at time 0.319, and it lowers its window size from 23 packets back to 14. Reno transmits data packets until time 12.912.

TCP Vegas also does not perform a handshake, and begins data transmission at 0 seconds. Its data transmission rate curve[2] is extremely similar to TRUMP's, except for some small deviations before time 0.3. Data transmission continues until time 12.018.

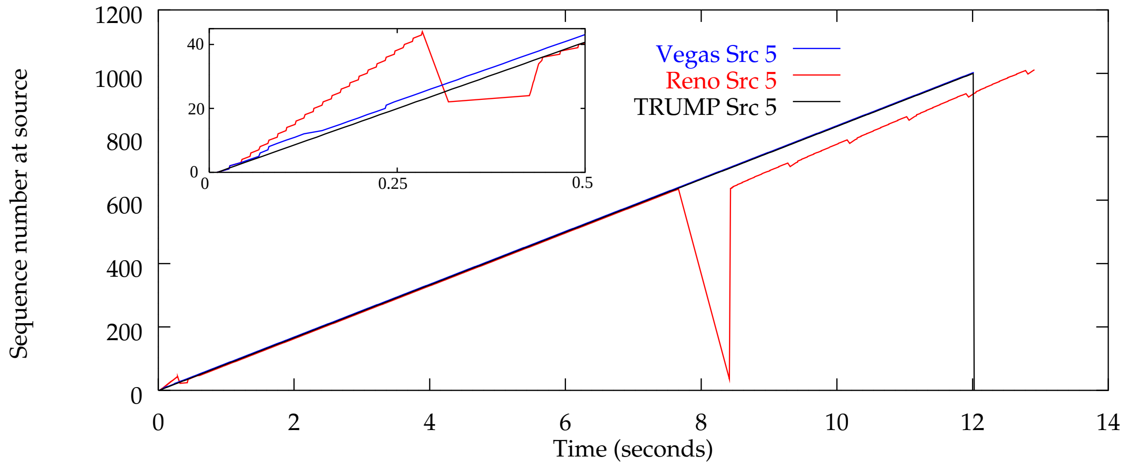


Figure 17: Sequence Numbers in Scenario 1

9.3.2 Router Queue Lengths

For TRUMP, buffer queue lengths in routers 2 and 3 never exceed 1 packet. No packets are dropped by any router. With Reno, buffer queue lengths in router 2 reaches the maximum value of 10 data packets several times as shown in Figure 18. Overall packet loss is 18 packets, all dropped by router 2. With Vegas, no packets are lost by any router, and the buffer queue length in router 2 sits at between 2 and 3 data packets for nearly all of the transmission time.

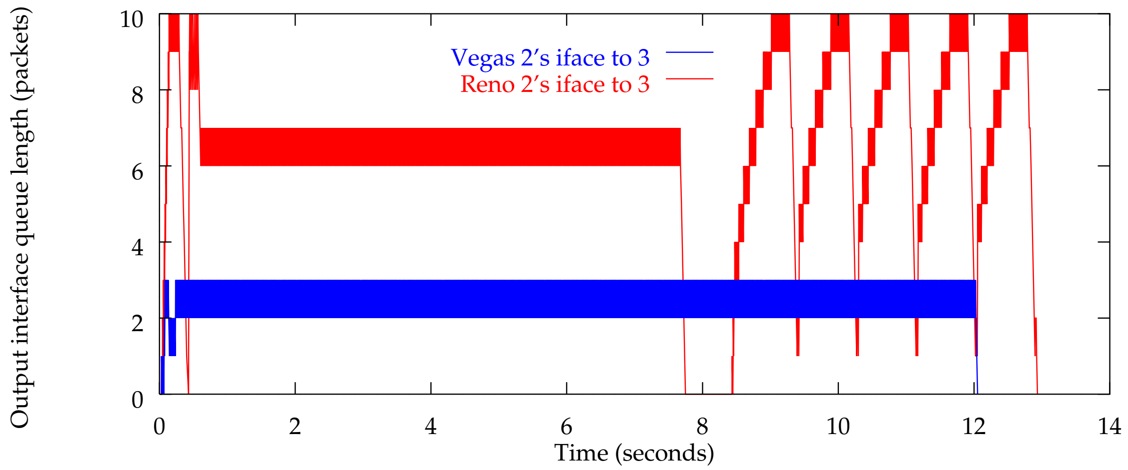


Figure 18: Queue Lengths in Scenario 1

[2]Strictly speaking, the derivative of the sequence number curve.

9.3.3 Round-Trip Times

The round-trip time for all TRUMP acknowledgments is 0.016 seconds, as shown in Figure 19. The round-trip time for Reno oscillates between 0.083 and 0.126 seconds, reflecting the queue size in router 2. In Vegas, the round-trip time rises quickly from 0.016 seconds to 0.047 at time 0.6, and maintains a constant value of 0.048 for most of the transmission time.

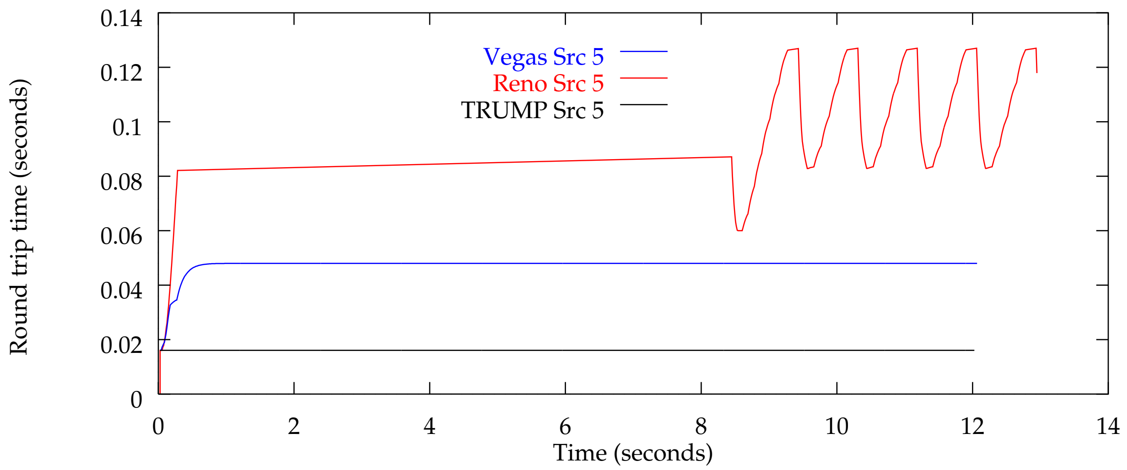


Figure 19: Round Trip Times in Scenario 1

In summary, TRUMP and Vegas do not lose packets in Scenario 1, but Reno does. TRUMP's and Vegas' data rates are extremely similar; Vegas, however, maintains a higher queue size in router 2, and this is reflected in the higher round-trip time.

9.4 Scenario 2

Scenario 2 is shown in Figure 20:

The scenario shows a number of nodes connected by full-duplex links. All links have a data rate of 10Mbps, except for the link between routers 2 and 3, which is a 64kbps link. The 10Mbps links have a latency of 1 μsec, and the 64kbps link has a latency of 1 second. The latter latency, although unrealistic, serves to set the overall round-trip time of this small network to that which is often experienced in a global-scale network.

There are four data flows, $5 \rightarrow 10$, $6 \rightarrow 11$, $7 \rightarrow 12$ and $8 \rightarrow 9$. All cross the 64kbps link, but all start at different times. Thus the link between routers 2 and 3 is the bottleneck in the scenario, and should cause congestion in those routers.

Note that all data flows have corresponding acknowledgment flows in the reverse direction. Note also that the data flow $8 \rightarrow 9$ is in the reverse direction to the other data flows.

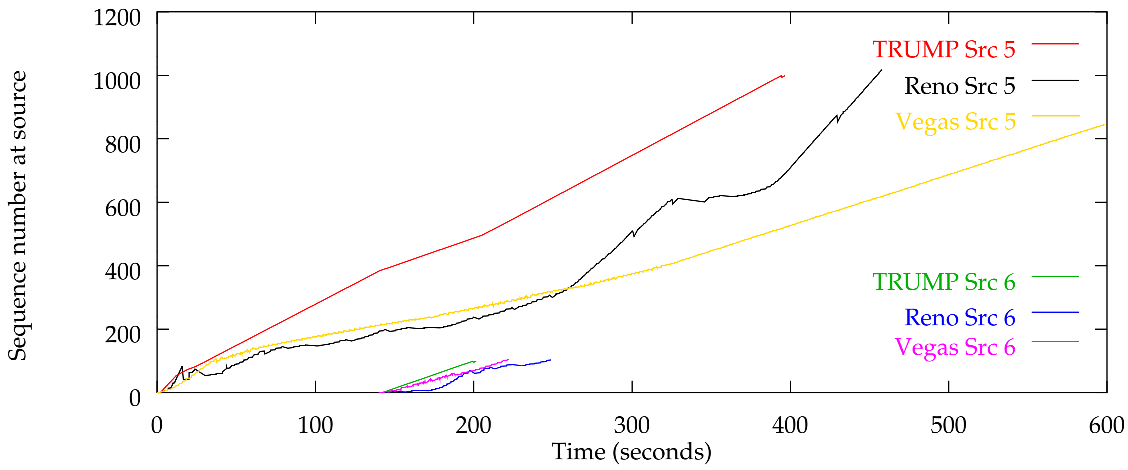


Figure 21: Sequence Numbers for Sources 5 and 6

TRUMP's sequence numbers rise smoothly, controlled by the rate values calculated by RBCC. Reno's rate is much more jagged, caused by its window size changes which reflect packets lost by the destination. Note in particular the early rate fluctuation for Reno's source 5. Vegas' rate is much smoother than Reno's, and its startup is much better behaved than Reno's: however, Vegas' source 5 does not detect that source 6 has terminated at time 222, and so its throughput is even worse than Reno source 5.

The plot for sources 7 and 8, Figure 22, shows similar characteristics. Both TCP Reno and Vegas transmits packets for source 8 (the reverse data flow) at a much slower rate than TRUMP/RBCC. This is surprising, given the fact that in this scenario TRUMP/RBCC did not lose any packets whatsoever: there must have been adequate bandwidth for source 8 which was not used by both Reno and Vegas.

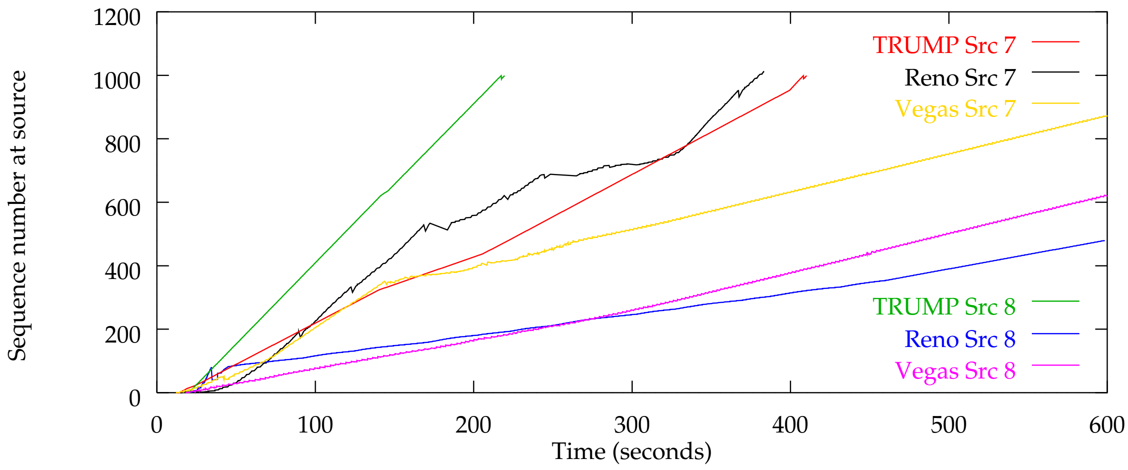


Figure 22: Sequence Numbers for Sources 7 and 8

9.4.3 Packet Loss

In Scenario 2, TRUMP sources do not lose any packets whatsoever. Reno's sources, on the other hand, continually lose packets throughout the life of their transmission. Most packet losses are at the beginning of the data flow (until the optimum window size is reached), and Reno slowly loses packets after that as it tries to open its sliding window to take advantage of any new excess bandwidth. Reno loses 108 packets in total. Figure 23 shows the cumulative packet losses by TRUMP, Vegas and Reno.

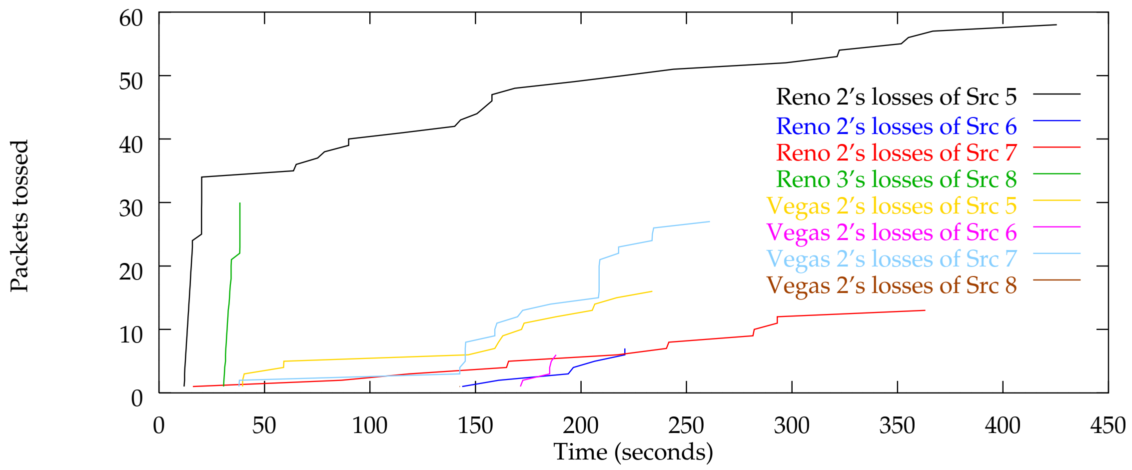


Figure 23: Cumulative Packet Losses by TCP

The Vegas improvements to TCP help reduce packet loss, with sources 5, 6 and 7 losing 16, 6 and 27 packets, respectively. This improvement seems more a result of the failure to utilise bandwidth after source 6 terminates, as against any real improvement to TCP.

9.4.4 Router Queue Lengths

In Scenario 2, routers 2 and 3 can only buffer 15,000 bytes (10 data packets). If the router cannot buffer the packet, it is immediately discarded (FCFS queueing and Drop Tail packet dropping are in use). Figure 24 shows the queue lengths (in packets) for router 2.

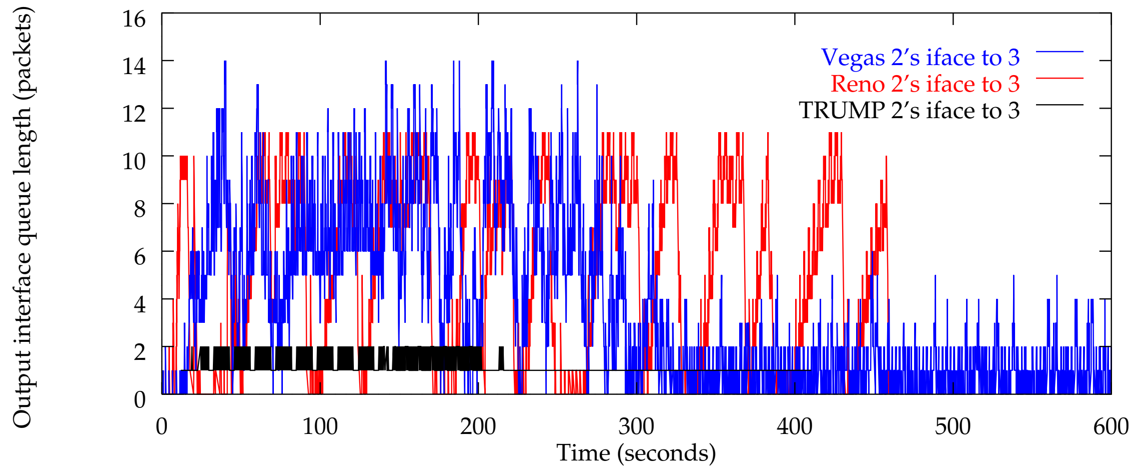


Figure 24: Average Queue Length in Router 2

This plot is cluttered. For router 2, there are several peaks in queue length where Reno's excessive rate has caused lost packets, due to its window resizing mechanism. TRUMP/RBCC keeps the queue length for router 2 between 1 and 2 packets over the entire simulation. Vegas keeps average queue lengths below Reno's when source 6 is idle (due to poor bandwidth utilisation), but causes many peaks above queued packets when source 6 is transmitting.

9.4.5 End-to-End Delays

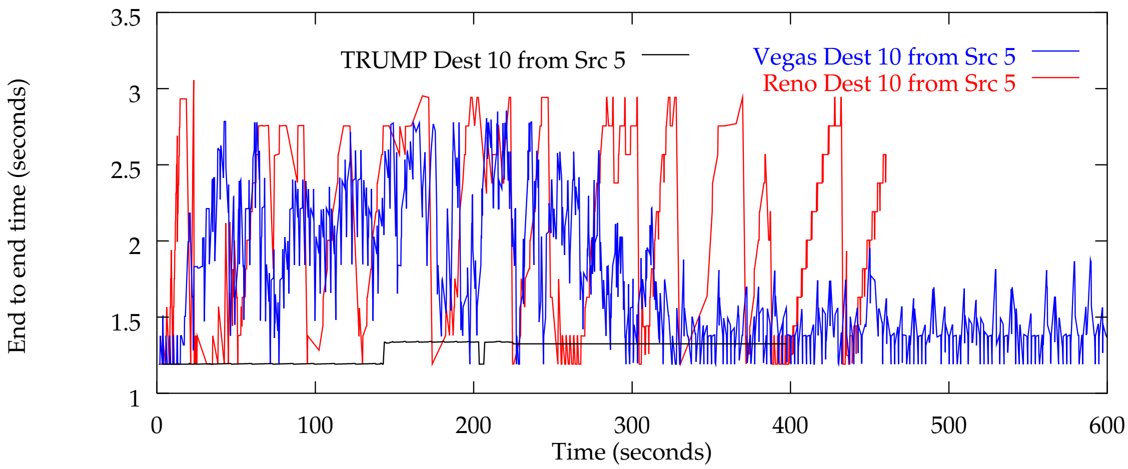


Figure 25: Source/Destination Delay between 5 and 10

In Scenario 2, the end-to-end delay for a packet from source 5 must be at least 1.188 seconds, given the link latencies and packet propagation time. Figure 25 shows the measured end-to-end delay of source 5 for TCP Reno, TCP Vegas and TRUMP. TRUMP has an end-to-end delay between 1.19 and 1.34 seconds; Reno's value oscillates between 1.19 and 3.06 seconds, due to the oscillation of the queue size in router 2; Vegas shows faster oscillations which have a higher deviation than Reno's, although the average end-to-end delay is lower. The end-to-end delays for sources 6 and 7 show the same behaviour. The low end-to-end delay deviation exhibited by TRUMP/RBCC is particularly suited for time-sensitive data such as real-time voice or video transmission.

A summary of per-source results for TRUMP, Vegas and Reno is given below.

Source Node	End Time	Avg Rate	End to end Avg	End to end S.Dev	Round-trip Avg	Round-trip S.Dev	Pkts Lost	Pkts Retx'd
TRUMP 5	396.6	30597	1.28	0.07	2.32	0.07	0	6
Vegas 5	598.4	23843	1.77	0.45	2.01	0.74	16	342
Reno 5	457.9	29037	2.06	0.59	3.00	0.47	58	89
TRUMP 6	201.6	20989	1.25	0.00	2.35	0.06	0	4
Vegas 6	222.3	24641	2.30	0.41	2.27	0.95	6	63
Reno 6	248.8	12905	2.01	0.65	2.98	0.52	7	12
TRUMP 7	410.2	30633	1.21	0.02	2.25	0.07	0	11
Vegas 7	599.9	29477	1.72	0.48	1.80	1.19	27	569
Reno 7	383.3	33322	2.19	0.58	3.15	0.47	13	17
TRUMP 8	219.5	60841	1.19	0.00	2.33	0.07	0	11
Vegas 8	599.8	27248	1.37	0.18	0.80	0.21	0	698
Reno 8	598.4	10628	1.38	0.41	2.63	0.43	30	33

Note that end-to-end and round-trip deviation for TRUMP is generally lower then Reno's and Vegas'. TRUMP's average data rate is higher than Reno's and Vegas', and although Vegas loses fewer packets than Reno, it retransmits substantially more packets than TRUMP or Reno.

9.4.6 Link Utilisations

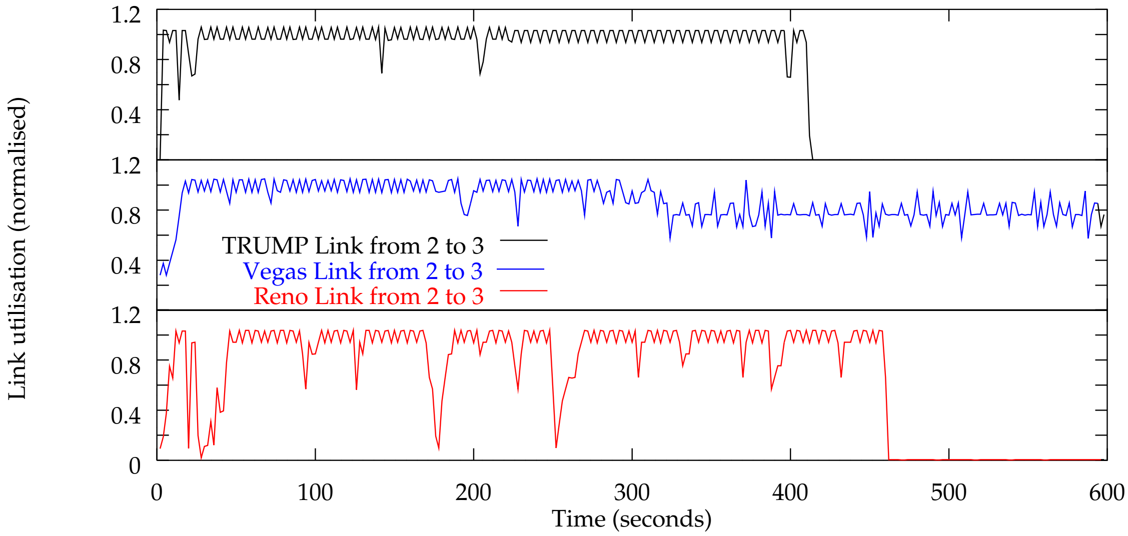


Figure 26: Link 2 → 3 Utilisation in Scenario 2

Another indication of network congestion, closely tied to average queue lengths, is the utilisation of a link. Together with the queue lengths, the link utilisation shows whether an output link is underloaded or overloaded.

Figure 26 shows the utilisation of the link in the direction of router 2 to router 3 for TCP Reno, TCP Vegas and TRUMP/RBCC. TRUMP fully utilises the link, with occasional dips due to rate recalculations as sources start and stop. Vegas performs nearly as well, but with utilisation falling off after source 6 terminates. Reno shows many dips due to window shutdown. The average utilisation for TRUMP, Vegas and Reno is 0.97, 0.87 and 0.83, respectively.

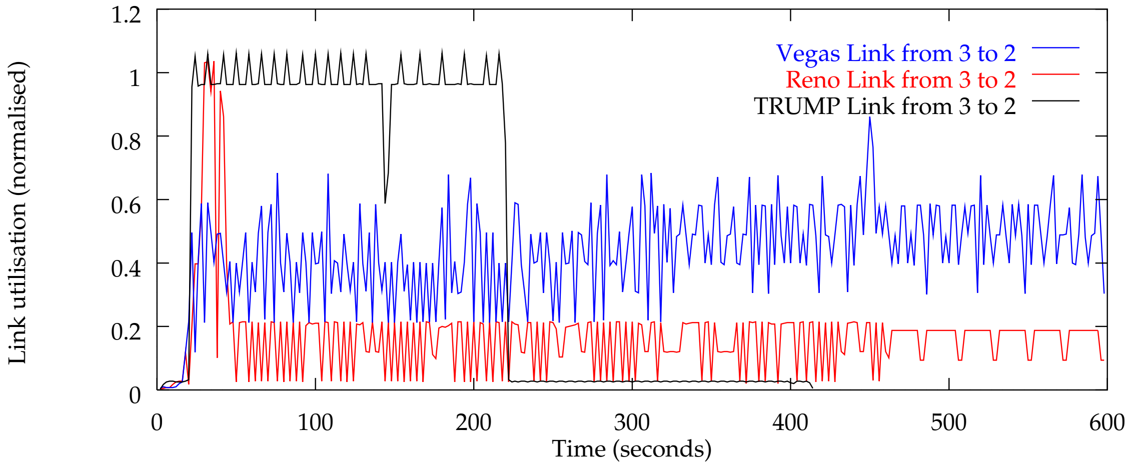


Figure 27: Link 3 → 2 Utilisation

The utilisation of the link in the direction of router 3 to router 2 shows that TRUMP fully utilises the link while source 8 is transmitting, with only acknowledgment traffic passing over the link when source 8 stops. Both Vegas and Reno, on the other hand, never fully utilise this link with traffic from source 8, although Vegas makes around twice the utilisation of the link compared to Reno.

9.4.7 Predicted and Measured TRUMP Transmission Rates

The optimum rates for TRUMP sources in Scenario 2 were given in Section 9.4.1. Figure 28 shows the actual rates returned from RBCC.

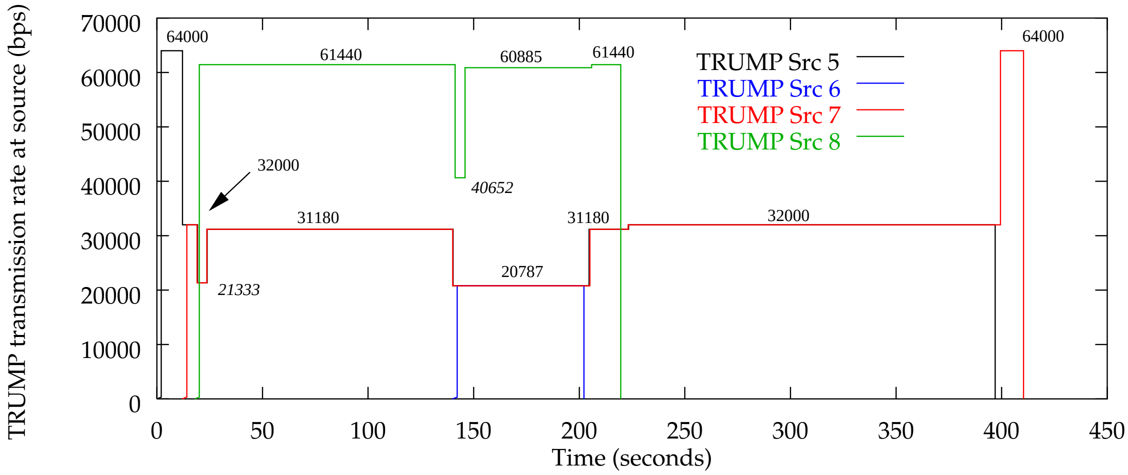


Figure 28: TRUMP's Rates for Scenario 2

As can be seen, the rates used by the TRUMP sources match nearly exactly with the optimal rates. There are slight delays due to round-trips which cause the TRUMP sources to adopt new rates after the optimal time. Initial bit rates are correctly found after TRUMP's two-way connection handshake. The close correspondence between the optimum rates and RBCC's calculated rates indicates that RBCC's rate calculation method is successful, and fair to all traffic sources.

9.4.8 Evaluation of Comparison Results

The comparison of TCP Reno, TCP Vegas and TRUMP/RBCC in Scenario 2 shows that TRUMP/RBCC gives quite different traffic characteristics to both TCP versions, with much less network congestion. RBCC determines source rates which will fully utilise output interfaces (where possible), while maintaining very low output queue sizes in routers. This has the effect of preventing long-term congestion, minimising buffering delays and gives end-to-end delays a very low variance, which is ideal for time-sensitive traffic. The rates calculated by RBCC are quite close to the optimum rates for Scenario 2.

9.5 Scenario 2a

Scenario 2 was modified to limit the desired transmission rates of flow $6 \rightarrow 11$ to 18,000 bps, and flow $7 \rightarrow 12$ to 32,000 bps. As with Scenario 2, the optimum transmission rates can be calculated:

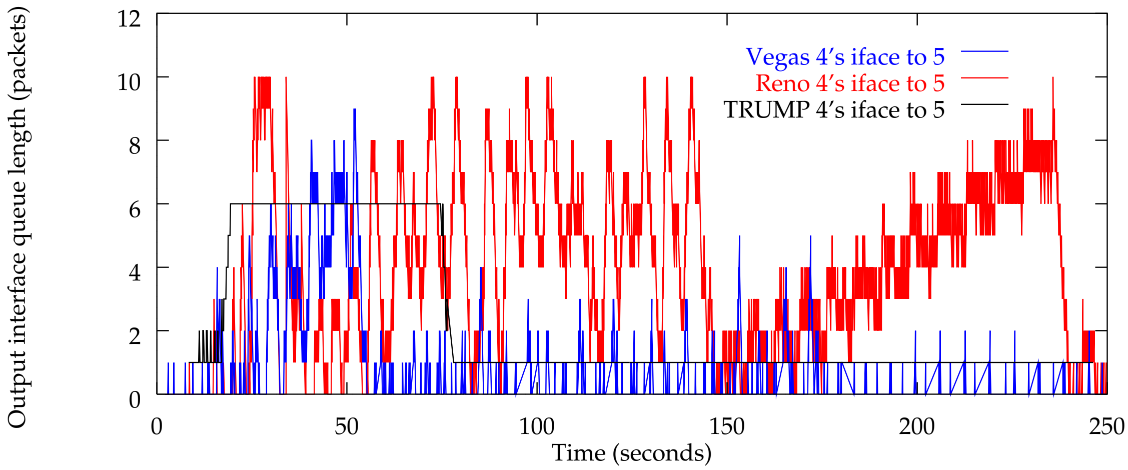


Figure 30: Queue Lengths in Scenario 3

With TCP Reno sources, the following routers are congested:

- Router 2 on the link to 3, with a peak of 2 packets at time 53.
- Router 3 on the link to 4, with 5 peaks to 2 packets, 3 peaks to 3 packets and one peak of 4 packets from time 40 to 200.
- Router 4 on the link to 3, with dozens of peaks above 1 from time 40 to 200.
- Router 4 on the link to 5. This interface is severely congested, with 2 or more packets queued for most of the time 60 to 200, as shown in Figure 30. There are 9 peaks of 10 packets queued.

As with TCP Reno, the following routers are congested with TCP Vegas:

- Router 2 on the link to 3, with several short peaks of 3 packets queued before time 200.
- Router 3 on the link to 4, with many peaks of 2 packets and above, and two short peaks of 4 packets queued before time 200.
- Router 4 on the link to 3, with some peaks of 2 packets and above, and one short peak of 5 packets queued at time 119.
- Router 4 on the link to 3, with dozens of peaks above 1 from time 40 to 200.
- Router 4 on the link to 5. This interface is severely congested from times 35 to 55, with a queue length of 1 to 9 packets, as shown in Figure 30.

9.6.3 Link Utilisations

Utilisation of the links 4 to 5 and 4 to 3 are much better for TRUMP than for TCP, as shown in Figures 31 and 32. In fact, neither of the TCP versions make full use of the link from 4 to 3, and TCP Vegas has much poorer utilisation than TCP Reno.

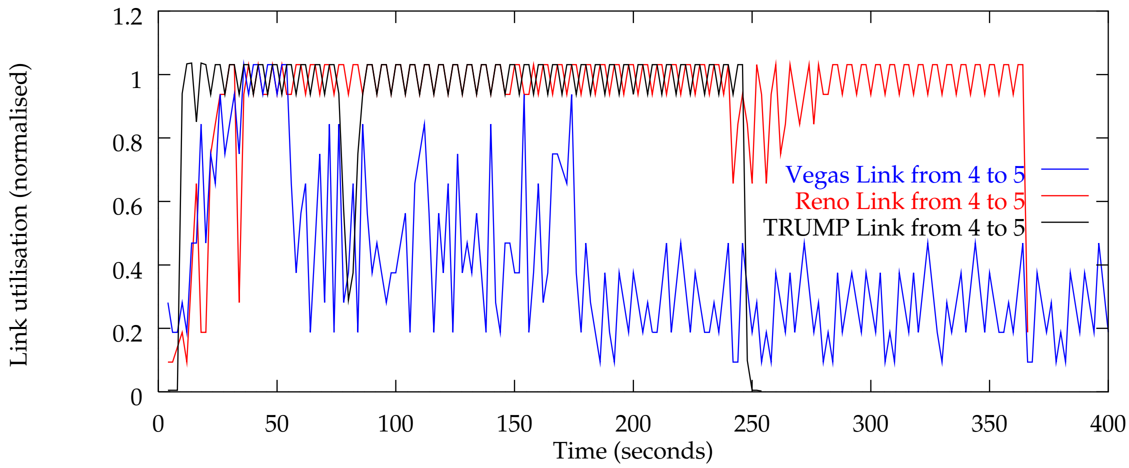


Figure 31: 4 $\rightarrow$ 5 Link Utilisation in Scenario 3

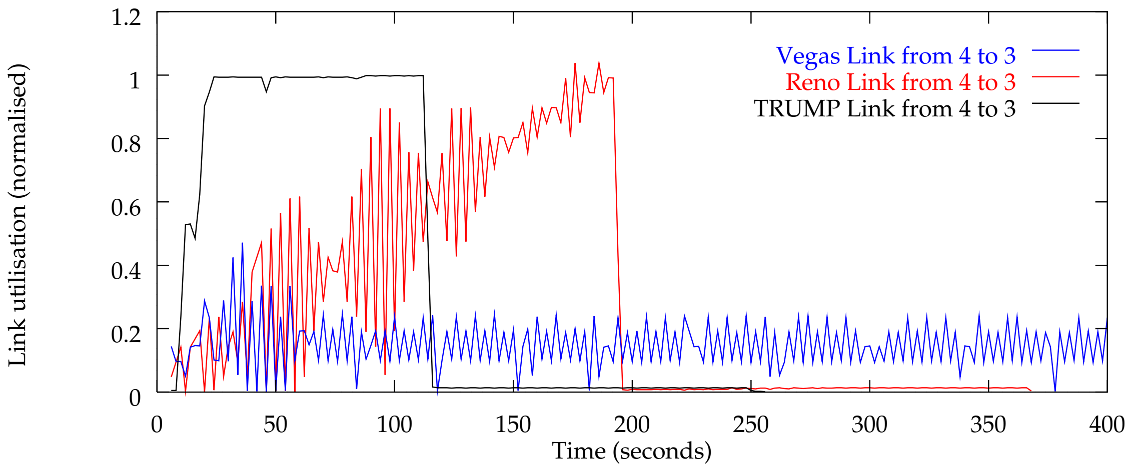


Figure 32: 4 $\rightarrow$ 3 Link Utilisation in Scenario 3

9.6.4 End-to-End & Round-trip Times

End-to-end times are smooth for TRUMP but not for TCP, as shown for source 6 in Figure 33. Average end-to-end times for TRUMP, Vegas and Reno are 3.41, 3.41, and 4.07 seconds, respectively. Note the large increase in TRUMP's end-to-end delay when router 4 has many packets queued.

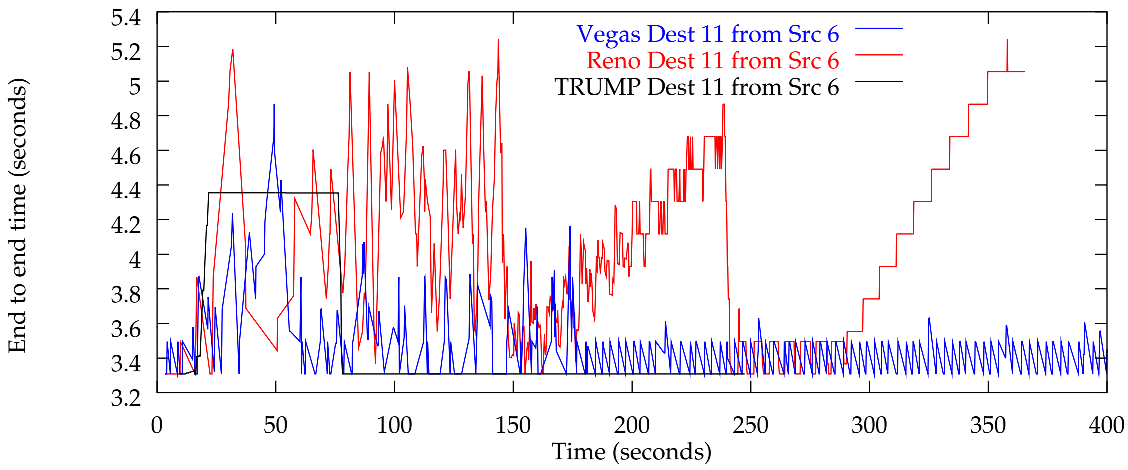


Figure 33: End-to-End Times in Scenario 3

9.6.5 TRUMP Transmission Rates

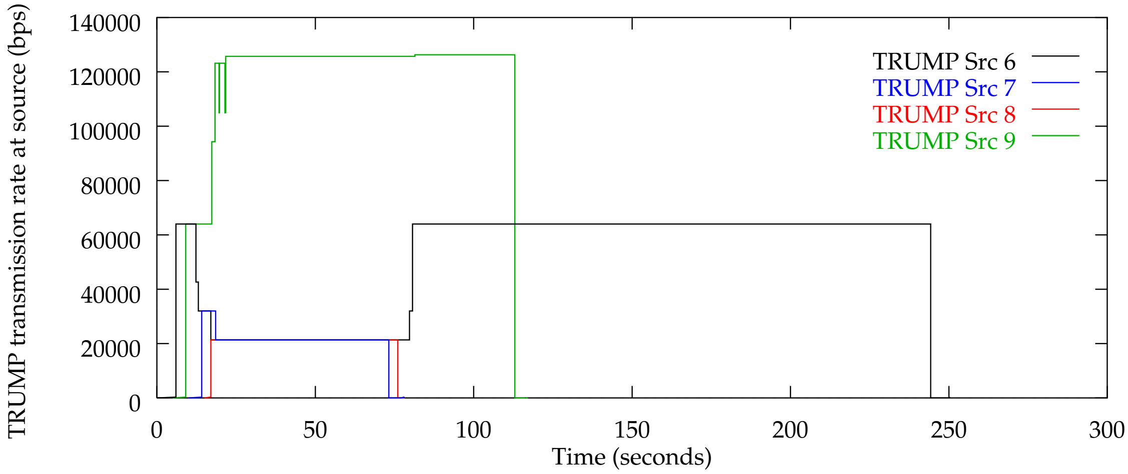


Figure 34: RBCC Rates in Scenario 3

Rates calculated by RBCC are also smooth and have no long-term oscillations. There are small fluctuations in individual rates when traffic flows start and stop, and this is due to the bandwidth-scaling congestion avoidance heuristic in RBCC's implementation.

9.7 Scenario 4

Scenario 4 is shown in Figure 35. This scenario is designed to see how well existing traffic sources react to other sources starting and stopping. Here, the links between routers 17 to 24 take turns being the main bottleneck at different times. The number of sources starting within such a short time should lead to short-term congestion.

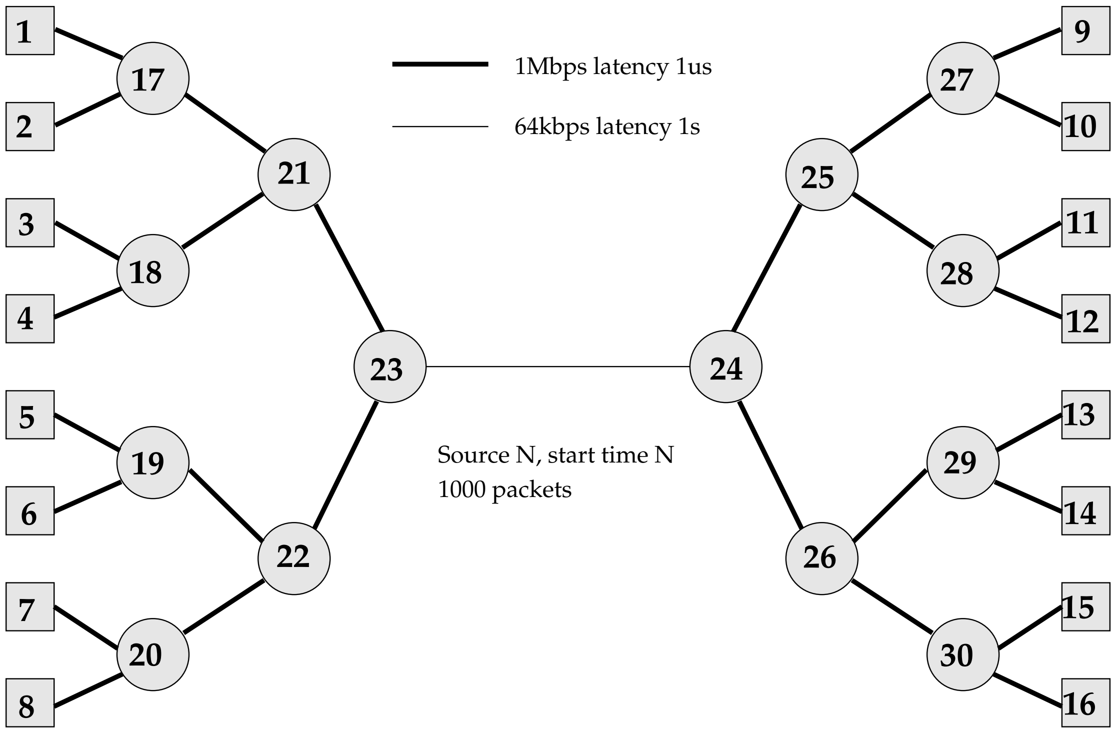


Figure 35: Scenario 4

For TRUMP, the main bottleneck is router 23, although router 21's buffers for the link to 23 peak at 2 packets at times 3 and 4. Router 23's packet buffer to 24 reaches 4 at times 5 and 6, but never exceeds 2 packets queued after time 8; this is shown in Figure 36.

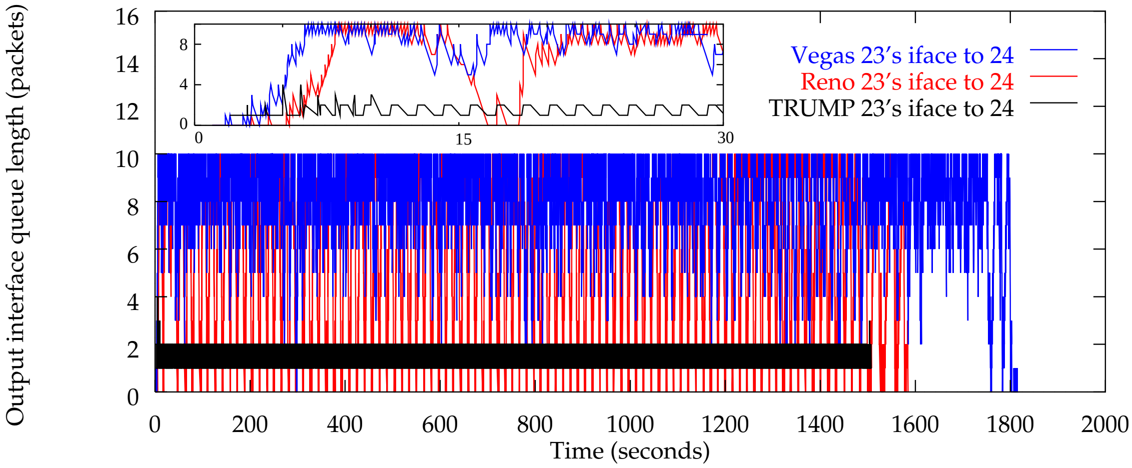


Figure 36: Queue Lengths in Scenario 4

For TCP Reno, the only congested router is 23, and it remains in a severely congested state while any source is transmitting, again shown in Figure 36. The output buffer's queue size fluctuates between 1 and 10 packets, with an average queue length of 6.74 packets. Needless to say, there is a large amount of lost traffic; 782 packets are dropped, spread evenly over all sources. This is approximately 9% packet loss.

TCP Vegas also congests router 23 primarily, as shown in Figure 36, and to the same extent. Compared to Reno, Vegas loses *2,702* packets, spread evenly over all sources. This is approximately 34% packet loss.

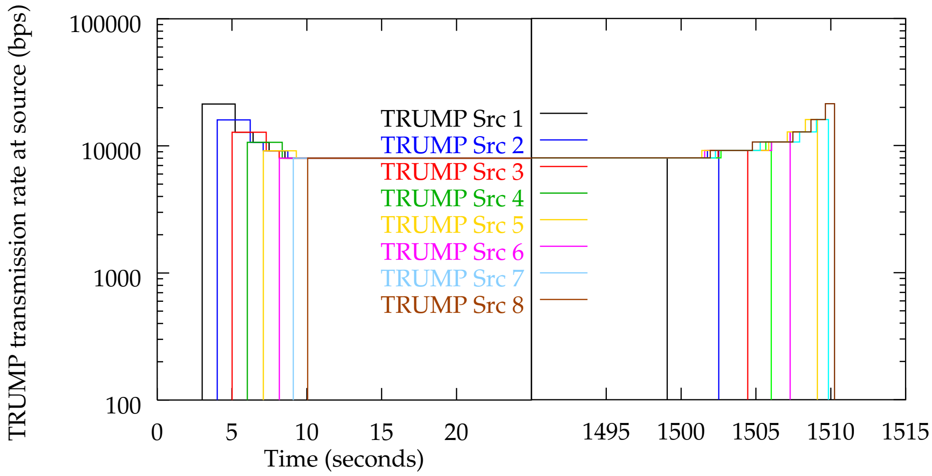


Figure 37: TRUMP Rates in Scenario 4

The transmission rates calculated by RBCC show a lovely smooth reduction in each sources' rate as new sources begin to transmit, as shown in Figure 37. The link 23 $\rightarrow$ 24 is 99% utilised while data transmission is in progress. End-to-end and round-trip times peak to high values initially, and then settle down to near-constant values from time 23 onwards, as is shown for source 6 in Figure 38.

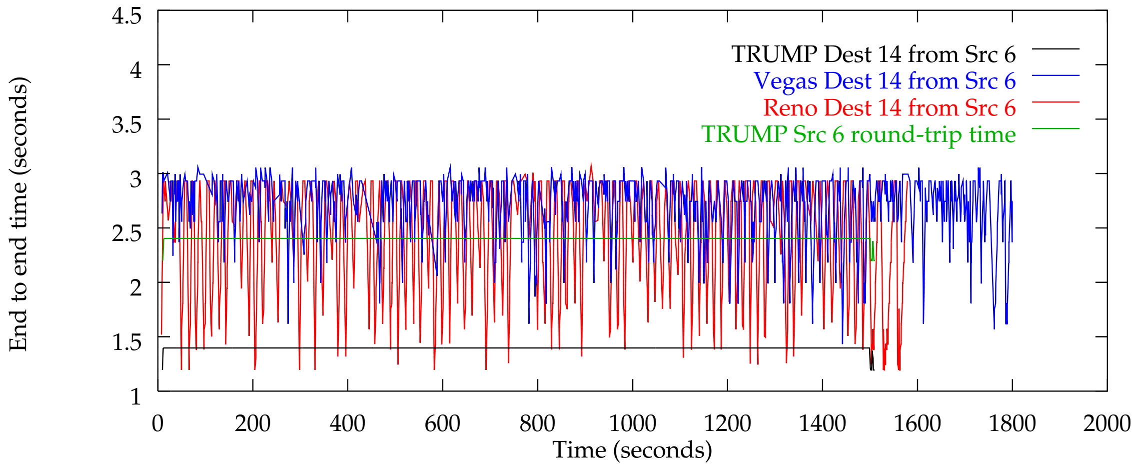
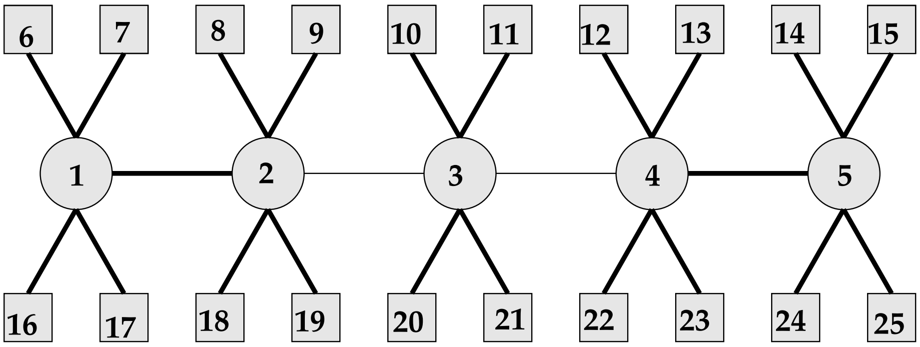


Figure 38: End-to-End and Round-trip Times in Scenario 4

For TCP Reno and Vegas, the link 23 $\rightarrow$ 24 is 93% and 99.7% utilised while data transmission is in progress, respectively. End-to-end and round-trip times are *extremely* peaky for all sources, as is shown for source 6 in Figure 38.

A summary of per-source results for TRUMP, Vegas and Reno is given below. Again note that the averages and deviations for both end-to-end and round-trip times are lower for TRUMP than for the TCP flavours. As noted before, Vegas loses more packets than Reno; it also retransmits many more packets than either Reno or TRUMP.



6 -> 10 start 0s
7 -> 24 start 20s
8 -> 21 start 50s
9 -> 25 start 60s
10 -> 23 start 61s

11 -> 16 start 1s
12 -> 19 start 21s
13 -> 20 start 40s
14 -> 22 start 41s
15 -> 17 start 49s

1-> 2 1Mbps latency 5ms
2 -> 3 100kbps latency 8ms
3 -> 4 64kbps 800ms
4 -> 5 1Mbps 10ms

Figure 39: Scenario 5

With TRUMP, several routers are congested at different times. Router 3 on the link to 2 has queue sizes above 2 from time 24 to 164, with three peaks to 5 packets queued around time 58. Router 2 on the link to 3 has short peaks to 3 packets queued from time 60 to 83. Router 4 on the link to 3 has two peaks to 3 packets queued at times 57 and 61. Router 3 on the link to 4 has peaks above 2 packets queued from time 61 to time 447. Some of these queue sizes are shown in Figure 40. No packets are lost by any RBCC routers.

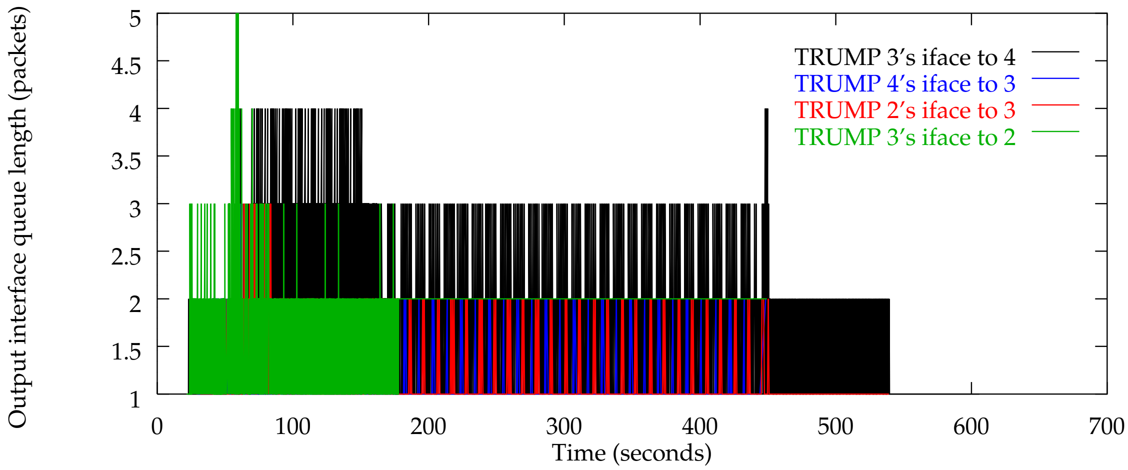


Figure 40: Queue Sizes in Scenario 5 by TRUMP

The rates calculated by RBCC are mostly smooth, with some small fluctuations for most sources during the times 20 to 80; these are due to the packet queue sizes crossing the RBCC threshold in some routers. The rates used by each TRUMP source are shown in Figure 41.

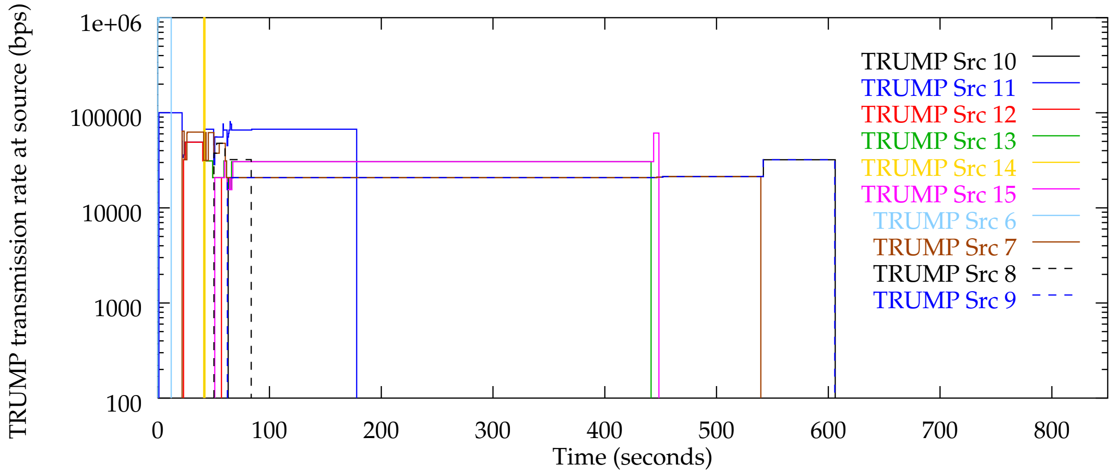


Figure 41: TRUMP Rates in Scenario 5

Average link utilisations for the low-speed links are given in the following table. The end-to-end times for TRUMP fluctuate a lot until time 80 and then show little deviation, except for sources 16 and 17. Round-trip times show some deviation for all sources at all times.

Link	TRUMP	TCP Vegas	TCP Reno
$2 \rightarrow 3$	0.43	0.37	0.41
$3 \rightarrow 2$	0.43	0.37	0.39
$4 \rightarrow 3$	0.70	0.79	0.60
$3 \rightarrow 4$	0.98	0.93	0.85

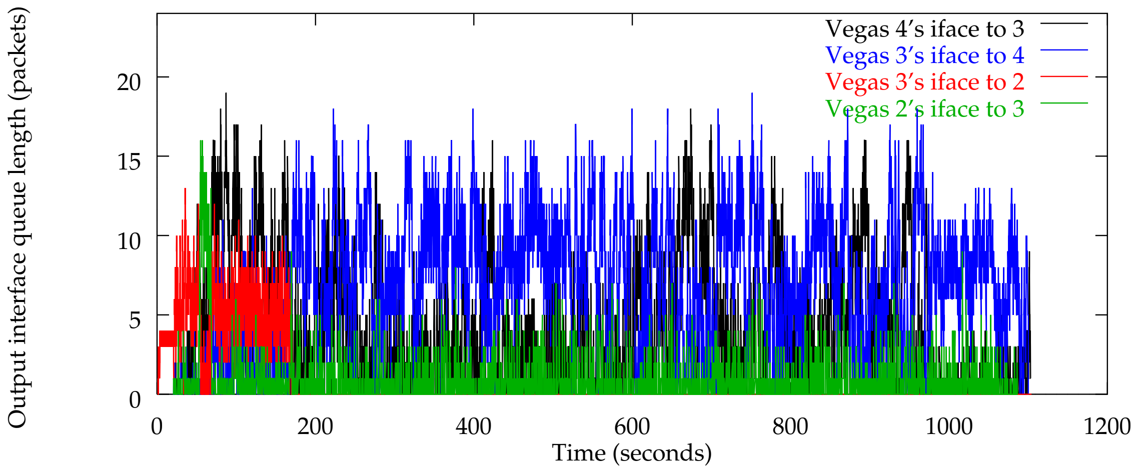


Figure 42: Queue Sizes in Scenario 5 by Vegas

In Scenario 5, TCP Vegas exhibits severe congestion and packet loss in routers 1, 2, 3, 4 and 5, with long-term queue lengths above 10 packets, as shown in Figure 42. 140 packets were lost, mainly by routers 3 and 4, with sources 7, 9, 10 and 15 losing the majority.

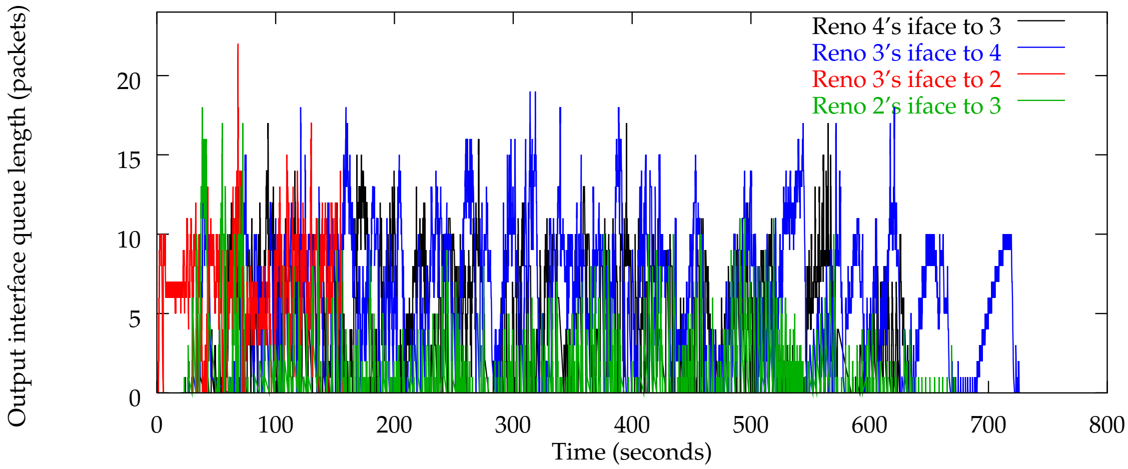


Figure 43: Queue Sizes in Scenario 5 by Reno

TCP Reno exhibits even worse congestion and packet loss in routers 1, 2, 3, 4 and 5 than Vegas, with long-term queue lengths also above 10 packets, as shown in Figure 43. 405 packets were lost, mainly by routers 3 and 4, with all sources except 16, 19, 20, 24 and 25 losing more than 10 packets.

The end times for all sources are given in the following table. Although some Vegas sources finish earlier than TRUMP (sources 8 and 11), many finish significantly later than TRUMP (sources 7, 9, 10, 12, 13 and 15). This demonstrates that TCP Vegas treats long-haul connections unfairly, giving more bandwidth to short round-trip time connections.

Source Node	End Time	Avg Rate	End to end Avg	End to end S.Dev	Round-trip Avg	Round-trip S.Dev	Pkts Lost	Pkts Retx'd
TRUMP 6	12.1	997051	0.02	0.00	0.03	0.00	0	2
Vegas 6	12.1	1000324	0.05	0.00	0.06	0.00	0	0
Reno 6	16.0	780865	0.05	0.04	0.09	0.02	0	31
TRUMP 7	539.1	23263	1.17	0.03	2.17	0.12	0	3
Vegas 7	1086.5	13097	2.13	0.52	3.33	0.55	33	155
Reno 7	671.4	20576	2.29	0.66	3.39	0.46	132	108
TRUMP 8	83.3	36418	0.16	0.05	0.27	0.08	0	0
Vegas 8	67.6	72338	0.80	0.28	0.95	0.11	0	0
Reno 8	74.7	73494	0.61	0.34	1.09	0.15	30	39
TRUMP 9	605.3	22205	1.43	0.13	2.38	0.21	0	5
Vegas 9	1066.1	24940	2.08	0.48	1.33	0.77	37	1087
Reno 9	604.8	23258	2.16	0.55	3.47	0.38	51	49
TRUMP 10	605.8	22213	1.15	0.02	2.01	0.06	0	5
Vegas 10	1102.6	24713	1.78	0.45	0.97	0.25	59	1135
Reno 10	725.6	20312	1.81	0.57	2.86	0.46	65	114

Source Node	End Time	Avg Rate	End to end Avg	End to end S.Dev	Round-trip Avg	Round-trip S.Dev	Pkts Lost	Pkts Retx'd
TRUMP 6	315.0	40894	4.84	0.12	8.86	0.12	0	46
Vegas 6	486.0	26246	4.93	0.39	8.88	0.75	1	20
Reno 6	503.5	25191	5.18	0.44	9.17	0.35	11	14
TRUMP 7	79.5	32157	1.19	0.00	2.20	0.00	0	5
Vegas 7	100.4	41549	1.56	0.33	1.54	0.76	0	138
Reno 7	1308.1	6610	1.33	0.39	2.62	0.47	0	63
TRUMP 9	153.6	32157	1.19	0.00	2.19	0.00	0	5
Vegas 9	168.5	44023	1.68	0.44	1.53	0.83	0	138
Reno 9	151.2	36528	1.89	0.54	2.75	0.33	11	20
TRUMP 11	189.0	32156	1.27	0.09	2.28	0.09	0	6
Vegas 11	204.3	46296	1.54	0.29	1.34	0.72	0	155
Reno 11	303.8	41912	1.53	0.38	2.42	0.39	19	31
TRUMP 13	223.9	32156	1.37	0.09	2.37	0.09	0	6
Vegas 13	201.9	45570	1.75	0.36	2.70	0.42	0	3
Reno 13	632.5	17057	1.39	0.37	2.63	0.49	26	45

As can be seen, source 6 is penalised by both TCP versions because it is a long-haul traffic flow. The short-haul traffic flows end much faster in TCP Vegas than in TRUMP, indicating that they are favoured.

With TRUMP, the only congested link is $4 \rightarrow 5$, with router 4's queue oscillating between 1 and 2 packets queued from time 156 to time 224. The transmission rates calculated by RBCC are smooth. End-to-end and round-trip times for TRUMP/RBCC are surprisingly flat, despite the congestion. In comparison, TCP Reno and Vegas exhibit network congestion in several routers, and some packets are lost; this is shown in Figure 45.

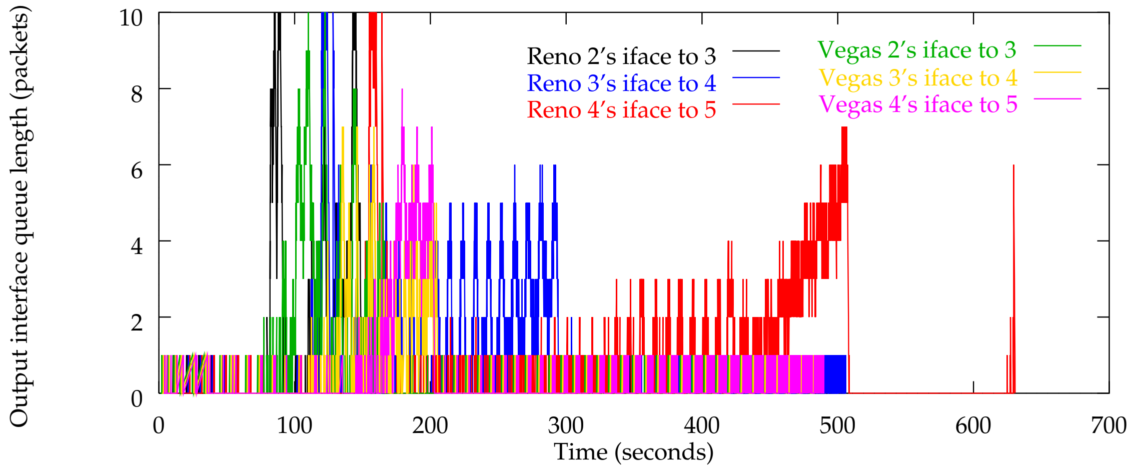


Figure 45: TCP Queue Lengths in Scenario 6

9.10 Scenario 7

Scenario 7 is the same as Scenario 2. However, in this scenario, the traffic flows vary their packet sizes evenly across a range of sizes:

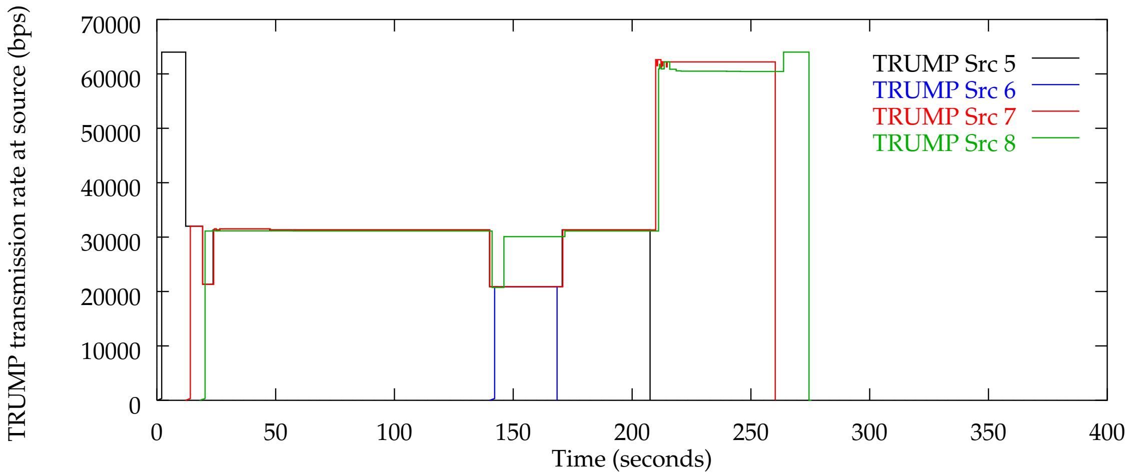


Figure 47: TRUMP Source Rates in Scenario 7

9.11 Scenario 8

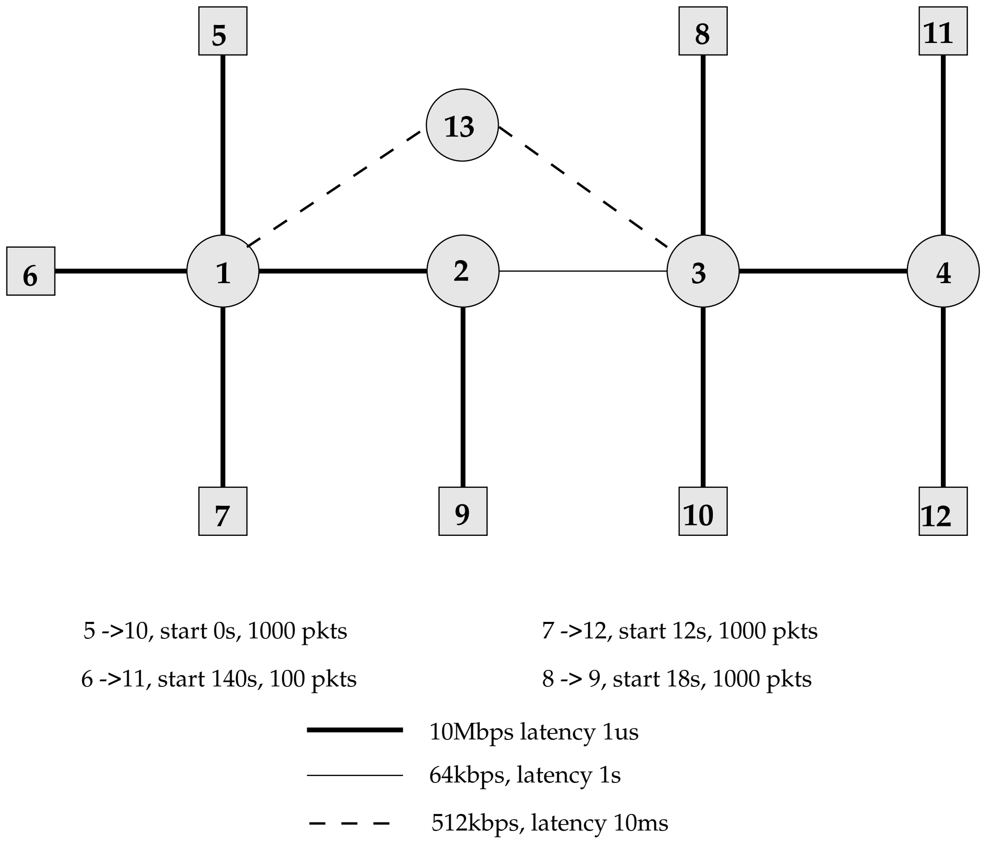


Figure 48: Scenario 8

Scenario 8 is shown in Figure 48. This is similar to Scenario 2, but there is an extra router, 13, and two extra links which connect router 13 to routers 1 and 3. Initially, the new links and router are not used. At time 100, router 1 changes its routing table to forward packets for destination 10 via router 13. This affects the data flow of source 5. The new route via router 13 offers more bandwidth than the original route (1 $\rightarrow$ 2 $\rightarrow$ 3).

As flow 6 $\rightarrow$ 11 doesn't come online until time 140, only the flows from sources 5, 7 and 8 are affected at the route change time of 100 seconds. The route change gives the flow

5 → 10 much more bandwidth; it should be able to use the full 512kbps across links (1 → 13 → 3). Flow 7 → 12 should then be able to use the bandwidth previously used by flow 5 → 10. Let us examine the route change on TRUMP, Vegas and Reno in turn.

The first TRUMP data packet from source 5 that crosses the new route has its **Rate** field updated with the new bandwidth, and this value reaches source 5 at time 100.24. The new rate calculated by RBCC is exactly 512kbps. Router 2 detects the lost of flow 5 → 10 at time 105.0, as described in Section 8.4. It calculates a new rate for source 7, which it receives at time 107.6. The new rate is 62,361bps, which provides for the acknowledgments of flow 8 → 9. The rates calculated by RBCC are shown in Figure 49.

TCP Vegas begins to open its window, prompted by round-trip time reductions, at time 100.4, and Reno begins to open its window at time 101.1, as shown in Figure 50.

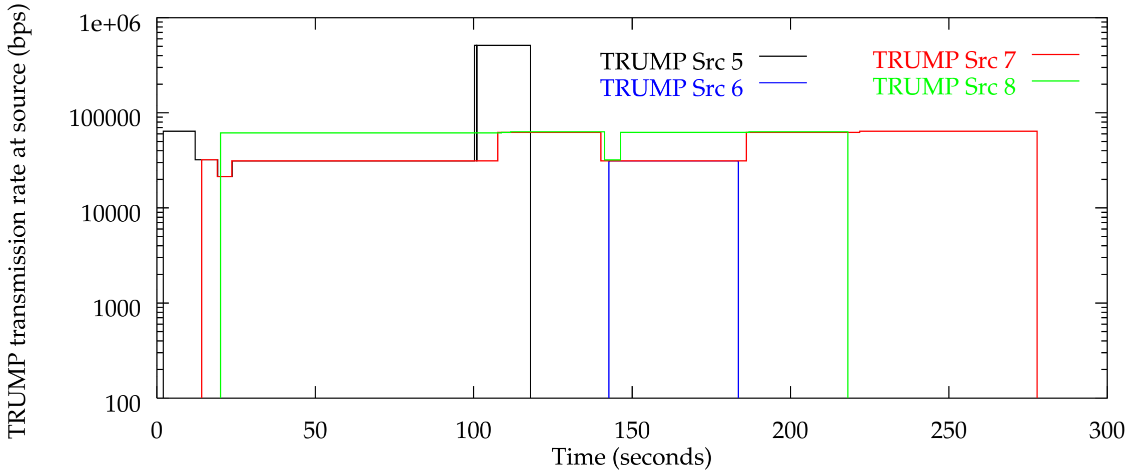


Figure 49: RBCC Rates in Scenario 8

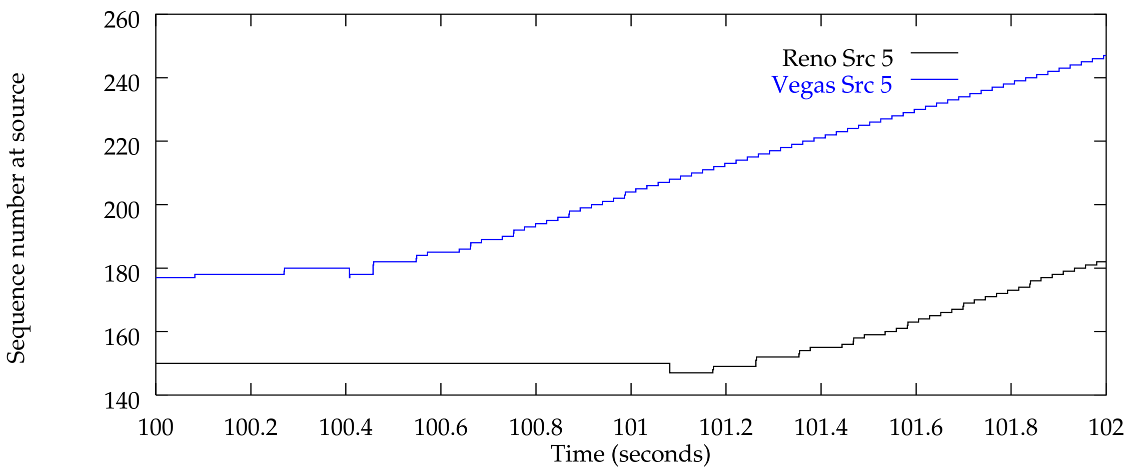


Figure 50: TCP Source 5 Sequence Numbers in Scenario 8

What is difficult to explain is the apparent transmission rate of source 7 in both TCP Vegas and Reno. Optimally, source 7's rate should double at the route change time of 100 seconds, halve when source 6 comes online at time 140 seconds, and redouble when source 6 finishes transmission. Reno appears to show this behaviour, as seen in the sequence number diagram in Figure 51; however, the initial rate is kept low by the Slow Start mechanism. Vegas seems to reach a stable rate just before time 50 seconds, increases its rate after the route change, slows down in response to source 6, but then never regains its original rate once source 6 finishes transmitting. This behaviour was also seen in Scenario 2, and appears to be a problem in TCP.

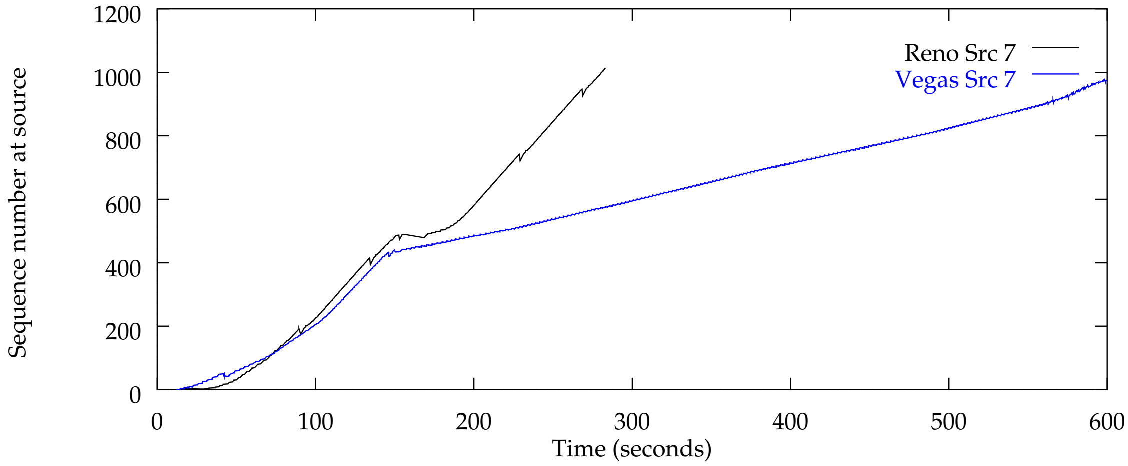


Figure 51: TCP Source 7 Sequence Numbers in Scenario 8

The following table gives the per-flow results for Scenario 8. Overall, source 5 goes faster for Vegas than TRUMP/RBCC, but Vegas loses packets. Despite a 5-second delay to detect the route change by router 2, TRUMP's source 7 still achieves a higher rate than either TCP, with no packet loss. And as with Scenario 2, source 8 achieves a much higher rate with TRUMP than either version of TCP.

Source Node	End Time	Avg Rate	End to end Avg	End to end S.Dev	Round-trip Avg	Round-trip S.Dev	Pkts Lost	Pkts Retx'd
TRUMP 5	118.2	103793	0.42	0.56	0.74	1.03	0	5
Vegas 5	119.8	110008	0.56	0.79	0.50	0.80	5	91
Reno 5	121.3	107848	0.53	0.74	0.56	0.86	88	77
TRUMP 6	182.7	31465	1.49	0.02	2.58	0.06	0	6
Vegas 6	224.3	33892	2.07	0.45	1.02	0.46	4	134
Reno 6	179.9	37852	1.80	0.54	2.64	0.34	7	12
TRUMP 7	275.9	46344	1.22	0.06	2.30	0.09	0	11
Vegas 7	599.7	33201	1.57	0.40	1.68	1.12	7	647
Reno 7	283.1	45282	2.03	0.59	2.98	0.49	7	8
TRUMP 8	218.1	61402	1.19	0.00	2.38	0.10	0	12
Vegas 8	484.6	38266	1.51	0.23	1.95	0.94	0	474
Reno 8	599.1	11688	1.35	0.39	2.63	0.43	30	33

9.12 Scenario 9

Scenario 9 is a modification to Scenario 6 (Figure 44), but with the $7 \rightarrow 8$ cross-flow starting at time 10, after source 6 learns of its initial rate. The links 1 to 4 impose a 3-second delay in the feedback of rate information to source 6 from its destination.

Section 7.1 suggests that, although selective acknowledgments lower the network load due to acknowledgment traffic, the effect of acknowledgment delays may induce higher network congestion. This scenario examines the effect of large selective acknowledgment sizes, combined with Rate Quenches, on the framework.

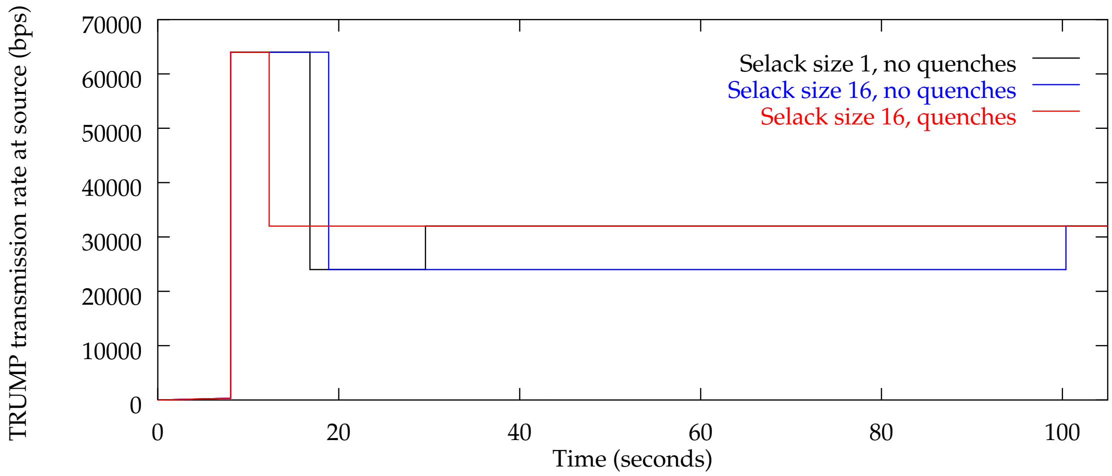


Figure 52: Source 6's Initial Rate Changes in Scenario 9

Figure 52 shows the rates used by TRUMP for source 6, for several selective acknowledgment sizes and Rate Quench combinations. For all three plots, source 6 obtains an initial rate at time 8 seconds.

The black plot shows source 6 using a selective acknowledgment size of 1, with no Rate Quenches being generated by the network. Source 7 starts at time 10, and requires some of the bandwidth in use by source 6. RBCC recalculates a new rate for source 6, and this arrives at the source at time 16.8 seconds. Because of the large round-trip time, the packet queue in router 1 exceeds the congestion avoidance threshold at time 13.6 seconds, and two packets for source 6 are lost around time 16 seconds. The link-delayed new rate for source 6 arrives at the source at time 17 seconds, and has been scaled by 0.75 by the network's congestion avoidance scheme. It is only after router 1's queues have fallen, at time 29, that source 6 receives its RBCC rate allocation unscaled.

The blue plot shows source 6's operation with a selective acknowledgment of 16, with no Rate Quenches being generated by the network. The effect of the larger acknowledgment size is to delay the reception of a new rate for source 6 until time 19 seconds. Again, because of router queue sizes, this has been scaled by the network's congestion avoidance scheme. The time until the avoidance scheme terminates scaling is also delayed, both by the selective acknowledgment delays and by the time to drain router 1's buffers.

The red plot shows source 6's operation with a selective acknowledgment of 16, and with Rate Quenches being generated by the network. The effect of Rate Quenches is dramatic. A new rate for source 6 is returned at time 12 seconds, and is not scaled as the buffer occupancy in router 1 has not exceed the congestion avoidance threshold.

Scenario 9 shows that high round-trip times do slow the congestion framework's reaction time, and delays due to large selective acknowledgment sizes increases this reaction time further. Rate Quenches appear to be a useful mechanism to circumvent these delays.

9.13 Summary

The simulation of nine hand-crafted network scenarios in this chapter show that the proposed rate-based congestion framework, with TRUMP and RBCC as component functions, appears to work very well at congestion control. Let us review the congestion characteristics for both TRUMP/RBCC and for the two flavours of TCP.