

Dakshi Agrawal, James Giles, and Dinesh C. Verma

T J Watson Research Center

IBM Corporation

Hawthorne, NY 10532

USA

Client Perceived Response Time and Its Relationship to Different Link and Page Characteristics

{agrawal, jamesgiles, dverma}@us.ibm.com

<http://www.research.ibm.com/people/a/agrawal>

Outline

- Motivation and Objectives
- Experimental Setup
- Client Perceived Response Time and Its relationship with Round Trip Time
- Some Observations
- Future Work and Conclusions

Motivation

Validate Linear Model

$$T = N\tau + P$$

where

T = Client perceived response time of a web page

τ = Round trip time between client and web server

N and P are constants that depend on link and web page characteristics.

An analytic model has many potential uses:

- ♠ Evaluate performance of edge-servers and content distribution networks analytically.
- ♠ Compare different schemes to improve the performance of websites.

Other Objectives

- Evaluate content distribution network effectiveness
- Itemize client perceived response time of typical web pages among different components such as connect time, delivery time etc.
- Understand causes of delays in downloading web pages
- Relationship of download time to page size, number of objects in the page etc.
- Compare different schemes to improve web performance

Experimental Setup



Notes:

- LAN 2 is capable of setting link bandwidth, link delay, and packet loss characteristics by using a kernel module rshaper.o. This module is only available for linux kernels.
- Measurements at the client are done by using Page Detailer. This software is available only for Windows.
- LAN 1 has a large bandwidth (100 Mbps), and delay and packet loss on it are negligible. Therefore network characteristics between the client machine and the web server are governed by LAN 2.
- All three machines have very light load.

Experimental Setup

(Client)

Client Configuration

- ThinkPad 770 with 233 MHz Pentium and 98MB RAM.
- Microsoft Windows NT Version 4.0
- Internet Explorer(IE) Version 5.5
- Page Detailer for Windows Version 3.5
 - Provides a fine grain itemization of total download time
 - Provides other fine grain details about a web transactions such as http headers, number of bytes in each downloaded object etc.
- WebExp: A VB tool that uses IE as a component
 - Can download a web page repeatedly using IE
 - Can cache (or not cache) web page content, cookies, DNS results.

(Firewall and Web server)

Experimental Setup

- Firewall and web server are 600 MHz Pentium and 128 MB RAM PCs running Red Hat 6.2. Linux kernel 2.2.12-20 is recompiled to include WAN module rshaper.o.
- Firewall is implemented by IP masquerading and IP forwarding using ipchains.
- Web server (Apache/1.3.9) uses locally stored copies of several popular web pages.

(What is one experiment?)

Experimental Setup

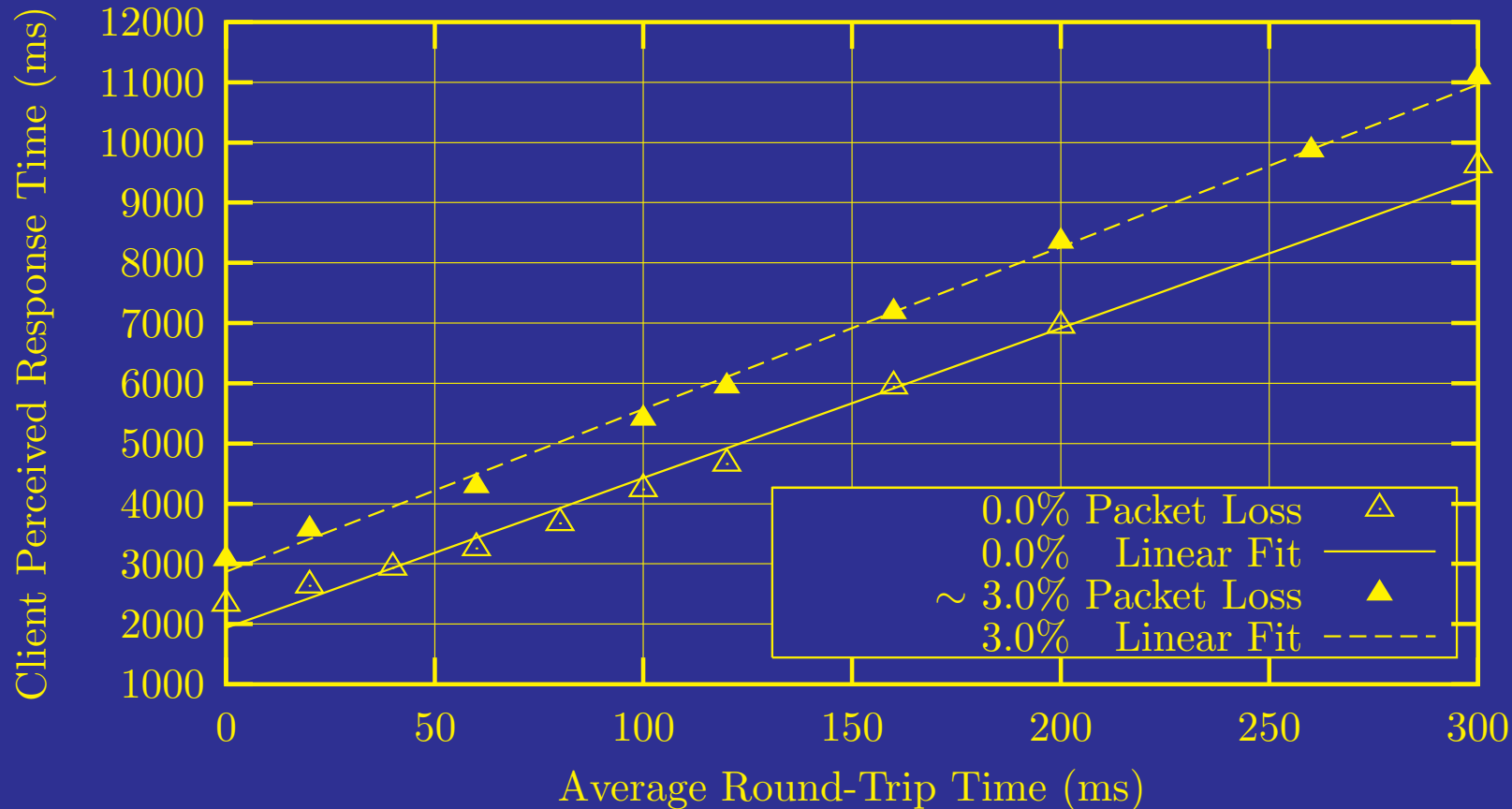
1. Set up link delay and packet loss rate between firewall and web server.
2. Verify link delay and packet loss rate using ping.
3. Run Page Detailer on Client machine.
4. Download a given web page 200 times using WebExp.
5. Save statistics collected by the Page Detailer and tag it by [web page, link delay, packet loss rate].

Repeat above steps for different link delays and packet loss rates.

(Validation of Linear Model)

Client Perceived Response Time

aol.com (95 KB, 43 Objects)



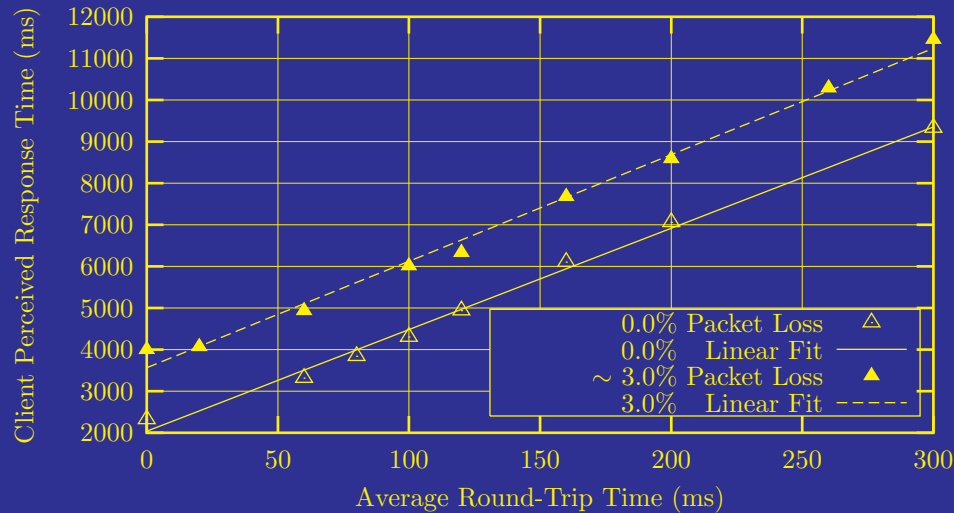
$$T = 24.9\tau + 1937$$

$$T = 27.0\tau + 2871$$

(Validation of Linear Model)

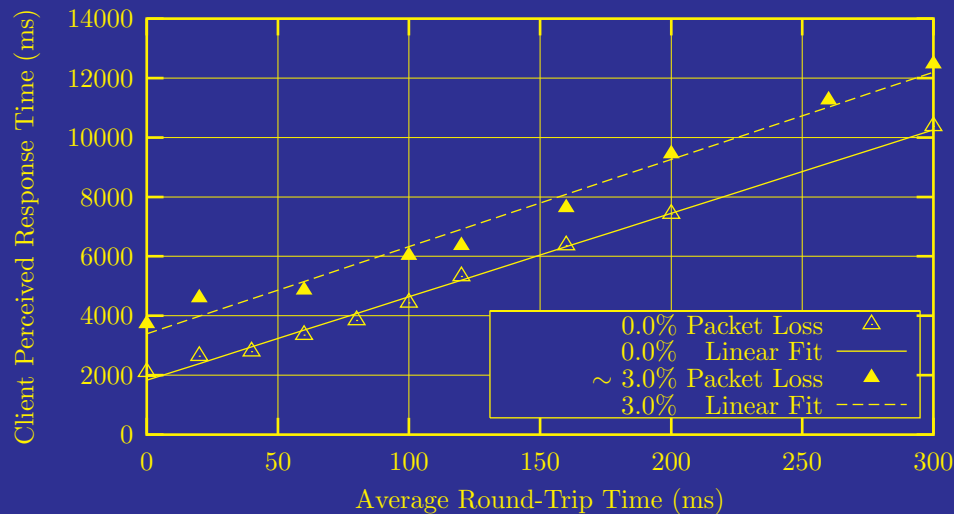
Client Perceived Response Time

zdnet.com (149 KB, 29 Objects)



$$T = 25.6\tau + 2040$$
$$T = 24.4\tau + 3565$$

ebay.com (123 KB, 42 Objects)

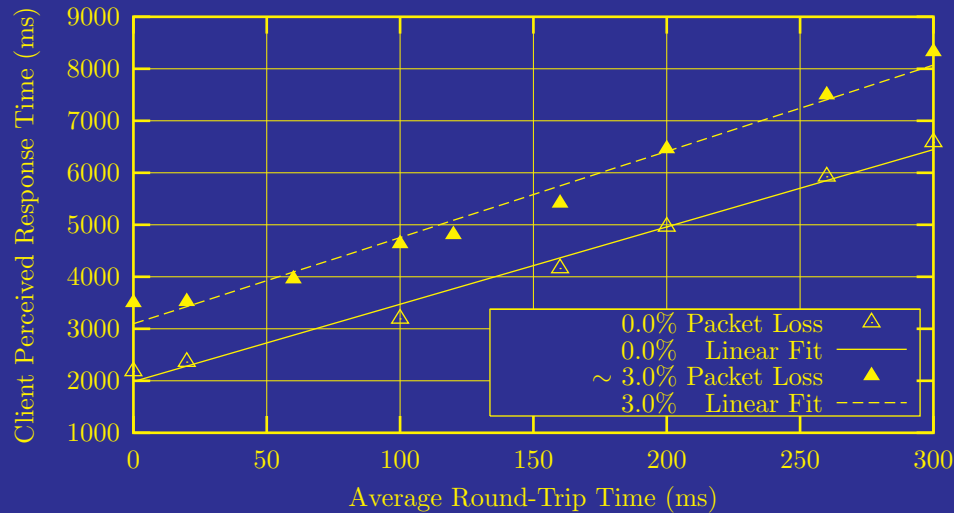


$$T = 28.1\tau + 1829$$
$$T = 29.3\tau + 3394$$

(Validation of Linear Model)

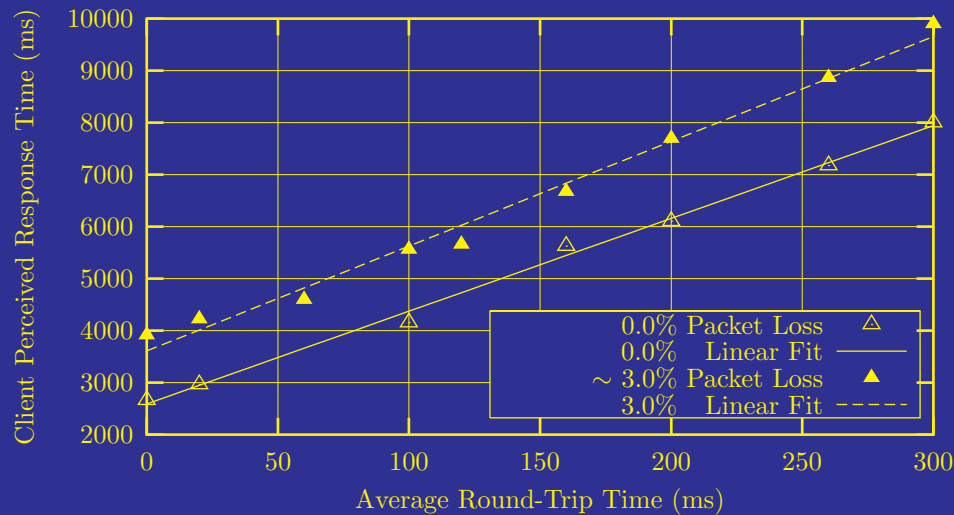
Client Perceived Response Time

ibm.com (68 KB, 21 Objects)



$$T = 14.9\tau + 1983$$
$$T = 16.6\tau + 3095$$

intranet.com (76 KB, 26 Objects)

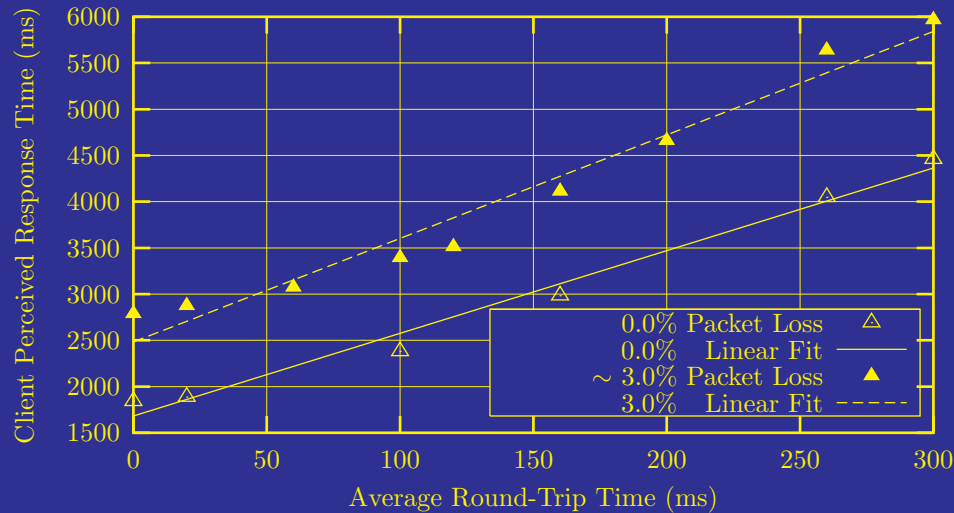


$$T = 17.8\tau + 2589$$
$$T = 20.2\tau + 3609$$

(Validation of Linear Model)

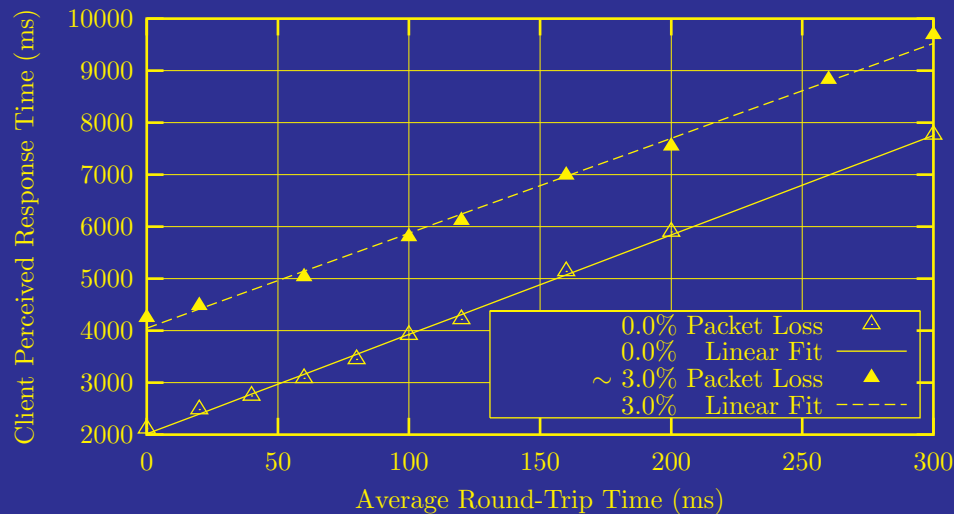
Client Perceived Response Time

schwabb.com (56 KB, 12 Objects)



$$T = 08.9\tau + 1681$$
$$T = 11.2\tau + 2482$$

microsoft.com (87 KB, 19 Objects)

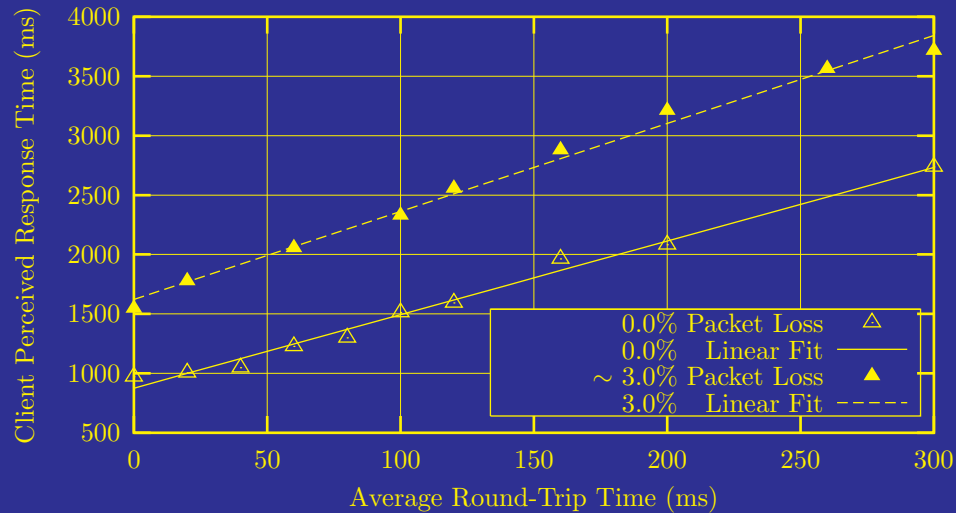


$$T = 19.1\tau + 2013$$
$$T = 18.3\tau + 4048$$

(Validation of Linear Model)

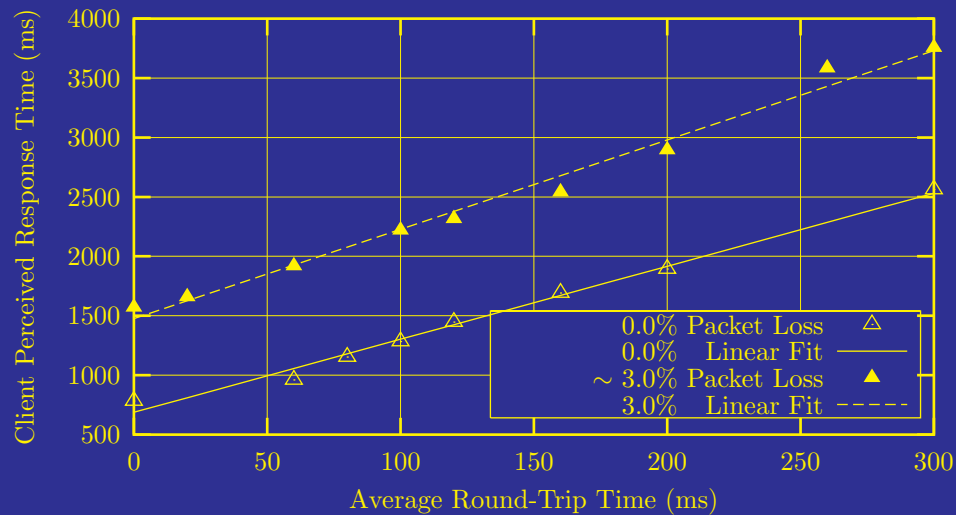
Client Perceived Response Time

lycos.com (34 KB, 6 Objects)



$$T = 6.2\tau + 876$$
$$T = 7.4\tau + 1621$$

yahoo.com (45 KB, 4 Objects)

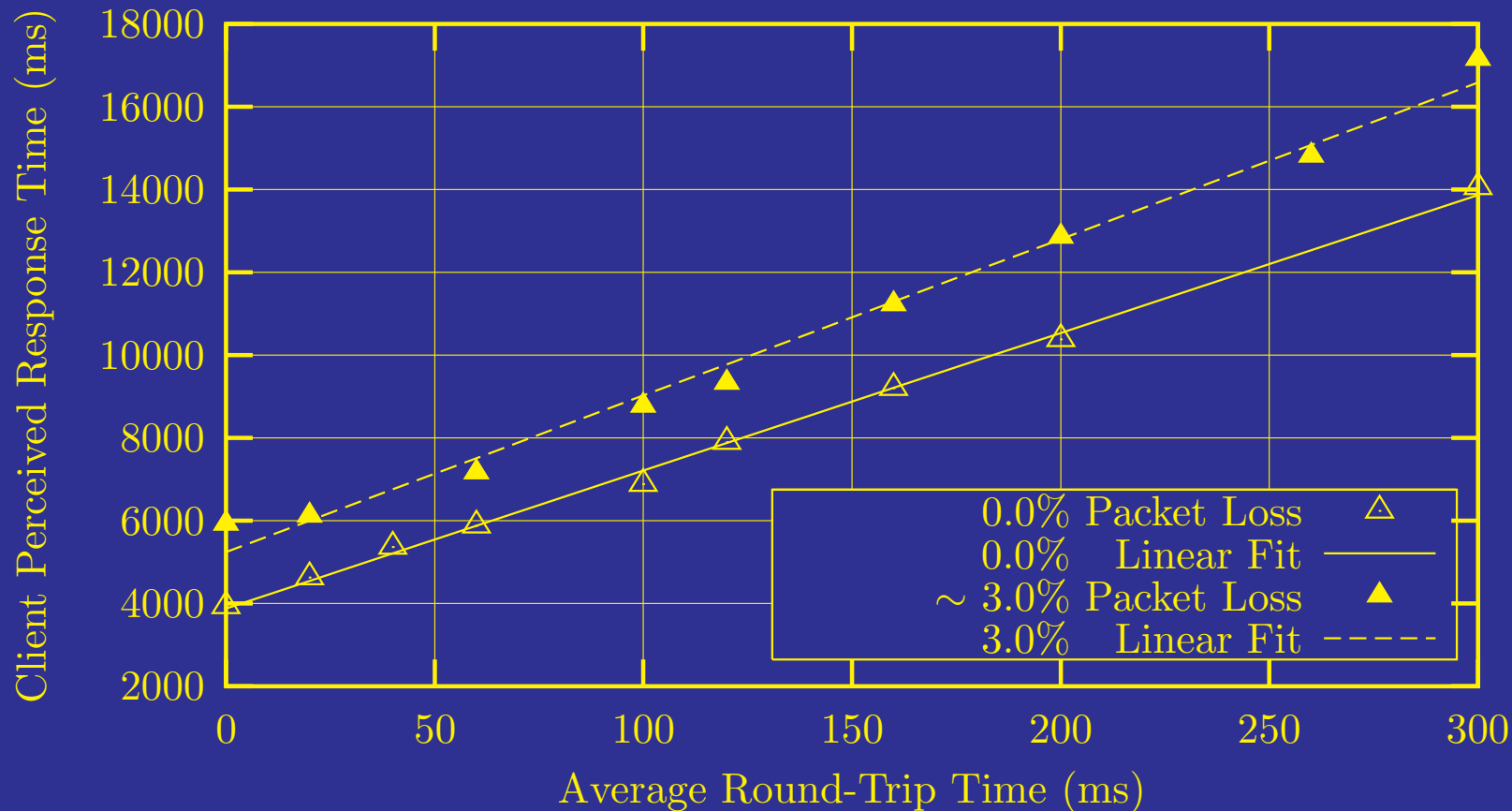


$$T = 6.1\tau + 688$$
$$T = 7.5\tau + 1475$$

(Validation of Linear Model)

Client Perceived Response Time

cnn.com (219 KB, 47 Objects)



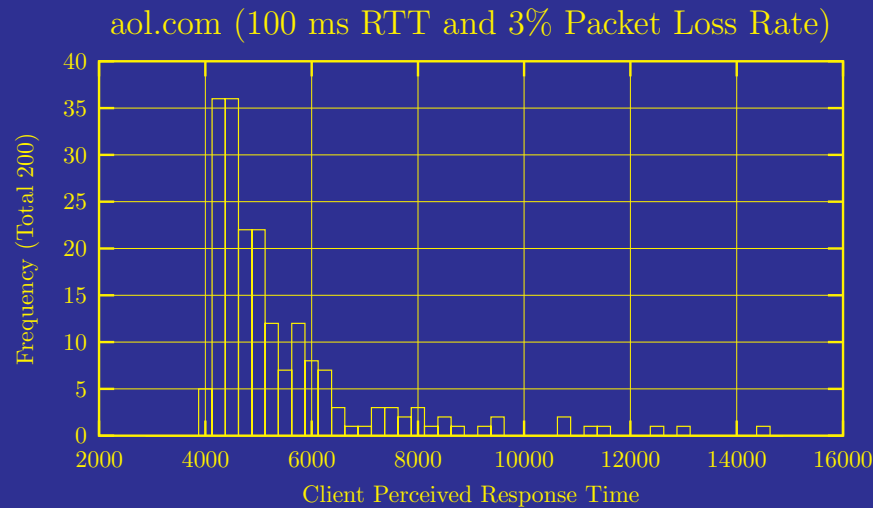
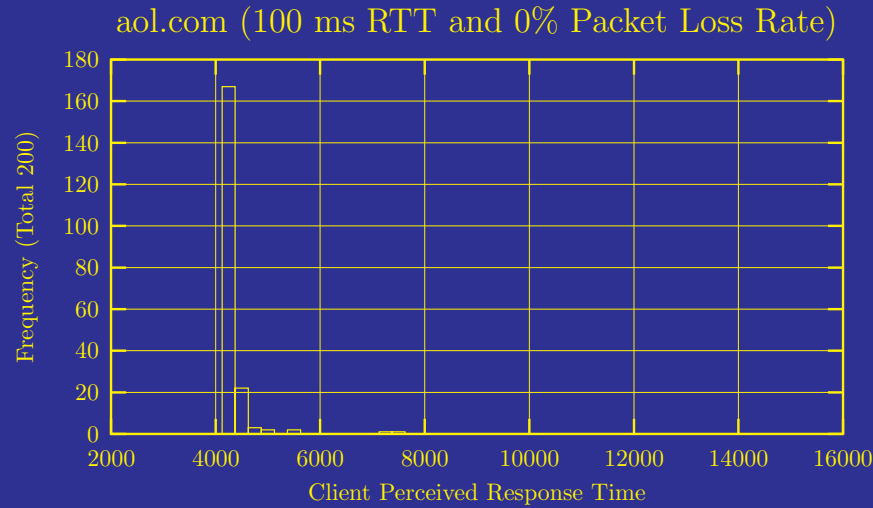
$$T = 33.3\tau + 3881$$

$$T = 37.8\tau + 5242$$

Some Observations

- The constant term (P) in the linear model of a web page does not have a strong correlation with either the page size or the number of objects in the web page.
 - ebay.com was ranked **third out of ten** sites in terms of both the page size and the number of objects, but its P value was ranked **seventh out of ten**.
 - w3.ibm.com was **sixth and fifth** in terms of page size and number of objects respectively, but its P value was **second out of ten**.
 - supports the hypothesis that a large part of P is due to processing at client and web server.
- The slope N of the linear model *is* strongly correlated with both the page size and the number of objects.

Some Observations



Notes:

Due to packet losses client perceived response time is more dispersed. There are major clusters around 3 and 6 seconds more than the response time for lossless case.

Components of Response Time

Some Observations

- In our setup, there are three major time components involved in an HTTP request
 - **Connection Time:** Time to get a TCP connection ready
 - **Server/Client (S/C) Response Time:** Time between the end of sending the HTTP request and the 'beginning' of receiving the response to request
 - **Delivery Time:** Time between the beginning and the end of HTTP response.
- In other cases, there could be components for SOCKS server, SSL connection setup, DNS name resolution, cookie setup etc.

(Breakdown of Transaction Time) Some Observations

- Breakdown of client perceived response time into its major components for a web page is a tough task due to
 - multiple embedded objects being downloaded over multiple connections to the server
 - Page Detailer cannot itemize the time during which communication between browser and communication socket is blocked
- To get an idea of the relative magnitudes of connection time, S/C response time, and delivery time, we can just simply add these time components for different objects within a web page ignoring any overlap.

(Breakdown of Transaction Time) Some Observations

RTT	cnn.com			ebay.com		
	Connect	S/C Response	Delivery	Connect	S/C Response	Delivery
0	255	3016	3401	321	2187	1963
100	500	7472	4919	388	5484	3604
200	660	12113	7709	764	9909	4630
300	855	17910	10642	830	14592	5801

RTT	cnn.com			ebay.com		
	Connect	S/C Response	Delivery	Connect	S/C Response	Delivery
0	4%	45%	51%	7%	49%	44%
100	4%	58%	38%	4%	58%	38%
200	3%	59%	38%	5%	65%	30%
300	3%	61%	36%	4%	69%	27%

Notes:

- The fraction of time occupied by server response time increases while the fraction of time occupied by content delivery decreases.
- It is not yet clear how these numbers translate into **effective** fractions.

(Breakdown of Transaction Time) Some Observations

- For faster client machines, client perceived response times are much smaller, however S/C response time still constitutes a major fraction of response time.
- Currently, major suspects for causing long response times are
 - Execution of javascript
 - Item sequences in a web page that do not fully exploit parallelism
 - Large page size
 - Large number of items
- Our server used persistent connections, but many others do not. This results into longer response time.
- Download of .cab files. real.com downloaded swflash.cab twice, once from activex.microsoft.com, and second time from akamai.net

Factors Not Considered

- Dynamically generated web pages
- Busy servers/clients
- Web pages composed from objects that come from multiple servers

(micro level tasks)

Future Work

- Put S/C time under a greater scrutiny.
- Find out reasons behind different server response times for different objects on the same page.
- Investigate reasons behind communication blockade between browser and socket.
- Plot other metrics such as median response time as a function of round trip time.