



Késsia Thais Cavalcanti Nepomuceno

UM ESTUDO EXPERIMENTAL SOBRE NAVEGAÇÃO WEB

Trabalho de Graduação



Universidade Federal de Pernambuco

RECIFE
2018



Universidade Federal de Pernambuco
Centro de Informática
Graduação em Ciência da Computação

Késsia Thais Cavalcanti Nepomuceno

UM ESTUDO EXPERIMENTAL SOBRE NAVEGAÇÃO WEB

Trabalho apresentado ao Programa de Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Bacharel em Ciência da Computação.

Orientador: *Djamel Fawzi Hadj Sadok*

RECIFE
2018

*I dedicate this work to all my family, friends and professors
who gave me the necessary support to get here.*

Resumo

Nos últimos anos, a Internet ficou sobrecarregada com informações devido à quantidade de dados e o número de usuários aumentando diariamente. Uma das atividades afetadas por esse crescimento é a navegação Web. A navegação Web é controlada principalmente pelos protocolos Transmission Control Protocol (TCP) e HyperText Transfer Protocol (HTTP). No entanto eles foram projetados para a Internet dos anos 90. Quando esses protocolos foram criados, não foi previsto o enorme crescimento que a Internet teria. Assim, eles podem ser incompatíveis com o cenário atual da navegação Web que exige maior segurança e velocidade. Visto isso, este trabalho tem como objetivo pesquisar e desenvolver mecanismos para avaliar o desempenho da Internet através dos protocolos de comunicação e análise envoltória de dados (do inglês Data Envelopment Analysis (DEA)). A Internet será avaliada em duas perspectivas. A primeira consiste na análise dos principais protocolos de transporte do modelo Open System Interconnection (OSI), TCP e Quick UDP Internet Connections (QUIC), onde foi possível constatar melhor performance do TCP. A segunda análise consiste na avaliação de páginas Web através do ranqueamento da performance dessas páginas que será obtido pelo DEA. Como resultado foi obtido as páginas que atingiram a fronteira de eficiência, ou seja, aqueles que tiveram melhor performance.

Palavras-chave: Análise Envoltória de Dados, Eficiência, Páginas Web, QUIC, TCP

Abstract

In recent years, the Internet has been overwhelmed with information due to the amount of data and the number of users increasing daily. One of the activities affected by this growth is Web browsing. Web browsing is controlled primarily by TCP and HTTP protocols. However they were designed for the Internet of the 90s. When these protocols were created, it was not predicted the tremendous growth that the Internet would have. Thus, they may be incompatible with the current scenario of Web browsing that requires greater security and speed. Thus, this work aims to investigate and develop mechanisms to evaluate the performance of the Internet through communication protocols and DEA. The Internet will be evaluated in two perspectives. The first one is the analysis of the main transport protocols of the OSI model, TCP and QUIC, where it was possible to see a better performance of ac tcp. The second consists of the analysis of web pages through the ranking of the performance of these pages that will be obtained by the DEA. As a result, the pages that reached the efficiency frontier were obtained, that is, those that had better performance.

Keywords: Data Envelope Analysis, Efficiency, Web Pages, QUIC, TCP

Lista de Figuras

3.1	Função de Distribuição Acumulada (FDA) dos objetos das páginas Web	24
3.2	Cenário dos protocolos	26
4.1	Page Load Time (QUIC - TCP) para o evento de <i>onLoad</i> com cache desabilitada.	32
4.2	Page Load Time (QUIC - TCP) para o evento de <i>onContentLoaded</i> com cache desabilitada.	32
4.3	Page Load Time (QUIC - TCP) para o evento de <i>onLoad</i> com cache habilitada.	33
4.4	Page Load Time (QUIC - TCP) para os eventos de <i>onContentLoaded</i> com cache habilitada.	33
4.5	Page Load Time (PLT) do QUIC e TCP para o evento de <i>onLoad</i> nos experimentos com cache desabilitada	35
4.6	Gráfico da fronteira de eficiência para os inputs	38
4.7	Gráfico da fronteira de eficiência para a combinação de inputs (eixo X) e output (eixo Y)	39

Lista de Tabelas

3.1	Fatores dos Experimentos	23
3.2	Combinação das configurações de Round Trip Time (RTT) e taxa de perda de pacotes	25
3.3	Recursos (inputs) e resultados (outputs) utilizados na análise envoltória dos dados	28
4.1	Porcentagens de páginas para as quais o QUIC apresentou desempenho melhor que o TCP sob a configuração de cache ativado (CA) e cache desativado (CD) .	34
4.2	Tabela de eficiência das páginas	35
A.1	Lista dos 100 sites mais acessados do mundo de acordo com o site Alexa. . . .	51

Lista de Acrônimos

DEA	Data Envelopment Analysis	18
FDA	Função de Distribuição Acumulada	9
HAR	HTTP Archive	26
HTTP	HyperText Transfer Protocol	17
HTTPS	Hyper Text Transfer Protocol Secure	25
OSI	Open System Interconnection	5
PLT	Page Load Time	17
QUIC	Quick UDP Internet Connections	18
RTT	Round Trip Time	20
TCP	Transmission Control Protocol	17
TLS	Transport Layer Security	20
UDP	User Datagram Protocol	20

Sumário

1	Introdução	17
2	Fundamentação Teórica	19
2.1	Protocolos de Comunicação	19
2.2	Análise Envoltória de Dados (DEA)	21
3	Metodologia	23
3.1	Testbed e Experimentos	23
3.2	Fatores e Métricas	27
4	Resultados	31
4.1	Comparação dos Protocolos	31
4.2	Análise da Performance das Páginas	34
5	Discussão	41
5.1	Comparação dos Protocolos	41
5.2	Análise de Performance das Páginas	42
6	Conclusão	45
	Referências	47
	Apêndice	49
A	TOP 100 Alexa Site	51

1

Introdução

Desde a sua criação, a Internet tem mantido um crescimento constante de usuários e um crescimento significativo do tráfego. De acordo com a previsão da Cisco Visual Networking Index TM (VNI), esse crescimento não mudará em breve. No ano de 2020, o relatório prevê um volume de tráfego IP quase três vezes superior ao observado em 2015. Além disso, mais de um bilhão de novos usuários da Internet devem se unir à comunidade global, crescendo de três bilhões em 2015 para 4,1 bilhões em 2020 [Cisco System Inc. \(2016\)](#).

O crescimento inadvertido da Internet requer algumas alterações em sua infraestrutura, e também em seus protocolos e aplicações. A navegação Web é uma atividade crítica afetada, considerando o fato de que as páginas Web estão cada vez mais sobrecarregadas com informações, como objetos dinâmicos ou simplesmente mais dados.

A navegação Web é controlada ou suportada principalmente pelos protocolos Transmission Control Protocol (TCP) e HyperText Transfer Protocol (HTTP). No entanto, o HTTP foi projetado no início dos anos 90. Ao projetar tais protocolos não foi prevista a evolução do tráfego Web. Para solucionar alguns dos problemas, os middleboxes na forma de cache e servidores proxy foram introduzidos para reduzir o tempo de carregamento das páginas Web. Além disso, o surgimento de dispositivos sem fio trouxe a necessidade de acesso mais rápido a Internet.

No contexto da navegação Web, o tempo necessário para carregar uma página é um dos fatores mais importantes que determina a experiência e a satisfação do usuário final. A métrica de tempo de carregamento da página, Page Load Time (PLT), impacta na maneira como um usuário navega e influencia o tempo que ele gasta em um site. Usuários que estão frustrados com um site de carregamento lento, tendem a visitar o site apenas uma vez [Big Commerce \(2014\)](#). Além disso, o PLT parece ter um impacto sobre o quão bem os usuários recomendam sites para outras pessoas [PR Newswire \(2008\)](#). Na verdade, alguns mecanismos de pesquisa, como o Google, usam o tempo de carregamento da página em seus algoritmos como um dos fatores para exibir seus resultados [Google Inc. \(2010\)](#), orientando os usuários para sites com PLT menor.

Como resultado, o PLT é de grande importância na navegação Web. No entanto, ele tende a aumentar cada vez mais devido ao crescimento da quantidade e variedade de dados da Internet. A comunidade da Internet teme fortemente que essa evolução possa levar ao colapso do

congestionamento no tráfego TCP [WIDMER; DENDA; MAUVE \(2001\)](#).

Para desenvolver uma alternativa que melhore o carregamento de páginas Web e torne a Internet mais rápida, pesquisadores da indústria e da academia apresentaram dois novos protocolos, o Quick UDP Internet Connections (QUIC) e o SPDY. Eles foram criados para substituir o TCP e o HTTP, respectivamente. Recentemente as especificações do SPDY foram usadas como base para o design do protocolo HTTP 2.

Embora esses protocolos tenham sido projetados para reduzir a latência na Internet. É necessário verificar e analisar o funcionamento dos seus recursos.

Neste trabalho é feita uma comparação de desempenho entre o QUIC e o TCP para estabelecer qual deles é mais rápido em condições de rede que refletem a Internet atual. A abordagem experimental utilizada consiste em comparar o tempo de carregamento de páginas Web usando os dois protocolos, pois é importante saber se os novos protocolos realmente operam como projetados e se atendem às demandas atuais da Internet.

Além disso, foi utilizada a técnica de análise envoltória de dados como ferramenta para estabelecer o ranqueamento da performance das páginas Web e também avaliar o tempo de carregamento dessas mesmas páginas. A partir disso foi possível analisar a eficiência das páginas mais acessadas com base no tempo de carregamento. Vale salientar que a técnica Data Envelopment Analysis (DEA) é bastante utilizada em avaliações de performance, por considerar vários inputs e outputs próprios do ambiente e também por não ser necessário precificar as variáveis envolvidas.

Esta monografia está organizada da seguinte forma: No capítulo 2 é fornecido a fundamentação teórica sobre a evolução da Internet, os protocolos de transporte e também análise envoltória de dados. O capítulo 3 apresenta o cenário de avaliação, descrevendo as ferramentas utilizadas e como os experimentos foram realizados. São apresentados também os fatores e métricas utilizadas e sua importância. O capítulo 4 apresenta os resultados dos experimentos, seguidos por uma discussão de tais resultados no capítulo 5. O capítulo 6 apresenta um resumo das lições aprendidas e conclui este estudo.

2

Fundamentação Teórica

Esse capítulo apresentará os principais protocolos de comunicação e a relevância deles para as telecomunicações. Também apresentará conceitos e aplicações de análise envoltória de dados.

2.1 Protocolos de Comunicação

TCP e HTTP são dois dos protocolos mais conhecidos na Internet, seus desenvolvimentos permitiram a troca padronizada de informações na Web, onde máquinas que obedecem a certas regras podem trocar informações entre si. No entanto, ao longo dos anos, a Internet teve uma grande evolução. Anteriormente, as páginas Web eram estáticas e simples, hoje em dia são muito mais complexas e dinâmicas [HTTP Archive \(2010\)](#).

Esses protocolos não foram projetados visando à latência, portanto, com toda evolução que a Internet teve nos últimos anos, eles podem deixar a desejar ao atender os requisitos da nova Internet. Quando foram projetados, não se pretendia carregar páginas Web rapidamente, mas apenas estabelecer comunicação entre diferentes máquinas. Embora esses protocolos tenham evoluído desde a criação da World Wide Web, eles atualmente representam o gargalo da nova conectividade com a Internet [BELSHE; PEON \(2012\)](#). Podemos listar os seguintes problemas que esses protocolos podem representar hoje em dia na “nova Internet”:

- Uma única conexão por solicitação, um segmento TCP não pode carregar mais de uma solicitação ou resposta HTTP;
- Somente clientes podem iniciar solicitações, não servidores;
- Cabeçalhos não compactados e redundantes.

Um dos gargalos dos protocolos HTTP e TCP é que eles dependem de várias conexões para simultaneidade. Isso causa uma série de problemas, incluindo pacotes adicionais para configurar a conexão, atrasos de inicialização e racionamento de conexão do cliente [BELSHE; PEON \(2012\)](#). Com relação a esses problemas, o Google e outras organizações propuseram

dois novos protocolos com o objetivo de eliminar ou minimizar os problemas mencionados e tornar a Internet mais rápida e segura. São protocolos desenvolvidos para operar nas camadas de aplicação/sessão e transporte, respectivamente: SPDY e Quick UDP Internet Connections (QUIC).

O QUIC é um protocolo de transporte experimental criado pelo Google e projetado para fornecer segurança e confiabilidade, juntamente com conexão e latência reduzidas [CARLUCCI; DE CICCIO; MASCOLO \(2015\)](#). O QUIC foi inicializado em 2012 com a motivação de reduzir o Round Trip Time (RTT) a zero e ajudar o SPDY, um protocolo também desenvolvido pelo Google para substituir o HTTP. O QUIC herdou alguns recursos do SPDY e solucionou alguns problemas, como *head-of-line-blocking* [MEDINA; SHEMLA; SOLT \(2004\)](#) e o *three-way handshake* [USHA RANI J MADHU M NAYAK \(2017\)](#).

O QUIC combina os melhores recursos do protocolo User Datagram Protocol (UDP) e TCP com ferramentas de segurança e tem como principal objetivo minimizar a latência da Internet [POSTEL \(1980\) ERMAN et al. \(2015\)](#). O QUIC também implementa na camada de aplicação alguns recursos que não são possíveis através do UDP, como controle de congestionamento e retransmissão. Não é possível construir o QUIC sobre TCP, porque o TCP já está implementado nos kernels do sistema operacional e nos firmwares e middlebox. Assim, devido à estrutura em torno do TCP, é improvável que ocorram modificações complexas nele [Google Inc. \(2014\)](#).

O SPDY é o protocolo de aplicação desenvolvido pelo Google em 2009 como uma alternativa para adicionar eficiência ao protocolo HTTP e diminuir a latência na Internet. Ele usa recursos como compressão de dados e envio, o que fornece solicitações para que os recursos sejam enviados mais rapidamente do servidor para o navegador [REGUNDWAR; SHUKLA; LOKHANDE \(2013\)](#). O SPDY procura uma navegação mais rápida através do uso de multiplexação, prioridade, compressão e Transport Layer Security (TLS) [DIERKS \(2008\)](#) e [ROSKIND \(2013\)](#).

De uma maneira didática, podemos dizer que o SPDY é dividido em duas camadas. A camada de enquadramento (*framing layer*) e a camada de aplicação (*application layer*). A camada de enquadramento multiplexa os dados com uma camada de transporte confiável. Ou seja, a camada de enquadramento usa um único canal para enviar os dados por meio do protocolo de transporte. A camada de aplicação descreve a mecânica na qual o SPDY substitui as solicitações e respostas HTTP. A princípio o SPDY, juntamente com o QUIC, pretende tornar a Internet mais rápida reduzindo o tempo de carregamento da página e resolvendo os problemas do TCP e HTTP mencionados acima. Eles possuem as seguintes características para atingir seus objetivos:

- Multiplexação: o QUIC multiplexa vários fluxos QUIC sobre a mesma conexão UDP. Um cliente pode enviar várias solicitações HTTP e receber várias respostas sobre o mesmo soquete UDP [CARLUCCI; DE CICCIO; MASCOLO \(2015\)](#).
- Solicita Priorização: Alguns objetos Web são mais importantes que outros e, portanto, devem ser carregados anteriormente. Os protocolos permitem que um cliente possa

especificar a ordem preferida na qual os recursos devem ser transferidos [WANG et al. \(2014\)](#) e [BRISCOE et al. \(2014\)](#).

- Push/Hint do Servidor: O recurso definido como push do servidor permite que o servidor envie recursos arbitrariamente sem uma solicitação explícita do cliente. Alternativamente, o SPDY pode enviar dicas (*hint*) para aconselhar os clientes a pré-buscar o conteúdo [ELKHATIB; TYSON; WELZL \(2014\)](#).
- Compactação de Cabeçalho: Usando a compactação de cabeçalho é possível reduzir a quantidade de informações redundantes do cabeçalho, cada vez que uma nova página é solicitada [ERMAN et al. \(2015\)](#).
- Criptografia universal [Google Inc. \(2014\)](#).

2.2 Análise Envoltória de Dados (DEA)

Análise Envoltória de Dados é uma técnica de otimização. É um método não paramétrico baseado em fronteiras lineares por partes ajustadas através de programação matemática. É utilizada para analisar as eficiências técnicas. Esse método envolve técnicas de programação linear a fim de construir uma superfície não paramétrica por partes (*piece-wise frontier*) envolvendo os dados [COELLI et al. \(2005\)](#). O método tem como objetivo determinar a eficiência de um conjunto de unidades de tomada de decisão (DMUs), considerando a existência de múltiplos inputs e outputs [CHARNES; COOPER; RHODES \(1978\)](#). Em outras palavras, é uma técnica de programação matemática que avalia a eficiência de unidades tomadoras de decisão (DMUs) como uma razão entre os recursos usados e resultados obtidos. Entre os resultados temos um score de eficiência para cada DMU que vai de zero a um, onde um significa que a unidade é eficiente, e zero significa que é extremamente ineficiente. Por exemplo, se uma unidade tem score 0,9 significa que é 90% eficiente, pode melhorar 10% e se tornar eficiente. Essa melhora pode ocorrer de diversas formas; diminuindo os recursos usados em 10% e mantendo o mesmo resultado ou aumentando em 10% o resultado e mantendo os recursos usados.

Na prática, DEA é utilizado para otimizar (maximizar ou minimizar) uma função linear de variáveis, chamada de função objetivo, sujeita a uma série de equações (ou inequações) lineares, chamadas restrições, ou seja, são problemas de otimização que a função objetivo e as restrições são lineares.

Com essa metodologia é possível comparar unidades grandes com pequenas pois compara uma unidade fictícia com as mesmas quantidades de recursos e resultados (inputs e output).

O objetivo deste trabalho é utilizar essa técnica para avaliação da eficiência em páginas Web. Através dos recursos utilizados e resultados obtidos.

O trabalho avalia a eficiência em páginas Web, com foco em seu PLT. Para isso, foi usada a técnica de análise envoltória de dados baseado no tamanho das páginas e quantidade de

objetos como inputs, e PLT como output.

As páginas consideradas 100% eficiente serão aquelas que terão menor PLT, maior quantidade de objetos e maior tamanho da página, dada sua razão entre inputs e outputs. As demais páginas se referenciarão a estas para avaliar sua performance; pois, mesmo se outras página possuem um PLT menor do que as páginas com eficiência máxima, ainda será necessário averiguar os inputs (quantidade de objetos e tamanho da página). Uma vez que eles são menores do que os das páginas 100% eficientes, não podemos categorizar as páginas avaliadas com eficiência máxima.

Os resultados apontam as páginas com melhor desempenho. Também foram encontradas as páginas que demandam melhorias para alcançar a fronteira de eficiência e onde podemos melhorá-las.

3

Metodologia

Para a execução dos experimentos, o ambiente foi configurado de forma que correspondesse o máximo possível à Internet de hoje em dia. Por meio de pesquisas, foram estabelecidas várias configurações que influenciam e representam a Internet do mundo real. Podem-se ver essas configurações na Tabela 3.1. Nos experimentos, foram utilizadas ferramentas confiáveis e consolidadas na comunidade de software. Nas próximas subseções, detalha-se a configuração dos experimentos, bem como as métricas e fatores utilizados.

Tabela 3.1: Fatores dos Experimentos

Fatores	Níveis
RTT (ms)	{20; 100; 200}
Packet Loss Rate	{0%; 0.5%; 1%; 2%}
Protocolos	QUIC; TCP
Cache	Habilitada; Desabilitada
Páginas Web	{Página#1; Página#2; ...; Página#100}

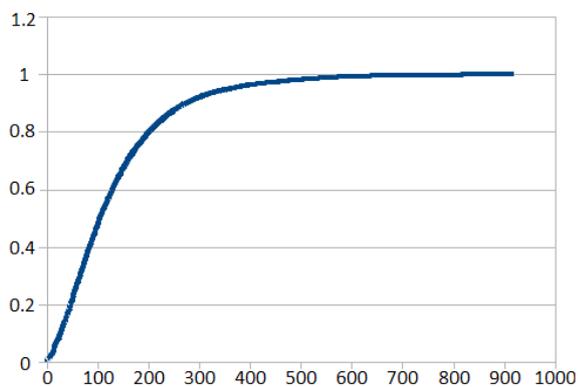
3.1 Testbed e Experimentos

O cenário que será descrito nesse capítulo foi projetado para refletir as condições reais do tráfego da Internet da maneira mais fiel possível. Portanto, foram usadas as cem páginas mais acessadas do mundo, de acordo com o ranking do site [Alexa site \(2017\)](#), para executar os experimentos. Os 100 principais sites do Alexa apresentam uma melhor representação da Internet, pois exibe os sites mais acessados do mundo e não estão restritos a um grupo. As páginas Web são de servidores diferentes, como página No. 1: Google.com; página No. 2: Facebook.com; página No. 3: Youtube.com; página No. 4: Baidu.com; página No. 5: Yahoo.com.

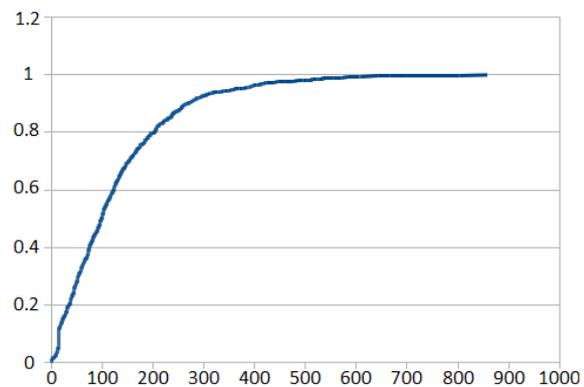
Antes de decidir que os experimentos seriam realizados com as cem páginas mais acessadas da Internet, foram escolhidas as primeiras dez mil páginas mais acessadas do site Alexa e foi analisado o número de distribuição de objetos (Figura 3.1a). A ideia desse passo inicial era encontrar uma quantidade significativa de páginas Web que permitisse conduzir os experimentos dentro de um período de tempo adequado. Então, foi diminuído o número

de páginas para mil páginas (Figura 3.1b), quinhentas páginas (Figura 3.1c) e cem páginas (Figura 3.1d). Ao longo da análise, foi observado que o número de distribuição de objetos manteve o mesmo comportamento. Como resultado, foi optado por executar os experimentos nas cem páginas Web mais acessadas de acordo com o Alexa.

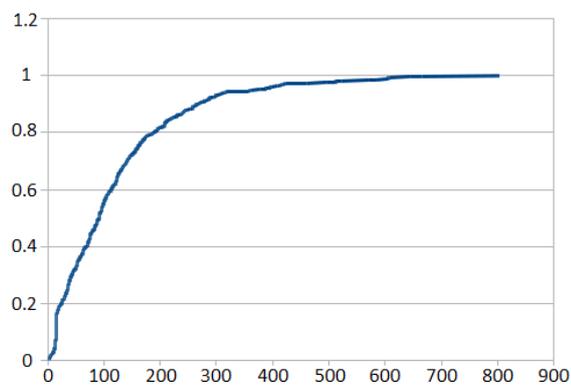
Para entender como o conteúdo é organizado e encontrar o número de objetos nas páginas do Alexa, foi usada a ferramenta Google BigQuery. O Google BigQuery contém um grande banco de dados com informações de páginas Web [Google Inc. \(2011\)](#). Assim, foi feito o upload das páginas do Alexa no Google BigQuery e, por meio de uma consulta, foi extraído o número de objetos dessas páginas, representado pelo eixo X dos gráficos da Figura 3.1a até a Figura 3.1d.



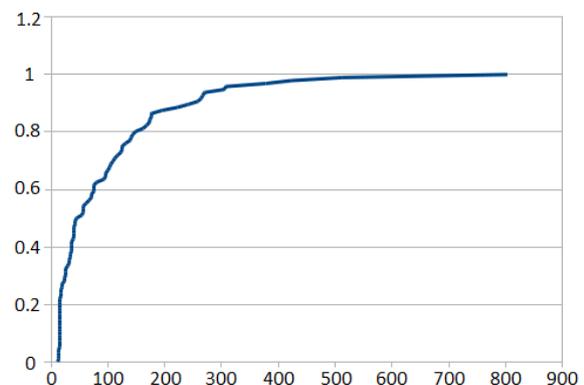
(a) Distribuição cumulativa dos objetos das 10k páginas Web mais acessadas de acordo com o site Alexa



(b) Distribuição cumulativa dos objetos das 1k páginas Web mais acessadas de acordo com o site Alexa



(c) Distribuição cumulativa dos objetos das 500 páginas Web mais acessadas de acordo com o site Alexa



(d) Distribuição cumulativa dos objetos das 100 páginas Web mais acessadas de acordo com o site Alexa

Figura 3.1: Função de Distribuição Acumulada (FDA) dos objetos das páginas Web

Com as cem páginas adquiridas, foi criado um script para gravar e reproduzir as páginas Web usando o Mahimahi, uma ferramenta que grava sites e os reproduz sob condições de rede emuladas [NETRAVALI et al. \(2015\)](#).

Para reproduzir as páginas, foi configurado todo o ambiente com máquinas virtuais usando o VMWare Player [VM Ware \(2017\)](#), uma ferramenta importante para obter um ambiente padrão em todos os testes, uma vez que garante a mesma configuração nos experimentos. Foi

usado o Ubuntu 16.04 como sistema operacional. As páginas foram executadas usando o Chrome versão 54.0.28.40.100 como cliente. Então, através do Mahimahi, as páginas foram executadas com os valores da taxa de perda de pacotes e RTT estabelecidos; essas configurações têm quatro e três níveis, respectivamente, totalizando 12 combinações de diferentes configurações que foram executadas (ver Tabela 3.1). Cada combinação foi realizada trinta vezes para cada uma das cem páginas. Pode-se ver a combinação dessas configurações (C1-C12) na Tabela 3.2.

Tabela 3.2: Combinação das configurações de RTT e taxa de perda de pacotes

Configuração	RTT	Taxa de Perda de Pacotes
Conf. 1	20 ms	0%
Conf. 2	20 ms	0.5%
Conf. 3	20 ms	1%
Conf. 4	20 ms	2%
Conf. 5	100 ms	0%
Conf. 6	100 ms	0.5%
Conf. 7	100 ms	1%
Conf. 8	100 ms	2%
Conf. 9	200 ms	0%
Conf. 10	200 ms	0.5%
Conf. 11	200 ms	1%
Conf. 12	200 ms	2%

A execução dessas páginas foi simulada com o protocolo QUIC ativado e desativado. Alternar entre esses estados, que é o foco principal dos experimentos, é possível através do servidor Web Caddy, um servidor Web de código aberto escrito em Go [Caddy Server \(2015\)](#). No momento em que os experimentos foram realizados, o FEC (forward error correction), a técnica de recuperação de perda de pacotes, foi desativado no Caddy pois havia sido desativado pelo QUIC [Github \(2016\)](#).

Como algumas das páginas gravadas têm recursos que usam HTTP, e QUIC requer Hyper Text Transfer Protocol Secure (HTTPS), foi utilizado um proxy para redirecionar o tráfego entre o cliente e o servidor. Para ser justo e não influenciar os resultados, os experimentos QUIC e TCP foram forçados a se conectar via proxy.

A Figura 3.2 resume as configurações do protocolo usadas durante os experimentos. Na camada de aplicação, foi usado HTTP 1.1 ou HTTP 2. O primeiro foi combinado com o protocolo TCP, enquanto o último foi usado ao analisar o protocolo QUIC. Os protocolos são combinados desta forma, considerando a comparação entre a estrutura que é esperada da Internet (QUIC e HTTP 2) com a Internet padrão de hoje (TCP e HTTP 1.1).

Para ter certeza de que o protocolo QUIC estava sendo usado quando configurado durante os experimentos, foi usado a ferramenta tcpdump para inspecionar os dados. Essa etapa é necessária porque, às vezes, enquanto uma solicitação é feita para usar o protocolo QUIC, ela pode não receber o sinal para seguir em frente e, em vez disso, usar o TCP.

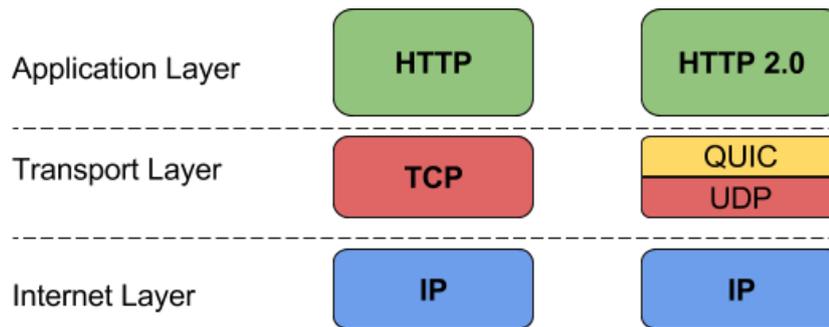


Figura 3.2: Cenário dos protocolos

Para adquirir o tempo de carregamento da página, foi capturado o registro de arquivos do HTTP, HTTP Archive (HAR), para cada uma das cem páginas em todas as combinações de fatores. O formato HAR fornece registros detalhados das ações executadas pelo navegador ao carregar a página [BUTKIEWICZ; MADHYASTHA; SEKAR \(2011\)](#). Para capturar o HAR foi usada a ferramenta Xdotool, responsável por simular os movimentos do teclado e do mouse e captura o arquivo na página Web [SISSEL \(2013\)](#). Foram coletados dois eventos importantes do arquivo HAR, o evento *onContentLoaded* e o evento *onLoad*. Para cada protocolo, foi obtido um total de 72.000 resultados de ambos os eventos. Para cada evento, foi calculada a média dos valores em cada configuração diferente e foi obtido o valor final de *onLoad* e *onContentLoaded* para cada página. Devido ao fato de que o protocolo QUIC só pode iniciar suas operações após a primeira conexão, os experimentos foram configurados com o cache de disco ativado e desativado. Será discutido este tópico no capítulo 5.

Por fim, para análise envoltória dos dados, foram avaliados os cem sites com os respectivos valores:

- Quantidade de objetos;
- Tamanho da página;
- Métrica *onLoad* com o protocolo TCP habilitado na conf.7;

Esses valores foram selecionados, pois afetam diretamente a performance do site, fazendo com que eles tenham um bom ou mau desempenho comparado aos outros. Uma vez que se tinham essas informações na planilha, foi executado o algoritmo para fazer o cálculo de score de eficiência para cada DMU, no caso, para cada site. Os inputs, ou recursos, usados foram os valores de quantidade de objeto e tamanho das páginas; o output, ou resultado, obtido foi o tempo de carregamento da página representado pela métrica *onLoad*; a eficiência resultante desses inputs e output será apresentado no capítulo 4. Através deles é possível visualizar as páginas com melhores desempenhos e as páginas que precisam de melhoria para alcançar a fronteira de eficiência.

3.2 Fatores e Métricas

Nos experimentos, quando as páginas gravadas foram reproduzidas, foram considerados os seguintes fatores que afetam o desempenho do carregamento das páginas Web:

- Tempo de ida e volta (RTT) com 20ms (*intra-coast*), 100ms (*inter-coast*) e 200ms (*3G ou cross-continent*). Tais valores são sugeridos por [WANG et al. \(2014\)](#);
- Perda de pacotes com 0%, 0,5%, 1% e 2%, estudos realizados alegam que os servidores do Google têm uma taxa de perda entre 1% e 2% [WANG et al. \(2014\)](#);
- As páginas gravadas. Em vez de usar o tamanho e a quantidade de objetos como fatores, eles foram substituídos pelos 100 sites mais acessados de acordo com o Alexa, conforme discutido anteriormente;
- Cache ativada e cache desativada. Todos os experimentos foram realizados com o cache desativado e, em seguida, todos os experimentos foram repetidos uma segunda vez com o cache ativado;
- E finalmente, foi variado o protocolo da camada de transporte, QUIC e TCP.

Embora a largura de banda seja um fator importante nas avaliações de rede, foi optado por não empregá-la no trabalho devido à natureza dos experimentos. A janela de controle de congestionamento do protocolo não aumenta até o nível de uso de toda a largura de banda, ou seja, a transmissão termina antes que a conexão a utilize completamente. Portanto, uma largura de banda maior ou menor não influencia os resultados.

Conforme mencionado no capítulo 3.1, foram usados os seguintes eventos HAR como métricas de avaliação:

- *onLoad* mede o tempo para renderizar completamente todos os componentes da página; [BUTKIEWICZ; MADHYASTHA; SEKAR \(2011\)](#);
- *onContentLoaded* mede o tempo necessário para começar a renderização do conteúdo da página [BUTKIEWICZ; MADHYASTHA; SEKAR \(2011\)](#).

Considerando que estão sendo coletados resultados de cerca de 100 sites em diferentes configurações de rede e protocolos, seria difícil mostrar informações precisas sobre determinadas páginas. No entanto, é justamente isso que este trabalho não está interessado em mostrar. O foco é a análise dos dois protocolos, QUIC e TCP, correspondentes às atuais e novas configurações da Internet.

Essas métricas também foram utilizadas como output para executar análise envoltória dos dados, como explicado na subseção acima.

A Tabela 3.3 apresenta os inputs e output utilizados para fazer a análise de eficiência das páginas, a primeira coluna são as páginas que no contexto de DEA simbolizam as DMUs. Em seguida tem-se a quantidade de objetos e tamanho de cada página (em bytes) como inputs, ou recursos, utilizados; por fim, tem-se o PLT representando o output, ou resultado.

Tabela 3.3: Recursos (inputs) e resultados (outputs) utilizados na análise envoltória dos dados

Páginas	Qtd. Objetos (<i>Input</i>)	Tamanho Pág. (<i>Input</i>)	PLT (<i>Output</i>)
google.com	16	640444	8693
facebook.com	36	487700	5806
youtube.com	131	1589924	6618
baidu.com	18	576698	3240
amazon.com	199	4639244	8586
wikipedia.org	20	53272	3177
qq.com	220	2314736	18427
google.co.in	15	641922	6153
twitter.com	35	129214	6075
live.com	16	132472	5067
taobao.com	126	2187887	15157
msn.com	92	758056	2871
yahoo.co.jp	57	512902	8779
sina.com.cn	496	19069095	16606
linkedin.com	18	325990	8404
google.co.jp	16	649383	6662
weibo.com	110	940945	12167
bing.com	56	941086	1717
yandex.ru	63	585770	7463
vk.com	39	448042	8969
hao123.com	148	1869267	8890
instagram.com	23	459604	10017
ebay.com	174	2624886	9332
google.de	17	678982	5484
amazon.co.jp	258	5060789	17499
mail.ru	132	789647	9531
google.co.uk	17	669256	5625
tmall.com	36	571077	16632
pinterest.com	31	1368820	11394
360.cn	165	1504194	9750
google.ru	19	642695	6513

Continua na próxima página

Tabela 3.3 – *Continuação da página anterior*

Páginas	No de Objetos	Tamanho da pág.	PLT
reddit.com	53	474662	8809
google.com.br	15	647513	6593
netflix.com	13	702609	15073
google.fr	17	670175	5704
sohu.com	385	1573258	12281
microsoft.com	84	2040864	10262
wordpress.com	20	263552	7994
google.it	18	679504	6115
blogspot.com	14	78667	8395
gmw.cn	141	1669451	14422
google.es	17	679515	6971
onlickads.net	12	152909	5255
tumblr.com	92	14845093	25104
ok.ru	41	578970	9322
imgur.com	102	1257262	6656
aliexpress.com	68	1423637	11290
xvideos.com	54	1279964	26609
apple.com	53	2069649	23528
stackoverflow.com	26	288626	6116
imdb.com	206	2144843	15093
google.com.mx	15	647486	5867
fc2.com	59	579111	11505
ask.com	23	276948	5079
amazon.de	302	6225062	12856
google.com.hk	14	525033	6571
alibaba.com	47	1239330	9128
google.ca	15	641062	6706
rakuten.co.jp	808	10521295	10760
tianya.cn	40	306612	8625
pornhub.com	136	2760955	7618
google.co.id	15	638836	6197
craigslist.org	10	162069	4454
github.com	14	1638025	8272
xinhuanet.com	128	2577879	7162
nicovideo.jp	112	1621744	7749
amazon.co.uk	254	5669366	14093

Continua na próxima página

Tabela 3.3 – *Continuação da página anterior*

Páginas	No de Objetos	Tamanho da pág.	PLT
soso.com	20	141242	3709
pixnet.net	182	3993262	17982
amazon.in	307	4408514	11568
blogger.com	14	78710	7894
kat.cr	184	387564	10922
outbrain.com	91	1507173	9354
go.com	82	1332187	13888
cnn.com	147	2861726	15879
google.pl	17	672061	6031
naver.com	101	1537049	6733
google.com.au	15	645025	6880
dropbox.com	208	4239393	21627
xhamster.com	96	869881	7843
cntv.cn	130	1021743	2627
flipkart.com	182	2150533	9504

4

Resultados

Neste capítulo, serão apresentados os resultados obtidos dos experimentos realizados seguindo as diretrizes descritas no capítulo 3. O objetivo é comparar o tempo de carregamento das páginas Web usando os dois protocolos, QUIC e TCP; após isso é feita uma análise da performance das páginas.

4.1 Comparação dos Protocolos

Nos Gráficos 4.1, 4.2, 4.3 e 4.4 é mostrado uma primeira visão de como o TCP se comportou melhor do que o QUIC ao considerar os eventos *onLoad* e *onContentLoaded*. Cada um dos quatro gráficos mostram 12 linhas (4 taxas de perda de pacote * 3 valores de RTT - ver Tabela 3.2) apresentando a diferença entre o tempo médio de carregamento da página para QUIC e TCP. A Equação 4.1 detalha como os valores plotados foram computados, onde Avg_Q e Avg_T são os valores médios para os eventos *onContentLoaded* e *onLoad* para os protocolos QUIC e TCP respectivamente, e Q_{t_n} e T_{t_n} são valores coletados para QUIC e TCP para cada um dos 30 experimentos. Em resumo, o eixo X nos Gráficos 4.1, 4.2, 4.3 e 4.4 representam as páginas da base Alexa, enquanto o eixo Y é a diferença no tempo de carregamento da página em milissegundos (QUIC-TCP).

$$Diff_{(quic-tcp)} = Avg_Q - Avg_T = \frac{\sum_{n=0}^{30} (Q_{t_n} - T_{t_n})}{30} \quad (4.1)$$

Nos Gráficos 4.1, 4.2, 4.3 e 4.4, ao apresentar o resultado como a diferença entre os PLTs do QUIC e TCP, é possível observar se as linhas plotadas estão abaixo ou acima de 0, afim de verificar qual protocolo teve melhor desempenho. Ao mostrar todas as doze linhas de uma só vez, é fornecido uma melhor visão geral da diferença de desempenho entre os protocolos, por outro lado, perde-se alguns detalhes refinados das páginas. Além disso, foi decidido ordenar os resultados coletados do menor valor (onde o QUIC teve melhor desempenho) até o maior valor (ou seja, quando o TCP teve um desempenho melhor). Enquanto que com esta representação perdem-se informações sobre as páginas específicas, é mais fácil visualizar o desempenho dos protocolos de forma geral. Além disso, como já mencionado, este trabalho não está interessado

no desempenho das páginas em particular, mas na performance dos protocolos.

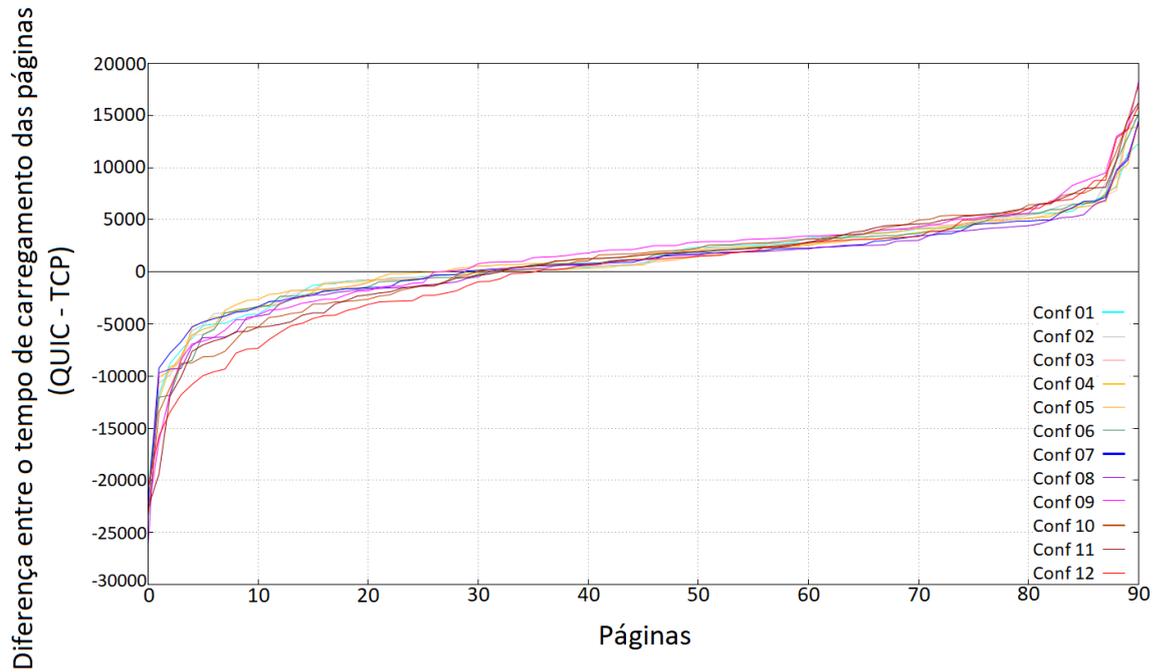


Figura 4.1: Page Load Time (QUIC - TCP) para o evento de *onLoad* com cache desabilitada.

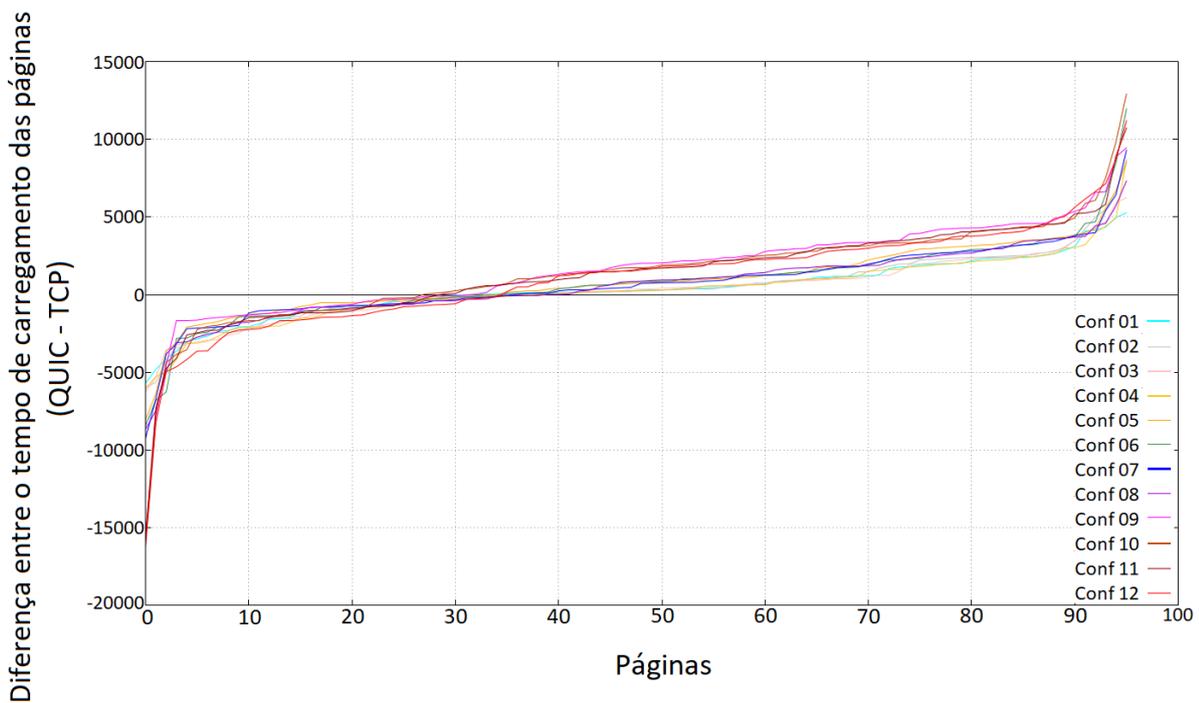


Figura 4.2: Page Load Time (QUIC - TCP) para o evento de *onContentLoaded* com cache desabilitada.

As Figuras 4.1 e 4.3 apresentam os resultados para o evento *onLoad* sem e com cache, respectivamente. As Figuras 4.2 e 4.4 mostram os resultados do evento *onContentLoaded* sem e com cache, respectivamente. Como se pode ver nos quatro Gráficos, 4.1, 4.2, 4.3 e 4.4,

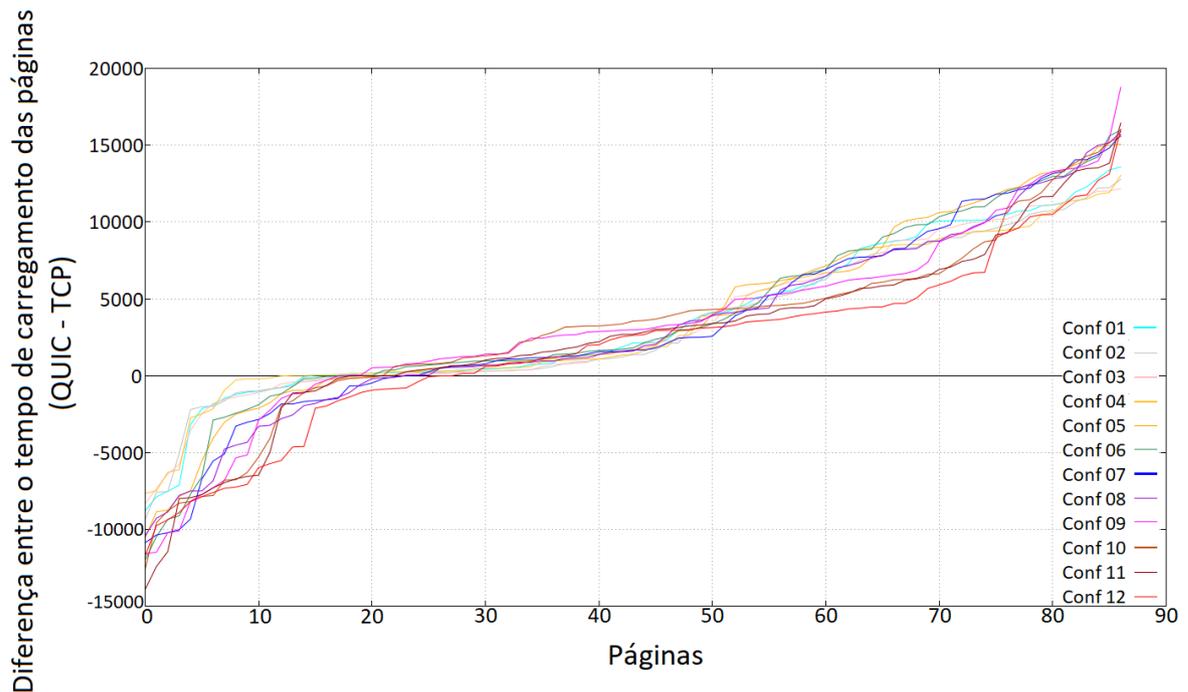


Figura 4.3: Page Load Time (QUIC - TCP) para o evento de *onLoad* com cache habilitada.

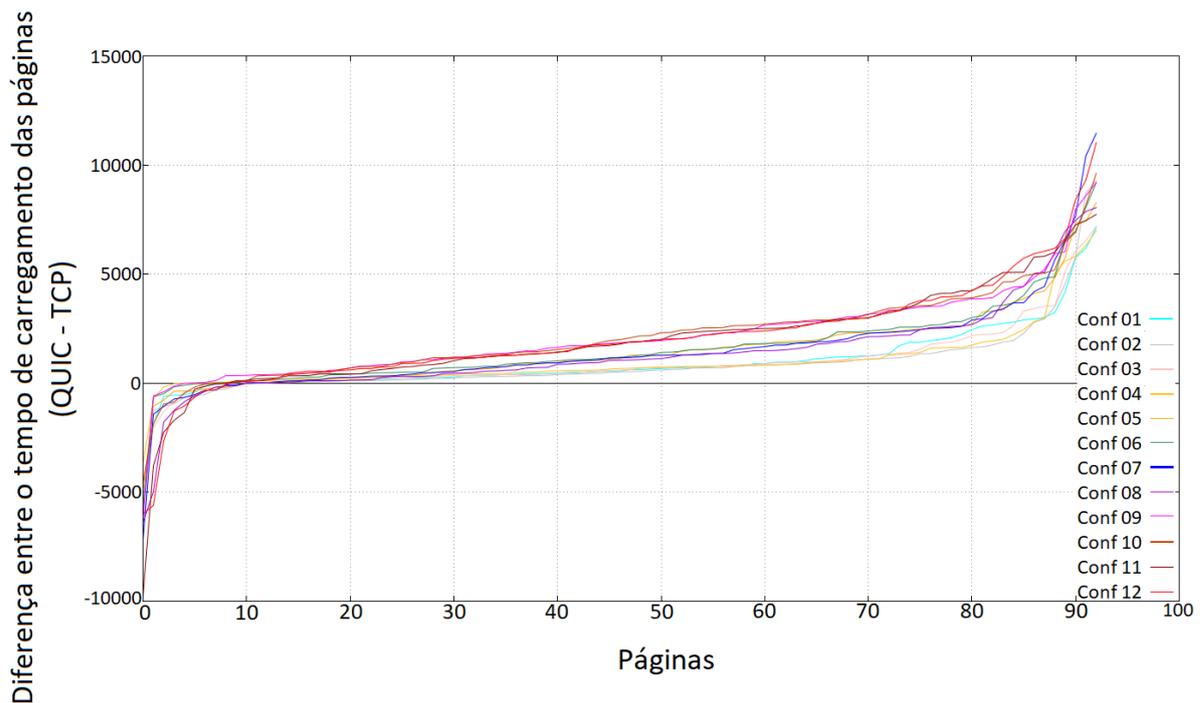


Figura 4.4: Page Load Time (QUIC - TCP) para os eventos de *onContentLoaded* com cache habilitada.

independentemente das configurações, o TCP apresentou melhor desempenho em pelo menos 60% das páginas avaliadas de qualquer configuração ou métrica.

Uma informação mais detalhada, incluindo a porcentagem de páginas onde o tempo de carregamento foi melhor para o QUIC do que o TCP, é mostrada na Tabela 4.1. O número

de páginas em que o QUIC superou o TCP variou de 5,37% com cache na 9ª configuração até 40,62% sem cache na 4ª e 8ª configurações. Considerando os três parâmetros mostrados na tabela (configuração, cache e evento) pode-se ver altas variações nos resultados coletados. Pode-se concluir que tanto a escolha da cache, quanto o cenário (configuração) têm um grande impacto nos resultados coletados. Por exemplo, ao fixar *onLoad*, habilitar cache e variar as configurações (segunda coluna da Tabela 4.1), os resultados diferem em mais de 16%. Da mesma forma, ao fixar *onContentLoad*, configuração 01, e variar cache, pode-se ver que o protocolo QUIC passou de menos de 10% das páginas (cache habilitado), que carregou mais rápido usando-o, para mais de 30% (cache desativado).

Tabela 4.1: Porcentagens de páginas para as quais o QUIC apresentou desempenho melhor que o TCP sob a configuração de cache ativado (CA) e cache desativado (CD)

Configuração	onLoad		onContentLoad	
	CA	CD	CA	CD
01	17.24%	35.16%	9.67%	34.37%
02	22.98%	34.06%	12.90%	37.50%
03	24.13%	34.06%	8.60%	39.58%
04	16.09%	35.16%	4.30%	40.62%
05	25.28%	29.67%	11.82%	36.45%
06	21.83%	35.16%	7.52%	36.45%
07	27.58%	31.86%	10.75%	37.50%
08	26.43%	34.06%	10.75%	40.62%
09	20.68%	28.57%	5.37%	33.33%
10	24.13%	32.96%	8.60%	28.12%
11	25.28%	35.16%	9.67%	30.20%
12	32.18%	39.56%	9.67%	36.45%

Finalmente, para ter uma ideia da dispersão dos dados, decidiu-se incluir um boxplot de uma parte dos experimentos A Figura 4.5 mostra esse boxplot para os resultados ao considerar a métrica *onLoad* sem cache. O eixo Y representa a métrica do tempo de carregamento da página de ambos os protocolos e o eixo X reflete as diferentes configurações nas quais os experimentos foram executados.

4.2 Análise da Performance das Páginas

A Tabela 4.2 mostra o resultado da análise de eficiência das páginas. A primeira coluna são as páginas, ou DMUs, em seguida tem-se a porcentagem de eficiência que essas páginas atingiram. A coluna "Ótima" simboliza a porcentagem de melhoria que cada página precisa incrementar para atingir 100% de eficiência. "Objetos (+)" é a quantidade de objetos que a página precisa acrescentar à quantidade de objetos já existentes pra atingir a fronteira de eficiência. Por exemplo, a página "google.com" precisa acrescentar 162,4458 objetos aos objetos já existentes para alcançar a máxima eficiência. A coluna "Tamanho (+)" simboliza o tamanho da página em

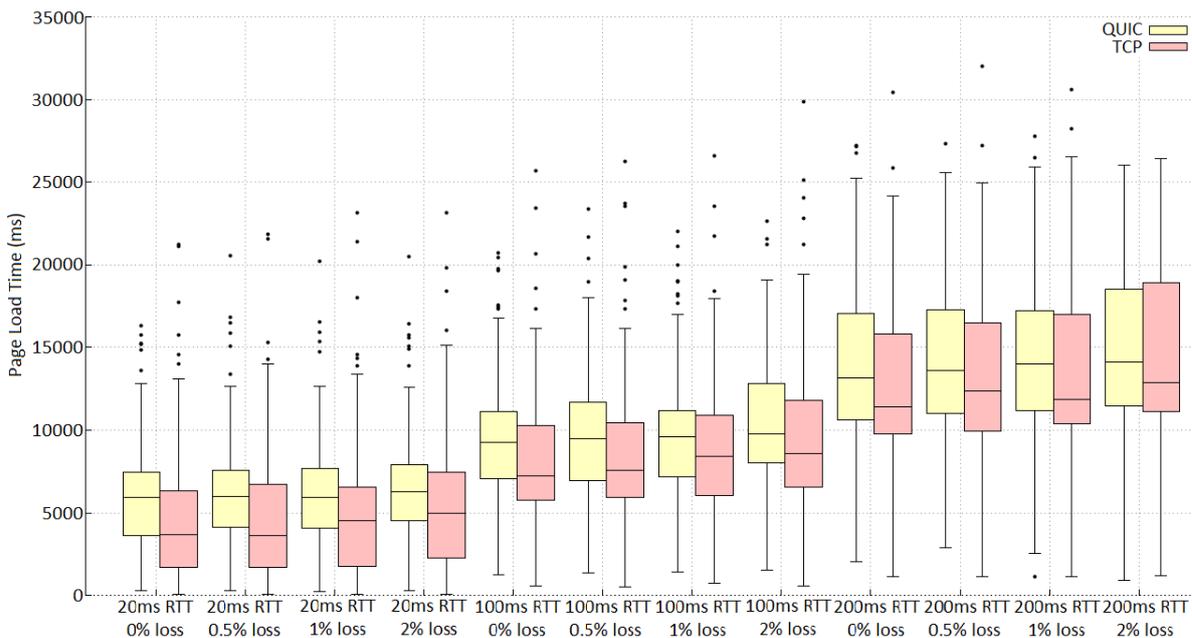


Figura 4.5: PLT do QUIC e TCP para o evento de *onLoad* nos experimentos com cache desabilitada

bytes que deve aumentar para se tornar eficiente, e por fim "PLT (-)" simboliza a quantidade em milissegundo que a página deve reduzir para alcançar eficiência máxima. Essas mudanças nos valores de inputs e output combinados com outros fatores proporcionam a eficiência ótima em cada página. No capítulo 5 serão analisados esses resultados.

Tabela 4.2: Tabela de eficiência das páginas

Páginas	Eficiência	Ótima	Objetos (+)	Tamanho (+)	PLT (-)
google.com	13,61%	86,38%	162	0	105
facebook.com	21,42%	78,57%	0	547324	653
youtube.com	54,75%	45,24%	2	0	555
baidu.com	56,44%	43,55%	98	0	612
amazon.com	100%	0%	0	0	0
wikipedia.org	15,38%	84,61%	0	675475	549
qq.com	19,57%	80,42%	0	1679670	694
google.co.in	22,10%	77,89%	174	0	90
twitter.com	13,46%	86,53%	0	1083611	819
live.com	8,60%	91,39%	0	422842	721
taobao.com	17,52%	82,47%	275	0	52
msn.com	74,19%	25,80%	5	0	243
yahoo.co.jp	15,32%	84,67%	0	578297	89
sina.com.cn	100%	0%	0	0	0
linkedin.com	8,47%	91,52%	85	0	623
google.co.jp	22,36%	77,63%	170	0	599

Continua na próxima página

Tabela 4.2 – *Continuação da página anterior*

Páginas	Eficiência	Ótima	Objetos (+)	Tamanho (+)	PLT (-)
weibo.com	13,61%	86,38%	0	3609626	1407
bing.com	100%	0%	0	0	0
yandex.ru	19,93%	80,06%	0	46424	490
vk.com	11,41%	88,58%	30	0	278
hao123.com	47,61%	52,38%	61	0	199
instagram.com	9,60%	90,39%	114	0	519
ebay.com	62,14%	37,85%	0	481475	744
google.de	24,04%	75,95%	97	0	331
amazon.co.jp	37,69%	62,30%	365	0	676
mail.ru	33,84%	66,15%	0	732181	1648
google.co.uk	23,70%	76,29%	96	0	472
tmall.com	4,27%	95,72%	134	0	719
pinterest.com	13,00%	86,99%	569	0	634
360.cn	37,54%	62,45%	6	0	149
google.ru	22,13%	77,86%	156	0	450
reddit.com	14,24%	85,75%	0	594068	118
google.com.br	22,29%	77,70%	174	0	530
netflix.com	5,66%	94,33%	690	0	878
google.fr	23,73%	76,26%	96	0	551
sohu.com	47,64%	52,35%	0	7219496	1521
microsoft.com	42,64%	57,35%	157	0	764
wordpress.com	6,85%	93,14%	6	0	214
google.it	23,39%	76,60%	165	0	52
blogspot.com	3,58%	96,41%	0	873791	511
gmw.cn	15,03%	84,96%	0	437044	1034
google.es	18,05%	81,94%	129	0	101
onclickads.net	7,14%	92,85%	0	682532	102
tumblr.com	64,75%	35,24%	1585	0	148
ok.ru	14,64%	85,35%	0	751488	734
imgur.com	43,29%	56,70%	6	0	593
aliexpress.com	13,53%	86,46%	305	0	529
xvideos.com	5,56%	94,43%	831	0	743
apple.com	9,06%	90,93%	135	0	52
stackoverflow.com	10,74%	89,25%	0	217473	53
imdb.com	21,96%	78,03%	0	1776714	1705
google.com.mx	22,93%	77,06%	102	0	714

Continua na próxima página

Tabela 4.2 – Continuação da página anterior

Páginas	Eficiência	Ótima	Objetos (+)	Tamanho (+)	PLT (-)
fc2.com	7,30%	92,69%	0	2590419	745
ask.com	14,10%	85,89%	22	0	734
amazon.de	54,30%	45,69%	307	0	378
google.com.hk	18,08%	81,91%	164	0	508
alibaba.com	26,33%	73,66%	101	0	540
google.ca	22,07%	77,92%	174	0	643
rakuten.co.jp	100%	0%	0	0	0
tianya.cn	10,25%	89,74%	0	75762	742
pornhub.com	73,34%	26,65%	38	0	748
google.co.id	21,99%	78,00%	173	0	134
craigslist.org	8,25%	91,74%	64	0	109
github.com	42,60%	57,39%	265	0	491
xinhuanet.com	68,48%	31,51%	37	0	291
nicovideo.jp	50%	50%	0	520856	879
amazon.co.uk	49,11%	50,88%	420	0	705
soso.com	15,38%	84,61%	0	103670	1081
pixnet.net	27,95%	72,04%	380	0	352
amazon.in	41,90%	58,09%	75	0	808
blogger.com	3,58%	96,41%	0	872593	11
kat.cr	22,77%	77,22%	0	8819384	161
outbrain.com	32,50%	67,5	0	67974	766
go.com	11,54%	88,45%	227	0	500
cnn.com	22,92%	77,07%	352	0	773
google.pl	23,80%	76,19%	96	0	878
naver.com	52,93%	47,06%	51	0	670
google.com.au	17,13%	82,86%	136	0	10
dropbox.com	20,14%	79,85%	583	0	107
xhamster.com	30,37%	69,62%	0	121213	870
cntv.cn	100%	0%	0	0	0
flipkart.com	51,41%	48,58%	0	603182	6

Na Figura 4.6 tem-se a fronteira de eficiência para os dois inputs utilizados, quantidade de objetos e tamanhos de cada página. O eixo X representa a quantidade de objetos, o eixo Y representa o tamanho das páginas. Os pontos que se encontram sobre a linha de fronteira de eficiência representam as páginas que atingiram 100% de eficiência, já os demais pontos precisam aumentar a quantidade de objetos, X, e/ou aumentar o tamanho da página, Y, para alcançar a fronteira de eficiência.

Na Figura 4.7 têm-se os recursos e resultado utilizados, input e output. O eixo X representa o output, no caso o PLT. Já o eixo Y simboliza a combinação dos dois inputs, quantidade de objetos e tamanho da página. Para uma página atingir a fronteira de eficiência ela precisa aumentar os recursos, inputs, ou então diminuir o resultado, output; ou ainda fazer uma combinação dos dois.

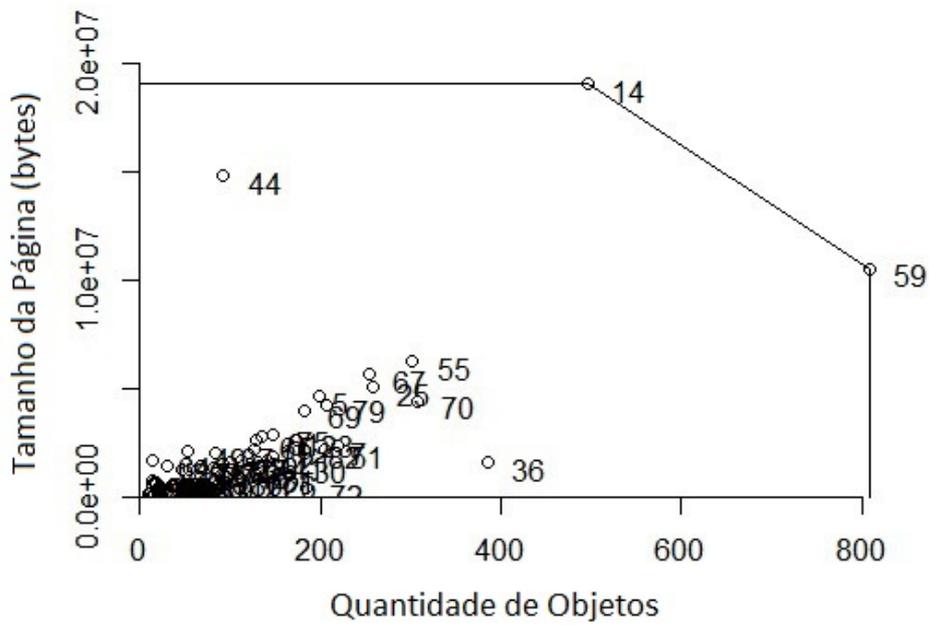


Figura 4.6: Gráfico da fronteira de eficiência para os inputs

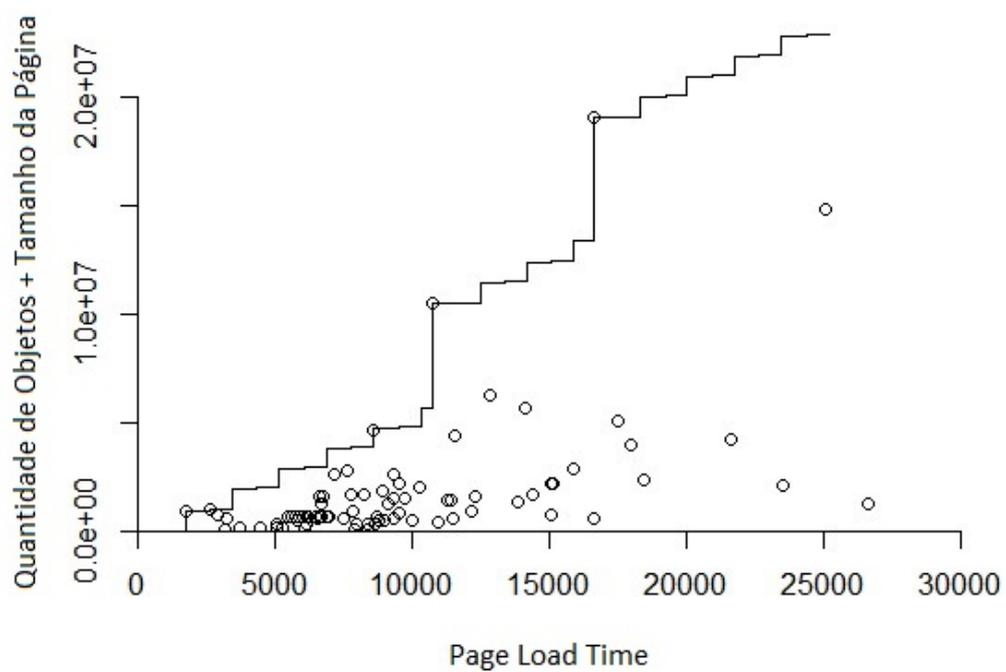


Figura 4.7: Gráfico da fronteira de eficiência para a combinação de inputs (eixo X) e output (eixo Y)

5

Discussão

Neste capítulo, é fornecida uma discussão adicional sobre os experimentos, bem como *insights* sobre o que aconteceu para explicar os resultados esperados e inesperados.

5.1 Comparação dos Protocolos

Em primeiro lugar, para ambos os grupos de experimentos em execução com ou sem cache, foi observado que a variação do RTT tem um impacto significativo na métrica de PLT. Como mostrado na Figura 4.5, ao fixar a taxa de perda de pacotes e variar o RTT em 20 ms, 100 ms e 200 ms, pode-se notar o aumento no tempo de carregamento da página.

No entanto, quando se fixa o RTT e varia-se a perda de pacotes para 0%, 0,5%, 1% e 2%, isso dificilmente influencia os resultados. Esses resultados estão de acordo com [COOK et al. \(2017\)](#), que também mostra o RTT como tendo um grande impacto no desempenho do QUIC quando comparado ao TCP, enquanto a perda de pacotes não é tão significativa.

A primeira execução dos experimentos foi realizada com o cache desabilitado, e se for considerado apenas este grupo de experimentos, pode-se ver que o TCP teve um desempenho melhor na maioria dos casos. Como a Tabela 4.1 apresenta, independentemente da configuração, os resultados mostram que o QUIC proporciona um melhor tempo de carregamento da página em menos de 40% das páginas Web consideradas.

Embora isso seja inesperado, uma razão concebível poderia ser que o QUIC não é o protocolo *mainstream* da Internet, portanto, para um cliente usar o QUIC na primeira conexão, ele inicialmente precisa executar um mecanismo de descoberta. Ou seja, quando a primeira conexão é realizada, o cliente usa o TCP para enviar solicitações para um servidor; quando um servidor suporta o QUIC, ele informa ao cliente e a conexão restante é mantida com o QUIC.

A fim de verificar essa suposição, foi executado um conjunto de experimentos com o sistema de cache ativado. Mais especificamente, ao usar cache, o navegador informaria previamente se a página pode ser atendida pelo QUIC e, portanto, evitaria o mecanismo de descoberta inicial. Pode-se ver os resultados nos Gráficos 4.1, 4.2, 4.3 e 4.4 e também na Tabela 4.1.

Surpreendentemente, foi observado que ativar a cache e usar o TCP produz uma melhoria de desempenho muito maior do que ativar a cache e usar o QUIC. Isso é refletido pelo fato de o número de páginas que foram carregadas mais rapidamente usando o QUIC foi ainda mais baixo. Podem ser mencionados possíveis razões pelas quais os resultados foram talvez contra intuitivos para o QUIC ao ativar o sistema de armazenamento em cache. Por exemplo, o mecanismo pode não ser tão eficiente para lidar com a cache e o protocolo QUIC quanto com o protocolo padrão, o TCP.

Como não foi possível identificar nenhum padrão em particular que pudesse categorizar diretamente as páginas que se beneficiavam mais ou menos com o QUIC ou o TCP, foi procurado por algum padrão nas fases do PLT (*Wait, DNS, Send, Connect, Blocked, Receive*), que são obtidos a partir do arquivo HAR [ODVARKO; JAIN; DAVIES \(2012\)](#). Foi descoberto que a fase *connect* prevaleceu como a fase mais demorada nas páginas carregadas com o protocolo TCP, o que significa que o TCP perde tempo ao criar conexões com o servidor. Assim, conforme detalhado no capítulo 2, a multiplexação do QUIC torna-se uma boa solução para evitar esse problema. No entanto, nas páginas que usam o QUIC, foi notado que a fase *receive* foi a mais demorada na maioria das páginas testadas, ou seja, o QUIC perde tempo ao ler a resposta do servidor (ou cache).

Finalmente, uma das razões que também podem explicar porquê os experimentos mostram resultados globais melhores para o TCP é o fato de que a lista de páginas não foi selecionada de tal forma que teoricamente funcionaria melhor com o QUIC. Em vez disso, foi escolhido o que foi considerado ser uma boa representação de páginas da Internet, considerando as páginas mais acessadas do mundo, que não são necessariamente otimizadas para o QUIC ainda.

5.2 Análise de Performance das Páginas

Através da técnica de análise envoltória dos dados foi obtido o score de eficiência das páginas, Tabela 4.2. A partir dos resultados é possível visualizar as páginas que alcançaram a fronteira de eficiência:

- amazon.com;
- sina.com.cn;
- bing.com;
- rakuten.co.jp;
- cntv.cn.

Essas páginas com máxima eficiência se encontram sobre a linha da fronteira de eficiência na Figura 4.6 e 4.7. Seus valores para melhoria na Tabela 4.2 estão zerados, pois sua

performance já se encontra no estado ideal. Para as demais páginas são mostrados quais recursos ou resultado elas precisam modificar para alcançar a fronteira de eficiência. Por exemplo, a página blogspot.com obteve 3,58% de eficiência portanto para incrementar 96,41% de eficiência a sua performance ela precisaria acrescentar 873791 bytes ao seu tamanho e reduzir 511 milissegundos do seu PLT.

Essa é uma análise confiável pois as páginas com seus respectivos inputs e output são comparadas entre si levando em conta a natureza de cada uma. É como se essas páginas fossem normalizadas para então ser comparadas. Com isso por mais diferentes que essas páginas sejam, tem-se análise das páginas sob uma mesma perspectiva.

6

Conclusão

Este trabalho realizou avaliação da performance da Internet a partir de duas perspectivas diferentes. Sendo a primeira a análise dos principais protocolos de transporte; e a segunda a análise de eficiência das páginas Web.

Há algumas discussões na comunidade sobre qual protocolo tem o menor tempo de carregamento de página. O tempo de carregamento de uma página Web afeta diretamente a experiência do usuário e refletirá sobre ações futuras dos usuários na Internet. Visando estas discussões, este estudo comparou o desempenho do QUIC com o HTTP 2 e o TCP com o HTTP 1.1. Foram incluídas neste trabalho as páginas da Internet mais acessadas em diferentes condições de rede para avaliar o desempenho dos protocolos. Primeiro, analisando a lista das páginas mais acessadas do mundo, foi descoberto que o número de distribuição de objetos permanece semelhante nos diferentes sites da lista do Alexa. Além disso, ao medir o tempo de carregamento das páginas com os eventos *onLoad* e *onContentLoad*, nos cenários descritos, foi notado que o QUIC apresenta um desempenho inferior ao TCP, cada um com os respectivos protocolos de aplicação. Ao inspecionar os resultados, foi possível observar um comportamento interessante em relação aos fatores, também observado em outros estudos [COOK et al. \(2017\)](#). Como discutido no capítulo anterior, a influência do RTT nos experimentos foi significativa, enquanto a influência da taxa de perda de pacotes foi inexpressiva. Embora o QUIC já esteja sendo usado em parte da Internet, ainda existem alguns obstáculos e pontos de melhoria até se tornar o protocolo principal da Internet. Uma vez que o ganho no desempenho não é tão claro. A estrutura da Internet, por ex. distribuição de servidores e formatos de páginas Web, está otimizada para os protocolos atuais; QUIC ainda está se adaptando, seu desempenho pode ser inferior ao esperado. Portanto, a implantação completa do QUIC ainda deve ser discutida antes de sua adoção.

Outra importante análise feita foi a medição de quais páginas apresentam boa performance e quais não apresentam. A partir disso foi possível encontrar pontos de melhoria nas páginas que não apresentaram um bom score de eficiência. Com os resultados de quais páginas possuem máxima eficiência é possível entender melhor o mecanismo de funcionamento da Internet e estudar fatores que a influenciam e como se pode melhorá-la. Como trabalhos futuros,

pretende-se continuar a avaliação de desempenho do protocolo QUIC. Para mais estudos de caso, planeja-se definir diferentes níveis dos fatores aqui estudados. Também planeja-se adicionar novos recursos de redes que influenciariam a métrica do tempo de carregamento da página. E também realizar uma nova análise da performance de eficiência das páginas Web.

Referências

- BELSHE, M.; PEON, R. SPDY protocol. **URL**
<http://www.chromium.org/spdy/spdyprotocol/spdy-protocol-draft3-1>, [S.l.], 2012.
- BRISCOE, B. et al. Reducing Internet latency: a survey of techniques and their merits. **IEEE Communications Surveys & Tutorials**, [S.l.], v.18, n.3, p.2149–2196, 2014.
- BUTKIEWICZ, M.; MADHYASTHA, H. V.; SEKAR, V. Understanding website complexity: measurements, metrics, and implications. In: ACM SIGCOMM CONFERENCE ON INTERNET MEASUREMENT CONFERENCE, 2011. **Proceedings...** [S.l.: s.n.], 2011.
- CARLUCCI, G.; DE CICCIO, L.; MASCOLO, S. HTTP over UDP: an experimental investigation of quic. In: ANNUAL ACM SYMPOSIUM ON APPLIED COMPUTING, 30. **Proceedings...** [S.l.: s.n.], 2015. p.609–614.
- CHARNES, A.; COOPER, W. W.; RHODES, E. Measuring the efficiency of decision making units. **European journal of operational research**, [S.l.], v.2, n.6, p.429–444, 1978.
- COELLI, T. J. et al. **An introduction to efficiency and productivity analysis**. [S.l.]: Springer Science & Business Media, 2005.
- COOK, S. et al. QUIC: better for what and for whom? In: IEEE INTERNATIONAL CONFERENCE ON COMMUNICATIONS (ICC), 2017. **Anais...** [S.l.: s.n.], 2017.
- DIERKS, T. The transport layer security (TLS) protocol version 1.2. **RFC**, [S.l.], 2008.
- ELKHATIB, Y.; TYSON, G.; WELZL, M. Can SPDY really make the web faster? In: NETWORKING CONFERENCE, 2014 IFIP. **Anais...** [S.l.: s.n.], 2014. p.1–9.
- ERMAN, J. et al. Towards a SPDY'ier mobile web? **IEEE/ACM Transactions on Networking**, [S.l.], v.23, n.6, p.2010–2023, 2015.
- Alexa site. **Alexa site**. [Online; accessed Jan-2017],
<http://www.alexa.com/topsites/>.
- Big Commerce. **Big Commerce**. [Online; accessed 11-Dec-2017],
<https://www.bigcommerce.com/blog/4-tips-improve-ecommerce-bounce-rate-right-now-sellmore-video/>.
- Caddy Server. **Caddy Server**. [Online; accessed Jan-2017],
<https://caddyserver.com/>.
- Cisco System Inc. **Cisco Visual Networking Index Predicts Near-Tripling of IP Traffic by 2020**. 2016.
- Github. **Pointers for implementing the client**. [Online; accessed Jan-2017],
<https://github.com/lucas-clemente/quic-go/issues/310#issuecomment-253555550>.

- Google Inc. **Using site speed in web search ranking**. [Online; accessed Jan-2017], <https://webmasters.googleblog.com/2010/04/using-site-speed-in-web-search-ranking.html>.
- Google Inc. **Google BigQuery**. [Online; accessed Jan-2017], <https://cloud.google.com/bigquery/>.
- Google Inc. **Chromium**. [Online; accessed 11-Dec-2017], <https://www.chromium.org/quic>.
- HTTP Archive. [Online; accessed 11-Dec-2017], <http://httparchive.org/>.
- PR Newswire. **'Customers Are Won or Lost in One Second,' Finds New Aberdeen Report**. 2008.
- VM Ware. **VM Ware**. [Online; accessed Jan-2017], <http://www.vmware.com/products/player/playerpro-evaluation.html>.
- MEDINA, E.; SHEMLA, D.; SOLT, Y. **Head of line blocking**. US Patent 6,829,245.
- NETRAVALI, R. et al. Mahimahi: accurate record-and-replay for http. In: USENIX ANNUAL TECHNICAL CONFERENCE. **Anais...** [S.l.: s.n.], 2015.
- ODVARKO, J.; JAIN, A.; DAVIES, A. HTTP Archive (HAR) format. **W3C draft**, [S.l.], 2012.
- POSTEL, J. User datagram protocol. **RFC**, [S.l.], 1980.
- REGUNDWAR, N. P.; SHUKLA, D. A.; LOKHANDE, P. A Studypaper on SPDY protocol: lets make the web faster [j]. **International Journal of Scientific and Engineering Research**, [S.l.], v.4, p.222–225, 2013.
- ROSKIND, J. **QUIC, Quick UDP Internet Connections**. [S.l.]: Google Draft, 2013.
- SISSEL, J. **xdotool–fake keyboard/mouse input, window management, and more**. [Online; accessed Jan-2017], <https://www.semicomplete.com/projects/xdotool/>.
- USHA RANI J MADHU M NAYAK, A. G. **Visualization of Three Way Handshake Mechanism of TCP/IP**. [S.l.]: International Journal for Research in Applied Science Engineering Technology - IJRASET, 2017.
- WANG, X. S. et al. How speedy is SPDY? In: NSDI. **Anais...** [S.l.: s.n.], 2014. p.387–399.
- WIDMER, J.; DENDA, R.; MAUVE, M. A survey on TCP-friendly congestion control. **IEEE Network**, [S.l.], v.15, n.3, p.28–37, May 2001.

Apêndice



TOP 100 Alexa Site

Tabela A.1: Lista dos 100 sites mais acessados do mundo de acordo com o site Alexa.

No.	Web Sites	No.	Web Sites	No.	Web Sites	No.	Web Sites
1	google.com	26	amazon.co.jp	51	xvideos.com	76	amazon.in
2	facebook.com	27	mail.ru	52	apple.com	77	blogger.com
3	youtube.com	28	google.co.uk	53	stackoverflow.com	78	kat.cr
4	baidu.com	29	tmall.com	54	imdb.com	79	outbrain.com
5	yahoo.com	30	pinterest.com	55	google.com.mx	80	bongacams.com
6	amazon.com	31	360.cn	56	fc2.com	81	go.com
7	wikipedia.org	32	google.ru	57	ask.com	82	googleusercontent.com
8	qq.com	33	reddit.com	58	amazon.de	83	cnn.com
9	google.co.in	34	t.co	59	google.com.hk	84	jd.com
10	twitter.com	35	google.com.br	60	alibaba.com	85	google.pl
11	live.com	36	netflix.com	61	google.com.tr	86	naver.com
12	taobao.com	37	google.fr	62	google.ca	87	google.com.au
13	msn.com	38	sohu.com	63	office.com	88	dropbox.com
14	yahoo.co.jp	39	paypal.com	64	rakuten.co.jp	89	chinadaily.com.cn
15	sina.com.cn	40	microsoft.com	65	tianya.cn	90	360.com
16	linkedin.com	41	wordpress.com	66	diplay.com	91	coccoc.com
17	google.co.jp	42	google.it	67	pornhub.com	92	xhamster.com
18	weibo.com	43	blogspot.com	68	google.co.id	93	adnetworkperformance.com
19	bing.com	44	gmw.cn	69	craigslist.org	94	cntv.cn
20	yandex.ru	45	google.es	70	github.com	95	haosou.com
21	vk.com	46	onclickads.net	71	xinhuane.com	96	adobe.com
22	hao123.com	47	tumblr.com	72	nicovideo.jp	97	china.com
23	instagram.com	48	ok.ru	73	amazon.co.uk	98	flipkart.com
24	ebay.com	49	imgur.com	74	soso.com	99	microsoftonline.com
25	google.de	50	aliexpress.com	75	pixnet.net	100	whatsapp.com