Network Congestion Avoidance Through Speculative Reservation

Nan Jiang, Daniel U. Becker, George Michelogiannakis, and William J. Dally
Department of Electrical Engineering, Stanford University
{njiang37, dub, mihelog, dally}@stanford.edu

Abstract

Congestion caused by hot-spot traffic can significantly degrade the performance of a computer network. In this study, we present the Speculative Reservation Protocol (SRP), a new network congestion control mechanism that relieves the effect of hot-spot traffic in high bandwidth, low latency, lossless computer networks. Compared to existing congestion control approaches like Explicit Congestion Notification (ECN), which react to network congestion through packet marking and rate throttling, SRP takes a proactive approach of congestion avoidance. Using a light-weight endpoint reservation scheme and speculative packet transmission, SRP avoids hot-spot congestion while incurring minimal overhead. Our simulation results show that SRP responds more rapidly to the onset of severe hot-spots than ECN and has a higher network throughput on bursty network traffic. SRP also performs comparably to networks without congestion control on benign traffic patterns by reducing the latency and throughput overhead commonly associated with reservation protocols.

1. Introduction

Congestion management is an important aspect of networking systems. In a shared communication medium, the presence of network congestion has a global impact on system performance. Network congestion is created when the offered load on a channel is greater than its bandwidth. In many traditional networks, the focus is on local communication bandwidth, and the network bisection channels are heavily oversubscribed due to cost constraints [1]. In these systems, the network bottlenecks usually occur on internal network channels due to under-provisioned global bandwidth. More recently, there has been a shift towards building system networks with full bisection bandwidth as new data center and cloud computing technologies increase the demand for global network communication [3, 4, 14–16, 18]. In networks with ample bisection bandwidth, congestion occurs almost entirely at the edge of the network.

Network endpoint hot-spots can occur in a wide range of network operations. Programming models commonly used in large computer systems, such as MapReduce, can have inherent hot-spot behavior [17]. Even if network traffic is uniform and random, multiple senders may temporarily send packets to the same destination and form a transient hot-spot [6]. Traffic that cannot be serviced by the over-subscribed destination is left in the router queues, causing network congestion. In lossy network systems like TCP/IP, congestion causes packet drops, but as a result the point of congestion remain somewhat isolated. However, many system area networks, such as Infiniband [2], are designed to be lossless and use tightly-controlled buffer allocation policies such as credit-based flow control. In these systems, the congested traffic remains in the network until it is delivered. As a result, congested packets back up into the rest of the network in a condition called *tree saturation* [21]. Without proper management and isolation of these congestion effects, traffic flow in the rest of the network will be adversely affected.

Many congestion control mechanisms for networking systems have been proposed [28]. Explicit Congestion Notification (ECN) is a popular mechanism that has been adopted by many networking systems [2, 22]. While the exact implementation of ECN differs from system to system, the underlying operating principle is similar. When the network detects congestion, it signals the sources contributing to the bottleneck to throttle down. The congestion signal is sent via an explicit message or piggybacked on acknowledgment packets from the destination. ECN has been well studied in the context of system area networks, particularly the InfiniBand Architecture (IBA) [11, 13, 23], and has shown to be effective in combating congestion. However, studies have also pointed out limitations such as reduced system stability, the need for parameter adjustment, and slow congestion response time [10, 20].

In this work we introduce the Speculative Reservation Protocol (SRP), a new congestion management mechanism for system area networks. In contrast to ECN, which only reacts to congestion after it has occurred, SRP avoids congestion by using bandwidth reservation at the destinations. Contrary to the common belief that network reservation protocols incur high overhead and complexity, SRP is designed with simplicity and low overhead in mind. Unlike previous reservation systems [19, 25], SRP uses a very light-weight reservation protocol with minimal scheduling complexity. The SRP reservation schedule is a simple mechanism that prevents the over-subscription of any network destination, eliminating hot-spot congestion. Furthermore, SRP avoids the latency overhead associated with reservation protocols by allowing sources to transmit packets speculatively without reservation. Speculative packets are sent with a short time-to-live and are dropped (and retransmitted later with reservation) if network congestion begins to form.

The speculative reservation protocol advances the state of the art in congestion control in the following ways:

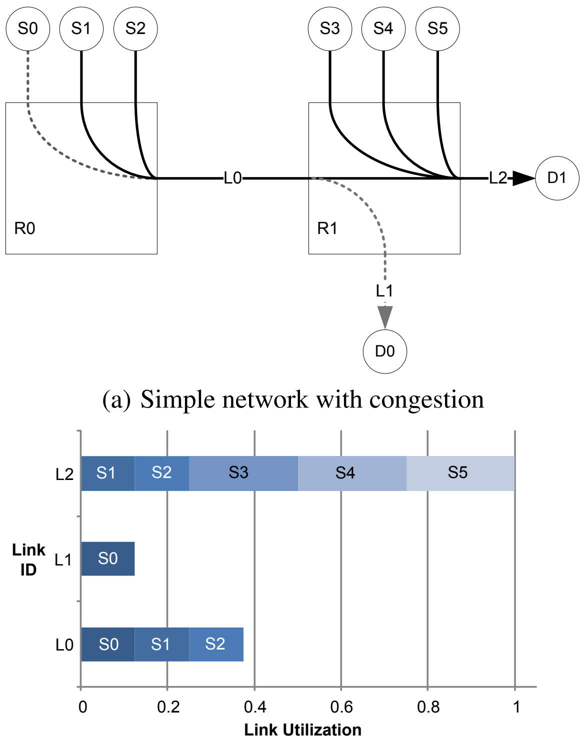
- SRP prevents the formation of congestion rather than reacting to congestion after it has already occurred.
- SRP has a very rapid transient response, reacting almost instantaneously to the onset of congestion-prone traffic, compared to the hundreds of micro-seconds it takes packet marking protocols such as ECN to respond.
- SRP has a low overhead and performs on par with networks without congestion control on benign traffic.
- SRP improves fairness between sources competing for a network hot-spot.

The remainder of the paper is organized as follows. In Section 2, we demonstrate the effect of tree saturation on networks without congestion control and describe the current solution using ECN. Section 3 describes in detail the operation of SRP. Section 4 specifies the experimental methodology used in this study. In Section 5, we present a comparison study of SRP, a baseline network, and ECN using several different test cases. In Section 6, we examine in detail the behavior and overhead of SRP. Related congestion control mechanisms are discussed in Section 7. Finally, we conclude the study in Section 8.

2. Motivation

Figure 1(a) shows a simple network congestion scenario that demonstrates the effects of tree saturation. Nodes S1 through S5 are contending for the hot-spot destination D1, whereas S0 is attempting to reach the uncongested destination D0. Each source tries to send at the maximum rate, which is equal to the rate of the links. Since S0 shares link L0 with S1 and S2, network congestion at L2 eventually backs up and affects the performance of S0 as shown in Figure 1(b). The hot-spot link L2 is at 100% utilization, with bandwidth divided between S1 through S5. Even though links L0 and L1 have spare bandwidth to support traffic from S0, this bandwidth cannot be utilized due to congested packets from S1 and S2 that are present in the input and output buffers of L0. We also note that the throughput of traffic from S1 and S2 is only half that of traffic from S3 through S5 due to the local fairness policies of the routers that grant equal throughput to each input port rather than to each traffic flow.

A congestion management algorithm used in many networking systems today is ECN. Figure 2 shows an example of the operation of ECN as implemented in Infiniband networks [2]. An ECN enabled router detects congestion by monitoring the occupancy of its input or output buffers. When a buffer's occupancy exceeds a certain threshold, the router marks the ECN field of packets passing through the buffer (in some systems the marking operation only occurs on ports identified as the root of the congestion). When the marked packet arrives at its destination, the ECN field is returned to the packet's source using a congestion notification packet. After the sender receives a notification, it incrementally reduces its transmission rate to that destination, relieving the congestion. In the absence of congestion notifications, the sender will gradually increase its injection rate to fully utilize the bandwidth of an uncongested network. To regulate the sender transmission rate, in the case of Infiniband, an inter-packet delay is added between successive packet transmissions to the same destination.



(a) Simple network with congestion

(b) Channel throughput of the congested network

Figure 1. Effect of hot-spot congestion in a network without congestion control.

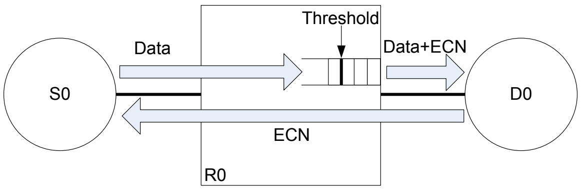
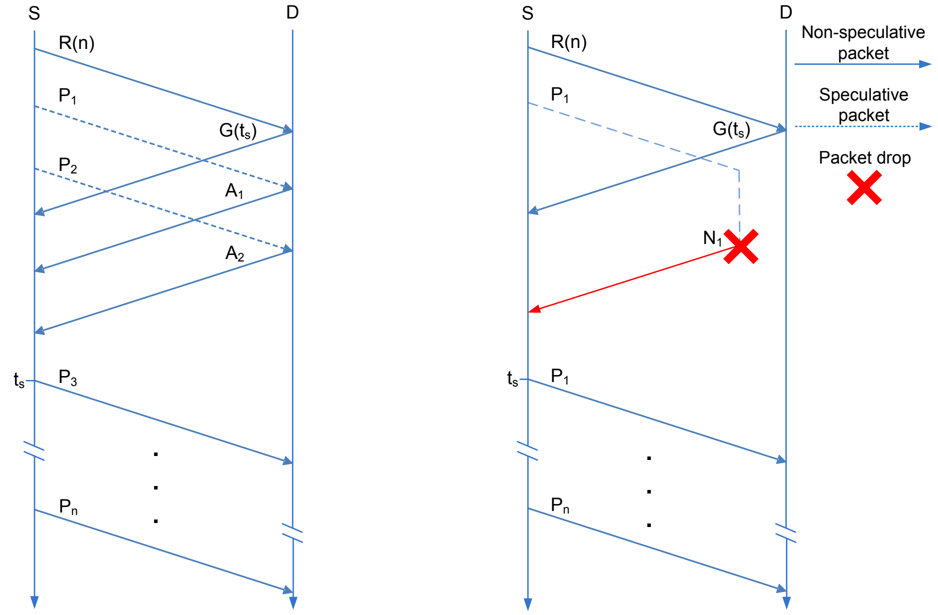


Figure 2. Operation of ECN.

With proper configuration, ECN has been shown to be effective in combating network congestion for long traffic flows [13]. However, due to the incremental nature of the algorithm, the reliance on buffer thresholds, and the round trip time of congestion information, ECN can have a slow response to the onset of congestion [10]. Furthermore, the set of parameters that regulates the behavior of the ECN algorithm needs to be carefully adjusted to avoid network instability [20].

3. Speculative Reservation Protocol

In contrast to packet marking mechanisms that react to congestion after it has occurred, SRP operates on the principle of congestion avoidance. SRP requires a reservation-grant hand-



(a) Normal network operation

(b) Congested network operation

Figure 3. Time diagrams of the operation of SRP under different network situations.

shake for communication between any sender and receiver to avoid over-subscribing the destination. To reduce the latency overhead associated with the reservation handshake, the sender can begin transmitting packets speculatively before the reservation grant returns. Speculative packets can be dropped by the network if congestion begins to form.

A time diagram of the normal operation of SRP is shown in Figure 3(a). In this scenario, no hot-spot is present in the network. The sender S initiates communication to the destination D by first issuing a reservation packet R. This reservation packet is small, has a high network priority, and travels on a separate control Virtual Channel (VC) [8] to guarantee rapid delivery. The reservation carries a reservation size, n, indicating the number of packets the source intends to transmit. The size is chosen to amortize the overhead of SRP across multiple packets while providing fairness and responsiveness to multiple flows.

After issuing the reservation, S begins speculatively sending packets P_1 and P_2 to D. These speculative packets travel on a low-priority VC and have a limited Time-to-Wait (TTW). A speculative packet is dropped by a routers if its total accumulated queuing time inside the network is greater than its TTW. We implement TTW tracking by timestamping packets upon arrival at a router's input port and then performing checks on this timestamp when the packet is at the head of the input buffer. Due to the unreliable nature of speculative packets, they require acknowledgments to notify the source whether they were successfully delivered or dropped.

Once the reservation packet arrives at D, the destination returns a small grant packet with a *starting time* payload, $G(t_s)$, based on its current reservation schedule. In addition, D updates its reservation schedule such that the next arriving reservation from any network source will be issued a starting time of no earlier than $t_s + n(1+\epsilon)\tau_p$. The constant τ_p is the time it takes to receive a single packet on the destination ejection channel. The parameter ϵ accounts for the bandwidth overhead of control packets. Aside from the reservation schedule, no other resources are reserved at the receiver beyond the normal operation of the network system.

When S receives the grant packet, it stops speculative transmission to D. After reaching time t_s, S resumes transmission to D in a non-speculative mode, starting with packet P_3 in the example in Figure 3(a). The non-speculative packets cannot be dropped and do not require acknowledgments. After S has transmitted all n packets successfully, any future transmission between S and D repeats this reservation process.

Figure 3(b) shows the time diagram illustrating SRP in a congested network with a hot-spot at node D. Initially, S behaves identically to the previous example by sending the reservation and a speculative packet. The reservation packet, having higher network priority, quickly arrives at D. The speculative packet, however, encounters a large queuing delay near D. When the queuing delay exceeds its TTW, the speculative packet is dropped by the router and a negative acknowledgment (NACK) is returned to S. In our implementation, the TTW is a fixed value based on the packet latency distribution of the network under high load uniform random traffic. The effect of using different TTW values is evaluated in Section 6.1. When S receives a NACK packet, it stops speculative transmission to the destination. S resumes packet transmission at t_s in non-speculative mode, starting with any packets that were previously dropped.

Due to the dropping protocol, out of order packet arrival is possible within each reservation. In the scenario of Figure 3(b), if the NACK packet returns after t_s, the retransmitted packet will arrive at D out of order. This can be prevented, at the cost of some bandwidth, by modifying the protocol such that after reaching t_s, S retransmits all outstanding speculative packets. This ensures in-order packet arrival at the cost of possible duplicate packets arriving at D.

SRP is designed to minimize latency and bandwidth overhead. Sending speculative packets makes the latency overhead of SRP nearly negligible. At low to medium network loads, the majority of speculative packets reach their destination, and SRP's latency is the same as that of the baseline network. Bandwidth overhead is the result of control packets and dropped speculative packets. To minimize control overhead, the reservation/grant/ACK/NACK packets are made much smaller than the data packets and the bandwidth consumed by each reservation is amortized across n data packets. The size of each reservation, n, is set by the message size subject to two reservation size limits. Messages smaller than a minimum reservation size, n_{min}, may bypass the reservation protocol to reduce overhead. Messages larger than the maximum reservation size, n_{max}, are sent using multiple successive reservations, one for each n_{max} packets. While this *chunking* is not necessary for correct operation, it prevents long messages from monopolizing a network destination which could create transient hot-spots in the network fabric.

At high network load, speculative packets are dropped more frequently due to increased queuing delays. Dropping speculative packets wastes network bandwidth and is a major source of overhead at high load. However, speculative packets never reduce the bandwidth available for non-speculative packets because they are sent on a separate, lower-priority virtual channel. Speculative drop overhead can be controlled by adjusting the speculative TTW and the reservation granularity. Bandwidth overhead behavior of SRP is explored in Section 6.1.

4. Experimental Setup

We characterize the performance and behavior of SRP using a cycle accurate network simulator based on Booksim [7]. We compare three networks: a baseline network with no congestion control, an SRP network, and a network implementing Infiniband style ECN.

The simulated network, unless otherwise specified, is a 256-node 2-level Fat Tree. 32-port routers are used on the first level, each with 16 down channels and 16 up channels. The second level of the Fat Tree uses 16-port routers with down channels only. The routers' operating frequency is set at 1GHz and the zero-load latency through each router is 26ns [24]. The network uses nearest-common-ancestor routing. A packet is first routed up the tree using randomly assigned up channels. When the packet reaches the router that is the lowest common ancestor of the source and the destination, it is routed down the tree deterministically to the destination node. Each network channel, including injection and ejection links, has a capacity of 10 Gb/s and a latency of 32ns.

The simulated routers use credit-based virtual cut-through flow control. In the baseline network, a single VC is used to transmit all network traffic. In the ECN network, a control VC is added for congestion notification packets. The SRP network has two control VCs: one is used by the reservation packets, and the other is used by the grant, ACK, and NACK packets. An additional low-priority data VC is added to SRP for the speculative packets. In both ECN and SRP networks, the control VCs have higher priority than the data VCs.

Network data packets comprise 2K bits and are partitioned into 32 64-bit flits. Network control packets consist of a single 64-bit flit. In all three networks, the main data VC input buffer implements Virtual Output Queuing (VOQ) to avoid Head-of-Line (HoL) blocking. All other VCs, including the speculative VC in the SRP network, use single FIFO input buffers. The input buffer size per VC is 16 packets. The router crossbar has a $2\times$ speedup over the network channels. Combined with VOQ, this speedup results in nearly 100% router throughput for random traffic. At the crossbar outputs, each VC has a 4-packet output buffer. Crossbar and output arbitration is performed using priority arbiters.

Two types of synthetic traffic patterns are used in the experiments. Hot-spot traffic is used to generate network congestion. We vary the over-subscription factor of the hot-spot by randomly selecting a subset of the network to transmit to the hot-spot. For benign traffic cases, Uniform Random (UR) traffic is used. A combination of these two traffic pattern is also used in some experiments. In the combined traffic pattern, a subset of nodes transmit exclusively to the hot-spot, while all other nodes are running UR traffic.

All network nodes generate traffic in units of messages. Messages range in size from a single packet to hundreds of packets, as specified for each experiment. When transmitting messages, network nodes use a mechanism similar to the Infiniband queue-pairs. The transmitting node creates a separate send queue for each destination. Similarly, receiving nodes create a separate receive queue for each source. The send queues at each node arbitrate for the injection channel in a round-robin fashion. A low-cost injection queue organization scheme has been proposed in [12]. However, the detailed Network Interface Controller (NIC) design is orthogonal to our study, and SRP can be modified to accommodate different node designs.

The congestion control parameters for ECN and SRP are listed in Table 1. Unless otherwise stated, this set of parameter values is used in all experiments. In the ECN network, if the total occupancy of the VOQ queues associated with a given output port exceeds b_{thres} and the downstream input buffer has credits available, the port is identified as the root of congestion and ECN is activated. This applies to all router output ports as well as the node ejection ports. When a node's send queue receives a congestion notification, it increases its inter-packet delay by IPD_+. If the send queue does not receive any congestion notification packets in an interval of length t_-, its inter-packet delay is decreased by IPD_-. The usage of the SRP parameters is described in Section 3.

Table 1. Configurable parameters for congestion control protocols.

Protocol	Parameter	Description	Value
SRP	ϵ	Reservation overhead adjustment	0.05
	TTW	Speculative packet Time-to-Wait	1.3μs
	n_{max}	Reservation granularity	16 packets
	n_{min}	Minimum reservation message size	4 packets
ECN	IPD_+	Inter-packet delay increment	800ns
	IPD_-	Inter-packet delay decrement	50ns
	t_-	Inter-packet delay decrement timer	2μs
	b_{thres}	Buffer threshold	90% input buffer capacity

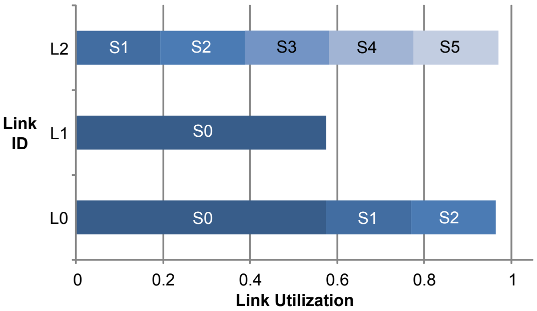


Figure 4. Speculative reservation resolving congestion on a simple network configuration.

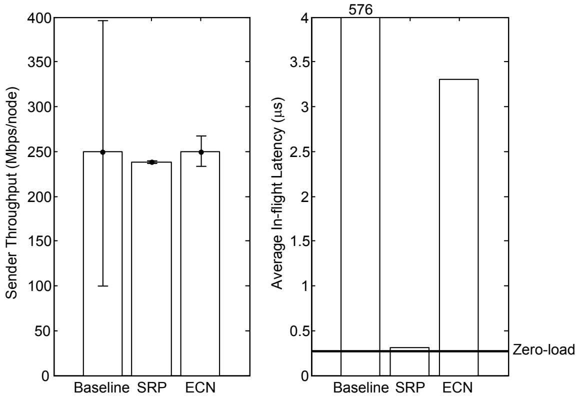


Figure 5. Network statistics for senders of a 40:1 hot-spot traffic pattern. The error bars on the throughput graph indicate the standard deviation.

5. Results

5.1. Simple Congestion Behavior

The initial assessment of SRP is performed on the simple network configuration presented in Section 2. The direction of traffic flow is shown in Figure 1(a). Figure 4 shows the throughput of data packets with SRP enabled. In contrast with the baseline, the utilization of the L0 and L1 links has increased significantly due to a 47% increase in the throughput of S0. Under SRP, the congested hot-spot packets are queued outside of the network. As a result, network resources are freed for use by packets from other flows, such as those from S0. Using SRP, the hot-spot link L2 has a 5% lower data throughput compared to the baseline due to the overhead adjustment factor.

In addition to resolving hot-spot congestion, SRP also provides improved fairness for traffic competing for the hot-spot node. Compared to the baseline network, SRP increases the throughput of traffic originating from S1 and S2 so that each of the five hot-spot senders now receives an equal share of the L2 bandwidth. Since each traffic source acquires reservations from the destination independently, an equal share of reservations are returned to each source, ensuring throughput fairness.

5.2. Hot-spot Traffic Behavior

Figure 5 shows the throughput and latency statistics of a 256-node Fat Tree network running a 40:1 hot-spot traffic over the course of 50ms simulated time. In this experiment, 40 nodes continuously transmit 8-packet messages to a single network destination. The set of nodes is selected randomly and used in all three networks. The baseline network shows a large variation in throughput for different hot-spot senders. Nodes that share the same first-level router with the hot-spot receive higher throughput than those from other parts of the network. With SRP enabled, the same set of hot-spot senders is able to acquire an equal share of reservations from the destination, resulting in very little variation in sender throughput. The average sender throughput for SRP is 5% lower than for other networks due to the overhead adjustment factor ϵ. Finally, the ECN network shows a higher variation in sender throughput than SRP, but still significantly outperforms the baseline.

In-flight packet latency is another indicator of network congestion. With a 40:1 over-subscription of the hot-spot, total packet latency (including source queuing delay) is unbounded. In-flight packet latency is the total packet latency less the source queuing delay. It correlates with the amount of network resources consumed by each packet. Figure 5 shows that the baseline network has a very high average in-flight packet latency, several thousand times that of the zero-load. This is symptomatic of tree saturation. Nearly every network queue on a path to the hot-spot is completely filled. Also, due to unfair allocation of network bandwidth, nodes closer to the hot-spot have a much lower latency than farther nodes, resulting in a latency distribution with high variance. With SRP, the average in-flight packet latency is close to the zero-load latency. This is achieved because SRP avoids congestion rather than correcting congestion. Hence, most packets traverse the network with little or no queuing delay. The average in-flight packet latency of the ECN network is much higher than that of SRP. This is because the ECN network requires some congestion to operate. No source throttling occurs unless queues periodically reach their thresholds. This is the penalty of reacting to congestion rather than avoiding it.

We further demonstrate the absence of hot-spot congestion by using the combined traffic pattern of hot-spot and background UR traffic. In these simulations we maintain the hot-spot traffic at 40:1 over-subscription and vary the rate of UR traffic injected by other nodes not participating in the hot-spot. We also simulate the networks without hot-spot traffic to establish a reference. Both hot-spot and UR traffic have a message size of eight packets. Figure 6 shows the offered vs. accepted throughput of UR and combined traffic for each network. In the baseline, the hot-spot leads to severe network congestion, causing the background uniform traffic to saturate at less than 10% network capacity. With SRP, the background uniform traffic remains stable even under very high network loads. A deeper look into the SRP network shows that at each node, the send queue to the hot-spot destination receives a very high reservation starting time that causes long stalls, while the send queues to other network destinations have a starting time that allows them to transmit immediately. This experiment shows that SRP completely eliminates the effect of the hot-spot on background traffic and represents a significant improvement over the baseline network, which cannot sustain any substantial background traffic when a hot-spot is active. The ECN network is also able to provide high background traffic throughput in the presence of a hot-spot. At steady state, the hot-spot send queues at each node become heavily throttled by congestion notifications, leaving network resources available for the background uniform traffic.

5.3. Transient Behavior

While both ECN and SRP are effective at managing long-term congestion, a key advantage of SRP, in addition to its lower in-flight latency, is its fast response to the onset of congestion. As soon as the hot-spot traffic is initiated by the senders, it becomes regulated by the reservation protocol, and any congestion in the network is completely avoided. While congestion may occur on the speculative VC, this condition is acceptable because the speculative packets have a low network priority and will quickly time out when they become blocked. We test the congestion response time of the networks using a step-function traffic pattern. In this traffic configuration, the network is warmed up with UR traffic (40% load). After 1ms simulated time, the traffic pattern is switched to a combination of background uniform (40% load) and 40:1 hot-spot traffic for the rest of the simulation. The message size for both traffic patterns is eight packets. The total packet latency and throughput of the uniform traffic is recorded each cycle before and after the transition in order to monitor the initial impact of the hot-spot. The results shown are the average of 100 simulations using different random seeds. Multiple simulations are required to get an adequate sample size for each transient point.

Figure 7 shows the network response of SRP and ECN to the step traffic pattern. The baseline result is not shown because the network becomes saturated by the hot-spot and never recovers. In the SRP network, the hot-spot onset has nearly no observable effect on the total packet latency or the throughput of the background uniform traffic. In contrast, the ECN network experiences a large latency spike and throughput reduction after the hot-spot forms. Over the course of the next millisecond, the latency and throughput of the ECN network recover to pre-transition levels. Near the end of the recovery period, the throughput curve spikes to compensate for the initial post-transition throughput deficiency.

This transient experiment demonstrates a fundamental limitation of ECN: network congestion must have occurred for ECN to become active. Packets sent during the ECN activation and source throttling period are destined to further increase congestion in the network. While ECN parameters can be modified for faster triggering on congestion, such as reducing the buffer threshold, this increases the number of false positives in traffic patterns that do not have congestion and thus can lower network throughput on benign traffic.

ECN's transient behavior improves as the hot-spot over-subscription factor is reduced. This is because there is a limit to the number of notifications that a node can generate in a given time period (one notification per packet received). With fewer hot-spot senders, every one of them receives more notifications in the same period of time, resulting in faster traffic throttling. In contrast, SRP works well across the entire range of hot-spot over-subscription factors with no need to adjust protocol parameters.

5.4. Uniform Traffic Throughput

In addition to avoiding network congestion, SRP also achieves good latency and throughput under benign traffic patterns. Figure 8 compares the latency-throughput curves of SRP and the baseline network for uniform random traffic with a message size of four packets. The latency curve of the ECN network is nearly identical to that of the baseline and is not shown here. At low loads, SRP has a negligible latency overhead compared to the baseline network. In this region of operation, most speculative packets are successfully delivered to the destination, and the reservation round-trip delay is masked by the overlapping speculative packet transmission. As network load increases, queuing delay on the speculative VC causes packet drops, and the latency effect of reservation becomes more noticeable. The SRP latency overhead peaks at 85% network load and is 25% higher than the baseline. At even higher network load, the latency overhead of SRP disappears as the baseline network begins to saturate.

In terms of network throughput, Figure 9 shows the saturation throughput of the three networks running uniform random traffic with various message sizes. In addition, we measure the networks' throughput using a combination of different message sizes. In the bimodal mode (Bi), traffic comprises equal fractions of short (4-packet) and long (64-packet) messages. In the mixture mode (Mix), the message size is uniformly distributed

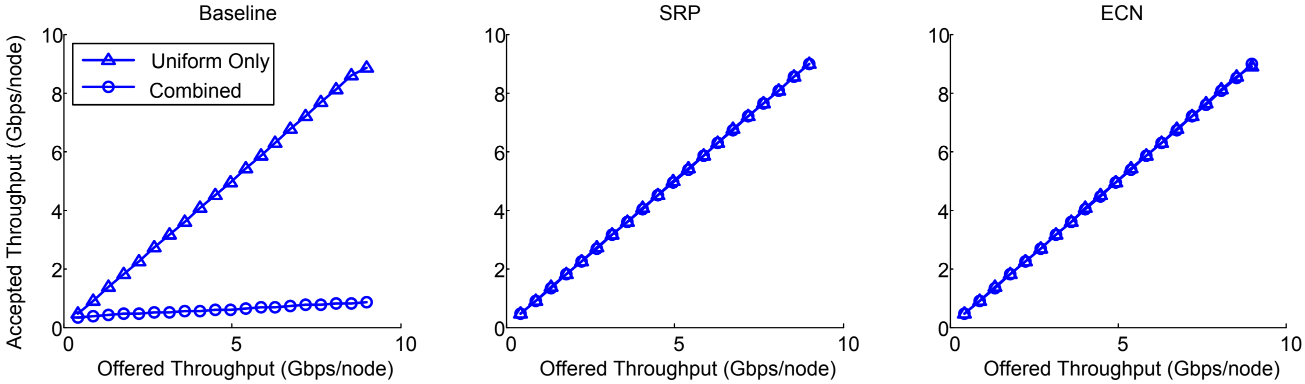
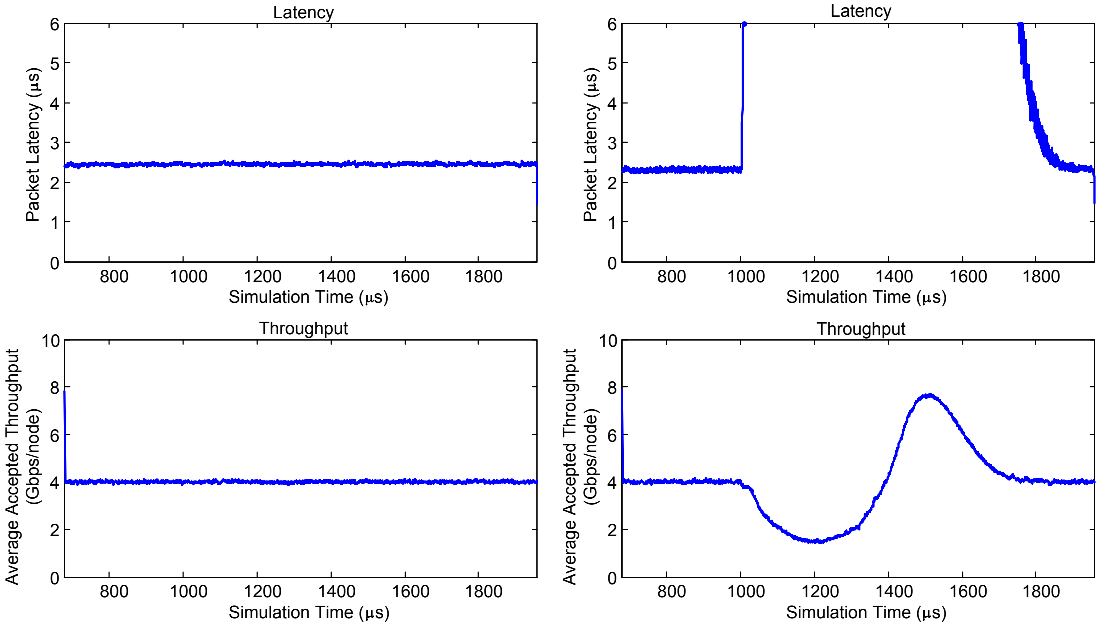


Figure 6. With SRP or ECN enabled, the offered vs. accepted throughput plot of the background traffic is nearly unaffected by the presence of a hot-spot.



(a) SRP network

(b) ECN network

Figure 7. Network response to the onset of a long-lived hot-spot.

between 4 and 64 packets. The results show that as the message size increases, the saturation throughput of the baseline network actually decreases. This is due to transient load imbalances caused by bursty traffic. When multiple large messages converge on a destination, a temporary hot-spot is formed. Without providing additional buffering, the network saturation throughput is reduced. The effect of large messages is mitigated in networks with congestion control because transient load imbalance can be resolved without additional buffers. By throttling the sources of the transient hot-spots, the network remains uncongested, allowing other packets to utilize the network resources. This is evident in the throughput bars for both SRP and ECN. As message sizes increases, the throughput of both networks remains high.

For messages smaller than the minimum reservation message size ($n_{min} = 4$ packets), the reservation protocol is not triggered, and the SRP network behaves essentially as the baseline network without any congestion control. In the simulation with a message size of four packets, the reservation protocol is activated and SRP increases saturation throughput by 6% and 3% compared to the baseline and ECN networks, respectively. For larger message sizes, SRP consistently outperforms the baseline network by eliminating all effects of transient load imbalance. For the larger message sizes, the saturation throughput of SRP essentially converges, suggesting that SRP will maintain high throughput for even larger message sizes. For both the bimodal and mixture simulations, SRP is able to maintain high network throughput. This demonstrates that the reservation scheduling of SRP is well behaved when interacting with different message sizes in the same network.

The ECN network has good performance on benign traffic with small message sizes. A properly set up ECN buffer threshold ensures that false congestion notifications are rarely generated under UR traffic even at very high network load. Therefore, for single and two-packet messages, ECN performs as well as the baseline. For larger message sizes, saturation throughput of

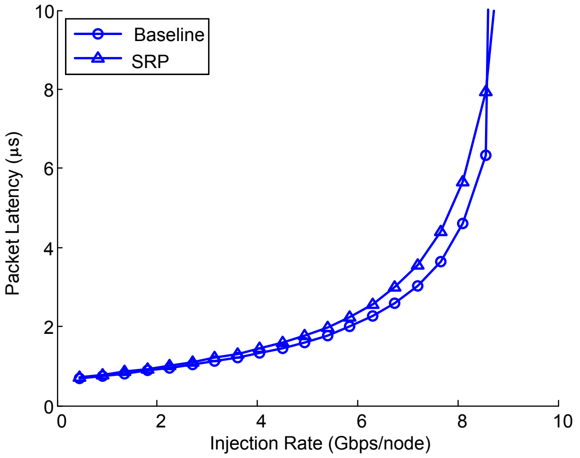


Figure 8. Latency vs. throughput plot of baseline and SRP networks under uniform random traffic with message size of 4 packets.

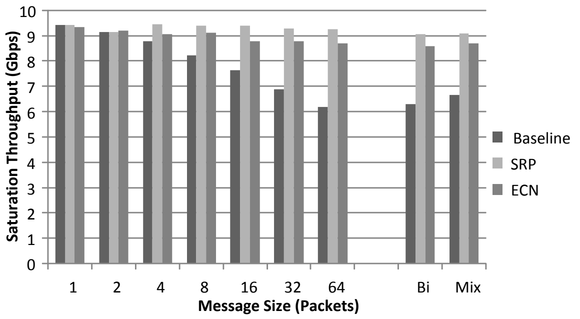


Figure 9. Network saturation throughput under uniform random traffic of different message sizes.

ECN is much higher compared to the baseline, but falls below that of SRP. With more bursty traffic, ECN cannot react as fast as SRP to transient load-imbalances, resulting in more network congestion. Furthermore, the ECN algorithm is designed such that it takes some time for a send queue to recover from a congestion notification. Some injection bandwidth is wasted during the recovery period. Similar to SRP, with increasing message size, the saturation throughput of ECN also converges.

6. Discussion

6.1. Reservation Overhead

The high performance of SRP is in part due to keeping its bandwidth and latency overhead at a negligible level. This is achieved by choosing appropriate values for the parameters in Table 1 as discussed below.

Figure 10 shows the average channel utilization of SRP and a baseline network running UR traffic with a message size of four packets. Network utilization with ECN falls in between SRP and the baseline and is not shown. Channel utilization overhead for SRP peaks after 70% network load and is consuming less than 2.5% additional bandwidth due to control and speculative packet drop overhead. The magnitude of the bandwidth overhead is a function of the network message size, highlighting the need for a minimum reservation message size, n_{min}. If the network were to use SRP for single packet messages, the maximum bandwidth overhead would increase to 20%. This would cause the SRP network to saturate earlier than the baseline for single packet messages. To avoid this high overhead, we introduce the minimum reservation message size and allow small messages to ignore the reservation protocol.

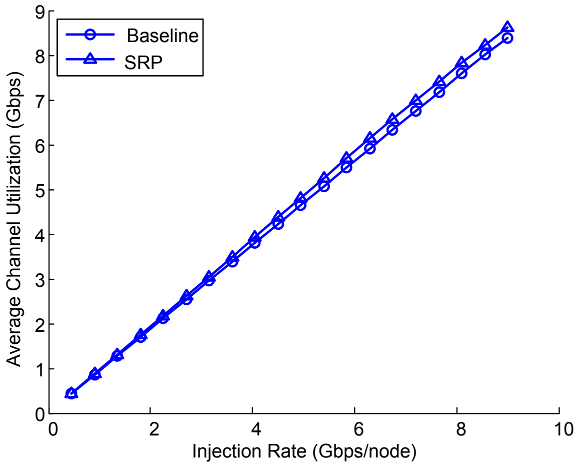


Figure 10. Comparison of channel utilization of SRP and baseline network under uniform random traffic with 4-packet messages.

The bandwidth overhead of SRP is largely due to dropped speculative packets. Figure 11 shows the speculative injection and drop rate as a function of the average network injection rate. This simulation is for UR traffic with 4-packet messages. As the injection rate increases, the injection rate of speculative packets also increases, peaking at about 50% network load. Because speculative packets have a lower network priority, normal data packets are preferentially admitted into the network when competing with a speculative packet. Above 50% network load, this causes the speculative injection rate to decrease. At higher load, the drop rate increases despite a lower speculative injection rate. The queuing delay at high load causes the fraction of speculative packets experiencing time out to increase significantly. Overall, the bandwidth wasted by the speculative packets is less than 1.5% of the total bandwidth. This is a small and justifiable overhead for the latency benefits provided by speculative packet transmission.

The speculative drop rate, and hence the bandwidth overhead, could be reduced by increasing the speculative TTW. Figure 12 shows the drop rate of SRP networks with speculative TTW values ranging from 0.6 to 2.6μs. Every doubling of the TTW reduces the peak drop rate by approximately 50%. A higher TTW will cause the speculative VC to become more congested in the presence of hot-spot traffic. However, it does not affect normal data packets or reservation packets, as these travel on separate, uncongested VCs. The downside of increasing speculative TTW is that it slows speculative retransmission.

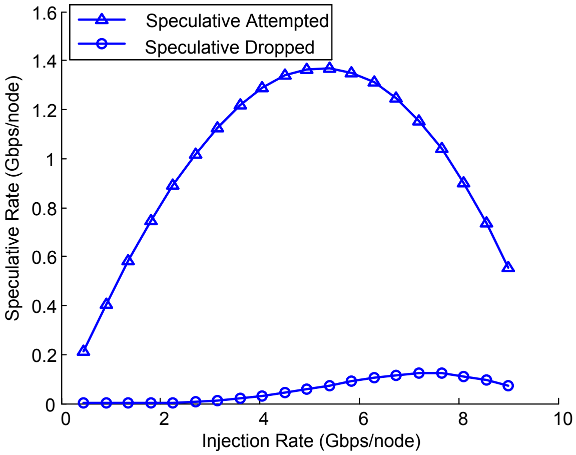


Figure 11. Rate of packets injected on the speculative VC and the rate of packet drops in the speculative VC.

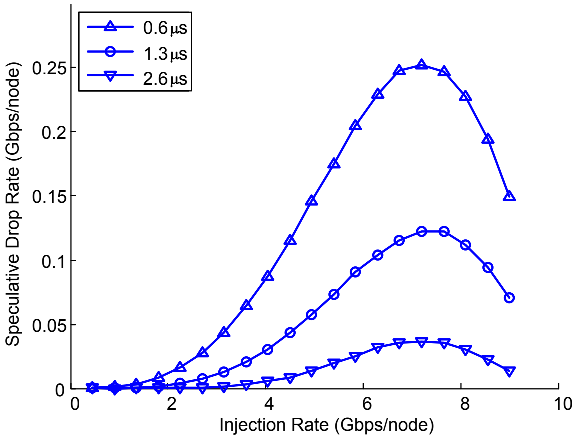


Figure 12. Effect of speculative TTW on the packet drop rate under uniform random traffic with 4-packet messages.

In our implementation, a speculative packet is only retransmitted once a NACK is received. With a high TTW, this may not happen until long after the rest of the message has been transmitted non-speculatively. An alternate fast retransmit protocol resends the outstanding speculative packets when all other packets in the reserved block have been sent. While this protocol eliminates the latency issues with high TTW, we chose not to use it because it introduces additional overhead caused by duplicate packet transmission when both the speculative and retransmitted packets reach the destination.

Bandwidth overhead can also be reduced by increasing n_{max}, the reservation granularity. Speculative packets are intended to cover the round-trip latency of the reservation handshake. With a higher network priority and dedicated control VCs, the reservation round-trip remains low even for very high network traffic load. Thus, the number of speculative packets that exist in the network is bounded by a constant times the number of reservation round-trips. For large message sizes, increasing the reservation granularity, n_{max}, reduces the number of reservation handshakes and hence the speculative drop rate. Figure 13 shows the effect of reservation granularity on the speculative drop rate for uniform random traffic with a message size of 256 packets. With each doubling of the reservation granularity, the maximum speculative drop rate is reduced by half. This matches the expected drop behavior of speculative packets. The benefits of large reservation granularity also need to be balanced with fairness. Large granularity allows long messages to monopolize the destination and starve flows with smaller messages.

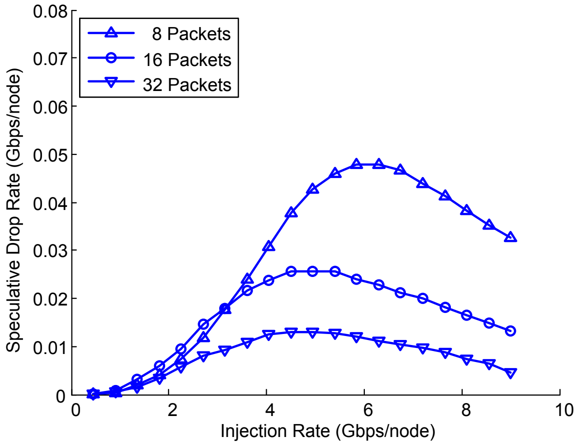


Figure 13. Effect of reservation granularity on the packet drop rate under uniform random traffic with 256-packet messages.

While statically configuring the speculative TTW and reservation granularity is simple to implement and works well, the optimal solution is to dynamically adjust these parameters based on current network conditions. If the sender knows the network is at high load but uncongested, it can chose to not transmit any packets speculatively or to send speculative packets with very high TTW to increase the probability of delivery.

6.2. Scalability

The SRP parameters listed in Table 1 need to be adjusted to accommodate different network configurations, particularly to account for different zero-load network latencies and expected network queuing delay. This has a direct impact on the overhead of SRP. With higher queuing delay, the speculative TTW needs to be increased proportionally to prevent premature timeouts. A larger network also increases the reservation handshake latency and increases the number of packets sent speculatively. Therefore, the reservation granularity needs to be increased to amortize the cost of speculation over a larger number of packets.

To demonstrate the scalability of SRP, we simulate a three-level Fat Tree with a total of 4096 nodes and increase the network channel latency to $128ns$. Using the same set of SRP parameters shown in Table 1, this larger network is able to effectively manage congestion formed by a 200:1 over-subscribed hot-spot traffic pattern. However, when running UR traffic with 4-packet messages, the SRP network has the same saturation throughput as the baseline. Contrast this with the data of the 256-node network shown in Figure 9, where SRP has a 6% higher saturation throughput than the baseline network for the same message size. The reduced saturation throughput is entirely the result of dropped speculative packets. Network throughput recovers by 3% when the speculative TTW is increased to match the higher network queuing delay.

6.3. Small Messages

Allowing small messages to bypass the reservation protocol eliminates SRP's ability to control network congestion caused by these messages. In practice, this is not an issue because most network congestion is caused by large traffic flows (e.g., MapReduce traffic [17]). However, if congestion control for small messages is required, it can be realized by using ECN packet marking to selectively enable SRP for some sources. When a congestion notification is received, SRP is enabled for small messages to that destination. Multiple small messages to the same destination can also be coalesced into a single reservation to amortize overhead.

6.4. Oversubscribed Networks

The focus of SRP is network congestion caused by oversubscribed network destinations. In networks with sufficiently provisioned bandwidth, this is the only cause of network congestion. However, in an under-provisioned network, congestion can arise due to overloaded network channels. The current SRP implementation reserves bandwidth only at the endpoints and cannot handle this case. Two fully reserved nodes which share a network channel will cause channel congestion. If the network has path diversity, adaptive routing can resolve the congestion by spreading the traffic across multiple network channels. With path diversity, the problem is actually one of channel load imbalance, not congestion, and the adaptive routing solution is orthogonal to the use of SRP. A true congestion control solution for networks with under-provisioned bandwidth would require reservations for the bottleneck network channels as well as the endpoints. Alternatively, an additional congestion notification mechanism, such as ECN, can be used to alert the sender of channel congestion. Because our focus is on fully-provisioned networks, we have not studied the problem of internal channel congestion, and it is beyond the scope of this work.

6.5. Fairness

As shown in Figure 5, SRP provides network-level fairness in a network whose routers only implement locally-fair allocation policies. This behavior may seem unintuitive because a reservation packet from nearby nodes can reach the destination faster than those from the far side of the network. However, the latency of the reservation round-trip only affects the number of speculative packets sent by each node, not the node's share of total bandwidth. As long as the throughput of reservation packets in the network is stable, each node will receive an equal share of data bandwidth. In our SRP implementation, the reservation packets have the highest network priority and comprise only a single flit. Thus, sustained congestion never occurs on the control VC, and long-term reservation fairness is guaranteed.

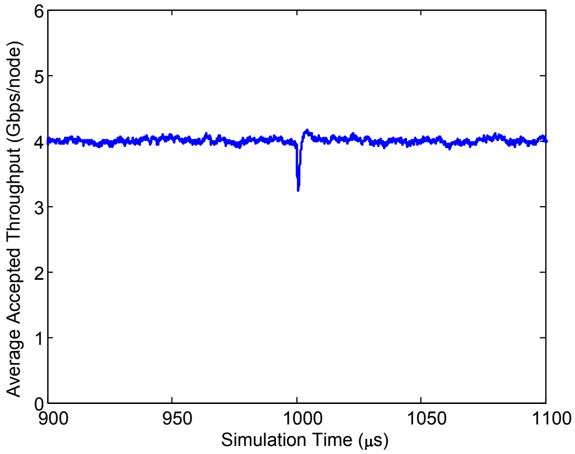


Figure 14. Throughput response of a SRP network under an impulse traffic pattern.

Because reservation packets are themselves unregulated, temporary congestion of the control VC can occur in rare circumstances. Figure 14 shows the response of SRP to an impulse traffic pattern. The network is warmed up under 40% load UR traffic for $1ms$. Then, simultaneously, every node in the network initiates a small (eight packets) hot-spot message to the same destination. After the impulse, all nodes return to UR traffic. This traffic pattern causes 256 reservation packets to be injected in a very short period, creating a temporary reservation hot-spot. Figure 14 shows a dip in network throughput for several micro-seconds after the impulse. However, because the reservation packets are small, the temporary congestion of the control VC quickly dissipates, and network throughput returns to normal. While this example demonstrates that a reservation hot-spot is possible, in realistic network settings, such simultaneous instantiation of network traffic rarely occurs and cannot be sustained.

6.6. Implementation

Implementing SRP requires modifying both the NIC and the network router. On the receiving side of the NIC, a single reservation schedule register is maintained that tracks the earliest grant time for the next incoming reservation. When a grant is issued, this register is incremented by the reservation size. On the sender side, the SRP modification depends on how the NIC supports traffic flows. In a system using queue-pairs, such as Infiniband, the NIC is modified to issue a reservation packet and transmit data packets initially in speculative mode. These speculative packets are retained in the NIC buffers until a positive acknowledgment is received.

The network routers are modified to add the ability to drop a packet and send a corresponding NACK when the packet's TTW expires. Packet drop can be easily handled by the input buffer logic when a packet arrives from the channel or when it reaches the head of an input queue. Generating a NACK packet can be handled by transforming the head flit of the dropped packet into a NACK.

In our SRP implementation, two control VCs and a single speculative VC are added to the network. The characteristics of these VCs allows for reduced implementation cost compared to normal data VCs. The control VCs are designed to handle only small packets and typically have a very low utilization factor. As a result, they have very low buffering requirements. The speculative VC will drop a packet when its cumulative queuing delay exceeds its TTW. Thus, the buffers of the speculative VC are sized to match the TTW. While the buffer cost of SRP is small, additional complexity is introduced into router allocators and arbiters to handle the additional virtual channels and packet priorities.

All of the experiments in this study use a network with a single data VC. In networks with multiple data VCs, the control and speculative VCs can be shared by all data VCs without a significant impact on performance. This is because the load on the control VCs is proportional to the amount of data injected by the node regardless of the number of data VCs. Allowing multiple data VCs to share the same speculative VC is acceptable because speculative packets can always fall back to their dedicated data VC when they expire. In case of multiple data VCs with different Quality-of-Service (QoS) requirements, reservation packets must be prioritized appropriately. In such scenarios, multiple control VCs may be necessary to avoid priority inversion due to head-of-line blocking.

7. Related Work

In this study, we have compared SRP to an ECN mechanism similar to the one used in IBA networks [2]. Over the years, several studies have evaluated the behavior and performance of IBA ECN. The ECN parameters used in our study were derived based on suggestions made in [20]. Pfister et al. characterized the behavior of the IBA ECN using a simulator on a wide array of network configurations and showed that ECN provides good congestion management in many simulations. However, the authors also noted network instability when the ECN parameters are not matched to the traffic pattern. The authors also did not address the short term behavior of ECN during the onset of network hot-spot congestion and instead focused on long-lived traffic flows.

Recently, Mellanox Technologies has incorporated ECN into their Infiniband routers [26]. Gran et al. studied the performance of IBA ECN in hardware on an InfiniScale IV router [13] and confirmed the effectiveness of ECN in hardware using synthetic traffic patterns and HPC benchmarks. Unfortunately, the hardware study was limited to seven nodes and two routers, in a configuration similar to the simple scenario we presented in Section 2. The transient behavior of ECN was also not addressed.

Improved ECN mechanisms for Infiniband networks have been proposed in [11, 23]. These methods differ from the standard IBA ECN mechanism in when packet marking occurs and how nodes respond when congestion is sensed. A summary and comparison study of three different ECN proposals is presented in [10]. Ferrer et al. performed a transient hot-spot study to demonstrate the response time of each method. While the study showed that the Marking and Validation Congestion Management method [11] yields the fastest response to congestion, all ECN methods still exhibit some negative impact on latency and throughput at the onset of a hot-spot.

ECN is also included in an extension to the TCP/IP standard [22]. It is used in conjunction with packet dropping to regulate the TCP congestion window size in order to reduce network congestion in Internet routers. More recently, a proposed enhancement to switched Ethernet adds a notification-based congestion control mechanism. Quantized Congestion Notification (QCN) is a new standard developed by the Data Center Bridging task group to provide congestion control. QCN allows Ethernet switches to send rate adjustment packets to the network sources in order to throttle their injection rates. Allowing the switches to directly send notification packets reduces the network response time to the onset of congestion.

Many proposed congestion control methods do not require packet marking. Some methods attempt to detect hot-spot traffic and isolate it in a separate queue, eliminating the congestion caused by HoL blocking of hot-spot traffic [5, 9]. While effective, these methods have a higher implementation cost due to the need for a large number of queues and VCs to handle an arbitrary number of hot-spots. Other methods broadcast congestion information to notify and regulate the injection rate of hot-spot traffic [27]. The calculation and transmission of global congestion information to each network node has limited scalability compared to ECN and SRP.

Other reservation protocols have been proposed for networking systems. Flit reservation [19] aims at increasing throughput and reducing latency of on-chip network routers by reserving network resources ahead of the packet's injection. Theoretically, the Flit-reservation protocol can prevent any network congestion because every packet's path is completely reserved. Such detailed reservation is too complex and carries too much overhead to implement in a large network, and it is not necessary to prevent network congestion. Song and Pinkston [25] use bandwidth reservation to prioritize the movement of congested packets inside the network. However, their mechanism only affects packets that are already in the network and has no control over the injection rate of the nodes responsible for the hot-spot.

8. Conclusion

This paper has introduced a Speculative Reservation Protocol (SRP) for avoiding network hot-spot congestion. In lossless networks that have adequately provisioned internal channel bandwidth, SRP uses a simple reservation mechanism to prevent the formation of congestion due to network hot-spots. In contrast, existing congestion control mechanisms like Explicit Congestion Notification (ECN) only become active once network congestion has already occurred. By keeping the network out of the congested state entirely, SRP provides better fairness and greatly improves transient response compared to ECN. Other design features of SRP, including reservation granularity and speculative packet transmission before reservation, focus aggressively on reducing the latency and bandwidth overhead of an already light-weight reservation protocol.

Experiments show that SRP responds almost instantaneously to the onset of congestion compared to the hundreds of microseconds it takes for ECN to react. SRP also provides near perfect bandwidth fairness among sources competing for a network hot-spot. For benign traffic patterns, SRP has at most 25% latency and 3% bandwidth overhead compared to a baseline network without congestion control. For bursty traffic patterns, SRP achieves a consistently higher saturation throughput than ECN and the baseline network by efficiently resolving transient hot-spots caused by bursty traffic. SRP represents an effective and viable alternative to the canonical packet marking congestion control mechanisms for high-speed computer networks.

Acknowledgements

This research was supported in part by the P. Michael Farmwald, the Professor Michael J. Flynn, and the Robert Bosch Stanford Graduate Fellowship. This research was furthermore supported by the National Science Foundation under Grant CCF-070234, and by the Stanford Pervasive Parallelism Laboratory.

References

[1] Cisco data center infrastructure 2.5 design guide.

[2] Infiniband trade association, infiniband architecture specification, volume 1, release 1.2.1, http://www.infinibandta.com.

[3] D. Abts, M. R. Marty, P. M. Wells, P. Klausler, and H. Liu. Energy proportional datacenter networks. In *Proceedings of the 37th annual international symposium on Computer architecture*, 2010.

[4] M. Al-Fares, A. Loukissas, and A. Vahdat. A scalable, commodity data center network architecture. *SIGCOMM Comput. Commun. Rev.*, 38, August 2008.

[5] A. Banerjee and S. W. Moore. Flow-aware allocation for on-chip networks. In *Proceedings of the 2009 3rd ACM/IEEE International Symposium on Networks-on-Chip*.

[6] T. Benson, A. Anand, A. Akella, and M. Zhang. Understanding data center traffic characteristics. *SIGCOMM Comput. Commun. Rev.*, 40, January 2010.

[7] W. Dally and B. Towles. *Principles and Practices of Interconnection Networks*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2003.

[8] W. J. Dally. Virtual-Channel Flow Control. *IEEE Transactions on Parallel and Distributed Systems*, 3(2), 1992.

[9] J. Duato, I. Johnson, J. Flich, F. Naven, P. Garcia, and T. Nachiondo. A new scalable and cost-effective congestion management strategy for lossless multistage interconnection networks. In *High-Performance Computer Architecture. 11th International Symposium on*, 2005.

[10] J.-L. Ferrer, E. Baydal, A. Robles, P. López, and J. Duato. On the influence of the packet marking and injection control schemes in congestion management for mins. In *Proceedings of the 14th international Euro-Par conference on Parallel Processing*.

[11] J.-L. Ferrer, E. Baydal, A. Robles, P. Lopez, and J. Duato. Congestion management in mins through marked and validated packets. In *Proceedings of the 15th Euromicro International Conference on Parallel, Distributed and Network-Based Processing*, 2007.

[12] J.-L. Ferrer, E. Baydal, A. Robles, P. Lopez, and J. Duato. A scalable and early congestion management mechanism for mins. In *Parallel, Distributed and Network-Based Processing, 18th Euromicro International Conference on*, 2010.

[13] E. Gran, M. Eimot, S.-A. Reinemo, T. Skeie, O. Lysne, L. Huse, and G. Shainer. First experiences with congestion control in infiniband hardware. In *Parallel Distributed Processing, 2010 IEEE International Symposium on*.

[14] A. Greenberg, J. R. Hamilton, N. Jain, S. Kandula, C. Kim, P. Lahiri, D. A. Maltz, P. Patel, and S. Sengupta. Vl2: a scalable and flexible data center network. In *Proceedings of the ACM SIGCOMM 2009 conference on Data communication*.

[15] C. Guo, G. Lu, D. Li, H. Wu, X. Zhang, Y. Shi, C. Tian, Y. Zhang, and S. Lu. Bcube: a high performance, server-centric network architecture for modular data centers. In *Proceedings of the ACM SIGCOMM 2009 conference on Data communication*.

[16] C. Guo, H. Wu, K. Tan, L. Shi, Y. Zhang, and S. Lu. Dcell: a scalable and fault-tolerant network structure for data centers. *SIGCOMM Comput. Commun. Rev.*, 38, August 2008.

[17] S. Kandula, S. Sengupta, A. Greenberg, P. Patel, and R. Chaiken. The nature of data center traffic: measurements & analysis. In *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference*, 2009.

[18] R. Niranjan Mysore, A. Pamboris, N. Farrington, N. Huang, P. Miri, S. Radhakrishnan, V. Subramanya, and A. Vahdat. Portland: a scalable fault-tolerant layer 2 data center network fabric. In *Proceedings of the ACM SIGCOMM 2009 conference on Data communication*.

[19] L.-S. Peh and W. Dally. Flit-reservation flow control. In *High-Performance Computer Architecture. Proceedings. Sixth International Symposium on*, 2000.

[20] G. Pfister, M. Gusat, W. Denzel, D. Craddock, N. Ni, W. Rooney, T. Engbersen, R. Luijten, R. Krishnamurthy, and J. Duato. Solving hot spot contention using infiniband architecture congestion control. In *High Performance Interconnects for Distributed Computing*, 2005.

[21] G. Pfister and V. A. Norton. "hot spot contention and combining in multistage interconnection network. *IEEE Trans. on Computers*, C-34, October 1985.

[22] K. Ramakrishnan. The addition of explicit congestion notification (ecn) to ip. *IETF RFC 3168*.

[23] J. Santos, Y. Turner, and G. Janakiraman. End-to-end congestion control for infiniband. In *Twenty-Second Annual Joint Conference of the IEEE Computer and Communications.*, volume 2, 2003.

[24] S. Scott, D. Abts, J. Kim, and W. J. Dally. The blackwidow high-radix clos network. In *Proceedings of the 33rd annual international symposium on Computer Architecture*, 2006.

[25] Y. H. Song and T. M. Pinkston. Distributed resolution of network congestion and potential deadlock using reservation-based scheduling. *IEEE Trans. Parallel Distrib. Syst.*, 16, August 2005.

[26] M. Technologies. Mellanox infiniscale iv switch architecture provides massively scaleable 40gb/s server and storage connectivity,. *Press release*.

[27] M. Thottethodi, A. Lebeck, and S. Mukherjee. Self-tuned congestion control for multiprocessor networks. In *High-Performance Computer Architecture. The Seventh International Symposium on*, 2001.

[28] C.-Q. Yang and A. Reddy. A taxonomy for congestion control algorithms in packet switching networks. *Network, IEEE*, 9(4), 1995.