Analysis and Simulation of a Fair Queueing Algorithm

Alan Demers, Srinivasan Keshav, Scott Sherker
Paper review by: **Ehsun Daroodi**
810181050

The rapid growth, in both use and size, of computer networks ha sparked a renewed interest in methods of congestion control. These methods have two points of implementation. The first is at source, where flow control algorithms vary the rate at which source sends the packets. The second is at the gateway. Congestion can be controlled at the gateway through routing and queueing algorithms. Queueing algorithms determine in which packets interact with each other which, in turn, affects the collective behavior of flow control algorithms.

Queueing algorithms can be thought as allocating three nearly independent quantities: bandwidth, promptness and buffer space. Currently the most common queueing algorithm is FSCF. FSCF essentially relegates all congestion control to the sources, since the order of arrivals completely determines the above three quantities. Congestion control algorithms should function well even in the presence of ill-behaved sources. Network with FSCF gateways doesn't have this property.

Nagle proposed an algorithm for fair queueing at the gateways. His algorithm uses different queues for different users and services them with round robin. The obvious flaw of his algorithm is that it doesn't consider the packet's length.

Pure round robin service provides a fair allocation of packet-sent but fails to guarantee a fair allocation of bandwidth because of variation in packet size. An ideal way is to use bit-by-bit round robin but that's not feasible. Instead, we emulate bit-by-bit round robin with an algorithm that assigns to each packet a timestamp and uses these quantities as sending order of packets. This order is non-preemptive. Namely, a new incoming packet never preempts a transmitting packet. Over sufficiently long conversations, this algorithm asymptotically approaches the fair allocation of bit-by-bit scheme.

Out algorithms can gives more promptness (less delay) to a user who utilizes less than its fair share of bandwidth. It is done by assigning timestamps to its packets that are less that their actual arrivals' time. This ability to provide a lower delay to lower throughput sources, completely independent of the window sizes of the sources, is one of the most important features of fair queueing.

Simulations show that: (1) fair queueing gateways by themselves don't provide adequate congestion control; they must be combined with intelligent flow control algorithms at the sources. (2) Fair queueing creates a firewall that protects well-behaved sources from their uncouth brethren. And (3) fair queueing algorithm when combined with currently available flow controls, delivers better satisfactory congestion control than formal FSCF.