清 华 大 学

综 合 论 文 训 练

题目：新型传输协议 QUIC 的评估与应用

系　　别：计算机科学与技术系

专　　业：计算机科学与技术

姓　　名：王豪

指导教师：徐明伟教授

2017 年 6 月 3 日

关于学位论文使用授权的说明

本人完全了解清华大学有关保留、使用学位论文的规定，即：学校有权保留学位论文的复印件，允许该论文被查阅和借阅；学校可以公布该论文的全部或部分内容，可以采用影印、缩印或其他复制手段保存该论文。

(涉密的学位论文在解密后应遵守此规定)

签　名：_____ 导师签名：_____ 日　期：_____

中文摘要

背景介绍：从 90 年代以来，基于 TCP 和 HTTP 的网络传输协议被广泛使用在互联网中。然而，最近 20 年的网页技术的飞速发展使得这些协议并不能满足传输的要求，成为了一种传输瓶颈。于是 Google 公司先后开发和探索了两种新型的传输协议：SPDY（2009 年）和 QUIC（2013 年）。其中 SPDY 是基于 TCP 传输实现的，在某些场景会遇到较大时延；而 QUIC 是基于 UDP 实现的有连接的新型传输协议，能克服 TCP 的局限性，有很广泛的应用前景。我的工作：对 QUIC 协议进行了较全面地分析，包括了基本思想、代码结构等。对 QUIC 协议和 TCP 协议进行了仿真试验和基于实际网络状况下的实验，对结果性能做了分析。此外，还通过实际构建一个可以通过 QUIC 协议的 web 服务器来尝试对 QUIC 协议的应用，并将其与原有的基于 HTTP 协议的下载网页过程进行对比。

关键词：QUIC；TCP；网络协议；协议测试；web 服务器搭建

ABSTRACT

An abstract of a dissertation is a summary and extraction of research work and contributions. Included in an abstract should be description of research topic and research objective, brief introduction to methodology and research process, and summarization of conclusion and contributions of the research. An abstract should be characterized by independence and clarity and carry identical information with the dissertation. It should be such that the general idea and major contributions of the dissertation are conveyed without reading the dissertation.

An abstract should be concise and to the point. It is a misunderstanding to make an abstract an outline of the dissertation and words "the first chapter", "the second chapter" and the like should be avoided in the abstract.

Key words are terms used in a dissertation for indexing, reflecting core information of the dissertation. An abstract may contain a maximum of 5 key words, with semicolons used in between to separate one another.

Keywords: QUIC; TCP; internet protocol; protocol testing; web server building

目 录

主要符号对照表

QUIC	Quick UDP Internet Connection
TCP	
UDP	
SPDY	
HPC	高性能计算 (High Performance Computing)
cluster	集群
Itanium	安腾
SMP	对称多处理
API	应用程序编程接口
PI	聚酰亚胺
MPI	聚酰亚胺模型化合物，N-苯基邻苯酰亚胺
PBI	聚苯并咪唑
MPBI	聚苯并咪唑模型化合物，N-苯基苯并咪唑
PY	聚吡咙
PMDA-BDA	均苯四酸二酐与联苯四胺合成的聚吡咙薄膜
ΔG	活化自由能 (Activation Free Energy)
χ	传输系数 (Transmission Coefficient)
E	能量
m	质量
c	光速
P	概率
T	时间
v	速度
劝学	君子曰：学不可以已。青，取之于蓝，而青于蓝；冰，水为之，而寒于水。木直中绳。輮以为轮，其曲中规。虽有槁暴，不复挺者，輮使之然也。故木受绳则直，金就砺则利，君子博学而日参省乎己，则知明而行无过矣。吾尝终日而思矣，不如须臾之所学也；吾尝跂而望矣，不如登高之博见也。登高而招，臂非加长也，而见者远；顺风而呼，声非加疾也，而闻者彰。

假舆马者，非利足也，而致千里；假舟楫者，非能水也，而绝江河，君子生非异也，善假于物也。积土成山，风雨兴焉；积水成渊，蛟龙生焉；积善成德，而神明自得，圣心备焉。故不积跬步，无以至千里；不积小流，无以成江海。骐骥一跃，不能十步；驽马十驾，功在不舍。锲而舍之，朽木不折；锲而不舍，金石可镂。蚓无爪牙之利，筋骨之强，上食埃土，下饮黄泉，用心一也。蟹六跪而二螯，非蛇鳝之穴无可寄托者，用心躁也。——荀况

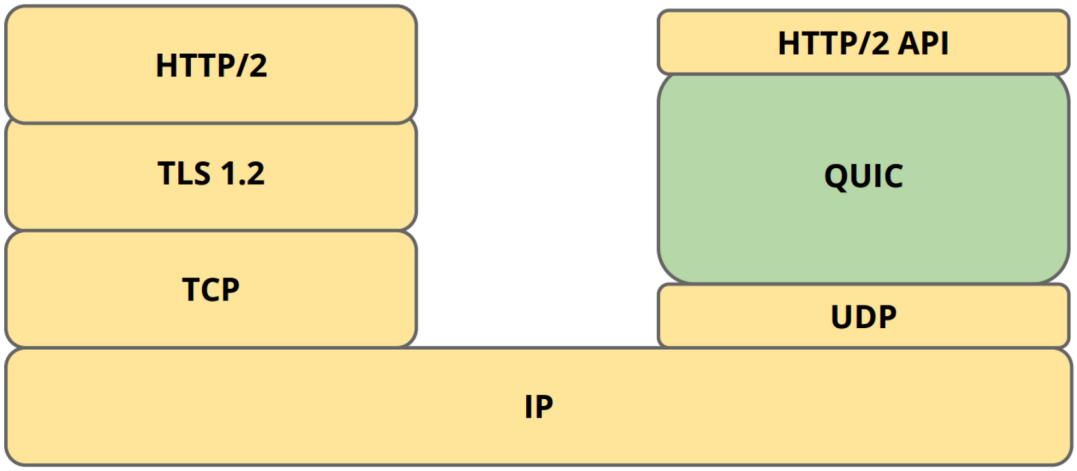
第 1 章　引言

1.1　研究背景

写相关 TCP、UDP、TLS、HTTP、SPDY 的介绍 QUIC 是一个由 Google 开发的新型传输协议，全称是 Quick UDP Internet Connections, 也就是快速 UDP 网络连接的意思。QUIC 解决了许多在现代 web 应用中遇到的传输层和应用层的问题，并且提供给 web 应用开发者提供了方便使用的接口。QUIC 非常接近于 TCP+TLS+HTTP2，但是是实现在 UDP 上的一个应用层协议，QUIC 在网络分层结构中的定位如下图所示。相比于 TCP+TLS+HTTP2，QUIC 主要的优点有：

- 低连接延迟
- 更好的拥塞控制算法选择
- 多路复用流来避免线头阻塞
- 前向纠错算法
- 连接迁移功能

具体来说，低连接延迟是指每次 QUIC 的连接建立（握手）只需要 1 到 0 个往返的时间，而相比较而言，TCP+TLS 则需要 1 到 3 个往返时间。在拥塞控制算法方面，QUIC 正在不断添加新的内容，现有的拥塞控制主要是基于 TCP 开发的，主要有 Cubic，此外 QUIC 相比于 TCP 的拥塞控制而言，还能提供更丰富的



拥塞控制信息，这得益于 QUIC 传输的过程中的 ACK 帧的内容包含了足够多的信息，包括收发包的延迟时间信息。QUIC 协议通过底层使用 UDP 协议避免了 TCP 连接过程中会遇到的线头阻塞问题，即 TCP 协议因为处理包有严格的顺序，所以一旦有包出错，后续的包会等待这个包处理完成，而 QUIC 则解决了这方面的问题。

1.2 相关工作介绍

几篇相关 quic 测试的文章综述

1.3 课题介绍及主要贡献

课题大背景是一个医疗大数据传输的项目，为了能够支持医疗大数据的有效传输，考虑到原有的 TCP 协议传输在可能在大数据传输方面的不足，课题组准备对新型传输的 QUIC 协议进行相关研究，而本篇文章的工作，就是对 QUIC 的性能评价以及测试服务器的搭建。我们的主要贡献有：

- 对 QUIC 的基本原理和源码的接口进行了分析
- 不同拓扑情况下进行了 QUIC 传输自定义数据的性能评估
- 对已有支持 QUIC 的 Web 服务器进行了传输性能评估
- 构建了可支持 QUIC 协议传输的 Web 服务器并进行了评估

1.4 文章结构

文章除去引言部分以外，在第二章先详细介绍了 QUIC 协议的基本思想和原理，其后分析了其服务端和客户端相关原语的接口代码的使用方式；第三章，根据提供的 QUIC 协议的结构搭建了简单的支持 QUIC 协议传输的应用程序，并基于 Mininet 进行仿真，对 QUIC 协议与 TCP 协议进行了不同拓扑和网络状况下的吞吐量性能评估；第四章，首先分析了访问已有的支持 QUIC 的服务器，对是否开启 QUIC 开关情况下的 Chrome 浏览器的网页下载速度进行了统计分析，更进一步，在自己构建的能够支持 QUIC 协议的 Web 服务器，对是否开启 QUIC 支持的服务器进行网页下载速度测试；在第五章，对整个项目进行总结并对未来工作进行了展望。

第 2 章　QUIC 协议的原理分析

本章将详细介绍一下 QUIC 协议的实现原理，包括 QUIC 协议的连接建立过程（包括握手方式及内容、连接的建立方式），对出现丢包情况下的特殊处理方式，出现拥塞时采取的拥塞控制算法，两种类型的流控算法以及其他有用的特性。

2.1　QUIC 原理简介

2.1.1　连接建立

QUIC 连接是由客户端发起的，流程的描述和流程图表示如下。

1. 客户端判断本地是否已有服务器的配置信息，如果有则跳转到第 5 步，否则继续。
2. 客户端向服务器发送空的客户端 hello 消息（CHLO），并请求服务器回传配置信息。
3. 服务端受到 CHLO 消息后，回复给客户端拒绝消息（REJ），并在其中附带自己的配置信息
4. 客户端收到服务端发送来的 REJ 消息之后，存储服务端的配置信息，并回到第 1 步。
5. 客户端向服务端发送完整的客户端 hello 消息，并开始正式握手，在这次的 hello 消息中将会包括加密连接的一些信息。
6. 服务端受到完整 CHLO 信息后，如果不同意连接就回复拒绝消息（REJ），则同第 3 步的操作；如果同意连接则连接成功，服务端回复服务端 hello 信息（SHLO），并在其中包括成功加密的信息。
7. 客户端接受服务端的信息，如果是 REJ 信息，则同第 4 步操作；如果是 SHLO 消息，则继续进行加密信息的交换。
8. 根据沟通成功的密钥进行通信，握手成功。

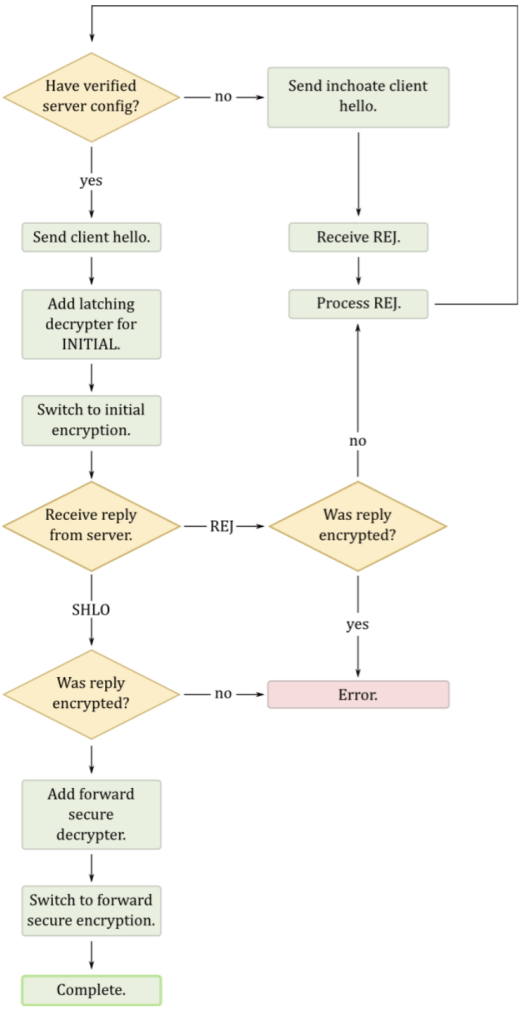

Have verified server config?
no
Send inchoate client hello.
yes
Send client hello.
Receive REJ.
Add latching decrypter for INITIAL.
Process REJ.
Switch to initial encryption.
no
Receive reply from server.
REJ
Was reply encrypted?
SHLO
yes
Was reply encrypted?
no
Error.
Add forward secure decrypter.
Switch to forward secure encryption.
Complete.

图 2.1　客户端握手流程

2.1.2 传输格式

2.1.3 丢包处理

QUIC 在处理丢包主要有两种方案。

- 前向纠错算法
- 丢包重传

2.1.3.1 前向纠错算法

前向纠错算法是通过在传输的包中添加冗余信息，来进行数据的校验和恢复，从而实现丢包的恢复，避免进行重传。在 QUIC 中，主要使用的前向纠错算法是异或算法（XOR）。这种异或算法的基本做法是将若干个包作为一组，额外添加一个冗余的异或包，这样便可以实现只要任何一个包丢失，都可以通过该异或包恢复出结果。这种异或算法的优点是计算成本低而且容易实现，而缺点是只能允许一组包里面有一个丢包出现，否则就不能恢复。

2.1.3.2 丢包重传

丢包重传是 TCP 在遇到丢包时选择的办法，在 QUIC 不能使用前向纠错算法进行丢包恢复的时候，也会使用丢包重传来进行恢复，在这一部分将主要介绍 QUIC 使用的丢包重传算法，以及分析它与 TCP 的不同之处。首先在 QUIC 的传输格式中，与丢包恢复相关联的帧有两个，一是 Stream 帧，因为 Stream 帧包含了应用的数据；二是 ACK 帧，包含了连接过程中传输数据时交互的确认信息。在 QUIC 中，发送方会设置一个重传计时器，计时器的设置与 TCP 相似；在收到 ACK 包之后（包括了最大已被确认的包序号 X），会忽略已经被接受放确认的包（即这些包的传输成功），并根据包中携带的时间戳来更新重传计时器的设置。把那些编号小于 X 并且没有被 NACK 的包记为 ACK，而对于那些编号小于 X 并且被 NACK 的包计入丢包计数器，在丢包计数器超过一定阈值之后，则会认定该包丢失，选择进行重传。与 TCP 的重传机制相比较，即便是重传的包，都会有恒定增加的包序号，从而避免了 TCP 中会出现的 ACK 包的歧义性。此外，在 QUIC 中有更多的 ACK 块，从而能使得 QUIC 在高丢包的环境下加快重传

第 3 章　QUIC 协议评估

本章的主要内容是将 QUIC 协议与 TCP 协议进行了传输任务的吞吐量进行对比，主要分为了在单机进行的仿真测试和真实网络流量情况下两个主机之间的传输测试。其中单机的仿真测试使用的拓扑包括了简单的一对一的拓扑，以及具有瓶颈电路的哑铃状拓扑。而在两个主机之间的传输测试包括了在一个机器上的两个进程分别运行的客户端和服务端之间的传输以及两个不同机器上运行的客户端和服务端之间的传输。

3.1　仿真方案设计

仿真的方案设计如下：由客户端在一段时间内，通过相应的协议 (QUIC 或者 TCP) 向服务端发送随机字符串，而服务端会输出已经成功收到的字节数，通过改变仿真的网络状况参数以及网络拓扑，还有客户端发送数据的流量模式（比如在泊松流的情况下）来对 QUIC 和 TCP 协议进行详尽的评估测试。

3.1.1　网络参数设置

网络参数主要与链路有关，主要参数有网络传输延迟（Delay）、链路带宽（BandWidth）、丢包率（Loss）；除此以外，在哑铃状拓扑中，其瓶颈电路的带宽（MidBw）也是一个重要的参数。在整个仿真过程中，这些参数会被设置为 xxx（用什么格式比较好）

3.1.2　网络拓扑

使用的网络拓扑有以下两种：一是简单的两个主机用一个交换机相连，在3.1所示，其中 h1 是 QUIC/TCP 服务端，h2 是 QUIC/TCP 客户端，S1 是交换机。

3.1.3　流量模式

二是一个哑铃型拓扑，3.2所示，其中 h1 是 TCP 服务端，h2 是 TCP 客户端，h3 是 QUIC 服务端，h4 是 QUIC 客户端，s1 和 s2 是两个交换机，其中 s1 和 s2 之间的链路是瓶颈链路。

图 3.1　简单拓扑

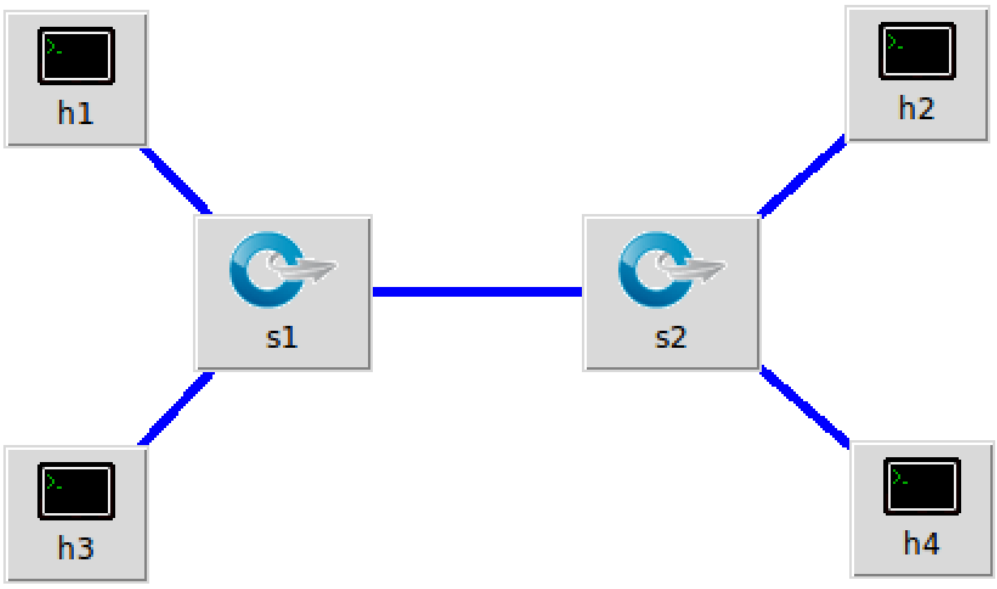


图 3.2　哑铃型拓扑

使用了两种流量模式：第一种是由客户端持续发送一定时间的随机生成的字符串。第二种是由客户端根据泊松流分布进行数据的发送。泊松流

3.2 仿真环境

在这一部分主要讲介绍一下本实验的仿真环境设置。主要包括整个代码运行的环境，以及仿真工具 Mininet 的简单介绍和接口使用，客户端和服务端的代码逻辑设计。

3.2.1 开发环境

开发环境使用的是由 Mac OS X 操作系统下的 VMware Fusion 软件下运行的 Ubuntu14.04，64 位桌面版的虚拟机。

3.2.2 仿真工具 Mininet

Mininet 是一个能够添加虚拟主机、交换机、控制器和链路的网络模拟器。Mininet 的主机能运行标准的 Linux 网络软件，更好的是他的交换机能支持更多自定义的路由算法和软件定义的工具。Mininet 拥有 Python 代码的接口，因此非常易于使用，在本章的实验中，将利用 Mininet 构造不同的网络拓扑，并修改相对应的网络参数，来对 QUIC 协议和 TCP 协议进行测试。下面以哑铃型拓扑为例，给出相应的 Python 代码。

```
class XTopo(Topo):
   def build(self, mid_bw, **opts):
      h1 = self.addHost('h1', ip="192.168.0.1/24", mac="cc:cc:cc:cc:cc:01")
      h2 = self.addHost('h2', ip="192.168.0.2/24", mac="cc:cc:cc:cc:cc:02")
      h3 = self.addHost('h3', ip="192.168.0.3/24", mac="cc:cc:cc:cc:cc:03")
      h4 = self.addHost('h4', ip="192.168.0.4/24", mac="cc:cc:cc:cc:cc:04")
      s1 = self.addSwitch('s1')
      s2 = self.addSwitch('s2')
      self.addLink(h1, s1, **opts)
      self.addLink(h2, s2, **opts)
      self.addLink(h3, s1, **opts)
      self.addLink(h4, s2, **opts)
      self.addLink(s1, s2, mid_bw)

   @staticmethod
   def create_net(mid_bw, **opts):
      return Mininet(topo=XTopo(mid_bw, **opts), link=TCLink)
```

3.2.3 服务端、客户端、仿真程序代码逻辑

客户端程序实现的功能是根据选择的流量模式，在连接到服务端之后，在一定时间内向服务端发送数据。服务端程序实现的功能是在每次成功接收到数据之后，统计已经接受到的数据字节数总和并在每次接收到数据的同时输出到终端。仿真程序（即基于 Mininet 的测试脚本）会定时监视服务端的输出结果，从而来进行数据传输的统计。紧接上一部分哑铃型拓扑的构造，在测试部分对 Mininet 的接口使用代码如下，其中 data1 和 data2 分别都是监控 TCP 和 QUIC 协议服务端得到的数据量大小，通过对 data1 和 data2 的统计，即可知道两种协议在当前网络状态下的传输吞吐量。

```
1   while h1.waiting and h3.waiting:
2       data1 = h1.monitor(timeoutms=RUN_TIME * 2 * 1000)
3       data2 = h3.monitor(timeoutms=RUN_TIME * 2 * 1000)
```

3.3 仿真结果

3.3.1 仿真结果表格

3.3.2 丢包率的影响

3.3.3 链路带宽的影响

3.3.4 瓶颈电路的影响

3.3.5 不同流量模式的结果

3.4 实际网络环境测试方案

3.4.1 这块的实验：相近网络环境下不同机器之间传输随机字符串所需要的时间

3.5 实际环境测试结果

3.6 本章小结

第 4 章　构建支持 QUIC 协议的 web 服务器

Google 将 chrome 中的 QUIC 代码分离出来，构建了一个

4.1　终端的 Http 传输测试

4.2　构建 web 服务器

4.3　客户端下载不同 web 的测试

4.4　本章小结

第 5 章　总结与展望

5.1　QUIC 的评估总结

5.2　QUIC 的拥塞控制算法探索

插图索引

表格索引

公式索引

参考文献

致　谢

衷心感谢导师徐明伟教授对本人的精心指导，他的言传身教将使我终生受益。

在美国麻省理工学院化学系进行九个月的合作研究期间，承蒙 xxx 教授热心指导与帮助，不胜感激。感谢 xx 实验室主任 xx 教授，以及实验室全体老师和同学们的热情帮助和支持！本课题承蒙国家自然科学基金资助，特此致谢。

感谢 ThuThesis，它的存在让我的论文写作轻松自在了许多，让我的论文格式规整漂亮了许多。

声　明

本人郑重声明：所呈交的学位论文，是本人在导师指导下，独立进行研究工作所取得的成果。尽我所知，除文中已经注明引用的内容外，本学位论文的研究成果不包含任何他人享有著作权的内容。对本论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确方式标明。

签　名：_____ 日　期：_____

附录 A　外文资料的调研阅读报告或书面翻译

在学期间参加课题的研究成果

综合论文训练记录表

学生姓名		学号		班级	
论文题目					
主要内容以及进度安排	指导教师签字： 考核组组长签字： 年　　月　　日				
中期考核意见	考核组组长签字： 年　　月　　日				

指导教师评语	指导教师签字：_____ 年　　月　　日
评阅教师评语	评阅教师签字：_____ 年　　月　　日
答辩小组评语	答辩小组组长签字：_____ 年　　月　　日

总成绩：_____

教学负责人签字：_____

年　　月　　日