# The Undismissible Rats: How Do Short Connections Affect Long-lived TCP Flows Under Moderate Traffic Load?

Rong Pan

Stanford University
Stanford CA 94305, USA
`rong@stanford.edu`

**Abstract.** It is well-known that the distribution of file sizes in the Internet has a long tail, and that the traffic mainly consists of small flows - "the mice", while a large portion of the bytes are sent by large flows - "the elephants". However, it is not yet well understood regarding how the mice and the elephants interact with each other when they share a common link. A simplified modelling assumption is that the mice appear as an uncontrolled and random background load while the elephants take the remaining link bandwidth [7].

In this paper, we find that under moderate traffic load, short connections, particularly large-sized mice whom we refer to as "rats", have a significant impact on the throughput of long-lived TCP flows. The occurrence of rats, who are in the slow-start phase of the TCP protocol, impair the ability of long-lived TCP flows to grab a link's available bandwidth. Our analysis shows that the throughput of the long-lived TCP flows is a function of the rats' arrival rate; and since the file sizes in the Internet are Pareto-distributed, this arrival rate is actually a function of the traffic's shape parameter.

## 1 Introduction

Studies have found that the file size distribution of Internet traffic, 90% of which uses the TCP protocol, has a heavy-tail [2], [4], [9]. In simple terms, it means that most TCP connections are "mice" (with short lifetimes, still in the slow start phase), but a few large TCP flows referred to as "elephants" (with long lifetimes, in congestion avoidance phase) generate most of the traffic. This behavior, known as "the elephants and mice phenomenon" is considered to be one of the few invariants of Internet traffic.

The Pareto distribution is shown to be an accurate model to describe the heavy-tailed distribution of file sizes [4]. The Pareto distribution has a cumulative distribution function as follows:

$$F(x) = 1 - (\frac{b}{x})^{\alpha} \quad for \ \ x \geq b, \tag{1}$$

where $\alpha$ is the shape parameter which determines the tail behavior and $b$ is the scale parameter. The expected value of a Pareto-distributed random variable equals to $E(x) = b(\alpha - 1)^{-1}$ when $\alpha > 1$.

Separately, the performance of long-lived TCPs has also been well-studied [5], [7], [10], [11]. Previous work has shown that the performance of long-lived

TCP flows depends on their round trip time (RTT) and packet loss probability. Assuming that the packet loss probability, $p$, is not too high and the receiver's window is not limited, the throughput $Th(p)$ of a long-lived TCP flow is given by

$$Th(p) \approx \frac{c}{RTT\sqrt{p}} \tag{2}$$

where $c$ is a constant. This equation plays a fundamental role in determining a long-lived TCP flow's throughput when the traffic's arrival rate $\lambda$ is relatively high. In this case, $p$ is independent of any particular flow's behavior. Given $p$, a flow with a longer RTT will get lesser throughput than a flow with a shorter RTT. However, we note that $p$ is not always independent of $Th$, especially when an elephant flow dominates a link's buffer space. For example, in the extreme case that only one long-lived flow is present on a link, $Th$ will be equal to the link capacity $C$ since the TCP protocol has the ability to fully utilize a link; and this is true regardless the value of RTT as long as there is enough buffering at the link. Here, Equation (2), while being true, only shows a secondary effect through which the drop probability can be obtained.

What kind of a role does Equation (2) play when the traffic mix consists of mice and elephants? Does the Pareto distribution play any role? It is not yet understood how the mice and the elephants interact with each other when they share a common link, i.e. how the bandwidth is distributed among the mice and the elephants. This task becomes more intricate because of the following fact: flows arrive and leave at random times; hence, the number of concurrent flows are varying over time. A modelling assumption, suggested by Kelly [7], is that the mice appear as an uncontrolled and random background load while the elephants take the remaining link bandwidth. This assumption is quite simplified; Kelly later indicates it might need more investigation [8].
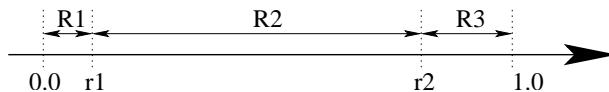


**Fig. 1.** Regions of Operations

This paper studies how the mice affect the performance of the elephants. We distinguish the following three regions that a long-lived TCP flow might operate in, shown in Figure 1. In Region $R1$, the link is lightly loaded. The boundary $r1$ could be around 10%.[1] Here, Kelly's assumption is valid. A long-lived TCP would dominate the behavior of the buffering queue and take most of link's bandwidth. Conversely in Region $R3$, the link's utilization is high. The boundary $r2$ could be around 70%.[2] In this region, since the aggregated arrival rate $\lambda$ is high, no single flow will dominate. Hence, the drop probability only depends on the mixed arrival rate of mice and elephants. Given $p$, a long-lived TCP flow's throughput can be found using Equation (2). However, in Region $R2$, which is the typical

---

[1] We only use these boundaries as a general guideline, the numbers chosen might not be accurate.

[2] Network operators start considering an upgrade to their equipment when a link's average utilization is above 70%.

region of operation for a network link, it is not yet understood how the mice and elephants share the link bandwidth. Our effort is focused on this normal, less-understood operating region.

The main contribution of this paper is to show that in Region $R2$ where a link is moderately loaded, the mice, especially those large-sized mice whom we refer to as "rats", play a rather negative role in sharing bandwidth with other TCP flows. These rats are big enough to just exit the slow-start phase of the TCP protocol, but not big enough to go through the saw-tooth like congestion avoidance. As a result, they undermine the ability of long-lived TCP flows to fill up a link's available bandwidth, one of the major strengths of the TCP protocol. Assuming that all flows share a common FIFO queue,[3] our analysis shows that the presence of rats causes extra delays and packet losses at the FIFO queue; and therefore force a long-lived TCP flow to back off and reduce its sending rate. As a result, the throughput of a long-lived TCP flow becomes a function of the rats' arrival rate $\lambda_{rats}$ as follows

$$Th \approx \frac{1.5}{RTT^2 \lambda_{rats}}. \tag{3}$$

Since we know that the file sizes are Pareto-distributed and recent studies have shown that Internet traffic can be modeled as a stationary stochastic process [1], we can rewrite Equation (3) as a function of the traffic's shape parameter $\alpha$:

$$Th \approx \frac{1.5}{RTT^2 \lambda} (\frac{b}{X})^{-\alpha}, \tag{4}$$

where $\lambda$ is the traffic arrival rate and $X$ is the minimum size of a rat. Equation (3) and (4), derived later in this paper, explain the role played by the mice in Region $R2$. Once again, they do not contradict the well-known Equation (2), which is valid in all regions. However, they do point to the fact that, in region $R2$, the drop probability becomes dependent on a flow's own behavior, i.e. $Th$, in this case.

We first study how mice and elephants interact with each other in Section 2, in which we introduce the notion of "rats". Section 3 uses a first-order mathematical model to analyze the role played by the rats. Then, we verify our findings using real trace statistics. We conclude in Section 4.

## 2 Interaction of mice and elephants

To gain insight into how short and long lived TCP flows share a link's bandwidth, we run a number of simulation experiments using the $ns2$ simulator [13].

### 2.1 Simulation Study

The simulation setup consists of one bottleneck link shared by multiple flows. The link uses a buffer that can buffer up to 250ms worth of packets, which means any flow with an RTT of less than 250ms has the ability to fully utilize

---

[3] There are proposals to buffer the mice and the elephants separately with some extra cost [3]. We do not study them in this paper.

a link. Note this implies that the size of the buffer is different for different link speeds. All packets are 1000 bytes long. All connections have a maximum receiver window of 1000 packets so that a receiver window is not a limiting factor for a flow's throughput. Also, a recent version of the TCP protocol, TCP/Sack, is used throughout this paper.

**The Basic Setup** In the basic setup, the bottleneck link bandwidth $C$ is 15Mbps and the link uses a droptail buffer with a size of 450 packets. Web sessions have a Pareto distribution with parameters $\alpha = 1.25$ and $b = 6.25$. Hence, the average file length, $E(x) = b(\alpha - 1)^{-1}$, is 25 packets.
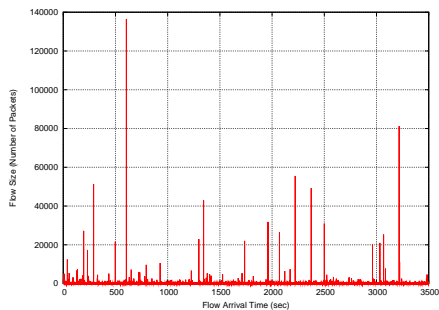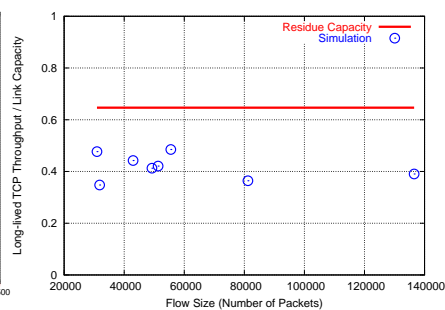


**Fig. 2.** Traffic

**Fig. 3.** Basic Setup: Long-lived TCP's Throughput

In the first experiment, the traffic of 100,000 web sessions consisting of mice and elephants, arrives randomly with an average demand of 5.3Mbps as shown in Figure 2. The mice have RTTs of either 30ms or 190ms (evenly distributed); and the elephants have an RTT of 190ms. The reason for the mice to have different RTTs is to reduce the phase effect under the droptail queue [6]. The rationale for the long-lived TCP flows having the same RTT is the following: since the link is moderately loaded, the degree of statistical multiplexing is not high. As a result, only a few elephants occur randomly in this simulation setup as shown in Figure 2.[4] It is easier and more accurate to calculate the average throughput of long-lived TCPs when they have the same RTT.

In this basic setup, the average traffic demand $\lambda$ is 5.3Mbps. Hence the link load, $\rho = \frac{\lambda}{C}$, is around 35%. According to statistical bandwidth sharing [1], the throughput of long-lived TCP flows, which use all the capacity not used by other flows, should be approximately equal to the residual capacity $C(1 - \rho)$ at the bottleneck link. However, Figure 3 shows that the throughputs *actually* achieved by the long-lived TCP flows, whose sizes are over 30,000 packets, are below the expected value. This figure illustrates that while the link has 65% of its bandwidth available, the long-lived TCP flows fail to fully utilize it, yielding an average throughput of 6.2Mbps, only 41% of the link capacity.

If we vary the arrival rate of the traffic, the behavior of long-lived TCP flows remain similar. We plot the average link utilization *when at least one long-lived*

---

[4] Although some of them might overlap a bit during their life-time, they are not on and off at the exact same time.

*TCP flow is present* in Figure 4.[5] One would expect the link to be 100% utilized since with enough buffering a long-lived TCP flow has the ability to use up a link's available bandwidth regardless of its RTT and the link's traffic load $\rho$. However, as shown in Figure 4, the long-lived TCP flows fail to fill up the available bandwidth when the link is under a moderate load except when the RTT is very short, such as 30ms in the plot. The worst link utilization (when long-lived flows present) goes down quickly with increasing RTTs (from 90ms to 230ms).

Clearly, the link is not always fully-utilized in Region $R2$ when long-lived TCP flows are present: in the worst case, the link's utilization is only around 60%. The plot also points out an interesting behavior: the link utilization goes down initially when $\rho$ starts to enter Region $R2$; stays low in Region $R2$; and increases again when the traffic demand is such that the link enters the congested region, Region $R3$. This figure demonstrates the fact that a long-lived TCP flow, under moderate traffic demands, could lose its strength to fully utilize a link.
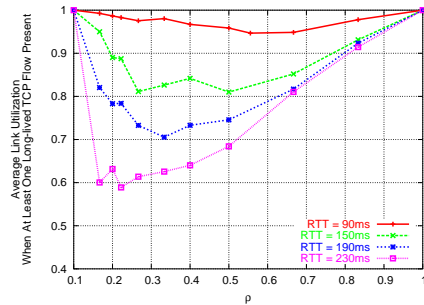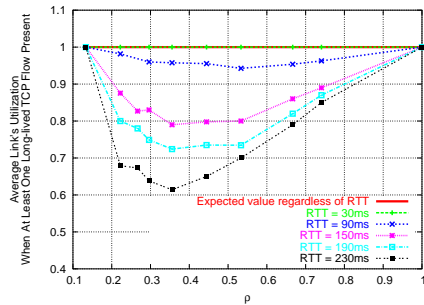


**Fig. 4.** Varying Traffic Demands: Average Link Utilization Under Long-lived TCP Flows

**Fig. 5.** With A Faster Link: Average Link Utilization Under Long-lived TCP Flows

**Link with A Higher Speed** Suppose we increase the link speed to 20Mbps and the buffer space to 600 packets (still 250ms worth of buffering), while keeping the same traffic arrivals as in Section 2.1. Figure 5 depicts that, once again, the long-lived TCP flows can fail to capture the available link bandwidth in Region $R2$. The link utilization shows a similar, U-shaped behavior.

**RED Queue instead of DropTail Queue** Suppose we alter the basic setup again by changing the queue management scheme to RED while keeping the other parameters the same. The $min_{th}$ and $max_{th}$ are set to be 150 and 300 packets. Figure 6 shows that the dynamics observed in the previous simulations does not change much when we use RED as the queue management scheme. This indicates that the phenomenon we observed might not be related much to the average queue length, rather it is related to the instantaneous queue length.

---

[5] Note the link utilization plotted here is the average link utilization when at least one long-lived TCP flow is present. It is not the overall, average link utilization over the entire simulation time. With $\rho = 0.35$, the average link utilization over the entire simualtion is approximately 0.35 as well.
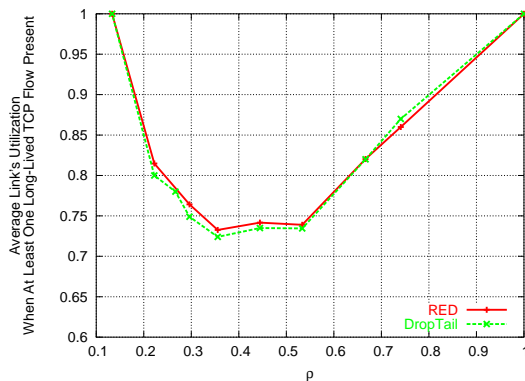
**Fig. 6.** With A RED-Queue: Average Link Utilization Under Long-lived TCP Flows

| $RTT_m$ | 30ms | 70ms | 110ms | 150ms | 190ms | 230ms |
|---|---|---|---|---|---|---|
| Throughput | 0.418 | 0.419 | 0.440 | 0.437 | 0.464 | 0.454 |

**Table 1.** Average Throughput of Long-lived TCP flows as a Function of Mice's RTTs

**Mice with Different Round Trip Times** If the mice have shorter or longer RTTs, would it affect the throughput of long-lived TCP flows? We now let the mice have two different round trip times with equal probability, $RTT_m$ and 190ms, where $RTT_m$ is a variable. We keep the RTT of 190ms for those flows whose sizes are above 30,000 packets. Table 1 shows that the average throughput of long-lived TCP flows depends little on the RTTs of mice. This behavior is rather different from that of a TCP flow operating in Region $R3$, where a flow can gain more bandwidth when competing against flows with shorter RTTs or lose bandwidth when competing against flows with longer RTTs as implied in Equation (2). This serves as another indication that the TCP protocol experiences a different throughput-limiting factor.

**Summary** We have also run other simulations such as traffic with a different shape parameter or two-way traffic which are omitted here. All the simulation studies demonstrate that, under a variety of scenarios in Region $R2$, a long-lived TCP flow can fail to capture a link's available bandwidth even though it has the ability to fully utilize a link when it is alone, i.e. when no mice flows are present. Here, neither Kelly's assumption nor Equation (2) can help explain the behavior of long-lived flows. The studies also show that the rates of mice flows, not the round trip times of mice flows, play a role in limiting a long-lived TCP flow's throughput.

### 2.2 A Key Observation

In the congestion-avoidance phase, a long-lived TCP flow regulates its window size according to network conditions which are directly related to the traffic load on a link. To examine the effect of the Internet traffic arrivals on the window size, we run a simulation where the flows whose sizes are over 30000 packets in

the basic setup are combined into one single FTP flow since this "long" FTP flow shows the evolution of the window size more visibly than those scattered long-lived flows in the original setup.
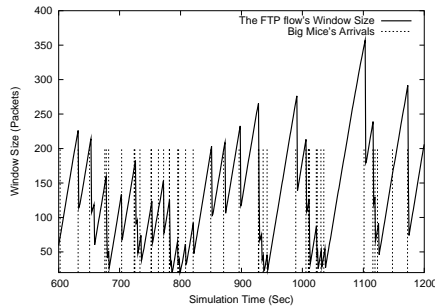


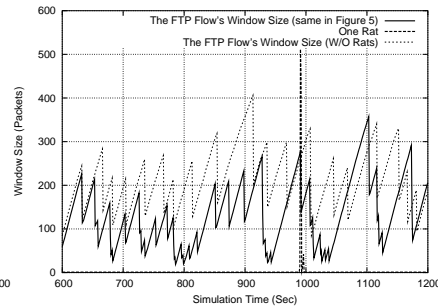**Fig. 7.** Window Size vs. Traffic Arrivals    **Fig. 8.** Window Size Comparison

The solid line in Figure 7 represents the window size of the FTP flow. The dotted lines indicate the arrivals of mice whose sizes are above 750 packets.[6] The plot illustrates that the time when the FTP flow cuts its window mostly coincides with the time when there is an arrival of a mouse whose size is above a few hundred packets. The arrivals of these big mice hinders the growth of the FTP flow's window. Hence, the throughput of the long-lived TCP flow becomes constrained. As a result, the FTP flow fails to capture the available link bandwidth.

### 2.3   The Notion of "Rats"

Since flows with a few hundred packets are significantly bigger than the common concept of "mice" - who send only a few packets, we introduce a new notion of "rats" to represent those flows whose sizes are big enough to disturb long-lived TCP flows, but small enough to avoid the saw-tooth type of window adjustment that a long-lived TCP flow would go through. The spike in Figure 8 shows an example of a rat's window size in the above simulation. If we remove rats and distribute their traffic load to flows under 600 packets (i.e. maintaining the same traffic arrival rate $\lambda$), the window size of the FTP flow becomes much larger, as shown in Figure 8. This serves as another evidence of the influence of rats. Since Internet traffic is Pareto-distributed, the rats are bound to occur. In other words, an elephant, whose life time is long, is certain to suffer from the disturbances caused by rats.

### 2.4   Discussion

One should be aware that the above observation does NOT indicate that long-lived TCP flows or small mice would not cause losses. As a matter of fact, they

---

[6] Note that the heights of the dotted lines in Figure 7 have no meaning.

would. However, these drops occur less frequently as compared to the drops caused by rats. Long-lived TCP flows occur scarcely under a moderate traffic demand as shown in Figure 2. Although these flows may overlap over a small period of time, the effect they have on each other is less than the effect brought by rats which arrive much more frequently. These rats, in the slow start phase, send many packets in one RTT time, and will overflow the buffer and cause damage to all other flows sharing the same link. Note that, here, the slow start threshold, $ssthresh$, is not a limiting factor since its intial value is set to be very high in most TCP implementations. For example, in Linux kernel 2.5.73, $ssthresh$ is initialized to be 0x7fffffff, which is practically infinity. Mice may cause drops as well, but because their sizes are small they can not overflow the buffer by themselves unless the buffer is already almost full when they arrive, which is rare since the link is under a moderate traffic demand.

## 3    Modeling and Analysis of Rats

This section develops a first-order mathematical model for analyzing the effect of rats on long-lived TCP flows. The model shows that the throughput of a long-lived TCP flow depends on the arrival rate of rats, $\lambda_{rats}$, and since the Internet traffic is Pareto-distributed, $\lambda_{rats}$ can be shown to be a function of the traffic's shape parameter $\alpha$.

### 3.1    Model

In our model of long-lived TCP's throughput, we assume a network setup as follows: an infinite long TCP connection is sharing a link with rats which arrive regularly. We also assume that the data buffer overflows and packets are dropped whenever a rat arrives. Note that we don't take into account the losses that are caused by mice or other competing long-lived TCP flows because these losses are rare compared to the losess caused by rats as discussed in Section 2.4.

Although multiple packets might be dropped within a window, TCP/Sack only cuts down its window by half instead of going into timeout. Hence one rat's arrival is equivalent to a single drop event. After a drop event, the TCP sender increases its congestion window by one packet per round trip time, until the next rat arrives.

**TCP throughput: a function of $\lambda_{rats}$** Suppose the inter-rat arrival time is $\lambda_{rats}^{-1}$, and the long-lived TCP connection has a rather constant round-trip time of $RTT$ seconds. Each time a rat arrives, the TCP sender has a congestion window of $W_{max}$ packets as shown in Figure 9.

By cutting its window by half for each rat's arrival and increasing its window by one per round-trip time subsequently, the TCP sender goes through $0.5W_{max}$ round-trip times for each drop event, i.e. each rat's arrival. Hence, the following equation holds:

$$\frac{W_{max}}{2} * RTT = \lambda_{rats}^{-1} \Rightarrow W_{max} = \frac{2}{RTT\lambda_{rats}}.$$
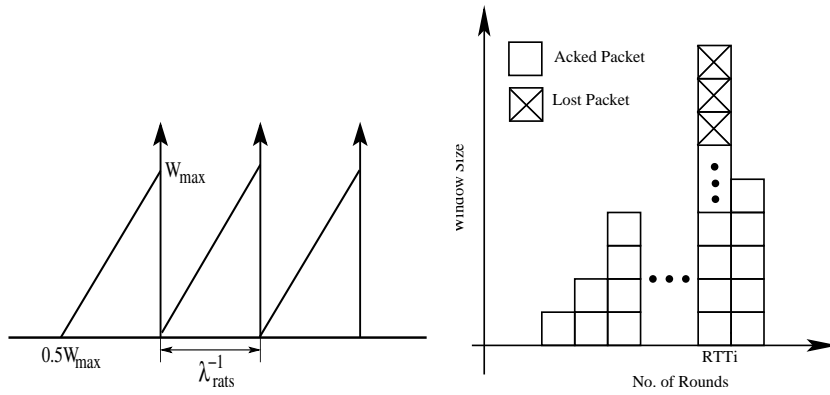
**Fig. 9.** Congestion Window Adjustment   **Fig. 10.** Evolution of Window Size

Since the average window size in a saw-tooth period equals $0.75W_{max}$, the throughput of a long-lived TCP, $Th_{tcp}$, can be expressed as follows:

$$Th_{tcp} = \frac{0.75W_{max}}{RTT} = \frac{1.5}{RTT^2\lambda_{rats}}. \tag{5}$$

**TCP throughput: a function of the shape parameter $\alpha$** As discussed in Section 1, previous work has shown that sessions arrive in the Internet as a Poisson process [1] and the file sizes in the Internet are Pareto-distributed [4]. The probability that a flow is a rat equals

$$P_{rat} = (\frac{b}{S_{rat}})^\alpha - (\frac{b}{S_{ele}})^\alpha \approx (\frac{b}{S_{rat}})^\alpha, \tag{6}$$

where $S_{rat}$ is the smallest size of a rat and $S_{ele}$ is the smallest size of an elephant. Since the Pareto-distributed file sizes imply that the probability is much higher for a flow being a rat than being an elephant, the above Equation can approximated by $(\frac{b}{S_{rat}})^\alpha$.

Given that the traffic arrives as a Poisson process of rate $\lambda$, we can obtain that the rats also arrive as a Poisson process with a rate of $\lambda_{rats} \approx \lambda(\frac{b}{S_{rat}})^\alpha$. Combining Equation (5) and $\lambda_{rats}$, we obtain that

$$Th_{tcp} \approx \frac{1.5}{RTT^2\lambda}(\frac{b}{S_{rat}})^{-\alpha}.$$

Certainly, a long-lived TCP flow's throughput is also constrained by the available link bandwidth, $C(1-\rho)$. Hence we can find the throughput of a long-lived TCP flow in Region $R2$ as

$$Th_{tcp} \approx \min( C(1-\rho), \frac{1.5}{RTT^2\lambda}(\frac{b}{S_{rat}})^{-\alpha} ). \tag{7}$$

Note that the above model does not take into account the throughput losses due to time out.

**The size of rats** Generally, rats are defined in this paper to be any connections which can cause the long-lived TCP flow to cut its congestion window size. Despite this loose definition of rats, we do follow a guideline in calculating the arrival rate of rats. We know that a rat, like any TCP flow, increases its congestion window geometrically in the slow start phase. For the $i$th round trip time, $RTT_i$, the congestion window size $W_i$ is $2^i$. The total number of packets sent up to $RTT_i$ equals

$$\sum_{m=0}^{i} 2^m = 2^{i+1} - 1 \approx 2^{i+1}. \tag{8}$$

This equation means that, if a flow has $N$ packets to send and it incurs no drop before the last round, then approximately $N/2$ are sent in the last round alone. Given that $N/2$ is close to the buffer size $B$ at an output link, the buffer is bound to overflow and some packets are lost. The lost packets are then resent. The evolution of the window size is illustrated in Figure 10. Although it normally takes a burst size of less than $B$ packets to overflow a buffer since other flows take up buffer spaces as well, $N \approx 2B$ serves as an upper-bound estimate of a rat's minimum size. The theoretical analysis of a rat's minimum size is left for future work.

### 3.2  Verification

We apply the analytical model to the simulations in Section 2. Table 2 shows the comparison results. Recall that $C$ is the bottleneck link capacity and $\lambda$ is sessions' arrival rate. The traffic load, $\rho$, equals the traffic arrival rate divided by the link capacity, and $U$ represents the link utilization when at least one long-lived TCP flow present. The buffer size, $B$, is set to be the bandwidth-delay product of the output link: 450 packets for a 15Mbps link and 600 packets for a 20Mbps link. The table clearly demonstrates that, under various traffic and network scenarios, our theoretical model is able to predict the performance of long-lived TCP flows accurately.

The model can not only predict a long-lived TCP flow's performance but also explain other phenomenon that we observed in Section 2: using a RED queue or varying mice's $RTT$ times does not change the TCP flows' performance. Since a rat is in the slow-start phase of exponentially expanding its window size, the sudden increase of its packet burst size would not change much the average queue length of the buffer at the bottleneck link. Therefore, packets are dropped because the buffer overflows, not because the average queue length is over a certain threshold so that RED needs to preemptively drop packets. As a result, the RED queue behaves like the droptail queue. Separately, we know from Equation (7) that the arrivals of mice only contribute to the traffic load, $\rho$, and therefore their RTT values have little effect on the elephants' performance.

Another interesting fact that the model can help explain is how a flow's throughput depends on its $RTT$. When a link is congested, the throughput of a long-lived TCP flow is inversely proportional to its $RTT$ as in Equation (2). When a link is moderately loaded, however, the dependency on RTT is quite different. When the $RTT$ value is small, the long-lived TCP flow's throughput is limited by the available link bandwidth, $C(1 - \rho)$, as the model predicts, which is consistent with what we have seen in Figure 4 when RTT = 30ms. When the $RTT$ value is big, the long tcp flow's throughput is inversely propotional to $RTT^2$. The drop in a flow's throughput is more drastic as shown in Figure 4.

| $C$ | $\lambda$ | $\rho$ | $B$ | $S_{rats}$ | $U(Simulation)$ | $U(Model)$ | $RTT$ |
|---|---|---|---|---|---|---|---|
| 15M | 26.7 | 0.356 | 450 | 750 | 1.0 | 1.0 | 30ms |
| 15M | 26.7 | 0.356 | 450 | 750 | 1.0 | 0.96 | 90ms |
| 15M | 26.7 | 0.356 | 450 | 750 | 0.86 | 0.83 | 150ms |
| 15M | 26.7 | 0.356 | 450 | 900 | 0.74 | 0.72 | 190ms |
| 15M | 26.7 | 0.356 | 450 | 900 | 0.61 | 0.61 | 230ms |
| 15M | 33.3 | 0.444 | 450 | 900 | 0.76 | 0.74 | 190ms |
| 15M | 33.3 | 0.444 | 450 | 900 | 0.64 | 0.65 | 230ms |
| 15M | 50 | 0.667 | 450 | 900 | 0.84 | 0.82 | 190ms |
| 15M | 50 | 0.667 | 450 | 900 | 0.76 | 0.79 | 230ms |
| 20M | 26.7 | 0.267 | 600 | 1200 | 0.69 | 0.73 | 190ms |
| 20M | 33.3 | 0.333 | 600 | 1200 | 0.67 | 0.70 | 190ms |
| 20M | 50 | 0.5 | 600 | 1200 | 0.70 | 0.74 | 190ms |

**Table 2.** Theory vs. Simulation: $\alpha = 1.25$

### 3.3 Real Traffic Statistics

In this section, we demonstrate that the phenomenon indeed still exists when we use real traffic statistics. We obtained traces of traffic from backbone routers at two different locations: New York and Philadelphia.[7] It has been studied that these traces have a heavy tail and the arrival process of sessions is Poisson. We applied the sampled trace statistics to the access link of 15Mbps. The average arrival rate to the access link is 4.48Mbps, which is equivalent to a traffic load of 0.3.
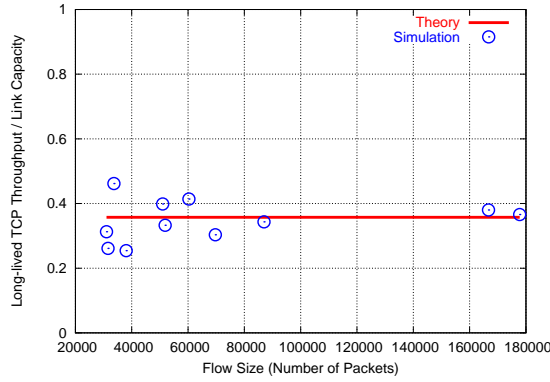


**Fig. 11.** Trace 1: Throughputs of Long-lived TCP flows

Figure 11 shows the simulation result using the traffic statistics of one trace. The circles in the figure represent the throughputs of long-lived TCP flows whose sizes are bigger than 30,000 packets. It is clear that these long-lived TCP flows fail to capture the available link bandwidth of 70%. The straight line in the plot shows our model prediction. Since there lacks enough data to fully characterize the shape parameter $\alpha$ for these traces, we measure the arrival rate of rats $\lambda_{rats}$

---

[7] Due to the proprietary nature of these traces, we omit the details about them.

directly from the sampled trace and apply Equation (5). The plot depicts that the simulation result using real traffic statistics, with $\lambda_{rats} = 0.062$, matches well with the analytical result. Most importantly, the plot shows that the phenomenon about the rats affecting elephants indeed exists for the real Internet traffic: an elephant would encounter multiple rats in its life-time and hence suffer from their presence.

## 4   Conclusion

In this paper, we have shown that short flows, especially "rats", have an important effect on the throughput of long-lived TCP flows. When a link is under moderate traffic demands, the occurrence of rats impairs the ability of the TCP protocol to fully utilize a link's available bandwidth, which leads to the link being under-utilized. We have also presented a theoretical model to explain the effect of rats. Our analysis shows that the throughput of a long-lived TCP flow is a function of the rats' arrival rate $\lambda_{rats}$. Since the file lengths of Internet traffic are Pareto-distributed and session arrivals are Poisson, $\lambda_{rats}$ is in fact a function of the traffic shape parameter.

## References

1. Ben Fredj, S., Bonald, T., Proutiere, A., Regnie, G., and Roberts, J., "Statistical Bandwidth Sharing: A Study of Congestion at Flow Level", *ACM SIGCOMM*, 2001.
2. CAIDA, "Traffic Workload Overview" and "Graphs of Ames Internet Exchange Traffic", *http://www.caida.org*, 2003.
3. Chen, X. and Heidemann, J. "Preferential Treatment for Short Flows to Reduce Web Latency", *Computer Networks*, 41(6), April, 2003.
4. Crovella, M. and Bestavros, A. "Self-Similarity in Worlf Wide Web Traffic: Evidence and Possible Causes", *IEEE/ACM Trans. on Networking*, vol.5, no.6, Dec. 1997, pp.835-846.
5. Floyd, S. and Fall, K., "Promoting the Use of End-to-End Congestion Control in the Internet", *IEEE/ACM Trans. on Networking*, August, 1999.
6. Floyd, S. and Jacobson, V., "On Traffic Phase Effects in Packet-Switched Gateways", *Internetworking: Research and Experience*, V.3 N.3, September 1992, p.115-156.
7. Kelly, F., "Mathematical modelling of the Internet", *Proceedings of the Fourth International Congress on Industrial and Applied Mathematics*, July, 1999.
8. Kelly, F., "Models for a self-managed Internet", *Philosophical Transactions of the Royal Society*, A358(2000), pp.2335-2348.
9. Leland, W., Taqqu, M., Willinger, W. and Wilson, D., "On the self-similar nature of Ethernet traffic", *ACM SIGCOMM*, 1993.
10. Misra, V., Gong, W. and Towsley, D., "A Fluid-based Analysis of a Network of AQM Routers Supporting TCP Flows with an Application to RED", *ACM SIGCOMM*, 2000.
11. Padhye J., Firoiu, V., Towsley, D. and Kurose, J., "Modeling TCP Throughput: A Simple Model and its Empirical Validation", *ACM SIGCOMM*, 1998.
12. Paxson, V. and Floyd, S., "Wide-area Traffic: The Failure of Poisson Modelling", *IEEE/ACM Trans. on Networking*, 3(3), June 1995, pp226-244.
13. Network Simulator, Version 2.1b8.