International Journal of Advanced Research in

Computer Science Engineering and Information Technology

Volume: 3 Issue: 2 25-Aug-2014,ISSN_NO: 2321-3337

TCP CONGESTION CONTROL OVER LOCAL AREA NETWORK USING MULTIPLE ROUTERS AND ALGORITHMS

[1] M.LAKSHMI , [2]A.SARANYA

[1]Assitant Professor, Dept.of.Computer science, ,MCC college.Pattukottai

[2]Research Scholar, Dept.of.Computer science, ,MCC college.Pattukottai

ABSTRACT− *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 applications that respond to congestion by rapidly reducing their transmission rates are likely to receive unfairly small bandwidth allocations when competing with unresponsive applications. 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.*

Keywords: TCP/IP.

I. INTRODUCTION

The negative impacts range from extreme unfairness against competing TCP traffic to the potential for congestion collapse. To promote the inclusion of end-to-end congestion control in the design of future protocols using best-effort traffic, we argue that router mechanisms are needed to identify and restrict the data flow control of best-effort flows in times of congestion. A flow that is not “TCP-friendly” is one whose long-term arrival rate exceeds that of any conformant TCP in the same circumstances. An unresponsive flow is one failing to reduce its offered load at a router in response to an increased data packet drop rate, and a disproportionate-data packet flow is one that uses considerably more bandwidth than other flows in a time of congestion.

1.1. Basic Principle of NBP:

The basic principle of NBP 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. NBP 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. This patrolling prevents congestion collapse from undelivered packets, because unresponsive flow's otherwise undeliverable packets never enter the network in the first place.

Although NBP is capable of preventing congestion collapse and improving the fairness of bandwidth allocations, these improvements do not come for free. NBP solves these problems at the expense of some additional network complexity, since routers at the border of the network are expected to monitor and control the rates of individual flows in NBP. NBP also introduces added communication overhead, since in order for an edge outer to know the rate at which its packets are leaving the network, it must exchange feedback with other edge routers. Unlike some existing Approaches trying to solve congestion collapse, however, NBP's added complexity is isolated to edge routers; routers within the core of the network do not participate in the prevention of congestion collapse. Moreover, end systems operate in total ignorance of the fact that NBP is implemented in the network, so no changes to transport protocols are necessary at end systems.

II.EXISTING SYSTEM:

Only the System is capable of preventing congestion collapse from undelivered packets. Router Does Not Support in this System. A data packet does not Work Properly.

III.PROPOSED SYSTEM:

TCP congestion control, which is implemented primarily through algorithms operating at end systems. Unfortunately, TCP congestion control also illustrates some of the shortcomings the end-to-end argument. As a result of its strict adherence to end-to-end congestion control, the current Internet suffers from main maladies: congestion collapse from undelivered packets.

End-to-end congestion control algorithms alone, however, are unable to prevent the congestion collapse and unfairness created by applications that are unresponsive to network congestion. The Internet's excellent scalability and robustness result in part from the end-to-end nature of Internet congestion control.

3.1. FEASIBILITY STUDY:

3.1.1. Technical Feasibility:

Technical Feasibility focuses on technology related issues, Practicality of available technical solution, risks involved and resources available. It also accesses that whether the available technology is mature enough to meet the system needs.

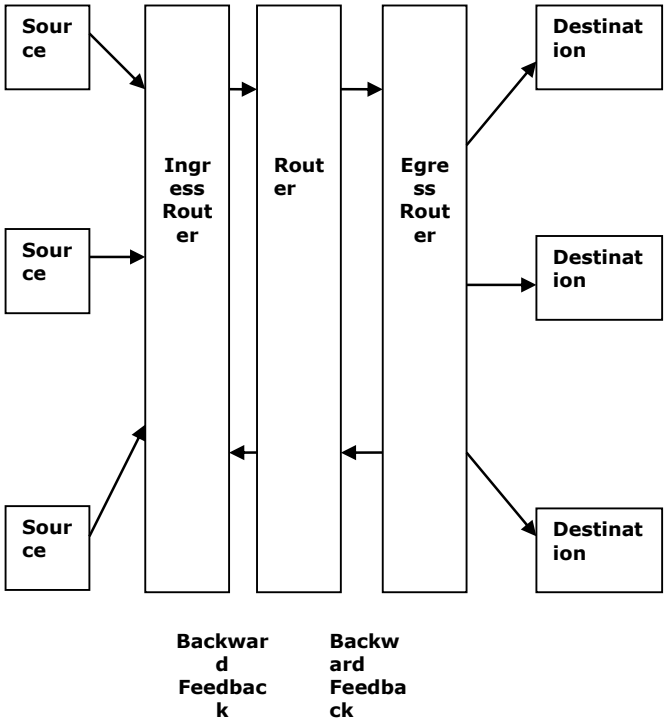
3.1.2. Operational Feasibility:

Operational Feasibility is active learning provides a way to integrate and interoperate business applications. Evaluating whether a system will work properly in the organization as well as the feedback of the end users about the problem. All the issues like performance, efficiency, providing information and services to the users, security etc. of the system are covered by operational feasibility. Usability analysis to test system interface is also performed on the proposed system's prototype and the system is sound to be operationally feasible.

3.1.3. Economic Feasibility:

Economic feasibility of the system is the measure of cost effectiveness of the system and includes cost benefit analysis, cost involved, income generated etc. while doing cost benefit analysis, fixed costs (cost of developing system) and cost

for operating the system are also taken into account. Major factors contributing to cost calculation are standards used for developing the service



oriented communication and the protocols need to implement the algorithm.

IV.MODULES

4.1.Source module:

The task of the module is to get the input from user and send the input in the form of the packets to the ingress router.

4.1.2. Ingress Router Module:

An edge Router operating on a flow passing into a network is called an Ingress Router. NBP prevents congestion collapse through a combination of per flow rate monitoring at egress router and per flow rate control at ingress router. Rate control allows an ingress router to police the rate at which each flows packet enters the network.

4.1.3.Router Module:

The task of this module is to accept the packet from the Ingress Router and send it to the Egress Router.

4.1.4. Egress Router Module:

An edge router operating on a flow passing out of a network is called an Egress Router. NBP prevents congestion collapse through a combination of per flow rate monitoring at egress router and per flow rate control at ingress router. Rate monitored using a rate estimation algorithm such as the Time sliding window (TSW).

4.1.5. Destination module:

The task of this module is to accept the packet from the Egress router and stored in a file in the Destination machine.

4.1.6. Simultaneous process:

Data and forward feedback are performing at the same time. The NBP feedback control algorithm determines how and when feedback packets are exchanged between edge routers. Feedback packets take the form of ICMP packets and are necessary in NBP for three reasons. First, they allow egress routers to discover which ingress routers are acting as sources for each of the flows they are monitoring. Second, they allow egress routers to communicate per-flow bit rates to ingress routers. Third, they allow ingress routers to detect network congestion and control their feedback generation intervals by estimating edge-to-edge round trip times.

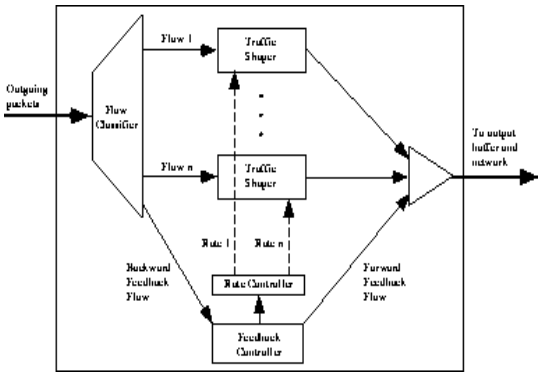
Key:

I-Ingress Router Name

R-Router Name

E-Egress Name

D-Destination Name



L-Message Length

EG-Egress Rate

Source Module:

The task of this Module is to send the packet to the Ingress router.

Input Parameters:

- Source Machine Name is retrieved from the OS.
- User types destination Machine Name.
- Message is typed by User.

Output Parameters:

- Data Packets.

Ingress Module:

The output ports of NBP ingress routers are also enhanced. Each contains a flow classifier, per-flow traffic shapers (e.g., leaky buckets), a feedback controller, and a rate controller. 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, which it periodically transmits to the network's egress routers. The rate controller adjusts traffic shaper parameters according to a TCP-like rate control algorithm, which is described later in this section.

Router

Module:

The task of this Module is to accept the packet from the Ingress router and send it to the Egress router.

Input Parameters:

- Data Packets from Ingress Machine.
- Forward feedback from the Router or Ingress Router.
- Backward feedback from the Router or Egress Router.
- Hop count

Output Parameters:

- Data Packets.
- Forward feedback.
- Incremented Hop count.
- Backward feedback

Egress Module:

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 feedback

V. CONCLUSIONS

NBP ensures at the border of the network that each flow's packets do not enter the network faster than they are able to leave it, while ECSFQ ensures, at the core of the network that flows transmitting at a rate lower than their fair share experience no congestion, i.e., low network queuing delay. This allows the transmission rate of all flows to converge to the network fair share. NBP requires no modifications to core routers or to end systems.

Only edge routers are enhanced so that they can perform the requisite per-flow monitoring, per-flow rate-control and feedback exchange operations, while ECSFQ requires a simple core-stateless modification to core routers. Simulation results show that NBP successfully prevents congestion collapse from undelivered packets. They also show that, while NBP is unable to eliminate unfairness on its own, it is able to achieve approximate global max-min fairness for competing

network flows when combined with ECSFQ, they approximate global max-min fairness in a completely core-stateless fashion.

VI. REFERENCES

[1] D. Bertsekas and R. Gallager, Data Networks, second edition, Prentice Hall, 1987

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

[3] 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.

[4] R. Jain, S. Kalyanaraman, R. Goyal, S. Fahmy, and R. Viswanathan, "ERICA Switch Algorithm: A Complete Description," ATM Forum Document 96- 1172, Traffic Management WG, August 1996

[5] W. Stevens, "TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms," RFC 2001, IETF, January 1997

[6] Lin and R. Morris, "Dynamics of Random Early Detection," in Proc. Of ACM SIGCOMM, September 1997, pp. 127–137.

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

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

[9] D. Clark and W. Fang, "Explicit Allocation of Best-Effort Packet Delivery Service," IEEE/ACM Transactions on Networking, vol. 6, no. 4, pp. 362– 373, August 1998.