An Experimental Study of Cellular Data Offloading in Urban and University Scenarios

Jeffrey Wang* and Ning Wang*
*Department of Computer Science, Rowan University, Glassboro, USA
Email: wangj7@students.rowan.edu and wangn@rowan.edu

***Abstract*—The capability's of our mobile devices, e.g., smartphones, allows for multiple connectivity interfaces and overall a better experience sending and receiving data. The limiting factor that slows down the process is latency through cellular and WiFi connections within different environments. By exploring the latency of mobile devices in different connection scenarios, it opens up tons of opportunity's to find more cost-efficient ways to connect to networks, download applications, and use the best possible interface within a given scenario. In this paper, we measured the latency data over the center city, Philadelphia, and the main campus, Rowan University. The results show that different scenarios have different latency pattern, and it is a necessity to propose cost-efficient communication methods.**

***Index Terms*—mobile communication, data offloading, latency measurement**

I. Introduction

With the progress of mobile devices, the data traffic amount for mobile users increases significantly. New mobile applications accessing Internet services are expected to further contribute to the trend. For example, with the increased computational power of these devices, novel application scenarios including augmented reality [1, 2] become realistic, hence leading to a higher bandwidth demand. To address this trend, a new technology consisting of mobile cellular offloading through WiFi networks has emerged. This technological development provides cloud and IT services, e.g., electronic newspapers, advertisements, road-situation reports, maps with traffic statistics, movie clips, etc., to improve performance quality.

There exist two types of wireless access methods. (1) Cellular-based access technologies such as 4G and Long Term Evolution (LTE): it plays a vital role in providing reliable and widely-covered Internet access to mobile users. (2) WiFi-based Access Points (APs): WiFi networks have shown their feasibility in content distribution, and they provide a cheap internet access method with limited coverage area. These APs are deployed intentionally by network service providers and government departments. For example, Xfinity users [3] can access Xfinity hotspots for free on a nationwide scale [4]. An illustration at center city, Philadelphia, is shown in Fig. 1.

In this paper, we explored how much data can be offloaded to the WiFi networks in the delay-sensitive applications in Urban and University scenarios. Email, weather updates, and user data synchronization applications belong to this category since people will not check them every second. The delay-sensitive application has more tolerance to latency, called access latency in [5]. The latency measurement result of this paper can be applied to this type of application to reduce cellular traffic. The study of this paper can be further applied to optimize the cellular data offloading decision [6, 7].

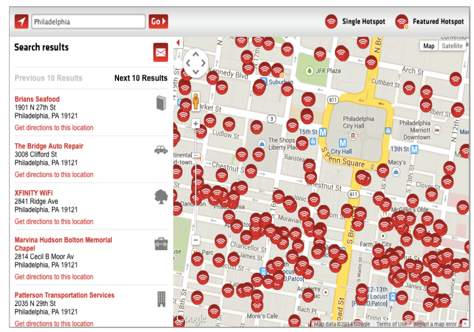


Fig. 1. Xfinity hotspot distribution at Center City, Philadelphia.

II. Experiments

We built a networked system, i.e., a smartphone and a backend server, to conduct experiments.

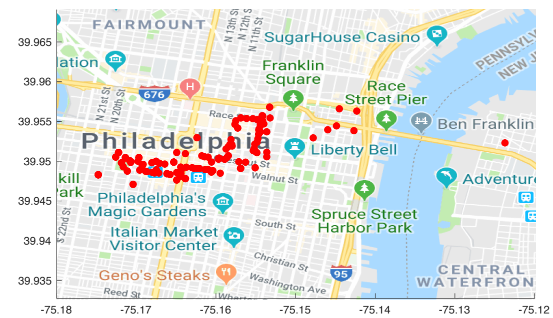
A. Testbed Information

A Java socket server hosted on an Linux machine running Ubuntu 14.04 with 8 2.8 GHz cores and 8 GB RAM was implemented to handle latency transmissions from an android smartphone. The server connected to Xfinity XB6, an IEEE 802.11n wireless AP and the data rate of the WLAN is configured to 1000 Mbps. The smartphone has 8 cores and 4 GB RAM, and the client under test has been rooted. Within the server side, the use of several interfaces or APIs allows for the client to establish a connection through the TCP layer.

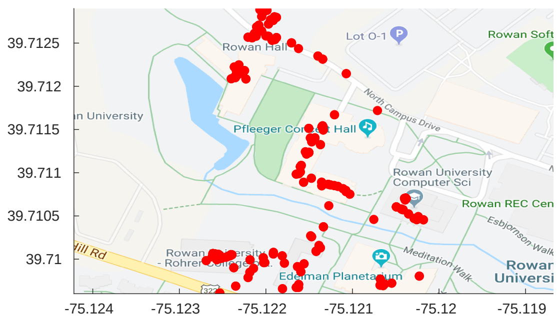
To minimize the workload of the test applications the android AsyncTask was implemented to run the applications in the background as we computed all RTT data offline by using timestamps of packets sending and received with built-in system time function System.currentTimeMillis().

B. Latency Measurement

When finding the RTT from client to server, We first sent data from the client and obtain the reply traces to obtain the RTT. The issue with the methodology was the reply traces to the client had many synchronization issues due to cellular providers. The subsequent method used was measuring specifically the time server side by sending two packets: one to establish the connection through a ping, and one possessing



(a) Center City, Philadelphia



(b) Main Campus, Rowan University

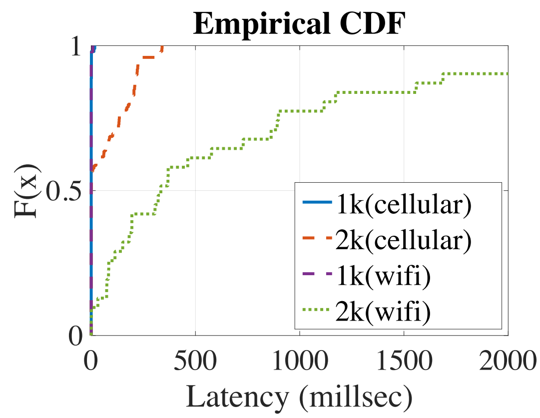
Fig. 2. Data collection during experiments

the actual data. The time was then measured for both packets to be received to give the RTT of the data. Both methodologies were thoroughly tested on both interfaces.

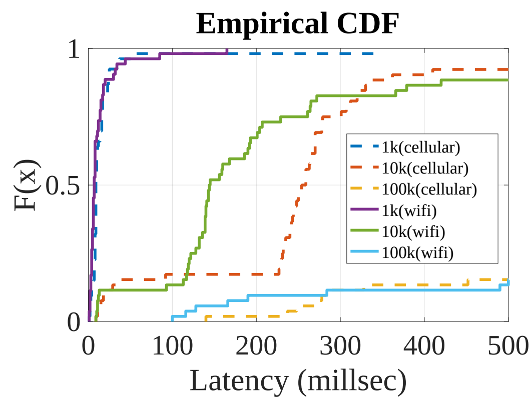
Since we are interested in varying latency's within different environments, we tested a dedicated server located within the Rowan University to ensure the data received would yield the most accurate results. We measure data transmission latency at two locations, Center City, Philadelphia, and Main Campus, Rowan University, as shown in Fig. 2. We used a range of different data sizes from 1kb to 100kb with a sampling frequency of 15 seconds to ensure packets get fully transferred. To handle the larger data sizes such as 100kb and above, a buffer was implement to stop the server from crashing as well as ensuring packets would fully get received.

III. Experimental Results

The Cumulative Distribution Function (CDF) results of latency measurement by using the proposed testbed are shown in Fig. 3. We observed that the data latency in both interfaces increases along with the packet sizes. In addition, the WiFi interface is more sensitive compared with the cellular interface in both scenarios. In Fig. 3(a), the WiFi interface has a much larger latency in the center city, Philadelphia. However, the WiFi interface has a smaller latency at Rowan University. As a result, the latency patterns at different locations are different. Another observation is that Internet data routing latency has an influence. The overall transmission latency at Rowan University is much smaller.



(a) Center City, Philadelphia



(b) Main Campus, Rowan University

Fig. 3. Experimental results

IV. Conclusion

In this paper, we established a networked testbed to measure the smartphone's connection latency in WiFi and cellular interfaces at two locations. The results show that different locations have different latency patterns. Our future work is to propose data offloading algorithm to optimally determine the data offloading strategies.

References

[1] J. I. Feijoo and G. A. Gomariz, "New challenges on crossplatform digital."

[2] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Communications Surveys & Tutorials*, 2017.

[3] https://en.wikipedia.org/wiki/Xfinity.

[4] D. K. Jha, J. P. Rula, and F. E. Bustamante, "exploring xfinity," in *Proceedings of the QEST*. Springer, 2016, pp. 136–148.

[5] X. Wang, M. Chen, Z. Han, D. O. Wu, and T. T. Kwon, "Toss: Traffic offloading by social network service-based opportunistic sharing in mobile social networks," in *Proceedings of the IEEE INFOCOM*, 2014.

[6] N. Wang and J. Wu, "Optimal data partitioning and forwarding in opportunistic mobile networks," in *Proceedings of the IEEE WCNC*, 2018.

[7] N. Wang and J. Wu, "Opportunistic wifi offloading in a vehicular environment: Waiting or downloading now?" in *Proceedings of the IEEE INFOCOM*, 2016.