Robust Rate based Congestion Control

Ganesh Gopalakrishnan {ggopalak@uiuc.edu}
Sneha K. Kasera {kasera@research.bell-labs.com}

1 Introduction

Routers need to implement congestion control measures to overcome the problem of reduced performance in the event of congestion. In this work, with a view towards robust implementation, we examine a new active queue management scheme, Acceptance and Departure Rate (ADR) that uses a combination of acceptance and departure rate measures to control link congestion in a router. Unlike many existing approaches, ADR does not use queue length measures which are not robust to changes in system capacity. The main advantage of our approach is the robustness of the system with varying network parameters including link capacity and network load.

2 Our Proposal

We use the acceptance and departure rates (ADR) to compute the fraction of input traffic that can be allowed into the system. The acceptance rate, defined as the number of bits accepted in the queue in a given time interval, provides a measure of offered load. The departure rate, defined as the number of bits transmitted on the link, a measure of processed load. A combination of both is essential in providing fast but stable response under different load conditions. Control based on the processed load provides stability and backlog clearance whereas the control based on offered load provides fast response.

In our timer-based ADR scheme, the system load (in terms of acceptance and departure rates) is measured periodically, based on which a *"fraction allowed"* (the fraction of the input traffic that can be allowed into the router queue) is computed. This fraction allowed is then used to drop packets in the next interval. In a timer-based scheme the algorithm to compute the fraction allowed need not be executed in the fast path. Only the actual packet drop decision based on the fraction allowed need be taken in the fast path. In addition to saving processing overhead in the fast path, a timer-based scheme gives the freedom to choose different and even complex policies for determining the fraction allowed without modifying the packet forwarding hardware.

In our algorithm, we periodically measure acceptance and departure rate and then normalize them with respect to line capacity. Next we compare the normalized acceptance rate with a threshold and determine the fraction allowed based on acceptance rate measure only (f_a). We also compare the normalized departure rate with another threshold and determine the fraction allowed based on departure rate measure only (f_d). The actual fraction of traffic allowed into system is chosen to be $f = \min(f_a, f_d)$. Furthermore, in order to prevent oscillations, the fraction allowed in an interval is smoothed by multiplying f with the fraction allowed in the previous interval. Once the fraction allowed is computed, we use a deterministic algorithm [Hajek 1985] for dropping packets. This deterministic algorithm demonstrates much less variability from the desired fraction allowed in comparison to other probabilistic drop mechanisms [Kasera 2001]. The randomness in packet arrivals ensures that no particular source is able to misuse the deterministic nature of the algorithm.

The timer-based ADR requires specification or determination of two timers, one associated with acceptance rate measurements and another with departure rate measurements. It also requires specification of the normalized acceptance rate and departure rate thresholds. In order to fully utilize the link, the departure rate threshold should be set close to 1. The acceptance rate threshold could be set to higher than one to allow some queuing during short bursts. The acceptance rate timer should be chosen to be less than the departure rate timer to allow the acceptance rate control to react faster to sudden bursts of load. The departure rate control should be applied over larger timer intervals. Both the timers should be set to less than the maximum queuing delay experienced in the router. The maximum queuing delay could be part of the router specification or measured from the expression B = C*D where B is the buffer size, C is the line capacity and D is the maximum queuing delay.

We use ns-2 simulator to evaluate the performance of ADR. In particular, we examine Link Utilization, Goodput, Loss and Delay. We have conducted extensive tests with varying network parameters and also compared ADR's performance with other AQM schemes. Our results show that ADR is fair and robust against varying system and network parameters like link Capacity, buffer size, network load as well as with different traffic types like TCP and UDP.

For a full description, analysis of ADR, visit http://www.uiuc.edu/~ggopalak/