ENSC 833-3: NETWORK PROTOCOLS AND
PERFORMANCE

CMPT 885-3: SPECIAL TOPICS: HIGH-PERFORMANCE
NETWORKS

FINAL PROJECT

Performance Comparison of TFRC and TCP

Spring 2002

Yi Zheng and Jian Wen

{zyi,jwena}@cs.sfu.ca

http://www.sfu.ca/~jwena/ensc835_project.htm

1. ABSTRACT

TCP congestion control is the most widely used congestion control protocol and a very successful one. It is a special case of Additive Increase and Multiplicative Decrease congestion control, with Increase parameter set to 1 and decrease parameter set 0.5. We call Additive Increase and Multiplicative Decrease congestion control as TCP (a, b) in this report, with a standing for increase parameter and b for decrease parameter. With the aim of transmitting voice and video smoothly, TCP Friendly Rate Control (TFRC) was developed..

In this report, the performance of TCP, TCP (a, b) and TFRC in a congested network environment are compared. The effects of Window Size, Queuing mechanisms and Transmission delays are studied.

2. INTRODUCTION

In today's Internet, hundreds of millions of computers are interconnected by networks while hundreds of thousands of computers join the Internet. Congestion control in packet networks has been proven to be an important problem, without congestion control, each computer or flow will insatiably try to seize more bandwidth while finally paralyze the whole Internet. A series of congestion control algorithms have been developed. How do these algorithms compare to each other, how to choose from these algorithms to implement in a specific network environment, how to efficiently utilize bandwidth while reducing loss rate and share the bandwidth fairly among flows raise a challenge for network researchers and designers.

TCP congestion control is the most widely used congestion control protocol and a very successful one. It is a special case of Additive Increase and Multiplicative Decrease congestion control, with Increase parameter set to 1 and decrease parameter set 0.5.

2.1 TCP (a, b) Congestion Control

TCP has a receiver-controlled window size. Sender uses a congestion window (cwnd). Transmission window= min (cwnd, receive-window). Sender adjusts cwnd by observing network congestion level. The principle of adjustment of cwnd is additive increase and multiplicative decrease. In the absence of congestion, increase cwnd by one mss per round trip time, upon packet loss, decrease cwnd by half.

TCP congestion control is featured by three phases: Slow Start, Congestion Avoidance and Fast Retransmit/Recovery.

Slow start phase:

At beginning of slow start phase, cwnd is set to 1*MSS (Maximum Segment Size), upon receiving every new ACK, the cwnd grows by one MSS, which results in doubling cwnd every round trip time.

Congestion Avoidance phase:

When RTO (Round Trip TimeOut) occurs, remember the half point of cwnd at the loss point("congestion-threshold", or Slow-Start-threshold, ssthresh)

When cwnd < ssthresh, increase cwnd exponentially,
When cwnd >= ssthresh, increase cwnd linearly,

Fast Retransmit/Recover phase:

If three duplicat ACK's detected,
-Retransmit segment identified by duplicate ACKs
-Set ssthresh to 0.5 times window size
-Set window size to ssthresh
-As long as dup acks keep coming, use the following window size equation:
  window + number of dup ACKs
-If ack changes, go back to avoidance
-If RTO occurs, back to slow start

TCP (a, b) generalizes TCP by parameterizing the congestion window increase value and decrease ratio. In TCP (a, b), a congestion epoch is defined as a period beginning with a congestion window of packets. The congestion window is increased additively to a congestion window of W, when a packet is dropped. The congestion window is then decreased multiplicatively back to 1-b)*W as show in Figure 1. Each congestion epoch consists of $(\frac{b}{a}W+1)$ round-trip times.

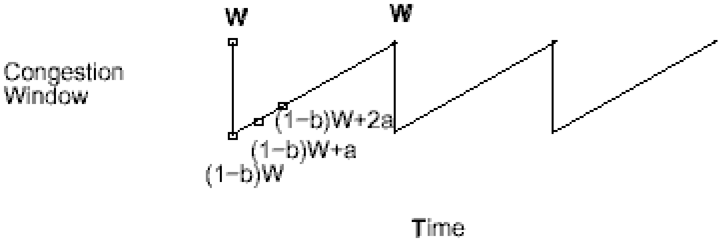

Figure 1. TCP (a, b) congestion window

Define S as the sending rate in packets per RTT (Round Trip Time), the average sending rate over one congestion epoch is T = $\frac{2-b}{2}W$ packets per second, which gives a total of $\frac{b(2-b)}{2a}W^2$ packets in one congestion epoch. The packet drop rate $p=\frac{2a}{b(2-b)W^2+a(2-b)W}$,

The precise version of the TCP (a, b) response function is

$$T=\frac{pa(b-2)+\sqrt{p(b-2)a(pba-8b-2pa)}}{4pbR}$$

2.2 TFRC congestion control

TFRC is designed to be reasonably fair when competing for bandwidth with TCP flows, where a flow is ``reasonably fair'' if its sending rate is generally within a factor of two of the sending rate of a TCP flow under the same conditions. However TFRC has a much lower variation of throughput over time compared with TCP, which makes it more suitable for applications such as telephony or streaming media where a relatively smooth sending rate is of importance. The penalty of having smoother throughput than TCP while competing fairly for bandwidth is that TFRC responds slower than TCP to changes in available bandwidth. Thus TFRC should only be used when the application has a requirement for smooth throughput, in particular, avoiding TCP's halving of the sending rate in response to a single packet drop.

TFRC is designed for applications that use a fixed packet size, and vary their sending rate in packets per second in response to congestion. Some audio applications require a fixed interval of time between packets and vary their packet size instead of their packet rate in response to congestion.

TFRC is a receiver-based mechanism, with the calculation of the congestion control information (i.e., the loss event rate) in the data receiver rather in the data sender. This is well suited to an application where the sender is a large server handling many concurrent connections, and the receiver has more memory and CPU cycles available for computation.

The throughput equation for TFRC is:

$$X = \frac{s}{R * sqrt(2 * b * p / 3) + (t\_RTO * (3 * b * p / 8) * p * (1 + 32 * p^2)}$$

Where: X: the transmit rate in bytes/second
s : the packet size in bytes
R: the round trip time in seconds
p : the loss event rate, of the number of loss events as a fraction of the number of packets transmitted
t_RTO : the TCP retransmission timeout value in seconds
b : the number of packets acknowledged by a single TCP acknowledgement

TFRC is a receiver-based mechanism, it works in the following way:

1.The receiver measures the loss event rate and feeds this information back to the sender.

2.The sender also uses these feedback messages to measure the round- trip time (RTT).

3.The loss event rate and RTT are then fed into TFRC's throughput equation, giving the acceptable transmit rate.

4.The sender then adjusts its transmit rate to match the calculated rate.

3. MAIN SECTIONS

3.1 Project goal and structure

The goal of our project is to investigate the fairness, smoothness, responsiveness, and aggressiveness of TCP, TCP (a, b) and TFRC.

The structure of our Simulations are as follows:

1. Study the effect of maximum window size on the bandwidth utilization and loss rate of TCP

2. Study the effect of number of flows on the bandwidth utilization and loss rate

3. Compare the bandwidth utilization and loss rate of TCP, TCP (a, b) and TFRC with network bandwidth solely occupied by one kind of these flows, as in Table 1:

	TCP	TCP (a, b)	TFRC
Bandwidth Utilization			
Loss rate			

Table 1. Comparison of bandwidth utilization & loss rate

4. Compare the bandwidth utilization and smoothness when the network bottle link are shared by two kinds of these flows, with same flow numbers for each kind, that is comparison as Table 2:

	TCP vs TCP (a, b)	TCP vs TFRC	TCP (a, b) vs TFRC
Throughput (mean)			
Smoothness(variance)			

Table 2. Comparison of Throughput and smoothness

5. Compare the effect of trunk link capacity on Friendliness metrics and loss rate of TCP, TCP (a, b) and TFRC, as Table 3:

	TCP vs TCP (a, b)		TCP vs TFRC		TCP(a,b) vs TFRC	
Trunk link 15Mb						
Trunk link 60Mb						

Table 3. Comparison of Throughput and smoothness

6. Study the effect of transmission delay on the bandwidth utilization and smoothness of TCP, TCP (a, b) and TFRC, comparison is as Table 4:

7. Compare the effects of Random Early Drop (RED) and DropTail on bandwidth utilization and smoothness of TCP, TCP (a, b) and TFRC, that is comparison as the Table 4:

	TCP vs TCP (a, b)	TCP vs TFRC
Throughput (mean)		
Smoothness (variance)		

Table 4. Comparison of transmission delay on Throughput and smoothness

In this part, we also study the effect of parameter ECN in RED on the loss rate of the network

3.2 Simulation scenario:

The basic scenario is as Figure 2, there will be some changes in different parts of our study, we will point out the difference.

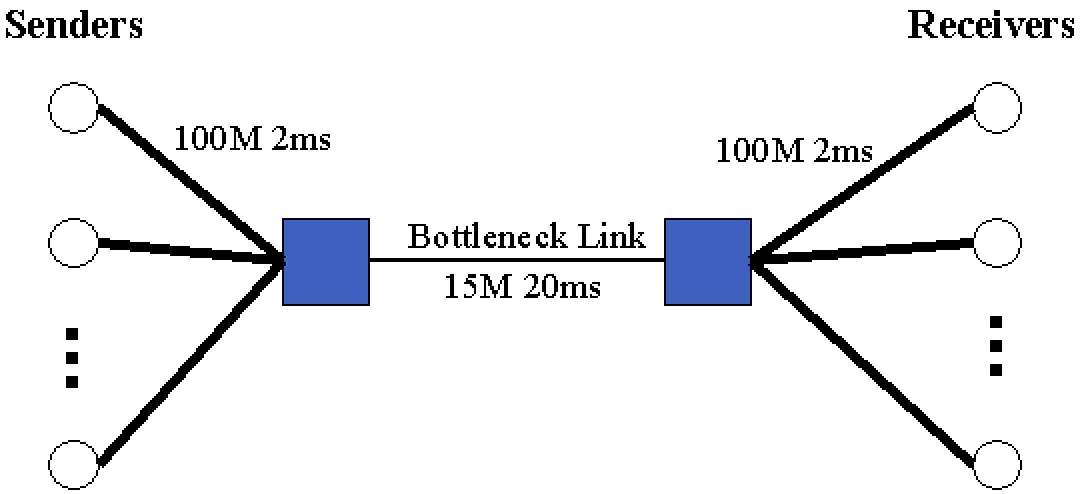

Figure 2. Simulation Scenario

In this scenario, a number of flows share the same bottleneck. The number of flows is varied in our simulations. The trunk link capacity is 15Mb or 60Mb, transmission delay on the trunk link is 20ms, the link capacity from sender to the entrance node of trunk link is 100Mb, transmission delay 2ms, and the link capacity from the outlet node of trunk link to receiver node is also 100 Mb, transmission delay 2ms. We also add some random flows as the background noise to approximate the real network.

Why choose TCP (1/5,1/8) and TCP (3/7,1/4) for comparison:

Applying TCP(a,b) Transmission rate function from $T = \frac{pa(b-2)+\sqrt{p(b-2)a(pba-8b-2pa)}}{4pbR}$

To TCP (1,0.5), we get $T_{1,1/2,R,p} = \frac{\sqrt{1.5}}{R\sqrt{p}}$, in order for TCP (a, b) to have the same long-term sending rate in relationship to the packet drop rate as TCP (1,0.5), we would like to have the same response functions: $T_{a,b,R,p} = T_{1,1/2,R,p}$, this gives $\frac{\sqrt{2-b}\sqrt{a}}{\sqrt{2b}R\sqrt{p}} = \frac{\sqrt{1.5}}{R\sqrt{p}}$, which is equivalent to: $a = \frac{3b}{(2-b)}$, this equation suggests that TCP(1/5,1/8) and TCP(3/7,1/4) should compete reasonable fairly with TCP(1,1/2).

3.3 Window size effect on bandwidth utilization & loss rate

Simulation Scenario: In this simulation, we use one TCP flow with trunk link 15Mb, transmission delay 20ms, the link capacity from sender to the entrance node of trunk link 100 Mb, transmission delay 2ms, and the link capacity from the outlet node of trunk link to receiver node is also 100 Mb, transmission delay 2ms. Queuing mechanism we use Random Early Drop (RED). We change the window size from 10 to 150, the simulation result is as Figure 3, the upper one is the bandwidth utilization as window size changes, the lower one is the corresponding loss rate:

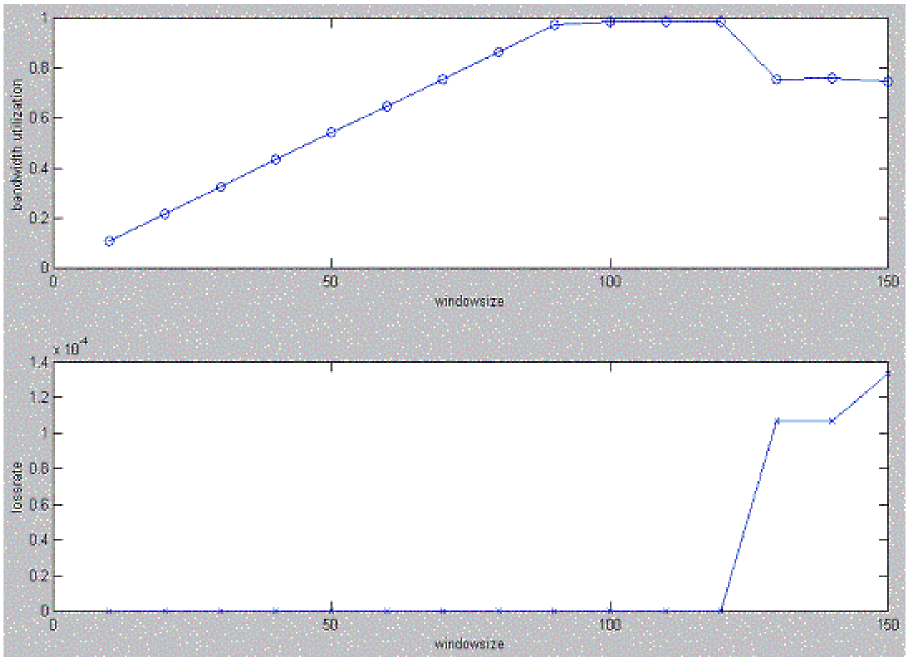

Figure 3. maximum window size effect on bandwidth utilization and loss rate

From Figure 3 we can see:

- When the maximum window size is small, the throughput of the TCP flow is limited by maximum window size, there is no congestion (loss rate zero), but bandwidth utilization is low. As the window size increase, the bandwidth utilization increases, loss rate still zero, when the window size reach 90, the bandwidth utilization is high and no packet loss. When window size reach 120, the bandwidth utilization reaches maximum and no packet loss. When window size is beyond 120, bandwidth utilization decreases and loss rate increases. When window size further increases, the loss rate increases but bandwidth utilization reaches an equalibrum.

Conclusion: Maximum window size affects the bandwidth utilization and packet loss rate

3.4 Throughput and loss rate comparison

Simulation Scenario: In this simulation, the trunk link 15Mb, transmission delay 20ms, the link capacity from sender to the entrance node of trunk link 100 Mb, transmission delay 2ms, and the link capacity from the outlet node of trunk link to receiver node is also 100 Mb, transmission delay 2ms. Queuing mechanism we use Random Early Drop(RED). The trunk link are all occupied by TCP flows. We change the number of flows from 10 to 70, the simulation result is as Figure 4, the upper one is the bandwidth utilization as flow number changes, the lower one is the corresponding loss rate:

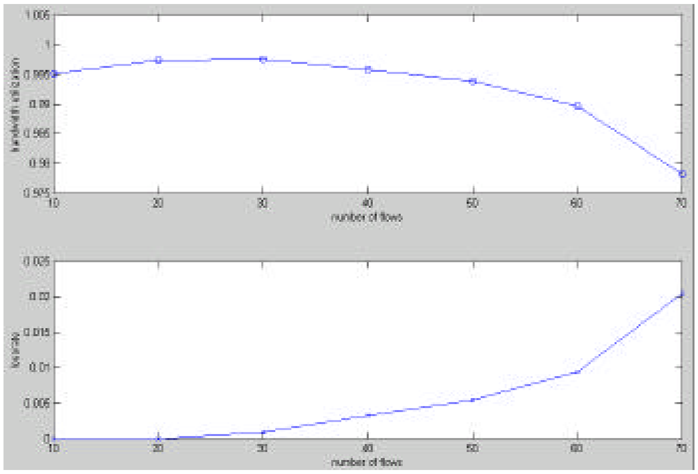

Figure 4. Effect of flow number on bandwidth utilization and loss rate

From Figure 4 we can see:

- As the number of flows increase from 10 to 20, the bandwidth utilization increases while there is no packet loss, the bandwidth utilization reaches maximum when TCP flow number is 20, as flow number further increases, the bandwidth utilization decreases and loss rate increases. We can see that congestion happens.

Conclusion: As flow number increases, congestion will be more severe

3.5 Throughput and loss rate comparison for TCP, TCP (a, b) and TFRC

Simulation Scenario: In this simulation, the trunk link 15Mb, transmission delay 20ms, the link capacity from sender to the entrance node of trunk link 100 Mb, transmission delay 2ms, and the link capacity from the outlet node of trunk link to receiver node is also 100 Mb, transmission delay 2ms. Queuing mechanism we use Random Early Drop (RED). The trunk link are all occupied by TCP flows, TCP (a, b) flows or TFRC flows. We change the number of flows from

10 to 70, the simulation result is as Figure 5, the upper one is the bandwidth utilization as flow number changes, the lower one is the corresponding loss rate:

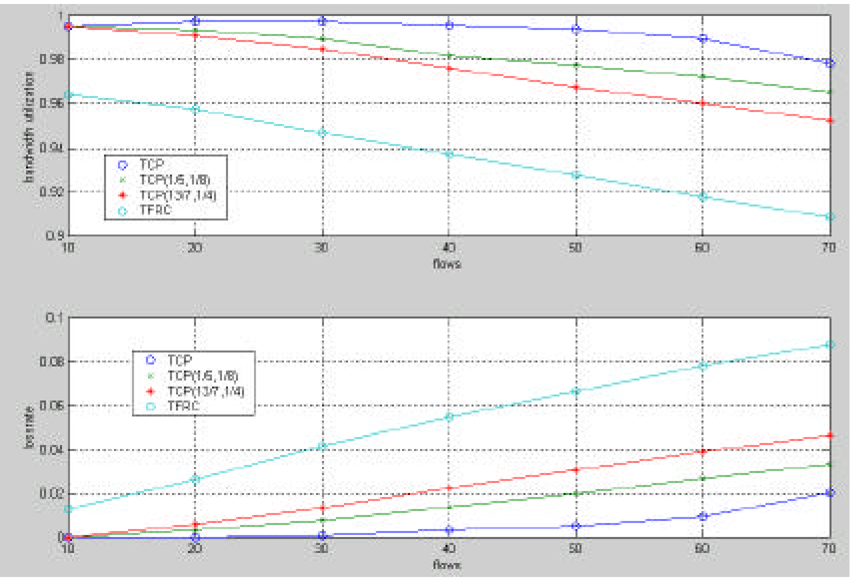

Figure 5. Comparison of bandwidth utilization and loss rate of TCP, TCP (a, b) and TFRC

From Figure 5 we can see:

The order of bandwidth utilization is TCP, TCP (1/5,1/8), TCP (3/7,1/4) and TFRC, with TCP utilize the bandwidth most efficiently. The loss rate order highest loss rate for TFRC, TCP (3/7,1/4) second highest, TCP (1/5,1/8) third highest and TCP the lowest.

As the flow number increase, the bandwidth utilization decreases and loss rate increases for all four kinds of flows.

Conclusion: When the bottleneck is shared by one kind of flows, TCP utilize the bandwidth most efficiently, TCP (1/5,1/8) second, TCP (3/7,1/4) third and TFRC last. As the flow number increase, the bandwidth utilization decreases and loss rate increases for all four kinds of flows.

3.6 Throughput and smoothness comparison

In this part, we will compare the throughput and smoothness of TCP, TCP (a, b) and TFRC when two kinds of them coexist in the network and share the bottleneck. The comparison is presented as follows:

- TCP (1/5,1/8) vs TCP (1,1/2)
- TCP (3/7,1/4) vs TCP (1,1/2)
- TCP (1/5,1/8) vs TFRC
- TCP (3/7,1/4) vs TFRC
- TFRC vs TCP (1,1/2)

Simulation Scenario: In this simulation, the trunk link 15Mb, transmission delay 20ms, the link capacity from sender to the entrance node of trunk link 100 Mb, transmission delay 2ms, and the link capacity from the outlet node of trunk link to receiver node is also 100 Mb, transmission delay 2ms. Queuing mechanism we use Random Early Drop (RED). The trunk link are all occupied by two kinds of flows, each kind occupies 8 flows. For example, in the first case, there are 8 TCP (1/5,1/8) flows and 8 TCP flows. In each of the following case, the left 4 figures are the throughput of one kind of flow, the right 4 figures are another kind. We choose every other flow to make our figure representative. The total simulation time is 90 seconds.

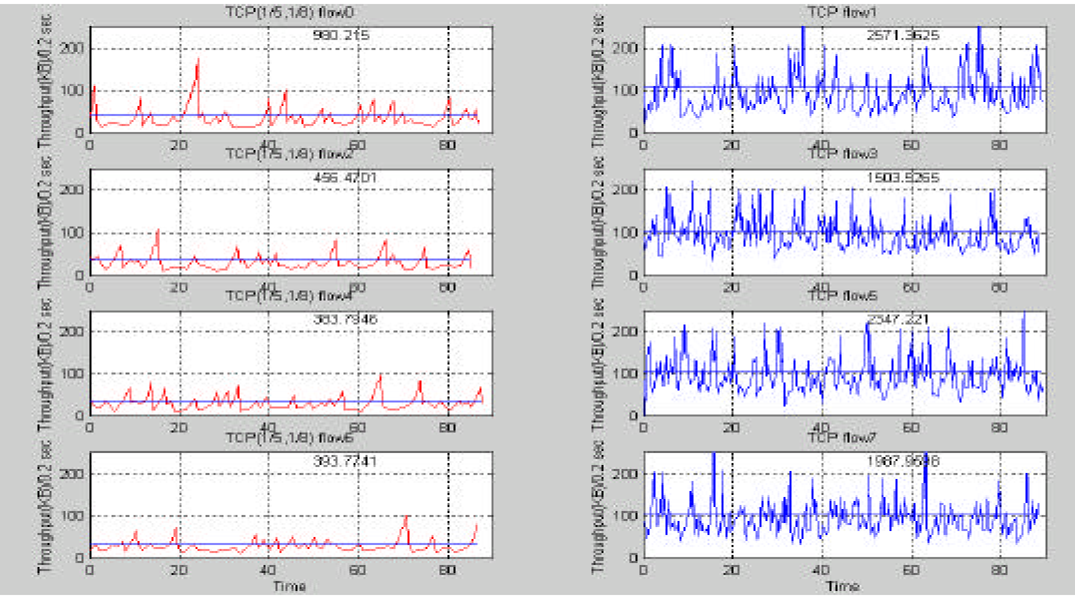

Figure 6. TCP (1/5,1/8) vs TCP

Case 1: TCP (1/5,1/8) Vs TCP (1,1/2)

From Figure 6 we can see:

The mean value of throughput of TCP (1/5,1/8) is about one half of TCP flows, but obviously these flows are smoother, as we can judge from the smaller throughput variance. From Figure 6 we can also see that the throughput of TCP (1/5,1/8) flows are similar and throughput of TCP (1,1/2) flows are similar.

Case 2: TCP (3/7,1/4) Vs TCP(1,1/2)

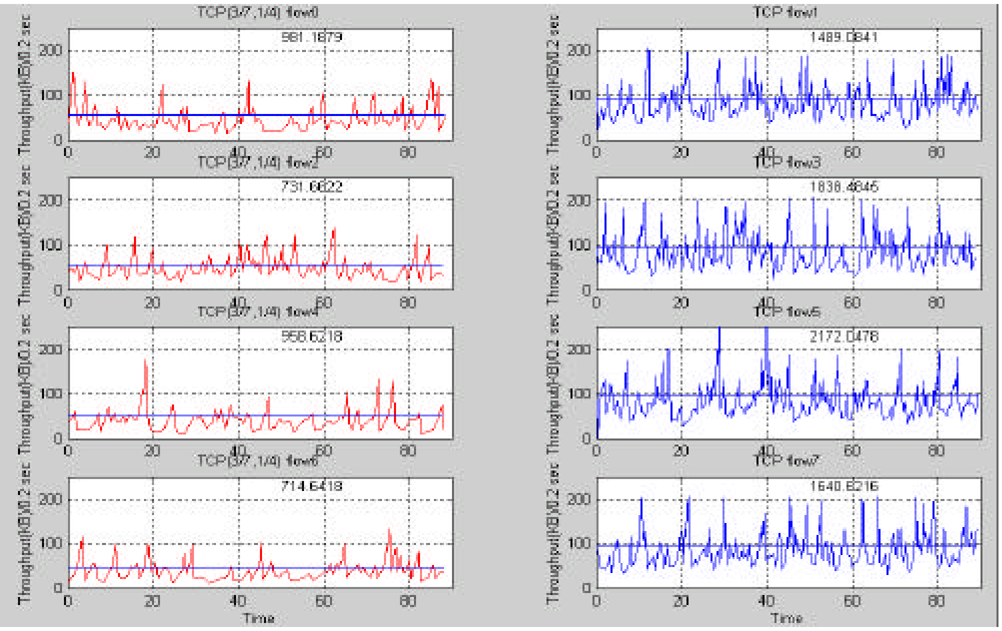

Figure 6. TCP (3/7,1/4) vs TCP

From Figure 6 we can see:

The mean value of throughput of TCP (3/7,1/4) is about one half of TCP flows, but obviously these flows are smoother, as we can judge from the smaller throughput variance. From Figure 6 we can also see that the throughput of TCP (3/7,1/4) flows are similar and throughput of TCP (1,1/2) flows are similar.

Comparing Figure 5 and Figure 6, we can see TCP(3/7,1/4) flows occupy more bandwidth compared with TCP(1/5,1/8) when coexist with TCP(1,1/2)

Conclusion:

TCP (a, b) flows occupy less bandwidth when coexist with TCP(1,1/2) flows but give smoother throughput.

Case 3. TCP (1/5,1/8) Vs TFRC

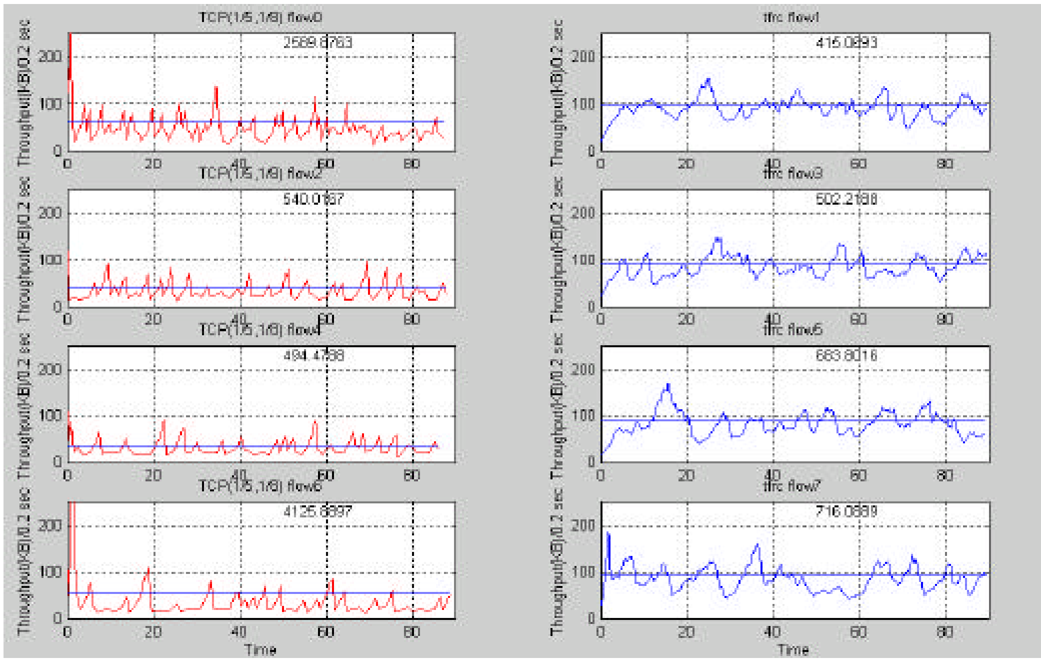

Figure 7. TCP (1/8,1/5) vs TFRC

From Figure 7 we can see:

The mean value of throughput of TCP (1/5,1/8) is about one half of TFRC flows, and less smooth than TFRC flows, as we can judge from the smaller normalized throughput variance. From Figure 6 we can also see that the throughput of TCP (1/5,1/8) flows vary somewhat while throughput of TFRC flows are similar.

Case 4. TCP (3/7,1/4) vs TFRC

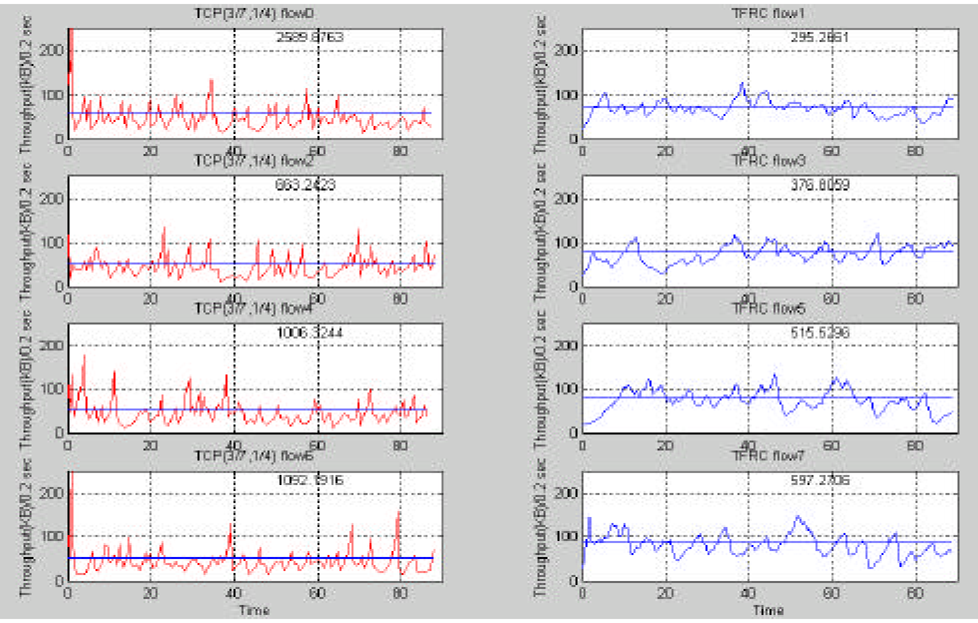

Figure 8. TCP (3/7,1/4) vs TFRC

From Figure 8 we can see:

The mean value of throughput of TCP (3/7,1/4) is about one half of TFRC flows, and less smooth than TFRC flows, as we can judge from the smaller normalized throughput variance. From Figure 7 we can also see that the throughput of TCP (3/7,1/4) flows vary somewhat while throughput of TFRC flows are similar.

Conclusion:

From Figure 6 and Figure 7 we can see that TCP (a, b) flows occupy less bandwidth when coexist with TFRC flows and give less smooth throughput than TFRC flows.

Case 5. TCP (1,1/2) Vs TFRC

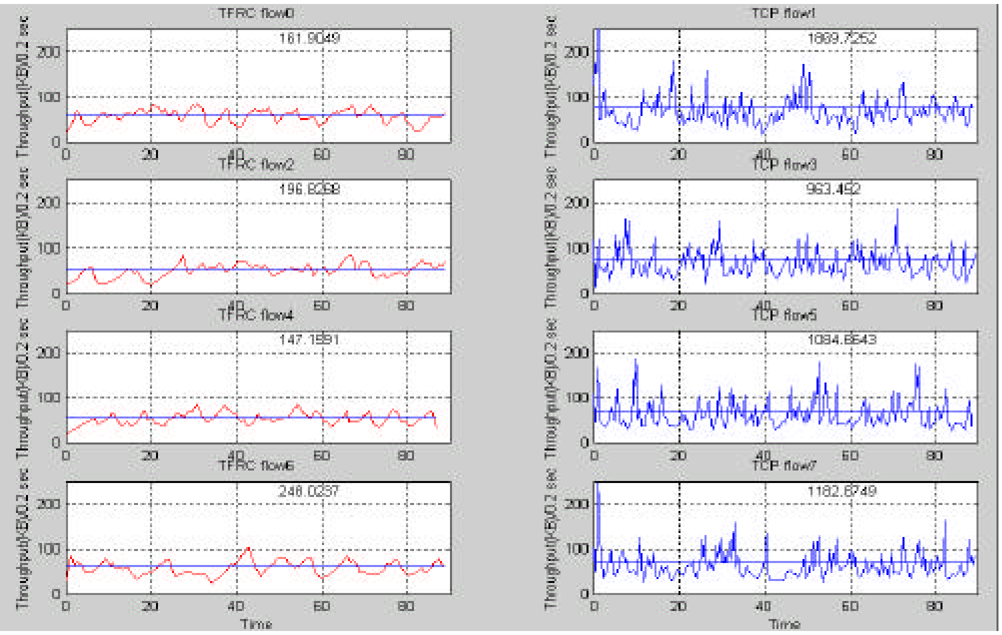

Figure 9. TFRC vs TCP

From Figure 9 we can see:

The mean value of throughput of TFRC flows is similar to throughput of TCP flows, but obviously smoother than TCP flows, as we can judge from the much smaller throughput variance. From Figure 6 we can also see that the throughput of TFRC flows are similar and throughput of TCP flows are similar.

3.7 Friendliness metrics with multiple flows and severe congestion

In this part, we compare the friendliness of TCP (a, b) and TFRC in competing with TCP for bandwidth as flow numbers increase and bottleneck capacity changes. The comparison is presented as follows:

	TCP vs TCP (a,b)		TCP vs TFRC		TCP(a,b) vs TFRC	
Trunk link 15 Mb						
Trunk link 60 Mb						

Table 5. Comparison of Throughput and smoothness

Simulation Scenario: In this simulation, the trunk link switch between 15Mb and 60Mb to compare the friendliness metrics and loss rate, transmission delay 20ms, the link capacity from sender to the entrance node of trunk link 100 Mb, transmission delay 2ms, and the link capacity from the outlet node of trunk link to receiver node is also 100 Mb, transmission delay 2ms. Queuing mechanism we use Random Early Drop (RED). The trunk link is all occupied by two kinds of flows, the flow number changes to give a comparison of Friendliness and loss rate.

Case 1. TCP vs TCP (3/7,1/4), bottleneck capacity 15Mb

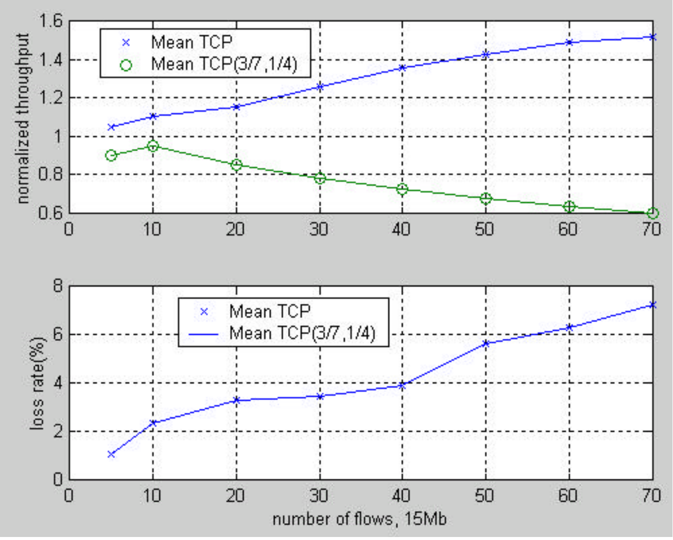

Figure 10. TCP vs TCP (3/7,1/4), 15Mb

From Figure 10, we can see:

When the number of flows is small, the congestion is not severe, the normalized throughput of TCP and TCP (3/7,1/4) flows are close, as the flow numbers increase, the congestion is more severe, as we can see from loss rate, and the difference of throughput between TCP flows and TCP (3/7,1/4) are larger.

Case 2. TCP vs TCP (3/7,1/4), bottleneck capacity 60Mb

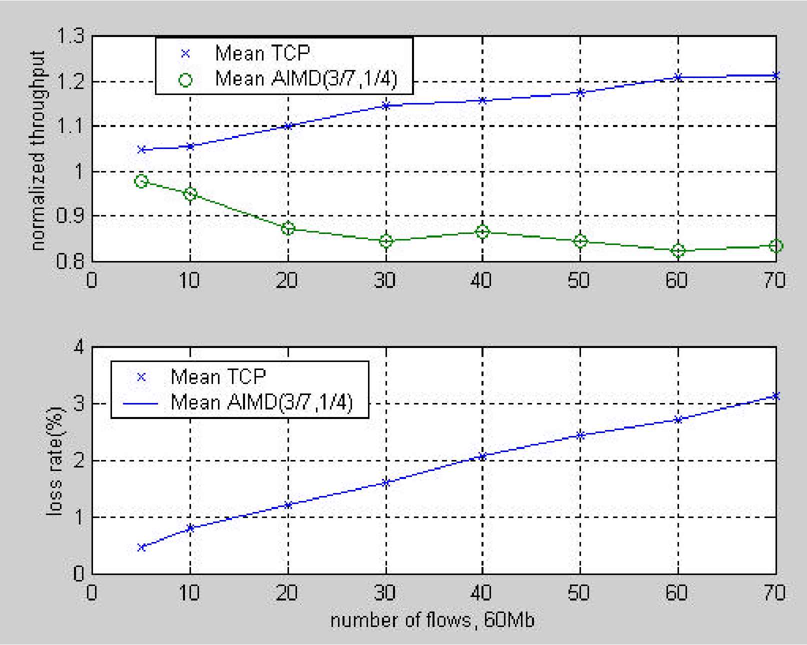

Figure 11. TCP vs TCP (3/7,1/4), 60Mb

From Figure 11, we can see:

When the number of flows is small, the congestion is not severe, the normalized throughput of TCP and TCP (3/7,1/4) flows are close, as the flow numbers increase, the congestion is more severe, as we can see from loss rate, and the difference of throughput between TCP flows and TCP (3/7,1/4) are larger.

Comparing Figure 9 and Figure 10, we can see:

As the bottleneck capacity increases from 15Mb to 60Mb, the congestion is less severe and the normalized throughput of TCP and TCP (3/7,1/4) becomes closer.

Conclusion:

When the number of flows is small, the congestion is not severe, the normalized throughput of TCP and TCP (3/7,1/4) flows are close, as the flow numbers increase, the congestion is more severe. Increase in the bottleneck capacity makes the competition for bandwidth more fairly while lessen the congestion situation.

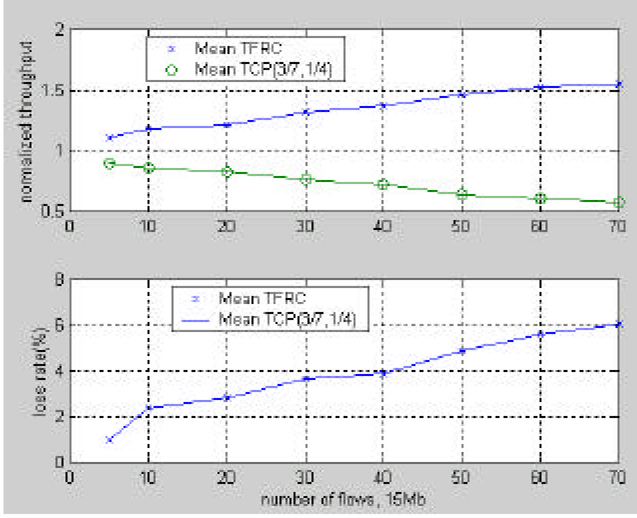

Figure 12. TFRC vs TCP(3/7,1/4), 15 Mb

Case 3. TCP (3/7,1/4) vs TFRC, bottleneck capacity 15Mb

From Figure 12, we can see:

When the number of flows is small, the congestion is not severe, the normalized throughput of TFRC and TCP (3/7,1/4) flows are close, as the flow numbers increase, the congestion is more severe, as we can see from loss rate, and the difference of throughput between TFRC flows and TCP (3/7,1/4) are larger.

Case 4. TCP (3/7,1/4) vs TFRC, bottleneck capacity 60Mb

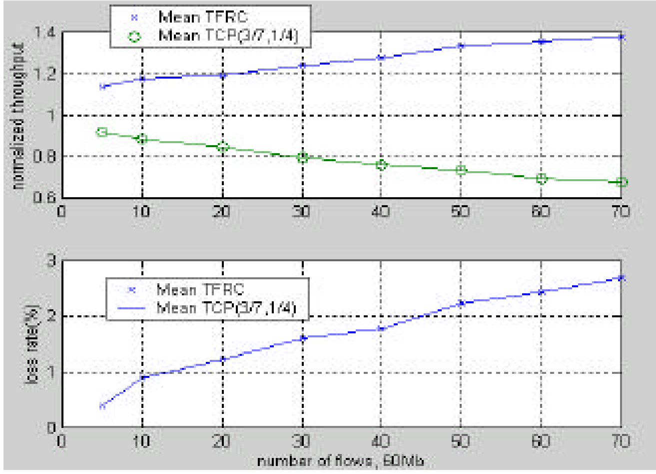

Figure 13. TFRC vs TCP (3/7,1/4), 60 Mb

From Figure 13, we can see:

When the number of flows is small, the congestion is not severe, the normalized throughput of TFRC and TCP (3/7,1/4) flows are close, as the flow numbers increase, the congestion is more severe, as we can see from loss rate, and the difference of throughput between TFRC flows and TCP (3/7,1/4) are larger.

Comparing Figure 11 and Figure 12, we can see:

As the bottleneck capacity increases from 15Mb to 60Mb, the congestion is less severe and the normalized throughput of TFRC and TCP (3/7,1/4) becomes closer.

Conclusion:

When the number of flows is small, the congestion is not severe, the normalized throughput of TFRC and TCP (3/7,1/4) flows are close, as the flow numbers increase, the congestion is more severe. Increase in the bottleneck capacity makes the competition for bandwidth more fairly while lessen the congestion situation.

Case 5. TCP vs TFRC, bottleneck capacity 15Mb

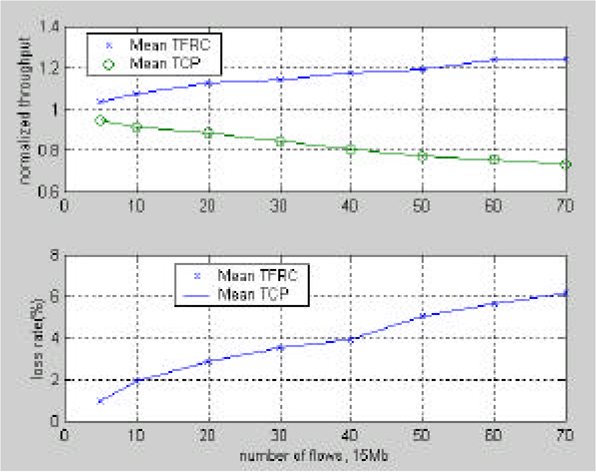

Figure 14. TCP vs TFRC, 15Mb

From Figure 14, we can see:

When the number of flows is small, the congestion is not severe, the normalized throughput of TFRC and TCP flows are close, as the flow numbers increase, the congestion is more severe, as we can see from loss rate, and the difference of throughput between TFRC flows and TCP are larger.

Case 6. TCP vs TFRC, bottleneck capacity 60Mb

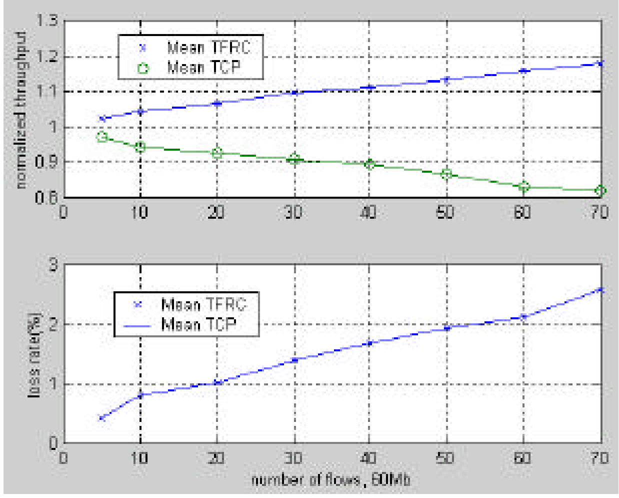

Figure 15. TCP vs TFRC, 15 Mb

From Figure 15, we can see:

When the number of flows is small, the congestion is not severe, the normalized throughput of TFRC and TCP flows are close, as the flow numbers increase, the congestion is more severe, as we can see from loss rate, and the difference of throughput between TFRC flows and TCP are larger.

Conclusion:

As we compare the Friendliness metrics and loss rate of TCP, TCP (a, b) and TFRC, we find that as the flow number increases, the difference between throughput becomes larger, and loss rate increases. The order for throughput from highest to lowest is TFRC>TCP>TCP (a, b). Throughput difference between TFRC and TCP is relatively smaller than throughput difference between TCP and TCP (a, b).

3.8 Transmission Delay effect

In this part, we discuss the effect of transmission on the throughput and smoothness of TCP, TCP (a, b) and TFRC. The comparison is presented as the following order:

- TCP (1/5,1/8) vs TCP (1,1/2)
- TCP (3/7,1/4) vs TCP(1,1/2)
- TCP (1/5,1/8) vs TFRC
- TFRC vs TCP (1,1/2)

Simulation Scenario:

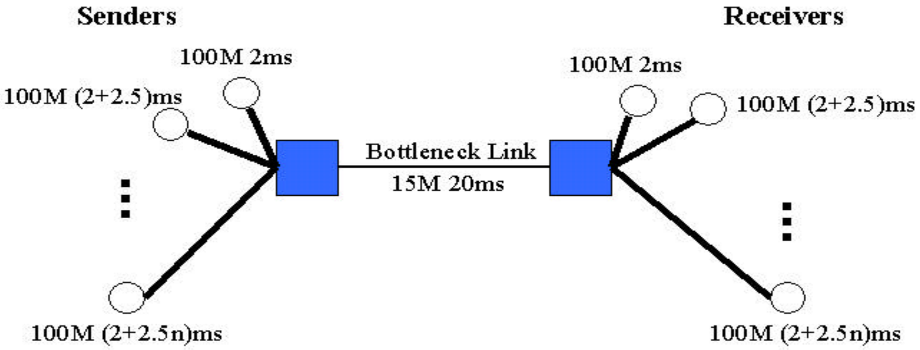

Figure 16. Simulation Scenario

In this simulation, the trunk link capacity is 15Mb, transmission delay on trunk link is 20ms, the link capacity from each sender to the entrance node of trunk link 100 Mb, the transmission delay between the sender to the entrance node of the trunk link is (2+2.5n)ms, that means the transmission delay for these flows increases 2.5ms between adjacent flow ID, and the link capacity from the outlet node of trunk link to receiver node is also 100 Mb, transmission delay between the outlet node of trunk link and receiver node is (2+2.5n)ms, that means the transmission delay for these flows increases 2.5ms between adjacent flow ID. Queuing mechanism we use Random Early Drop(RED). The trunk link is all occupied by two kinds of flows.

Case 1. TCP (1/5,1/8) vs TCP (1,1/2)

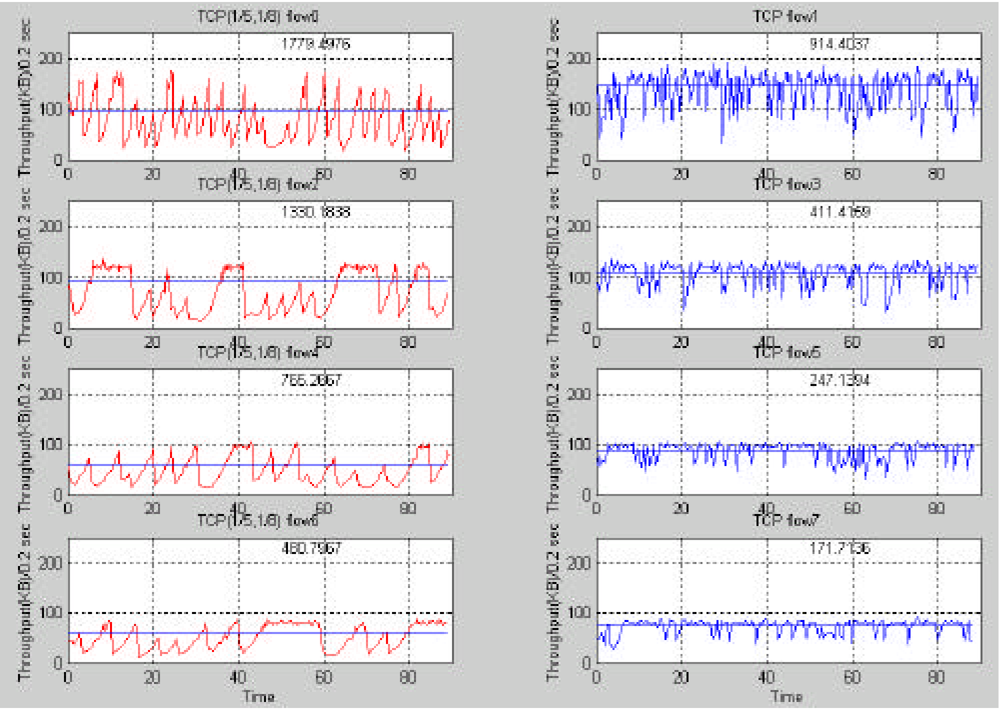

Figure 17. TCP (1/5,1/8) vs TCP

From Figure 17 we can see:

As the transmission delay increases, the bandwidth occupancy decrease, throughput gets smaller and smoother, as we can see from the variance of throughput. In the situation of large transmission delay, TCP's throughput is better than TCP (1/5,1/8).

Case 2. TCP (3/7,1/4) vs TCP

From Figure 18 we can see:

As the transmission delay increases, the bandwidth occupancy decrease, throughput gets smaller and smoother, as we can see from the variance of throughput. In the situation of large transmission delay, TCP’s throughput is better than TCP (3/7,1/4).

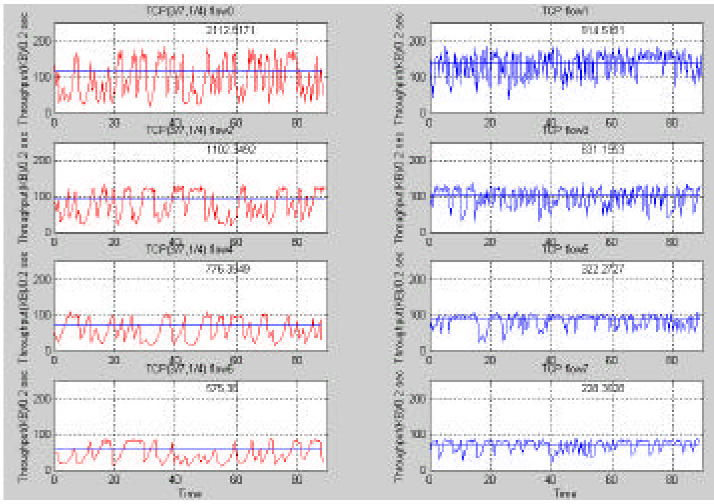

Figure 18. TCP (3/7,1/4) vs TCP

From Figure 18 we can see:

As the transmission delay increases, the bandwidth occupancy decrease, throughput gets smaller and smoother, as we can see from the variance of throughput. In the situation of large transmission delay, TCP’s throughput is better than TCP (3/7,1/4).

Case 3. TCP (/7,1/4) vs TFRC

From Figure 19 we can see:

As the transmission delay increases, the bandwidth occupancy decreases for both TCP 3/7,1/4) flows and TFRC flows, throughput gets smaller and smoother, as we can see from the variance of throughput. In the situation of large transmission delay, TFRC’s throughput is larger and smoother than TCP (3/7,1/4).

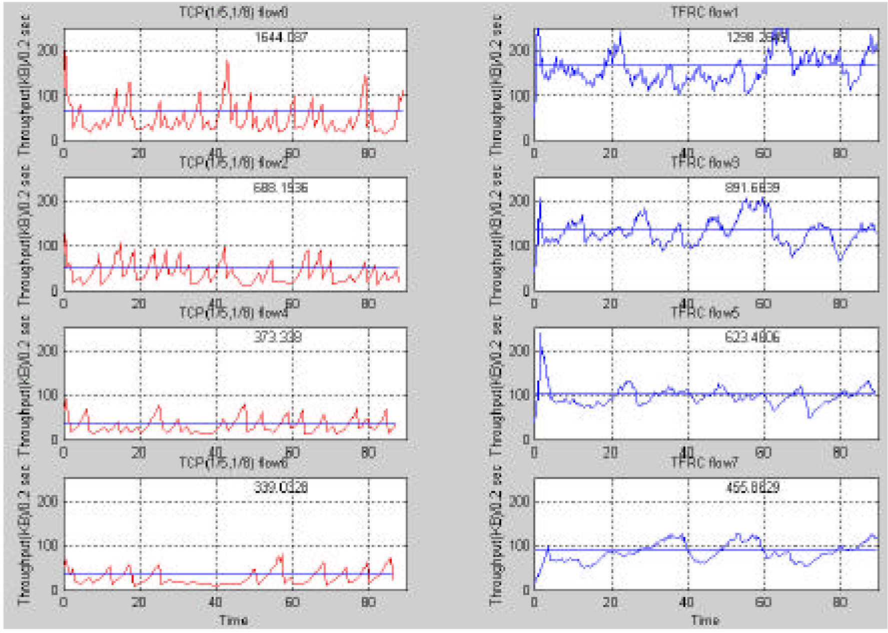

Figure 19. TCP (1/5,1/8) vs TFRC

Case 3. TCP vs TFRC

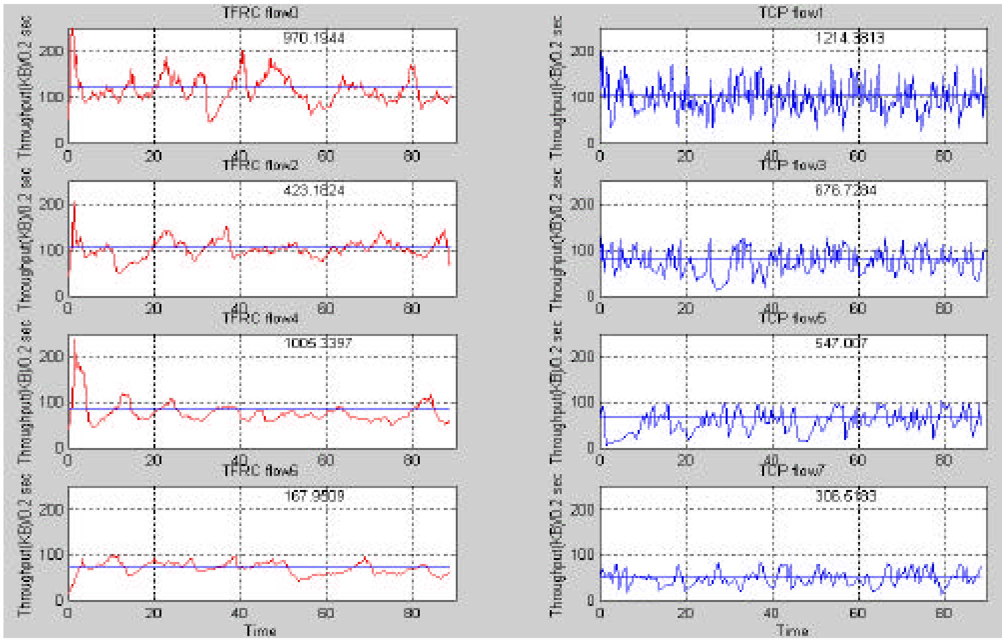

Figure 20. TCP vs TFRC

From Figure 20 we can see:

As the transmission delay increases, the bandwidth occupancy decreases for both TCP flows and TFRC flows, throughput gets smaller and smoother, as we can see from the variance of throughput. In the situation of large transmission delay, TFRC's throughput is larger and smoother than TCP.

Case 4. TFRC vs TCP (1/5,1/8)

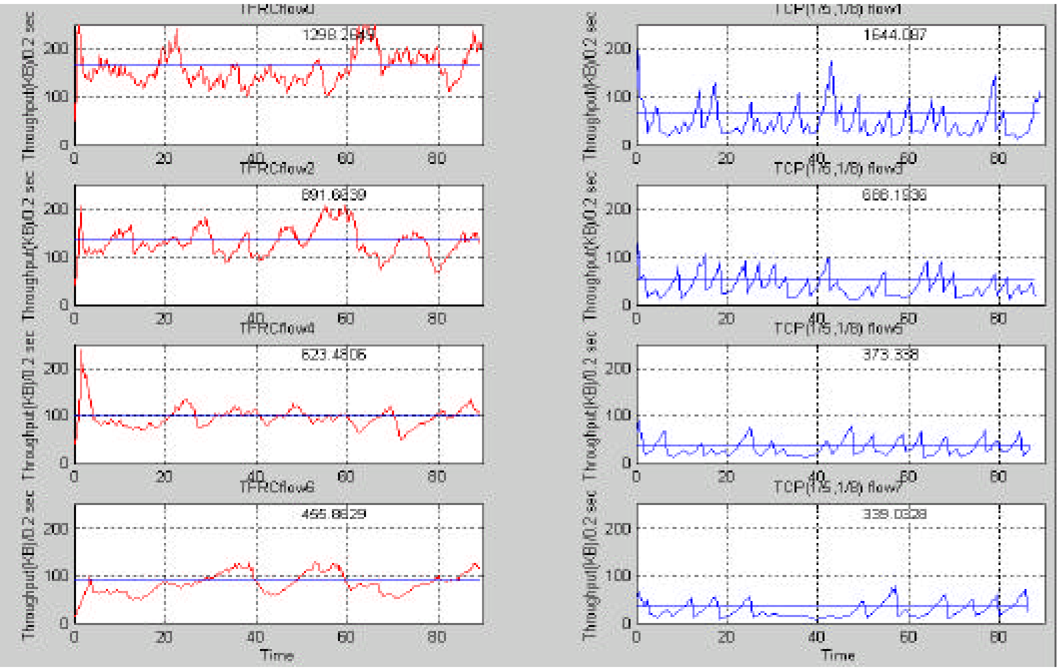

Figure 21. TFRC vs TCP (1/5,1/8)

From Figure 21 we can see:

As the transmission delay increases, the bandwidth occupancy decreases for both TCP (1/5,1/8) flows and TFRC flows. The throughput of TFRC is larger and smoother than TCP (1/5,1/8), as we can see from the variance of throughput.

Conclusion:

As the transmission delay increases, the occupancy of bandwidth decreases, and throughput becomes smoother. From simulations, we can see that the throughput of TFRC is larger and smoother than TCP with equal transmission delay, while throughput of TCP are larger and smoother than TCP (a, b). The throughput of TFRC is close to that of TCP, which means TFRC compete fairly with TCP.

3.9 Effects of Queuing algorithms

In this part, we compare the queuing effects of RED and Droptail, from theoretical analysis, RED feedback congestion information in advance of real congestion, which is helpful in alleviate congestion situation. We compare the effects of queuing mechanisms according to Table 6.

	RED vs DropTail
TCP vs TCP (a,b)	
TCP vs TFRC	

Table 6. comparison of effects of Queuing algorithms

Simulation Senario: In this simulation, the trunk link is 15Mb, transmission delay 20ms, the link capacity from sender to the entrance node of trunk link 100 Mb, transmission delay 2ms, and the link capacity from the outlet node of trunk link to receiver node is also 100 Mb, transmission delay 2ms. Queuing mechanism switches between Random Early Drop (RED) and DropTail to give a comparison of throughput and smoothness.

Case 1. TCP vs TCP (1/5,1/8), Droptail vs RED

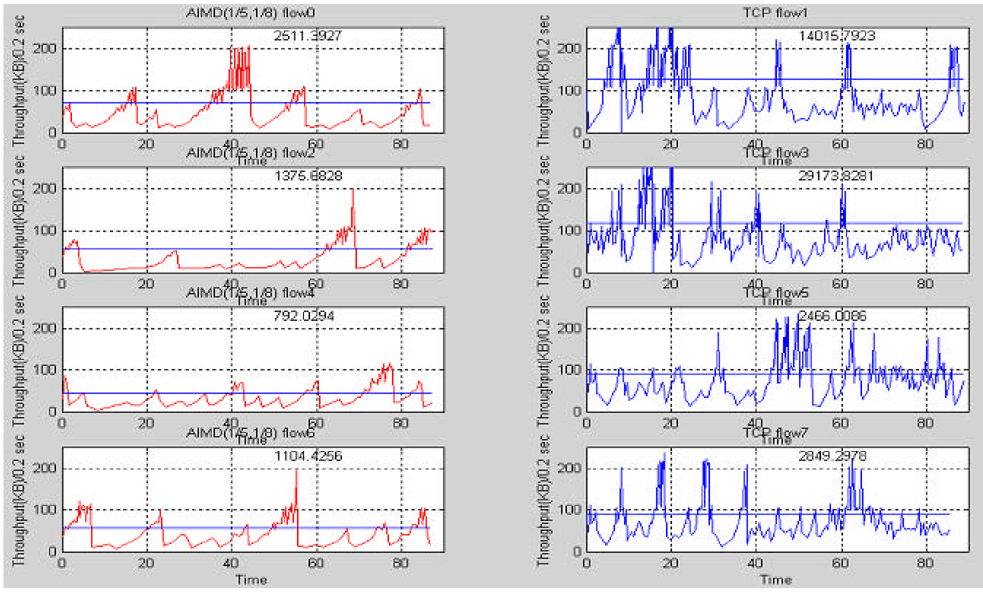

Figure 22. TCP(1/5,1/8) vs TCP, DropTail

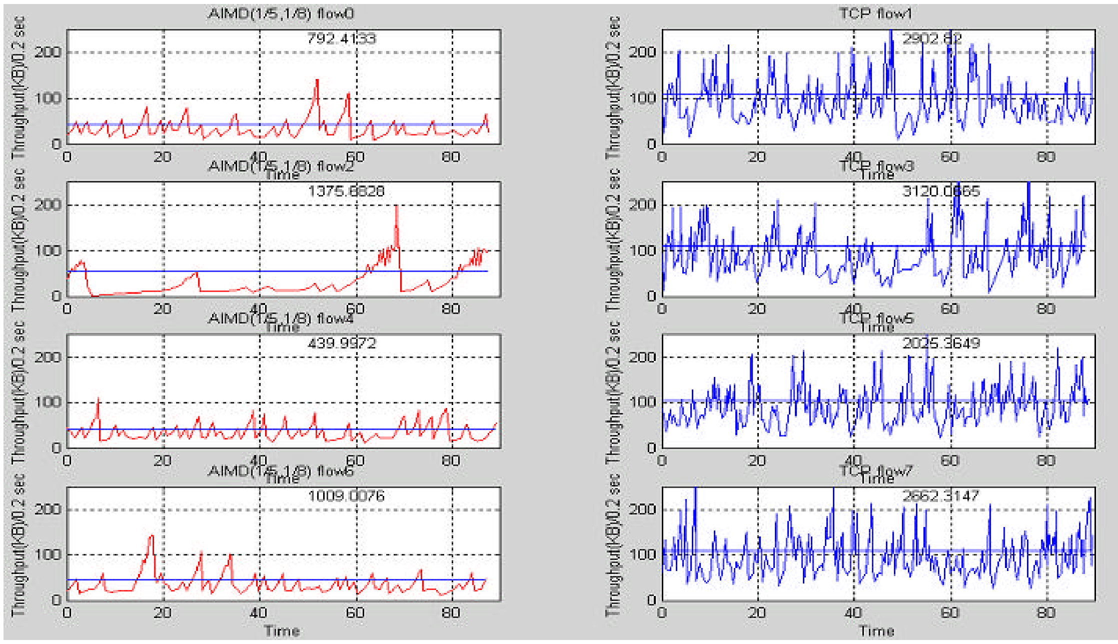

Figure 23. TCP(1/5,1/8) vs TCP, RED

From Figure 22 and Figure 23, we can see

In network using RED queuing, the throughput is smoother than in the network which uses DropTail queue.

Case 2. TCP vs TCP (1/5,1/8), Droptail vs RED

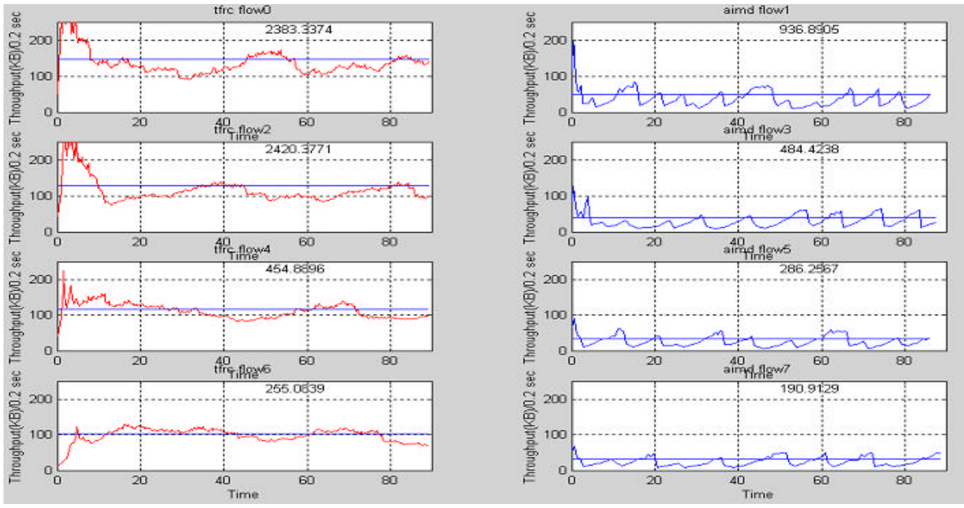

Figure 24. TCP (1/5,1/8) vs TFRC, DropTail

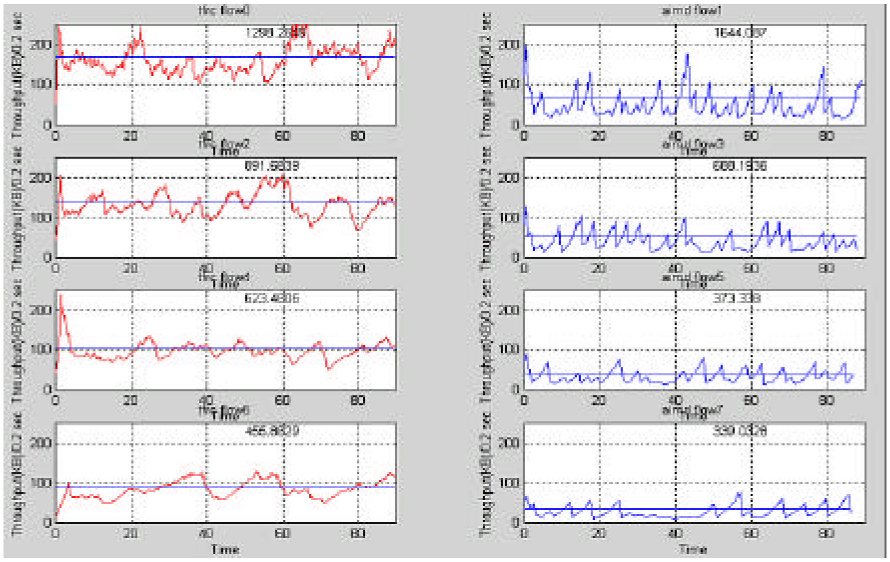

Figure 25. TCP (1/5,1/8) vs TFRC, RED

From Figure 24 and Figure 25, we can see

In network using RED queue, the throughput is smoother than in network which uses DropTail queue for both TCP (1/5,1/8) and TFRC flows.

3.10 Effect of ECN on loss rate

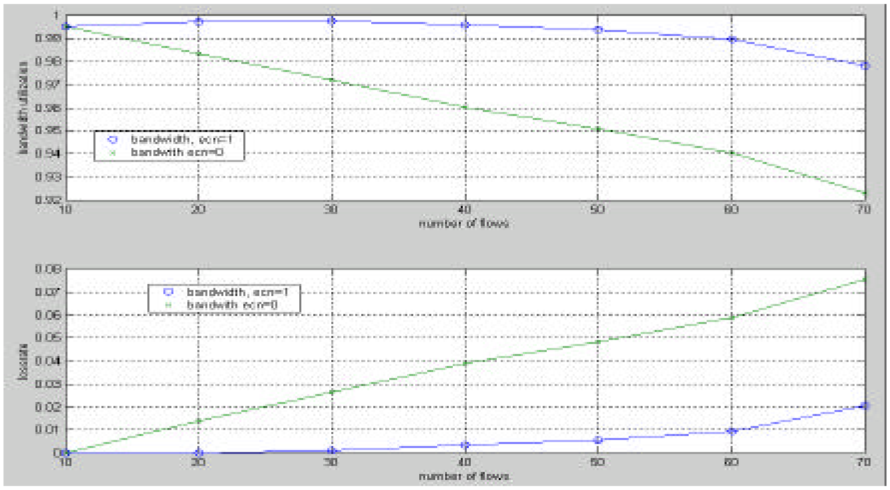

Figure 26. TCP (1/5,1/8) vs TFRC, RED

From Figure 26, we can see

When we set ECN to 1, the bandwidth utilization is higher than when ECN is set to 0, and the loss rate decrease when ECN set to 1. As the number of flows increase, this tendency is more obvious.

4. CONCLUSION

- Window size affect the bandwidth utilization and loss rate of flows
- The number of flows affect the bandwidth utilization and loss rate and congestion level
- TCP (3/7, 1/4) and TCP (1/5, 1/8) flows are smoother than the TCP (1,1/2) flows, but less smooth than the TFRC flows
- Throughput of TCP (1/5, 1/8) is smaller than TCP(3/7, 1/4) but smoother than the latter
- Comparing TCP (3/7, 1/4) and TCP(1/5, 1/8) with TCP(1,1/2), throughput of TCP(1,1/2) is higher
- TCP (a, b) and TFRC compete fairly with TCP (1,1/2), while avoiding TCP(1,1/2)'s reduction of the sending rate in half in response to a single packet drop, with TFRC achieving the best performance
- Different Queuing algorithms have different effects on throughput, RED better than Droptail.
- Transmission Delay affects the bandwidth utilization of flows, as transmission delay increases, the bandwidth occupancy decreases

REFERENCE

[1] Jamal Golestani, A Class of End-to-End Congestion Control Algorithms for the Internet , Proceedings of ICNP, 1998.
[2] S. Kunniyur and R. Srikant, "End-To-End Congestion Control: Utility Functions, Random Losses and ECN Marks", Longer version of the paper that appeared in Proceedings, INFOCOM 2000, Tel-Aviv, Israel, March 2000. Also submitted to IEEE Transactions on Networking
[3] S. Kunniyur and R. Srikant, "Fairness of Congestion Avoidance Schemes in Heterogeneous Networks", Proceedings, International Teletraffic Congress-16, Edinburgh, Scotland, 1999
[4] Yair Bartal, J. Byers and D. Raz, Global Optimization using Local Information with Applications to Flow Control, STOC, October 1997.
[5] TCP Friendly Rate Control (TFRC): Protocol Specification, Handley, M., Pahdye, J., Floyd, S., and Widmer, J. Internet draft draft-ietf-tsvwg-tfrc-02.txt, work in progress, May 2001.