Modeling the Performance of Short TCP Connections

Neal Cardwell

Stefan Savage, Tom Anderson

Department of Computer Science and Engineering
University of Washington

October 26, 1998

Overview

- Motivation: Why model short TCP flows?
- Review of TCP mechanics
- Alternative performance models
- Simulation results
- Measurement results
- Conclusions

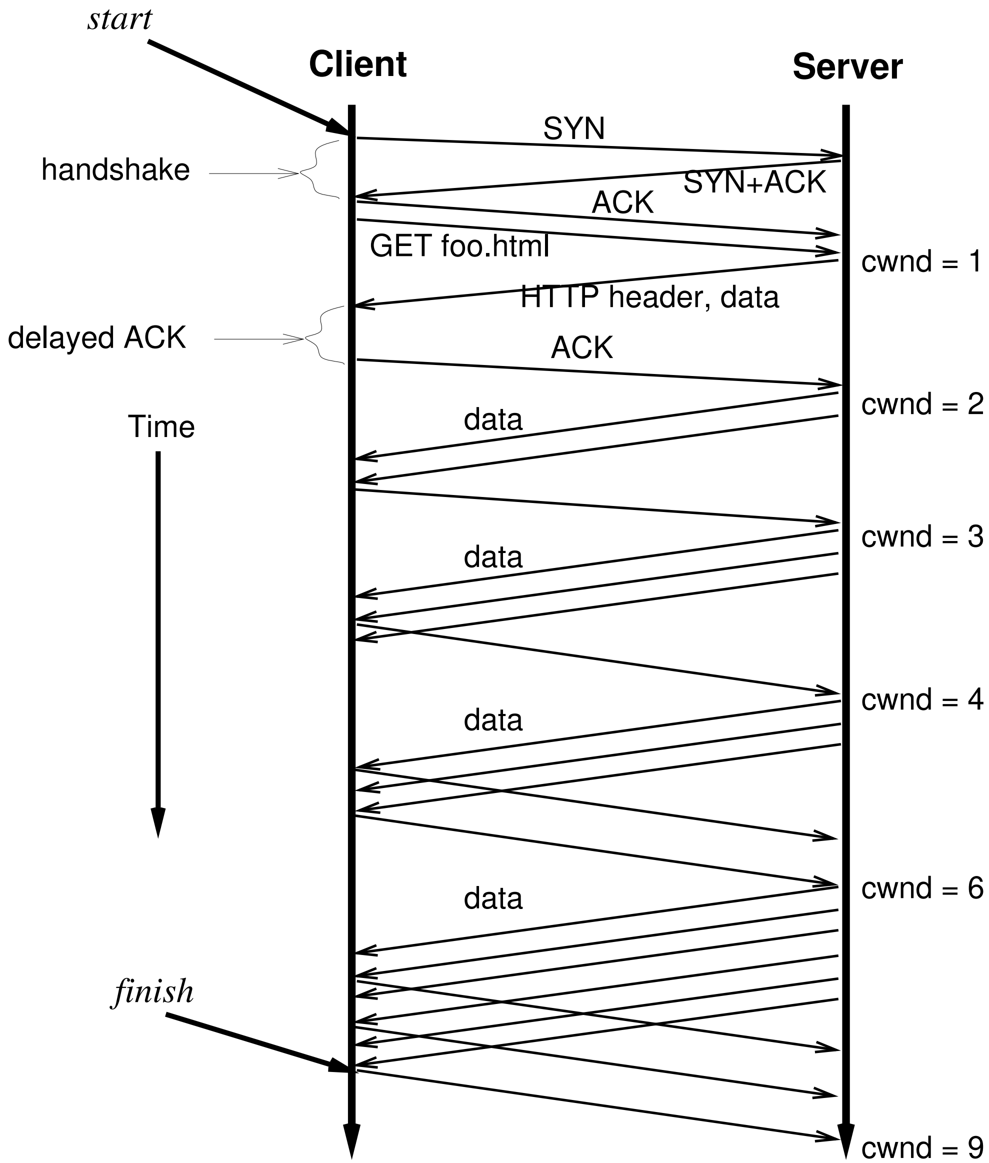
Why model short TCP flows?

- Why TCP?
  - Most popular apps use TCP: WWW, e-mail, news, telnet, FTP
  - Most Internet traffic is TCP
- Why *short* TCP flows?
  - Most TCP is HTTP, with 1 object, 10KB per flow
- Why *model* performance?
  - Understand effect of RTT, bandwidth, loss rate...
  - Apply to design of TCP, networks, routing, apps...

TCP Review

- Reliable, ordered, bi-directional byte stream
- Three-way handshake for connection establishment
- Sliding window flow control & congestion control
- Congestion avoidance
  - Slow start: exponential growth from $cwnd = 1$
  - Congestion avoidance: linear growth from $ssthresh$
- Congestion control
  - Retransmission timeout (RTO): slow start from $cwnd = 1$
  - Triple duplicate ACKs: congestion avoidance from $cwnd = cwnd/2$

A Typical TCP Connection



Model I: Optimal Performance

generalizing [Mahdavi97, HOT97]:
w = initial $cwnd$, in segments
r = rate of exponential increase of $cwnd$ during slow start
$cwnd_i$ = $cwnd$ (in segments) in RTT number i
$data_i$ = segments sent by end of RTT number i

$$
\begin{aligned}
data_i &= \sum_{k=1}^{i} cwnd_k \\
&= w + w \cdot r + w \cdot r^2 + w \cdot r^3 + \cdots + w \cdot r^{i-1} \\
&= w \cdot \frac{r^i - 1}{r - 1} \\
&\vdots \\
i &= \log_r \left(\frac{data_i (r-1)}{w} + 1 \right) \\
&\vdots \\
t &= \log_r \left(\frac{data(r-1)}{w \cdot MSS} + 1 \right) \cdot RTT + RTT + t_{delack}
\end{aligned}
$$

Model II: Adapting Steady-State Model

Steady-state TCP bandwidth model from [Padhye+98]:

$$B \approx \min\left(\frac{W_{max}}{RTT}, \frac{MSS}{RTT\sqrt{\frac{2bp}{3}} + T_0 \min\left(1, 3\sqrt{\frac{3bp}{8}}\right) p(1+32p^2)}\right)$$

Applied to short flows:

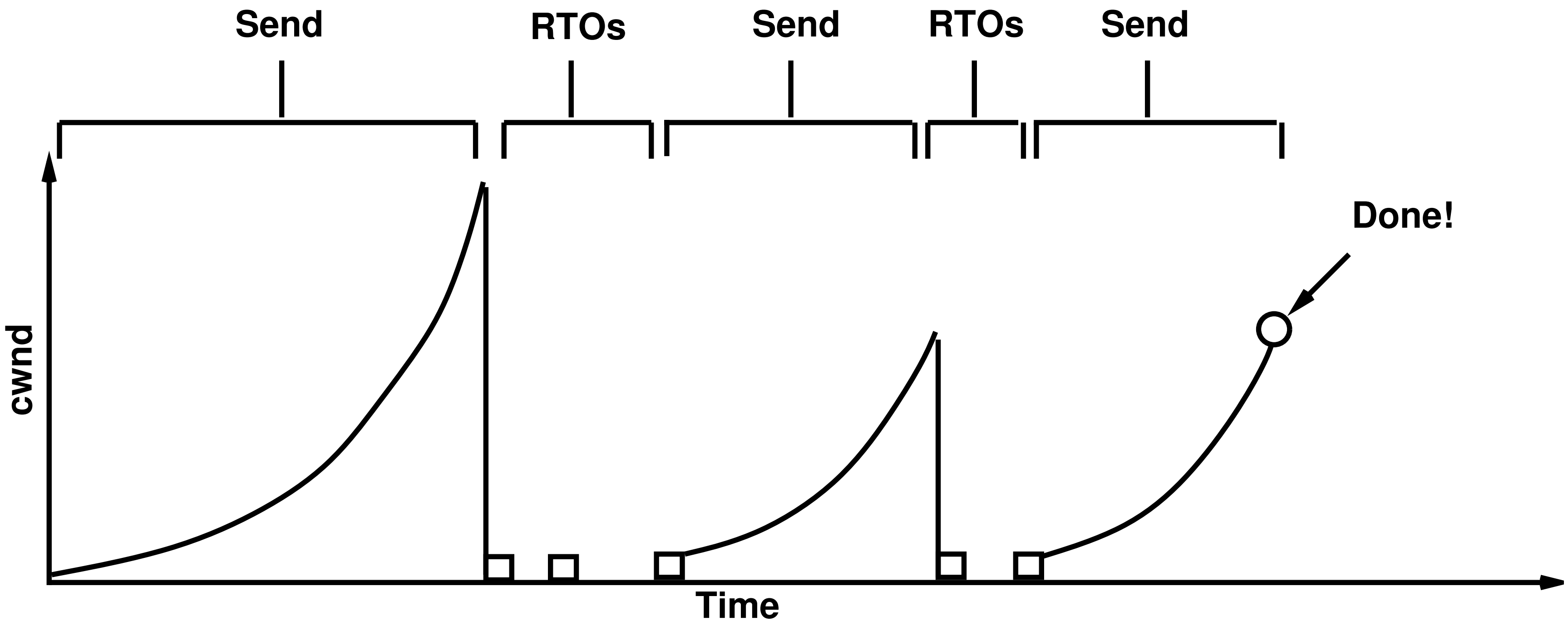
$$t = data/B(p) + RTT + t_{delack}$$

A New Approach

Observations:

- Optimal model for short flows ignores losses
- Steady-State model ignores finite-ness of data
- Combine approaches:
  - Start with finite amount of data
  - Calculate expected number of runs of RTOs
  - Break data into consecutive chunks
  - Calculate cost of RTO runs, sending data

Visualizing the New Model



Simplifications: No triple-duplicate ACKs, always in slow start

Model III: Short Flows w/ Loss

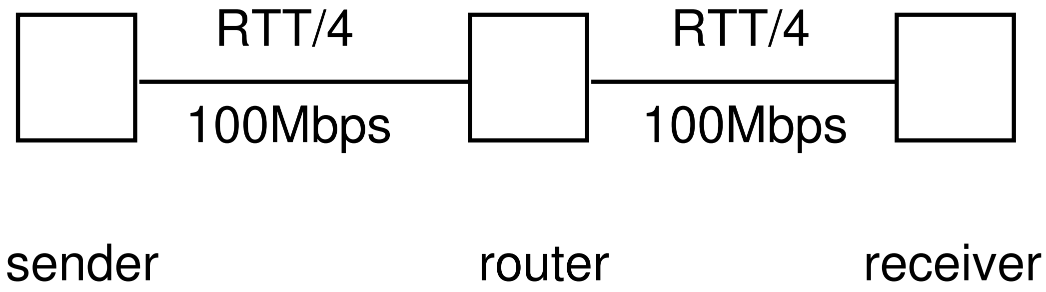
- Number of losses: $p = \frac{l}{data+l} \Rightarrow l = \frac{data \cdot p}{1-p}$
- Number of RTOs [Padhye+]: $n = Q(p) \cdot l$
- Average num RTOs in a row: $s = \frac{1}{1-p}$
- Number of runs of RTOs: $u = n/s$
- Cost of RTOs [Padhye+]: $t_{RTO} = u \cdot T_0 \cdot \frac{f(p)}{1-p}$
- Number of sending phases: $v = u + 1$
- Cost of each sending phase:

$$t_{xfer} = v \cdot \log_r \left(\frac{data(r-1)}{v \cdot w} + 1 \right) \cdot RTT$$

- Total time:

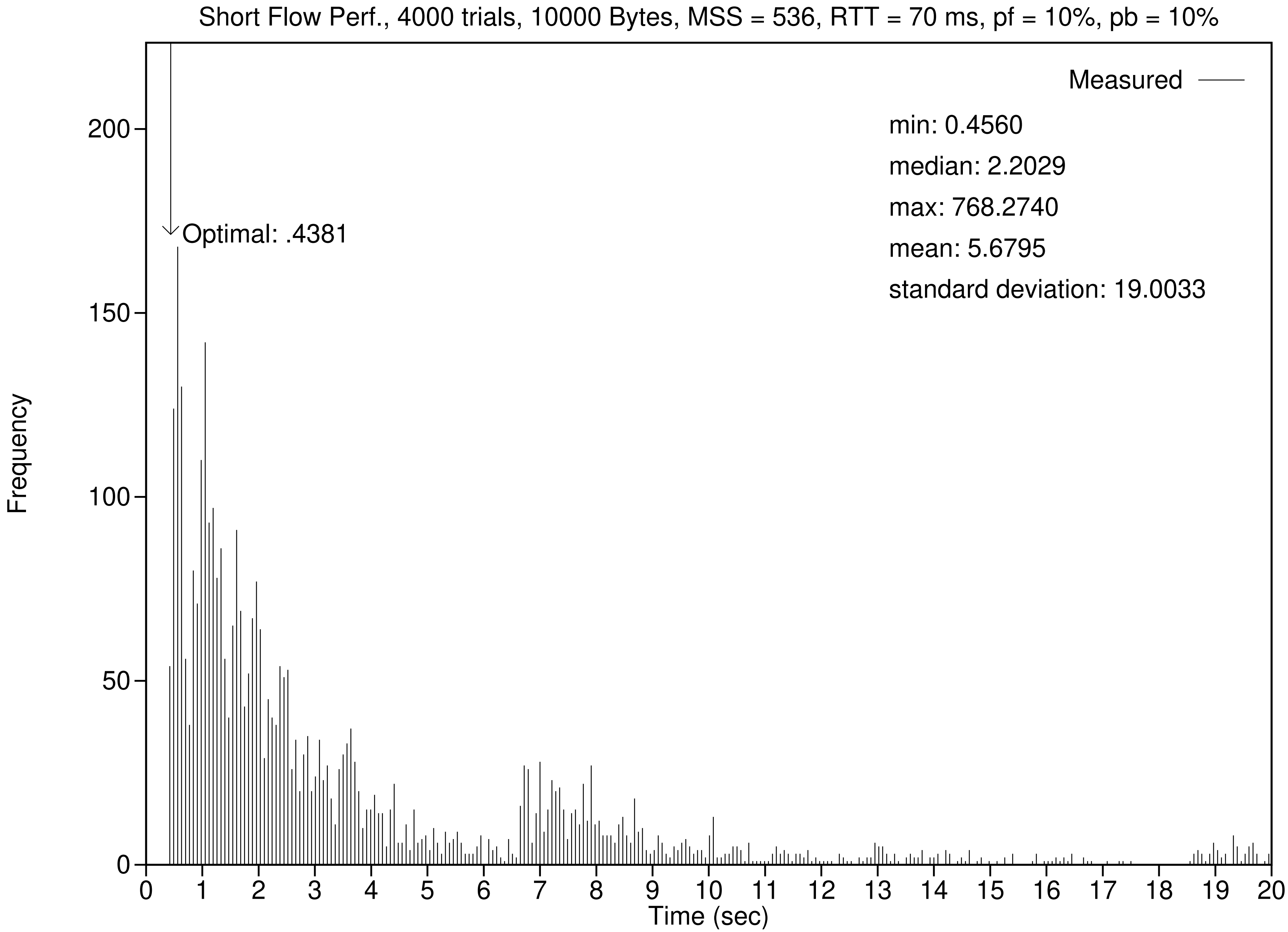
$$t = RTT + t_{delack} + t_{RTO} + t_{xfer}$$

Simulations

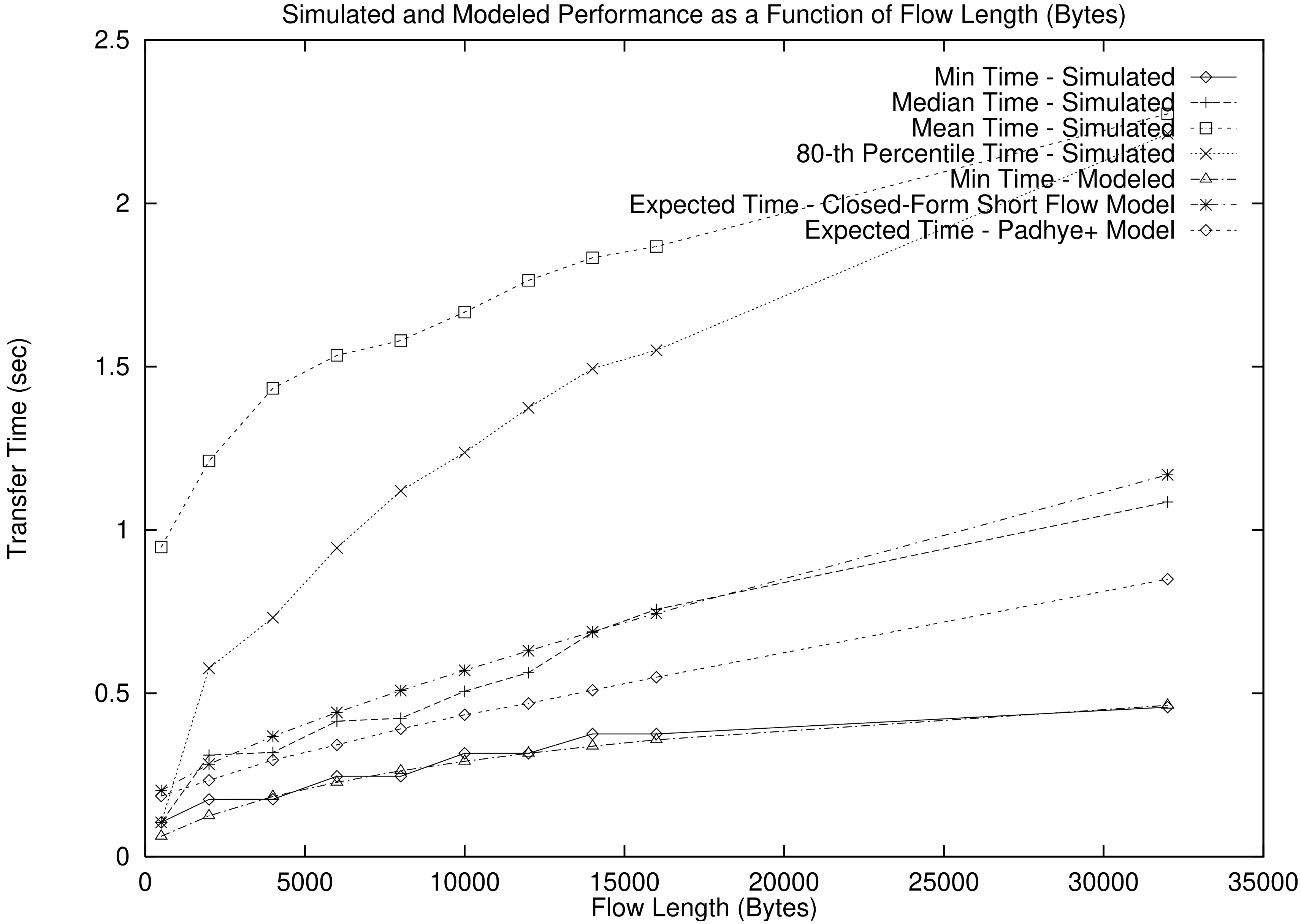


- `ns` network simulator, `FullTCP` model (like BSD Reno)
- Router configured for i.i.d. losses in both directions
- Default: 4000 trials, $MSS = 1460$B, 10KB data, $RTT = 70$ ms, $p_f = p_b = 5\%$

High Variance and Heavy Tails

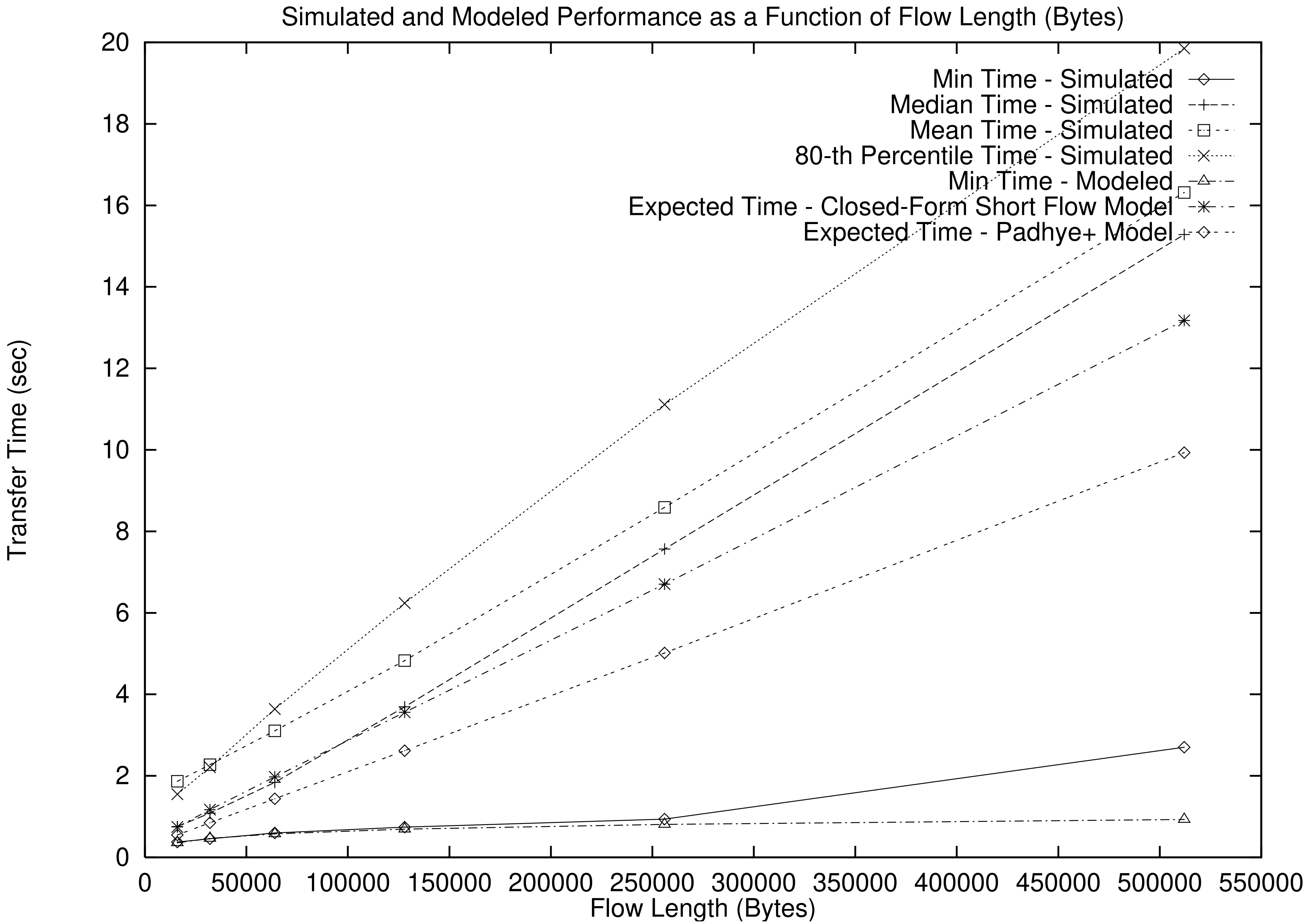


Short Flows



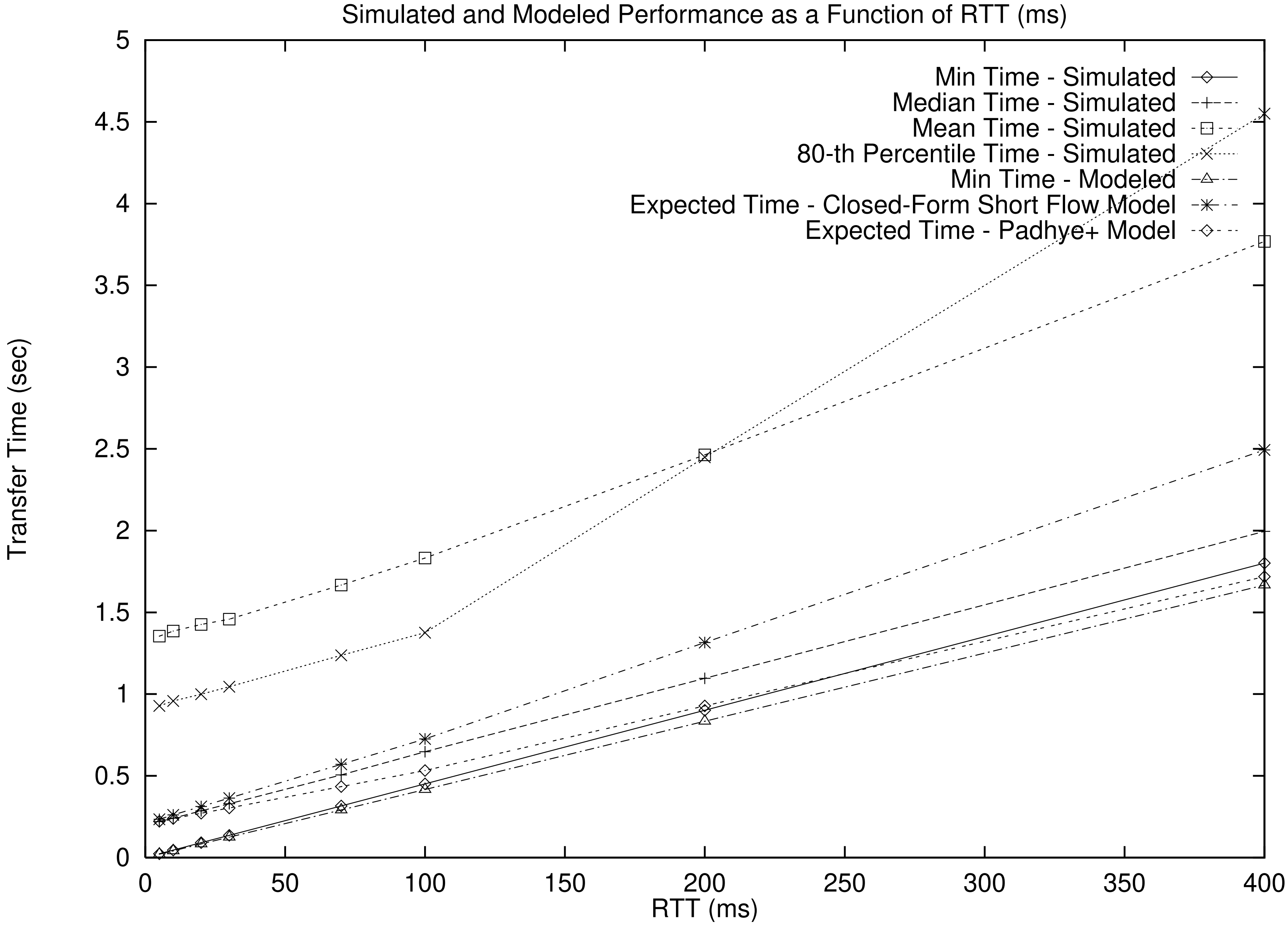
Simulated and Modeled Performance as a Function of Flow Length (Bytes)
Transfer Time (sec)
Flow Length (Bytes)
Min Time - Simulated
Median Time - Simulated
Mean Time - Simulated
80-th Percentile Time - Simulated
Min Time - Modeled
Expected Time - Closed-Form Short Flow Model
Expected Time - Padhye+ Model
0
0.5
1
1.5
2
2.5
0
5000
10000
15000
20000
25000
30000
35000

Long Flows



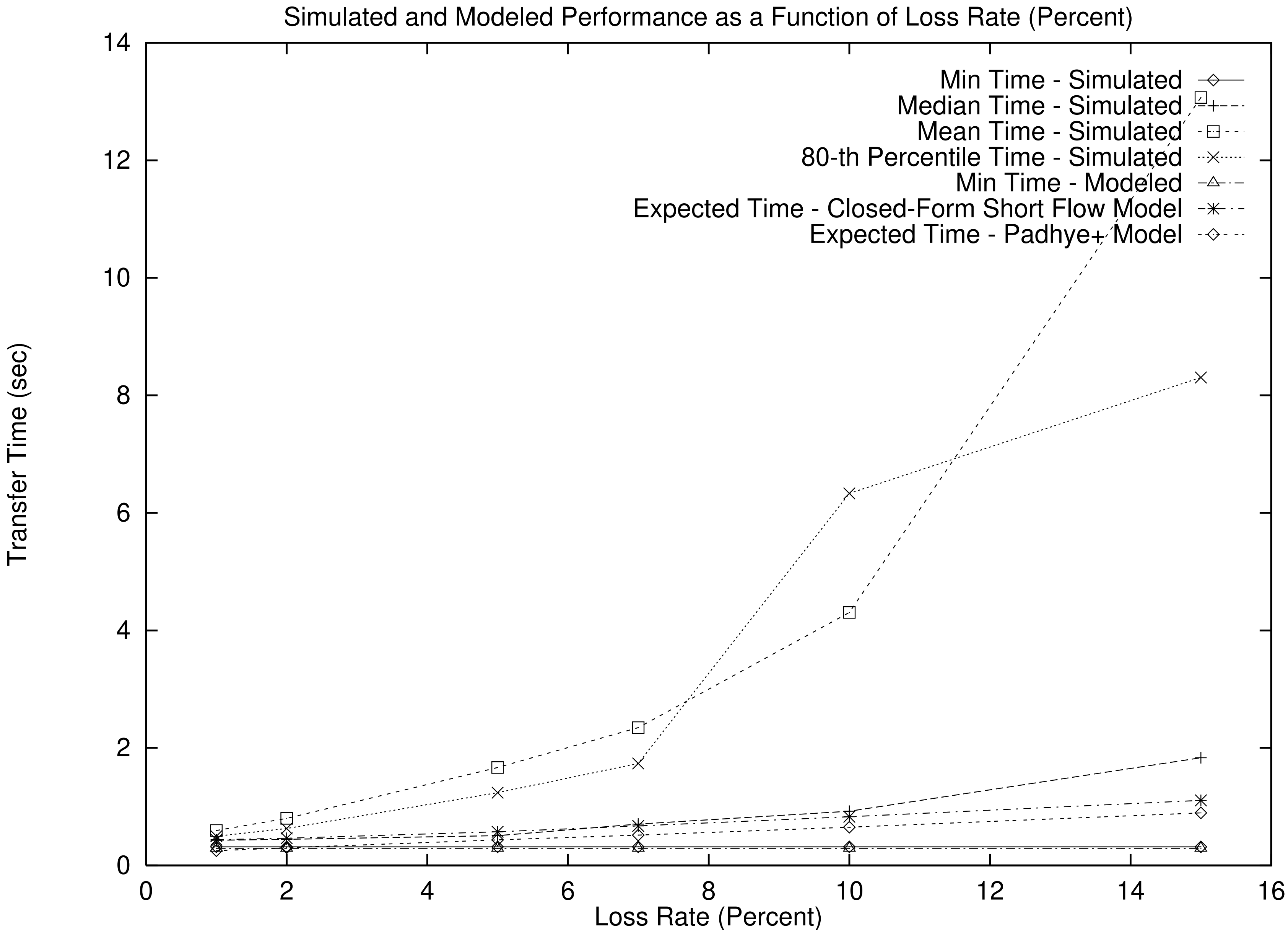
Simulated and Modeled Performance as a Function of Flow Length (Bytes)
Transfer Time (sec)
Flow Length (Bytes)
Min Time - Simulated
Median Time - Simulated
Mean Time - Simulated
80-th Percentile Time - Simulated
Min Time - Modeled
Expected Time - Closed-Form Short Flow Model
Expected Time - Padhye+ Model
0
2
4
6
8
10
12
14
16
18
20
0
50000
100000
150000
200000
250000
300000
350000
400000
450000
500000
550000

RTT

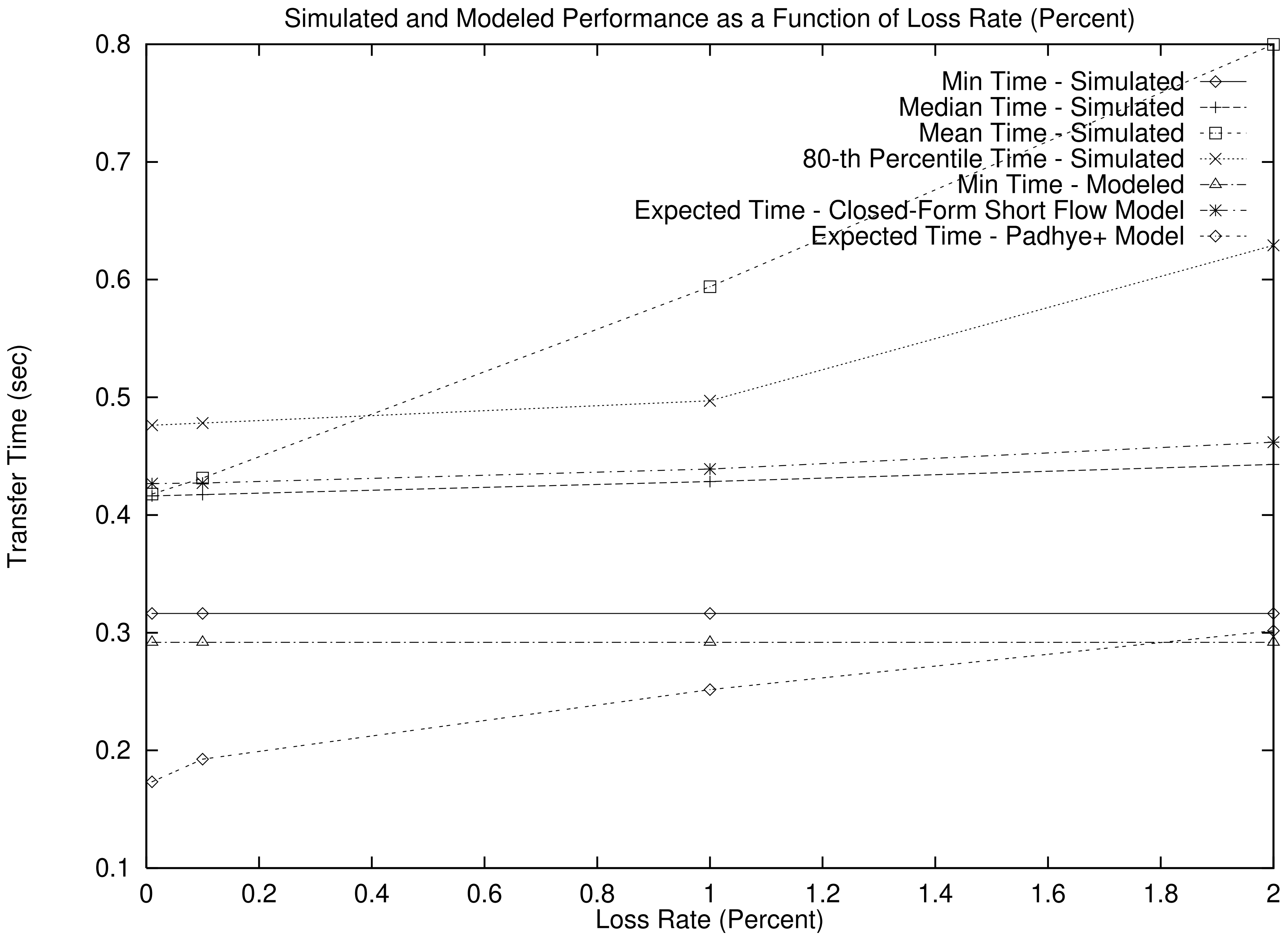


Simulated and Modeled Performance as a Function of RTT (ms)
Min Time - Simulated
Median Time - Simulated
Mean Time - Simulated
80-th Percentile Time - Simulated
Min Time - Modeled
Expected Time - Closed-Form Short Flow Model
Expected Time - Padhye+ Model
Transfer Time (sec)
RTT (ms)
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5
0
50
100
150
200
250
300
350
400

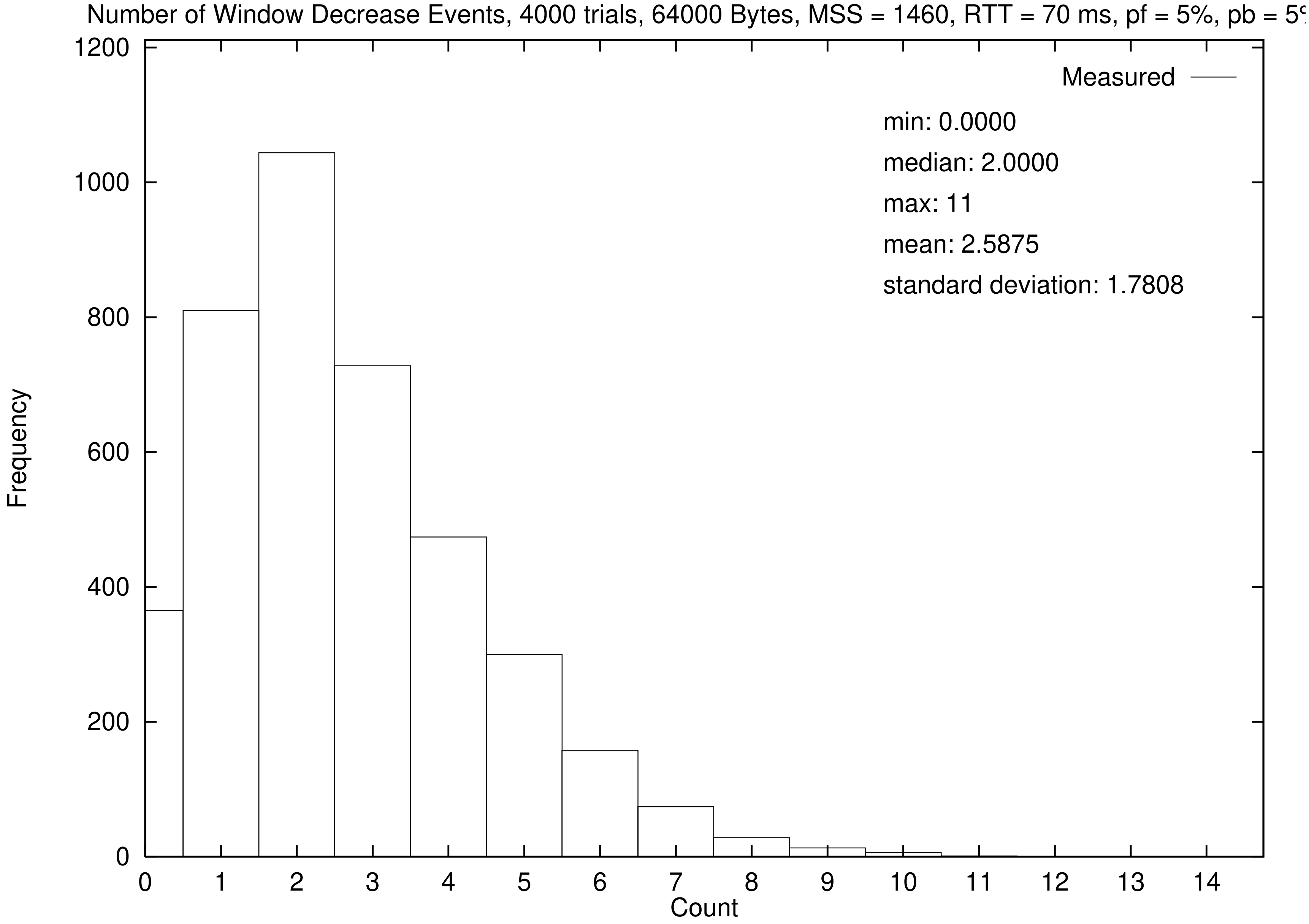
High Loss Rates



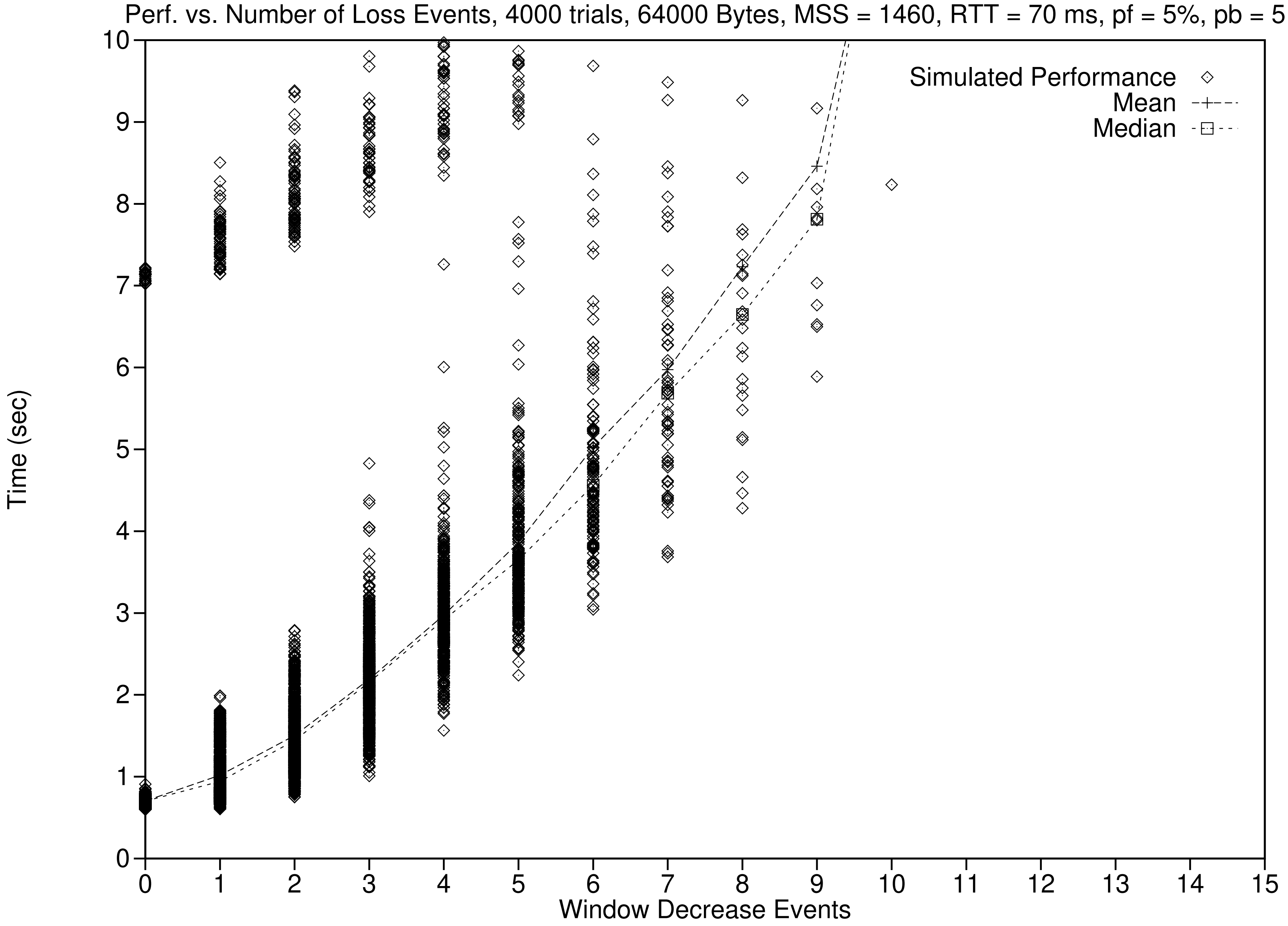
Low Loss Rates



Distribution of Number of Losses



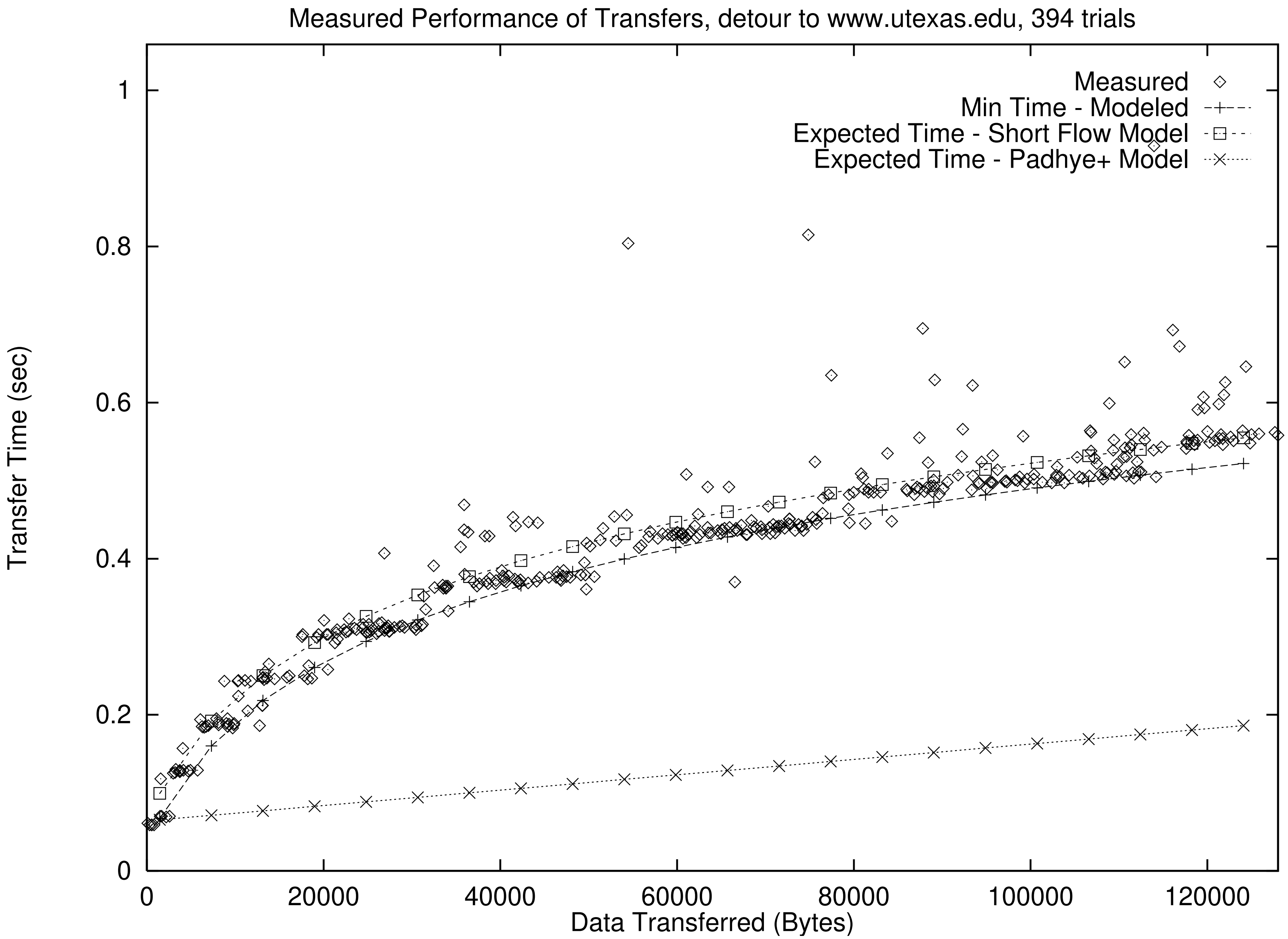
Performance & Number of Losses



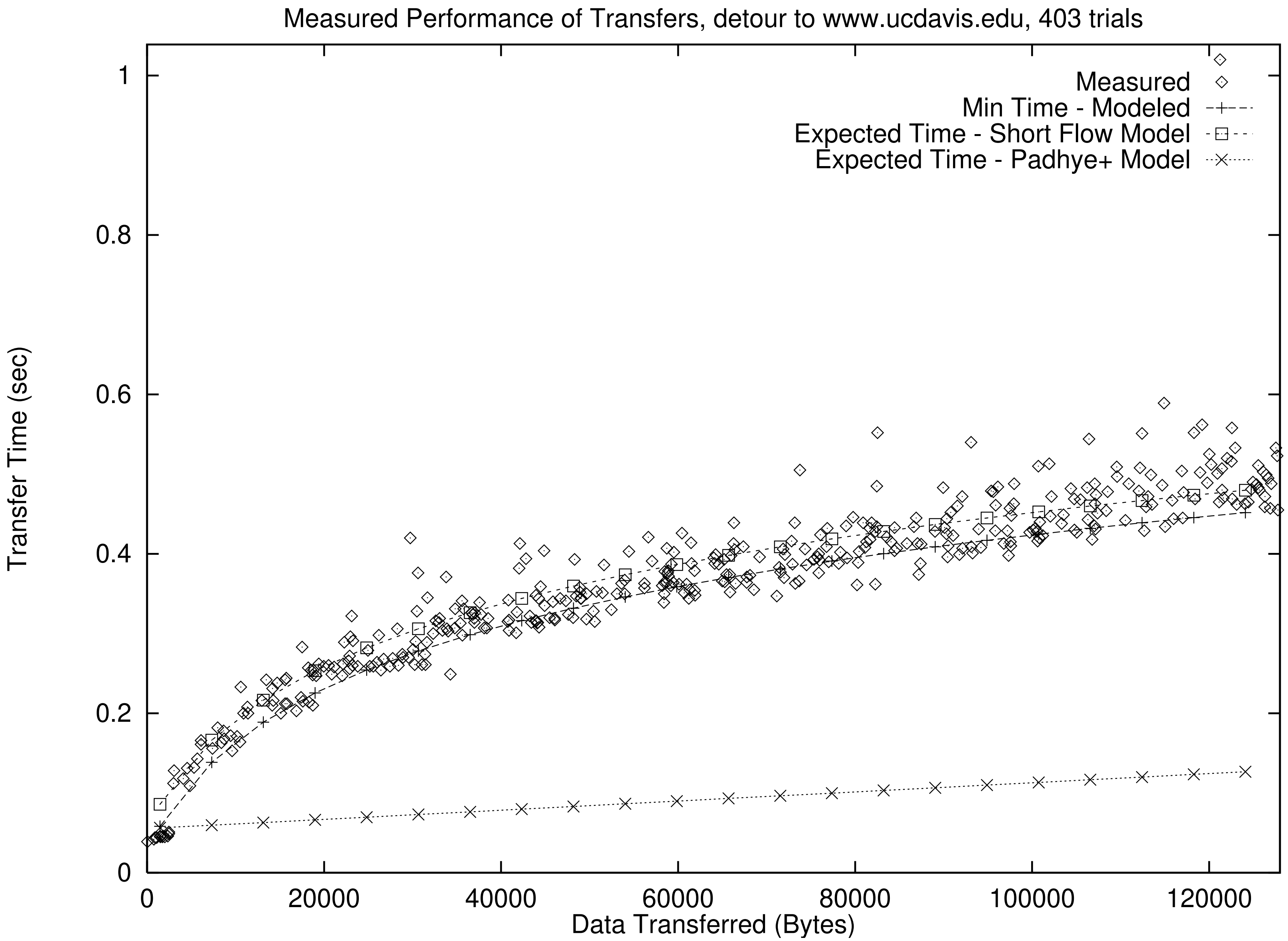
Measurements

- Short TCP transfers (uniform random 1..128,000 bytes)
- From Linux 2.0.35 PC at UW to four US sites running `discard`
- Used tcpdump to find transfer time, MSS, RTT, RTO, and loss rate.
- Exponential inter-measurement times, 60 sec mean
- 9:30AM to 5:00PM PDT on a Friday; 300-400 trials

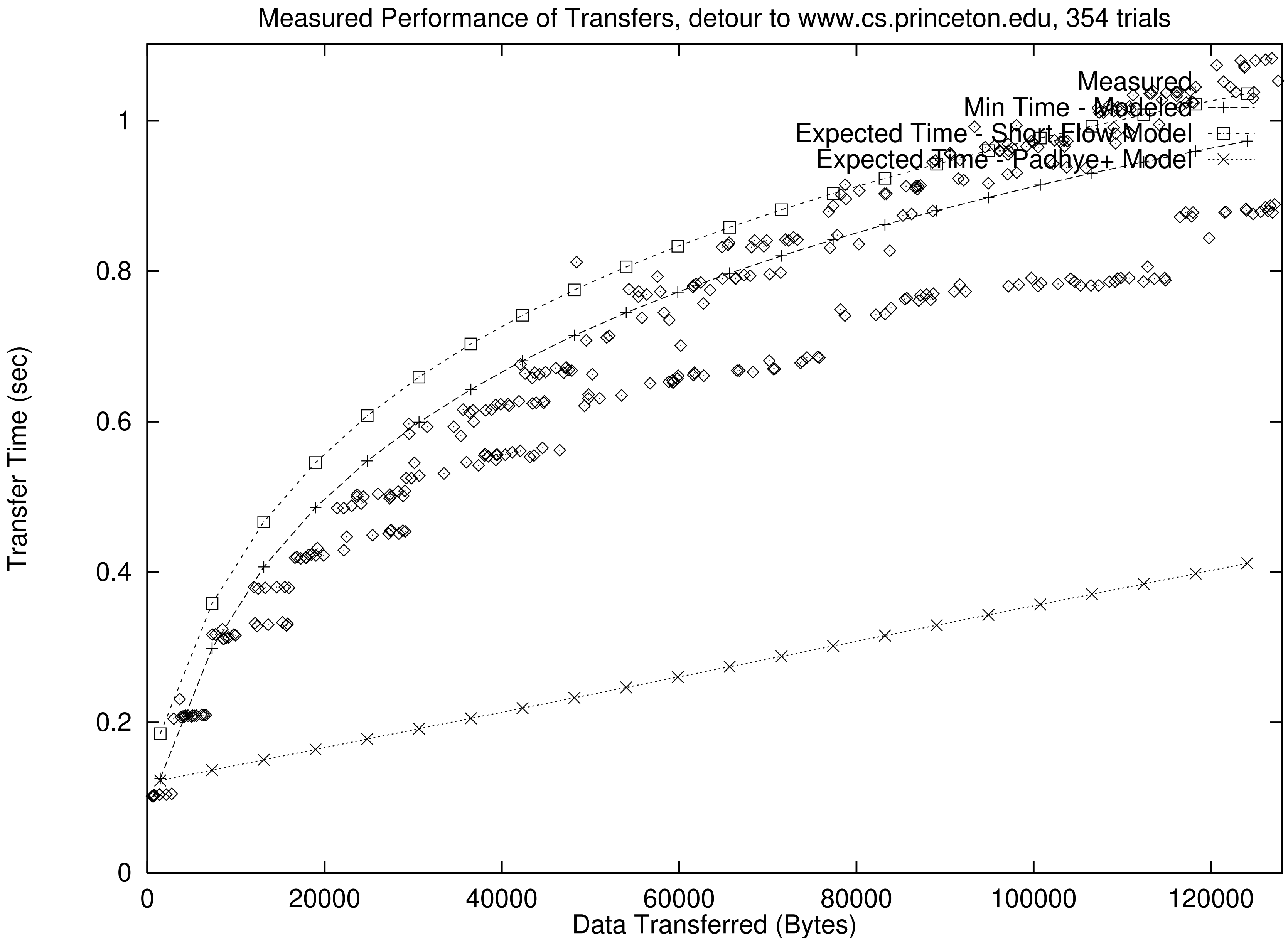
detour.cs to www.utexas.edu



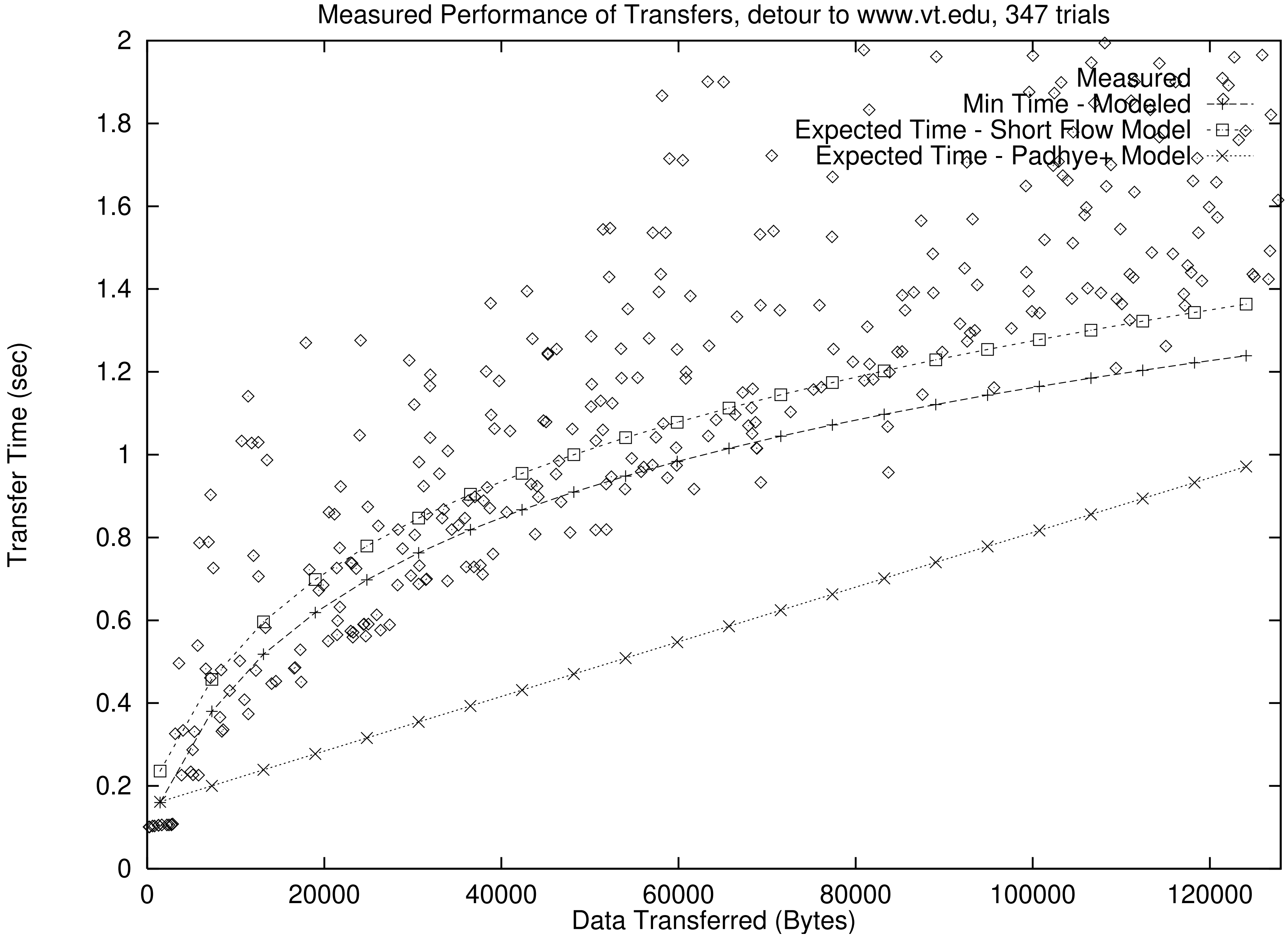
detour.cs to www.ucdavis.edu



detour.cs to www.cs.princeton.edu



detour.cs to www.vt.edu



Conclusions

- Lossless model (I) fits lossless scenarios well
- Steady-state model (II) can be adapted for short flows; fits for simulations but not real scenarios with low p
- New short flow model (III) approximates typical performance of short flows across several simulated and real scenarios

The End

So long ago when we were taught
That for whatever kind of puzzle you've got,
You just stick the right formula in,
A solution for every fool
– *Emily Saliers, "Least Complicated"*