Towards Efficient Cellular Traffic Offloading via Dynamic MPTCP Path Configuration with SDN

Qi Zhao, Muhao Chen, Pengyuan Du, Tuan Le and Mario Gerla
Department of Computer Science, UCLA

2/19/2019 @ ICNC 2019
Outline

- Introduction
- Problem Identification
- Solution Design
- Evaluation
- Conclusion
Introduction

- Trend of Future mobile wireless system
- The explosion of mobile communications
  - 18-fold increase of Mobile data traffic over the past 5 years
  - 60% of the total is rich multimedia content

Exabytes per Month

<table>
<thead>
<tr>
<th>Year</th>
<th>Exabytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>2016</td>
<td>7</td>
</tr>
<tr>
<td>2017</td>
<td>11</td>
</tr>
<tr>
<td>2018</td>
<td>17</td>
</tr>
<tr>
<td>2019</td>
<td>24</td>
</tr>
<tr>
<td>2020</td>
<td>35</td>
</tr>
<tr>
<td>2021</td>
<td>49</td>
</tr>
</tbody>
</table>
Introduction

- Modern mobile devices are commonly equipped with multiple network interfaces
  - WiFi & Cellular
  - Traffic offloading & Network handover
Introduction

- Many solutions have been proposed
  - Mobile IP
    - Provides smooth handover **BUT** uses one network at a time
  - Multipath TCP
    - Simultaneously utilizes multiple communication paths
    - Smoother reaction to network failure
    - Work over today’s networks
Outline

- Introduction
- Problem Identification
- Solution Design
- Evaluation
- Conclusion
Problem Identification

- MPTCP still has performance issue
  - Lowest-Delay-First scheduler will send the packets to the link with lowest RTT first and then other links with higher RTT
  - Users only have their own local view of the network
  - With multiple users in network, the link with lowest RTT will be congested though there may be more available bandwidth on other network links

How to select path(s) for each MPTCP user in the same network?
How much bandwidth should each MPTCP user utilize for each path?
Outline

- Introduction
- Problem Identification

**Solution Design**

- Evaluation
- Conclusion
Solution Design

- Quantify the Quality of Links
  - Throughput ratio tells that how the overall performance of MPTCP will be compared to SPTCP
  - Bandwidth capacity and delay ratios can be used to predict the overall performance of MPTCP compared to SPTCP instead of the actual value of bandwidth capacity and delay

```
<table>
<thead>
<tr>
<th>Link 1</th>
<th>Link 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>20.835 Mbps / 20 Mbps</td>
<td>10.234 Mbps / 10 Mbps</td>
</tr>
</tbody>
</table>
```

![Graph showing data rate over time for different link configurations]
Solution Design

- Generating Data for Model Training
  - Same topology with a pair of hosts and two links between them
  - Only change the bandwidth and delay ratios of two links
    - Link delay ratio: range(1, 1000)
    - Link bandwidth ratio: [0.1, 0.125, 0.25, 0.5, 1.0, 2.0, 4.0, 6.0, 8.0, 10.0]
Solution Design

- **Model Training**
  - Radial Basis Function kernel based SVM with 10-fold cross-validation

- **Model Evaluation**
  - Link delay ratio: randomly select 100 from range(1, 1000)
  - Link bandwidth ratio: [0.2, 0.4, 0.6, 0.8, 1.2, 2.5, 4.5, 6.5, 8.5, 9.5]

MAE = 0.0573
Variance = 0.0062
Periodically inspect the network and feeds the control module several information to predict the best performance path selection.

Perform prediction and dynamic adjustment.

Deploy the path selection on edge switches by setting up OpenFlow flow tables.

**System Architecture Design**

- **SDN controller**
- **LTE base stations**
- **WiFi APs**

**Informations:**
- **Controller**
- **Information Collect**
- **SVM Model**
- **Path Configuration**

**Techologies:**
- **OpenFlow**
- **OvS**
- **Linux TC**
- **wmediumd**
- **MPTCP**
- **Linux kernel**

**Mobility support**
Software Defined Networking

- MPTCP can only manage their sub-flows locally
- SDN controller manages sub-flows **globally and faster**
Outline

- Introduction
- Problem Identification
- Solution Design
- Evaluation
- Conclusion
Mininet-WiFi-based Emulation Testbed

- **Testbed platform:**
  - Linux Ubuntu 14.04 with 8GB RAM
  - Mininet-WiFi, an extension of Mininet, installed
  - MPTCP v0.92 and Open vSwitch installed

- **Experiment with 3 different protocols** to evaluate the performance of our proposed solution
  - Single-path TCP (SPTCP) – *baseline*
  - Multi-path TCP only (MPTCP)
  - Multi-path TCP with SDN control (our system)
Evaluation Scenario

- 5 Mobile users & 1 host
- 1 OVS switch – LTE station
  - 50ms backhaul delay
  - 8Mbps backhaul bandwidth
  - 50ms propagation delay
  - Station link bandwidth
    - [1, 2, 3, 4, 5]Mbps for Sta1 - 5
- 1 OVS AP – WiFi AP
  - 802.11g wireless network
  - 8Mbps backhaul bandwidth
- 60s total time and users moves into the AP’s range at 30s
Experiment Results - SPTCP

- Communication interruption for all stations at around 30s
- Average throughput almost remains the same
Experiment Results - MPTCP

- Throughput still suffers when new link joins in
- Average 6.832Mbps before users have WiFi access and 13.926Mbps afterwards
Experiment Results - Our System

- All stations connect to and utilize the WiFi link almost immediately.
- Average 15.054Mbps after users have WiFi access, which is 8% higher than MPTCP case.

Station 5 uses WiFi only after 30s based on the SVM model.
Outline

- Introduction
- Problem Identification
- Solution Design
- Evaluation
- Conclusion
Conclusion

- A demonstration of using machine learning algorithm to perform network transmission path selection with accurate prediction for MPTCP
  - SVM model based path selection module
  - Support dynamic path adjustment

- A deployable system architecture to maximize the network resource utilization, particularly for WiFi and LTE heterogeneous wireless networks, without tremendous network infrastructure and protocol modification
Thank you!

Q&A