router which helps in maintain the flow of data packets between two networks and prevent from data loss and congestion in network respectively. Packets send through source module are accepted by the in router module and pass to the router module keeping in sense the feedback received from the egress router and flow and rate controlled by using the Leaky Bucket Algorithm and Rate Control Algorithm respectively, As router module passed the packets to out router module which is already get congestion free and with proper bandwidth from the feedback send by the egress router to ingress router and the same time out router module pass the packet to destination. Here implemented BGP is responsible to deliver the data at destination without any loss in packet and with max-min available bandwidth [5]. Destination module uses the sliding window algorithm to maintain fair bandwidth allocation to the incoming packets. Here forward feedback and backward feedback mechanism is more important to prevent the network from congestion control in the network.

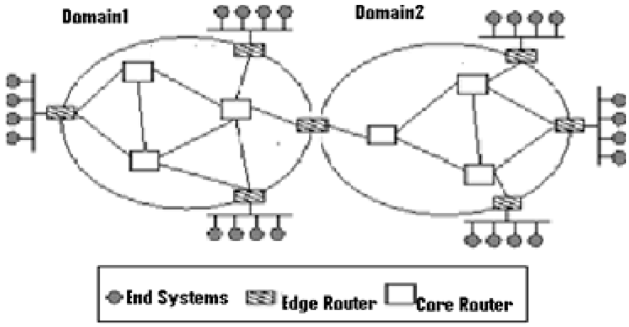


Figure 1: The core-stateless internet architecture assumed by BGP

2. Existing System

As a result of its strict adherence to end-to-end congestion control, the current Internet suffers from two maladies: Congestion collapse from undelivered packets and unfair allocations of bandwidth between competing traffic flows. The first malady congestion collapse from undelivered packets arises when packets that are dropped before reaching their ultimate continually consume bandwidth destinations. The second malady unfair bandwidth allocation to competing network flows arises in the Internet for a variety of reasons, one of which is the existence of applications that do not respond properly to congestion. Adaptive application (e.g., TCP-based applications) that respond to congestion by rapidly reducing their transmission rate are likely to receive unfairly small bandwidth allocations when competing with unresponsive applications. The Internet protocols themselves can also introduce unfairness. The TCP algorithm, for instance, inherently causes each TCP flow to receive a bandwidth that is inversely proportional to its round-trip time. Hence, TCP connections with short round-trip times may receive unfairly large allocations of network bandwidth when compared to connections with longer round-trip times. The impact of emerging streaming media traffic on traditional data traffic is of growing concern in the Internet community. Streaming media traffic is unresponsive to the congestion in a network, and it can aggravate congestion collapse and unfair bandwidth allocation.

3. Proposed System

To address the maladies of congestion collapse we introduce and investigate a novel Internet traffic control protocol called Congestion Free Router (CFR). The basic principle of CFR is to compare, at the borders of a network, the rates at which packets from each application flow are entering and leaving the network. If a flow's packets are entering the network faster than they are leaving it, then the network is likely buffering or, worse yet, discarding the flow's packets. In other words, the network is receiving more packets than it is capable of handling. CFR prevents this scenario by "patrolling" the network's borders, ensuring that each flow's packets do not enter the network at a rate greater than they are able to leave the network. Patrolling prevents congestion collapse from undelivered packets; because unresponsive flow's otherwise undeliverable packets never enter the network in the first place.

3.1 Rate Control Mechanism

In this paper, we consider the rate control problem with the objective of maximizing the total user utility. It takes into account the possible differences in user requirements, and also provides a framework for achieving a wide range of fairness objectives. We propose a simple algorithm for achieving the optimal rates for this problem. The algorithm can be implemented in a distributed way and does not require the network to know the user utility functions. In our algorithm, the network communicates to the user the number of congested links on the user's path, and the user (end-host) adjusts its rate accordingly, taking into account its utility function and the network congestion feedback. We show through analysis and experimentation that our algorithm converges to the optimum rates. Effective rate control of *elastic* traffic sources is required in order to control congestion in a communication network. Elastic traffic sources are those which do not require axed rate of service and can adjust their transmission rates based on the congestion level of the network. Examples of elastic traffic sources include internet traffic sources using TCP and sources using ABR service in ATM networks. A rate control strategy should ensure that the network is used efficiently, while guaranteeing that the traffic offered to the network is such that the congestion at the network resources remains within an acceptable level. Besides these, it is also desirable that the rate control algorithm would ensure that the available network resources are shared by the competing streams of traffic in some fair manner. There can be many different measures of fairness, one of the most well-known being max-min fairness [1]. Most of the notions of fairness explored in the literature treat all users equally. The differences in rate allocations are only due to the different path bandwidths and processing capability limitations. However, users in general have widely varying bandwidth requirements, and therefore it is desirable that any fair rate allocation scheme would take into account this heterogeneity in user requirements.

3.2 Leaky Bucket Mechanism

The leaky bucket algorithm is used to regulate the traffic flow from the input port to the output port. We assume leaky bucket as a bucket with a small hole at the bottom. Hence, the bucket at a controlled rate from the hole at the bottom. Also we assume that the limit of the bucket is infinity. Hence there is no case of bucket getting filled and the packets getting lost due to the limit of the bucket enter the network. Linking these two functions together are the feedback packets exchanged between ingress and egress routers; ingress routers send egress routers forward feedback packets to inform them about the flows that are being rate controlled, and egress routers send ingress routers backward Feedback packets to inform them about the rates at which each flow's packets are leaving the network.

3.3 Feedback Control Mechanism

The feedback control algorithm in BGP determines how and when feedback packets are exchanged between edge routers. Feedback packets take the form of ICMP packets and are necessary in BGP for three reasons. First, forward feedback packets allow egress routers to discover which ingress routers are acting as sources for each of the flows they are monitoring. Second, backward feedback packets allow egress routers to communicate per-flow bit rates to ingress routers. Third, forward and backward feedback packets allow ingress routers to detect incipient network congestion by monitoring edge-to-edge round-trip times. The contents of BGP feedback packets are shown in Figure 4. Contained within the forward feedback packet are a time stamp and a list of flow specifications3 for flows originating at the ingress router. The time stamp is used to calculate the round trip time between two edge routers, and the list of flow specifications indicates to an egress router the identities of active flows originating at the ingress router. (An edge router adds a flow to its list of active flows whenever a packet from a new flow arrives; it removes a flow when the flow becomes inactive.) In the event that the net- work's maximum transmission unit size is not sufficient to hold an entire list of flow specifications, multiple forward feedback packets are used. When an egress router receives a forward feedback packet, it immediately generates a backward feedback packet and returns it to the ingress router. Contained within the backward feedback packet are the forward feedback packet's original time stamp, a router hop count, and a list of observed bit rates, called *egress rates*, collected by the egress router for each flow listed in the forward feedback packet. The router hop count, which is used by the ingress router's rate control algorithm, indicates how many routers are in the path between the ingress and the egress router. The egress router determines the hop count by examining the time to live (TTL) field of arriving forward feedback packets. When the backward feedback packet arrives at the ingress router, its contents are passed to the ingress router's rate controller, which uses them to adjust the parameters of each flow's traffic shaper. In order to determine how often to generate forward feedback packets, an ingress router keeps, for each egress router, a timer which determines the frequency of forward feedback packet generation[4].

3.4 Border Gateway Protocol

BGP is an exterior gateway routing protocol (EGP) that is the standard routing protocol used to facilitate routing exchanges throughout the Internet. Our network utilizes BGP version 4 (BGPv4) to exchange routing information with directly connected members, associate members, partners and international peering partners. Members' external routing devices need to be able to make decisions about where to send traffic. If it is assumed that the members' ISP provides them with a default route (i.e. a route to use where the destination is unknown), then the routing device will need to know each of the possible destinations that are reachable via the network in order to make an effective routing decision. The list of routes available via the network is dynamic, and it is therefore necessary for the member to be able to hold the entire routing table in their routing device. BGP is a network layer congestion-avoidance protocol that is aligned with the core-stateless approach [3]. The core-stateless approach, which has recently received a great deal of research attention allows routers on the borders (or edges) of a network to perform flow classification and maintain per-flow state but does not allow routers at the core of the network to do so. As in other work on core-stateless approaches, we draw further distinction between two types of edge routers. Depending on which flow it is operating on, an edge router may be viewed as ingress or an *egress* router. An edge router operating on a flow passing into a network is called an ingress router, whereas an edge router operating on a flow passing out of a network is called an egress router. Note that a flow may pass through more than one egress (or ingress) router if the end-to-end path crosses multiple networks. BGP prevents congestion collapse through a combination of per-flow rate monitoring at egress routers and per-flow rate control at ingress routers. Rate monitoring allows an egress router to determine how rapidly each flow's packets are leaving the network, whereas rate control allows an ingress router to police the rate at which each flow's packets enter the network. Linking these two functions together are the feedback packets exchanged between ingress and egress routers; ingress routers send egress routers *forward* feedback packets to inform them about the flows that are being rate controlled, and egress routers send ingress routers *backward* feedback packets to inform them about the rates at which each flow's packets are leaving the network[4]. By matching the ingress rate and egress rate of each flow, BGP prevents congestion collapse within the network. This section describes three important aspects of the BGP mechanism: 1) the architectural components, namely, the modified edge routers, which must be present in the network; 2) the feedback control algorithm, which determines how and when information is exchanged between edge routers; and 3) the rate control algorithm, which uses the information carried in feedback packets to regulate flow transmission rates and thereby prevent congestion collapse in the network.

3.4.1. Architecture of BGP

The only components of the network that require modification by BGP are edge routers; the input ports of egress routers must be modified to perform per-flow monitoring of bit rates, and the output ports of ingress routers must be modified to perform per-flow rate control. In addition, both the ingress and the egress routers must be modified to exchange and handle BGP feedback packets. The input ports of egress routers are enhanced in BGP. Fig. 3 illustrates the architecture of an egress router's input port. Data packets sent by ingress routers arrive at the input port of the egress router and are first classified by flow [5]. Flow classification is performed by ingress routers on every arriving packet based upon a flow classification policy. An example flow classification policy is to examine the packet's source and destination *network addresses*, and to aggregate all packets arriving on an ingress router and destined to the same egress router into the same BGP flow (i.e., a macro-flow).

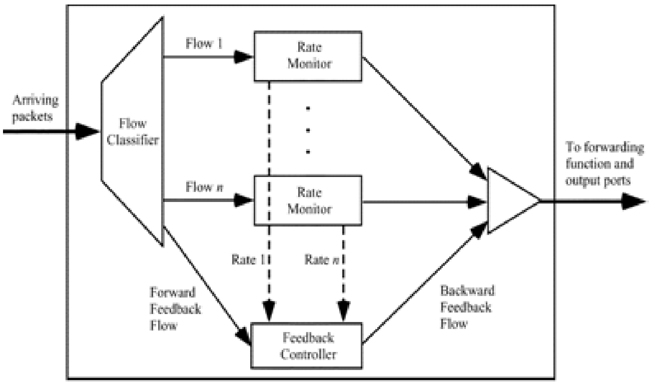
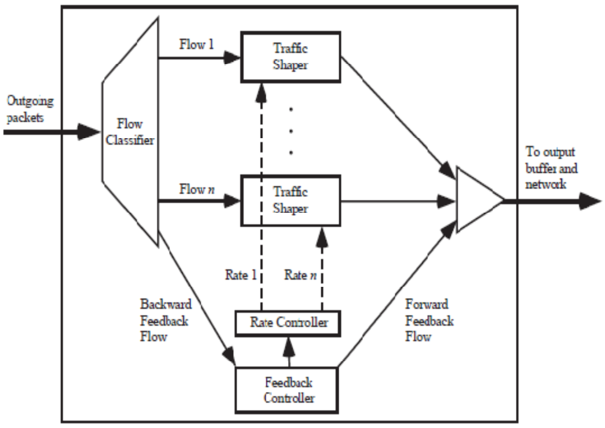


Figure 2: Input port of BGP Egress router



Figures 3: Output port of BGP ingress router

Other flow classification policies can be used, for instance, in the case of IPv6, flows may be classified by examining the packet header's flow label, whereas in the case of IPv4, it could be done by examining the packet's source and destination addresses and port numbers. After classifying packets into flows, each flow's bit rate is then rate monitored using a rate estimation algorithm such as the Time Sliding Window (TSW) algorithm [2]. These rates are collected by a feedback controller, which returns them in backward feedback packets to an ingress router whenever a forward feedback packet arrives from that ingress router. The output ports of ingress routers are also enhanced in BGP. Each output port contains a flow classifier; per-flow traffic shapers (e.g., leaky buckets), a feedback controller, and a rate controller (see Fig. 3). The flow classifier classifies packets into flows, and the traffic shapers limit the rates at which packets from individual flows enter the network. The feedback controller receives backward feedback packets returning from egress routers and passes their contents to the rate controller. It also generates forward feedback packets that are transmitted to the network's egress routers. To prevent congestion collapse, the rate controller adjusts traffic shaper parameters according to a TCP-like rate-control algorithm, and the rate-control algorithm used in BGP is described later in this section [4].

3.4.2 Implementation

The various modules in the protocol are as follows:

A. Source Module:
The task of this Module is to send the packet to the In Router.

B. In Router Module:
An edge router operating on a flow passing into a network is called an In Router. CFR prevents congestion collapse through a combination of per flow rate monitoring at Out Router and per flow rate control at In Router routers. Rate control allows an In Router to police the rate at which each flow's packets enter the network. In Router contains a flow classifier, per-flow traffic shapers (e.g., leaky buckets), a feedback controller, and a rate controller.

C. Router Module:-
The task of this Module is to accept the packet from the In Router and send it to the Out Router.

D. Out Router Module:-
An edge router operating on a flow passing out of a network is called an Out Router. CFR prevents congestion collapse through a combination of per flow rate monitoring at Out Router and per flow rate control at In Router [6]. Rate monitoring allows an Out Router to determine how rapidly each flow's packets are leaving the network. Rate monitored using a rate estimation algorithm such as the Time Sliding Window (TSW) algorithm. Out Router contains a flow classifier, Rate monitor, and a feedback controller.

E. Destination Module:-
The task of this Module is to accept the packet from the Out Router and stored in a file in the Destination machine.

4. Conclusion

1. Unlike existing internet congestion control approaches, which rely on end-to-end control, GBP is able to prevent the congestion collapse from undelivered packets.
2. GBP requires no modifications to core routers or to end systems. Buffering of packets in carried out in the edge routers rather than in the core routers.
3. The packets are sent into the network based on the capacity of the network and hence there is no possibility of any undelivered packets present in the network. Only edge routers are enhanced so that they can perform the requisite per-flow monitoring, per-flow rate control and feedback exchange operations.
4. The feedback-based traffic control mechanism, stability is an important performance concern in GBP. Fair allocation of bandwidth is ensured using the Network Border Patrol and this avoiding the congestion in the network

References

[1] Floyd and K. Fall, "Promoting the Use of End-to-End Congestion Control in the Internet," IEEE/ACM Transactions on Networking, August 1999, To appear.

[2] J. Padhye,V. Firoiu, D. Towsley, and J. Kurose, "ModelingTCP Throughput:A Simple Model and its Empirical Validation," in Proc. of ACM SIGCOMM, September 1998, pp. 303–314.

[3] B. Suter, T.V. Lakshman, D. Stiliadis, and A.Chaudhary, "Design Considerations for Supporting TCP with Per-Flow Queuing," in Proc. of IEEE Infocom'98, March 1998, pp. 299–305.

[4] B. Braden et al., "Recommendations on Queue Management and Congestion Avoidance in the Internet," RFC 2309, IETF, April 1998.

[5] Demers, S. Keshav, and S. Shenker, "Analysis and Simulation of a Fair Queueing Algorithm," in Proc. Of ACM SIGCOMM, September 1989, pp.1–12.

[6] Parekh and R. Gallager, "A Generalized Processor Sharing Approach to Flow Control – the Single Node Case," IEEE/ACM Transactions on Networking, vol. 1, no. 3, pp. 344–357, June 1993.

[7] Stoical, S. Shenker, and H. Zhang, "Core-Stateless Fair Queuing: Achieving Approximately Fair Bandwidth

[8] Parekh and R. Gallagher, "A Generalized Processor Sharing Approach to Flow Control- the single node case," IEEE/ACM Transactions on networking, vol. 1, no. 3, pp. 344-357, June 1993.

Author Profile

Akram H. Shaikh received BE degree in Computer Science & Engineering from Amravati University in 2010 during 2010 to 2013 He worked as Assistant Professor in Amravati University. He is now Pursuing the M.E. in Computer Engineering from Amravati University.

Bharat K. Chaudhary received M.E. Degree in Computer Science and Engineering from Government College of Engineering. Aurangabad. He is currently now working as a Assistant Professor in Amravati University