Leveraging Wi-Fi Signals to Monitor Human Queues

Yan Wang and Yingying Chen, *Stevens Institute of Technology*
Richard P. Martin, *Rutgers University*

EDITOR'S INTRO

A common daily task for most of us is the process of waiting at a store check-out, in a post-office, or in a security line at an airport. Knowing how long we're likely to wait in a line can help us decide if we really want to commit the time. Or, if we have no choice, it can help us determine if the wait time will be problematic—perhaps making us miss our flight. This article applies pervasive computing in an ingenious way to help resolve the human-queuing problem. *—Roy Want*

The goal of pervasive computing research is to create devices or systems that can leverage computationally rich environments to support daily human-computer interactions. The use of smartphones and their data-intensive apps has created novel opportunities to exploit such network traffic for monitoring and optimizing real-world processes. Research has shown that we can use cellular call data records and cellular signal traces to infer large-scale transportation patterns[1] and the level of congestion on roadways, respectively.[2] Similarly, pervasive wireless infrastructures (such as Wi-Fi and Bluetooth) provide increasing convenience in our day-to-day activities. Furthermore, we have found that we can use signal readings from the Wi-Fi traffic consumed by smartphones to monitor a finer-scale yet common process in our daily lives—human queues.

We often wait in long lines at many different places—including at retail stores, banks, theme parks, hospitals, and transportation stations. We've found a way to exploit the existing Wi-Fi infrastructure to extract unique Wi-Fi signal patterns from the smartphones of people in line to estimate the wait time. The major advantage of our approach is that it can work under real-world queue scenarios in various environments without requiring a specialized infrastructure or incurring manpower overhead. Furthermore, our solution only requires that a small fraction of people waiting in line use Wi-Fi on their smartphones.

Our approach can work under real-world queue scenarios in various environments without requiring a specialized infrastructure.

MONITORING HUMAN QUEUES

Figure 1 presents an abstraction of a human queue in a typical environment. The *waiting period* is the time between arrival and receipt of service. During the *service period*, people might pay for items or accept treatment, depending on the service. People exit the queue during the *leaving period*. Note that our concept is interpreted loosely—people don't need to stand in a line but could sit in a waiting room, and they might not be served in a strict first-in, first-out order.

Real-time quantification of the waiting and service times in such queues can help optimize service processes across various industries by helping managers, service providers, travelers, and even customers change their behavior and processes as needed. Managers could make staffing decisions based on the service length derived from the queue measurements. For example, during certain hours of the day, the wait for service might grow longer at a coffee shop due to increased demands for espresso drinks compared to other items. In such a case, it might be more effective to change the staffing to use skillful baristas as opposed to simply adding staff.

Similarly, a hospital emergency department might want to have experienced nursing staff help with triage when waiting times for patients become too long. Airport checkpoints experiencing abnormally long delays could divert screeners from queues with shorter waiting times. Customers also can benefit from accurate queue measurements. For example, knowing when the checkout lines in a warehouse are expected to be shorter could help a customer better arrange his or her schedule.

Published by the IEEE CS ■ 1536-1268/14/$31.00

Existing solutions to the queue-monitoring problem often rely on cameras[3] or special sensors (such as infrared[4] or floor-mat sensors[5]) at multiple locations. Bluetooth signals from smartphones have also been exploited to measure travel times in airports and for vehicle traffic. However, these solutions require multiple sensors to fully monitor a single queue, which increases installation and system costs. In addition, these techniques, which use wireless networks, are too coarse-grained to differentiate between the waiting and service time.

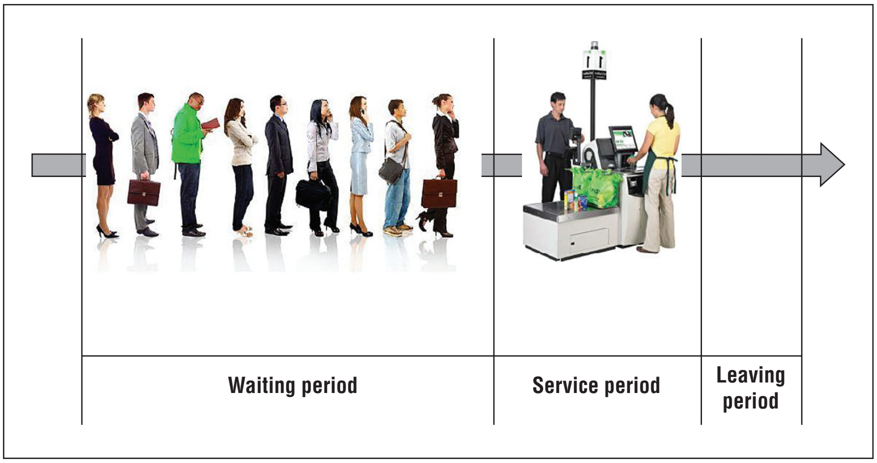


Figure 1. Important time periods and corresponding positions in a human queue.

A SINGLE WI-FI SIGNAL-BASED APPROACH

Our approach uses a single Wi-Fi monitor, close to the front of the queue, to measure the received signal strength of packets emitted from phones. Intuitively, the received signal power should follow a known pattern, increasing as a smartphone user moves toward the service point and the phone moves closer to the monitor. When the person is receiving the service, the signal power should be strong and relatively stable. Finally, the received signal power drops dramatically when the person exits the service point.

Figure 2 presents the received signal strength (RSS) trace of a smartphone in a queue collected from a single Wi-Fi monitor at the service desk in a coffee shop. The captured RSS trace reflects the pattern of the distance between the smartphone user and the service desk.

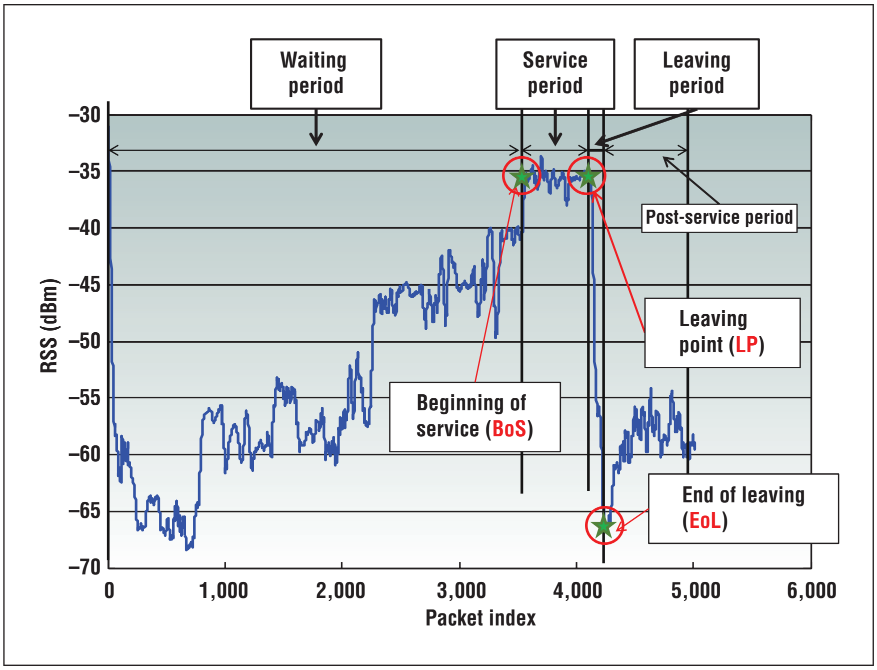


Figure 2. Illustration of special queue-related patterns embedded in the signal trace collected from a smartphone in a queue.

Detection Challenges

Accurately discerning the time points—when a person begins and ends service—with a single-point monitoring system is challenging, because the multipath, shadowing, and fading components of a wireless signal are quite dynamic due to the movements of the person and surrounding people. Here, we summarize the major challenges to implementing our system.

Tracking queues. Our low infrastructure approach—using only a single Wi-Fi monitor—can't uniquely determine the phone's position. Our solution should be able to identify the unique characteristics presented in the smartphone's signal traces to perform queue parameter estimation without needing to explicitly localize the phone.

Dealing with real environments. Although the distance between the smartphone and the service desk dominates the received signal, the RSS is affected by various factors, including user movements, changing environments, signal interference, and the multi-path (Rayleigh fading) effect typical of indoor environments. Also, different holding styles and vibrations of smartphones also cause noisy RSS readings. Moreover, people standing in queues cause large signal attenuation, because over 60 percent of the human adult body is water, which heavily absorbs Wi-Fi signals. Thus, the system

SMARTPHONES

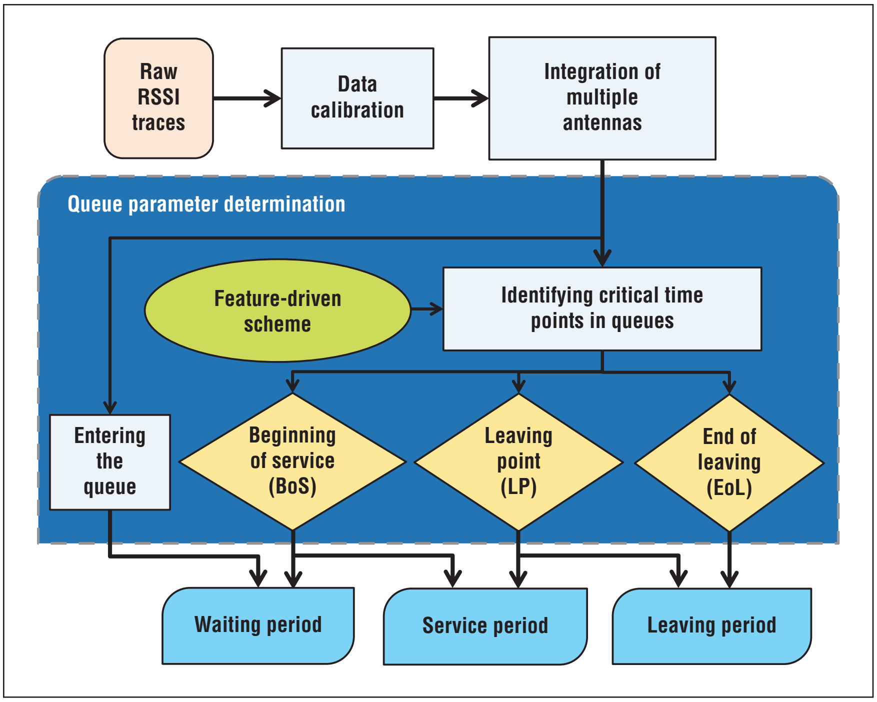


Figure 3. Flow overview of our queue-monitoring system.

should be designed in such a way that it can cope with noisy signal readings.

Identifying queue-related signal traces. The received signal traces extracted from smartphones can't be directly used to estimate queue parameters, because trends of the received signal could occur after the queue process that are similar to the RSS trend within the queue. An effective data-calibration mechanism is required to identify which segment of the RSS trace includes only the important periods of the queue process.

System Overview

Our system comprises three main subtasks: *data calibration*, *integration of multiple antennas*, and *queue parameter determination*. Figure 3 shows the flow of our system. The Wi-Fi monitor discovers the smartphone when the user enters the queue and starts to passively record the phone's RSS readings. Our system first applies data calibration to the RSS trace, which aims to preserve the unique trend presented in the raw RSS trace while removing high frequency noise. The system also identifies segments of the RSS trace containing related time periods for measuring queue parameters.

To filter out signal outliers and obtain a reliable Wi-Fi signal trace, the system further integrates multiple antennas, which already exist in many Wi-Fi access points. This subtask combines the selected RSS traces from two antennas in the Wi-Fi monitor to generate an integrated signal trace that fortifies the unique pattern of the signal associated with important time periods of the queue.

Finally, queue parameter determination implements a feature-driven scheme to infer the critical time points in the queue. The critical time points—the beginning of service (BoS), leaving point (LP), and end of leaving point (EoL)—are used to estimate the queue parameters, including the waiting period, service period, and leaving period. Specifically, the waiting period is the time interval between the BoS and the starting time of the trace, whereas the service period is the time interval between the BoS and the LP. The leaving period is the time interval between the LP and EoL.

FEATURE-DRIVEN SCHEME

We developed a *feature-driven* scheme for our system to identify the critical time points in a human queue. Because the RSS value changes dramatically when people leave the queue after service, the features associated with the leaving period are the most obvious and easy to extract. Therefore, when designing the feature-driven scheme, we employed a time-reversed strategy, which directly applies the features associated with the leaving period in the RSS trace to determine the EoL first and then the LP and the BoS.

In particular, we identified three features extracted from the RSS trace associated with the leaving period:

- the leaving period has the longest consecutive negative-slope segments of the selected RSS trace;
- the received signals before the leaving period are stable with the highest amplitude of the selected RSS trace; and
- the leaving period experiences the largest decrease of the signal in the selected RSS trace.

There are two major components in the scheme: the *EoL estimator* and *LP/BoS estimator*. The EoL estimator exploits the three features just listed to determine whether a consecutive negative-slope segment is likely to be a leaving period, which indicates the starting time of the segment to be EoL. The LP/BoS estimator exploits the observation that the distributions of Wi-Fi signal changes (that is, the slopes of RSS) before and after the LP and BoS are significantly different to identify the LP and the BoS, respectively.

EVALUATION

We investigated the Wi-Fi device density in a coffee shop for one month to determine whether a sufficient amount of Wi-Fi users are present in human queues to facilitate queue measurements. We used a Wi-Fi monitor, placed close to the service desk, to passively monitor Wi-Fi packets sent by mobile devices (see Figure 4). A mobile device is determined to be within the queue if the received signal strength (RSS) amplitude of the device is greater than an empirical threshold (–45dBm). Meanwhile, we manually counted the arrival time of each new customer and the length of the queue as the ground-truth for comparison. More than 30 percent of the customers were using Wi-Fi (mostly via smartphones) while waiting in line for coffee, indicating that the number of Wi-Fi signal-emitting devices in the queue would be sufficient for real-time human queue measurements.

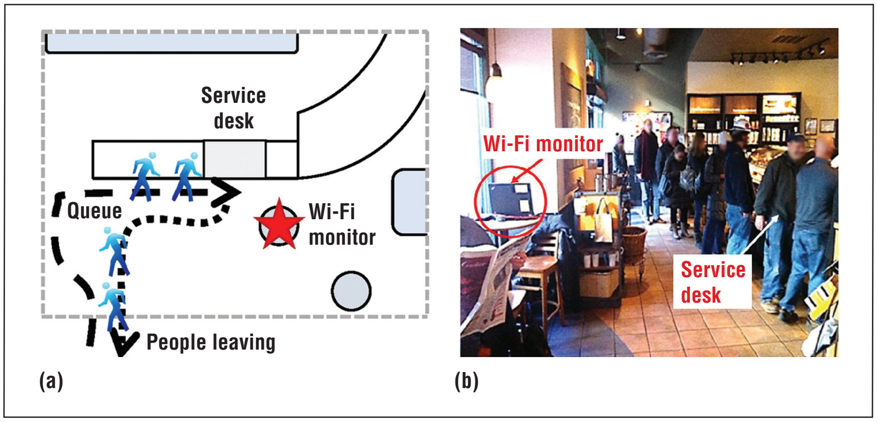


Figure 4. System evaluation: (a) an illustration of the experimental setups and (b) the actual coffee shop environment.

We further collected 72 RSS traces in the coffee shop over the month, with volunteers in the queue carrying various smartphones, including an HTC 3D, an HTC EVO 4G, and a Nexus One. When using our system, the median error of the LP and BoS estimation was approximately 4 seconds, and the median errors of the derived waiting and service time periods was less than 7.5 seconds, which is only approximately 11 percent of the ground truth. This indicates that our system is effective in measuring human queues with high accuracy using only a single-point Wi-Fi signal monitor and a sample of phones in the queue.

Our Wi-Fi-based solution for queue measurements could enable a wealth of new applications, such as bottleneck analysis, shift assignments, and dynamic workflow scheduling. With improved accuracy, compatibility, and security, the Wi-Fi-based solution could create a minimum infrastructure approach to provide real-time queue information instead of using cameras or special sensors. This queue-monitoring system could also potentially use existing Wi-Fi access points without adding the additional single monitor.

By using the already-deployed Wi-Fi infrastructure in various environments, we offer low-cost and highly flexible pervasive queue monitoring. Our approach not only benefits queue monitoring in small public areas but also provides convenient solutions for daily work practices in transportation, such as dynamically tracking the congestion at bus and train stations, which would be a critical basis for adjusting the bus and train schedules or boarding and payment processes.

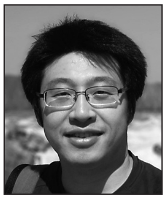
ACKNOWLEDGEMENT

We thank Marco Gruteser, Jie Yang, and Hongbo Liu for their contributions toward developing the Wi-Fi signal-based human queue measuring system.

REFERENCES

1. S. Isaacman et al., "Human Mobility Modeling at Metropolitan Scales," *Proc. 10th Int'l Conf. Mobile Systems, Applications, and Services* (MobiSys 12), 2012, pp. 239–252.
2. G. Chandrasekaran et al., "Vehicular Speed Estimation Using Received Signal Strength from Mobile Phones," *Proc. 12th ACM Int'l Conf. Ubiquitous Computing* (Ubicomp 10), 2010, pp. 237–240.
3. V. Parameswaran et al., "Design and Validation of a System for People Queue Statistics Estimation," *Video Analytics for Business Intelligence*, vol. 409, 2012, pp. 355–373.
4. U. Stilla et al., "Airborne Monitoring of Vehicle Activity in Urban Areas," *International Archives of Photogrammetry and Remote Sensing*, vol. 35, no. B3, 2004, pp. 973–979.
5. D. Bauer et al., "Simple Sensors Used for Measuring Service Times and Counting Pedestrians," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2214, no. 1, 2011, pp. 77– 84.

Yan Wang is a PhD candidate at the Stevens Institute of Technology. Contact him at ywang48@stevens.edu.



Yingying Chen is an associate professor in the Department of Electrical and Computer Engineering at the Stevens Institute of Technology. Contact her at yingying.chen@stevens.edu.

Richard P. Martin is an associate professor in the Department of Computer Science at the Rutgers University. Contact him at rmartin@cs.rutgers.edu.