DEMO: EApp: Improving Rural Emergency Preparedness and Response

Karyn Doke, Nachuan Chengwang, Andrew Boggio-Dandry*, Petko Bogdanov, Mariya Zheleva
Department of Computer Science, *Department of Electrical & Computer Engineering, University at Albany, SUNY
{*kdoke, nchengwang, aboggio-dandry, pbogdanov, mzheleva*}*@albany.edu*

***Abstract*—Large-scale emergencies, both natural and man-made, are increasingly incurring devastating losses in terms of infrastructure and human lives. Rural areas, with their unique socio-economic structure, are particularly vulnerable to such losses. While the U.S. critically relies on its rural areas for water, food supply and energy, rural areas emergency preparedness and response severely lags behind that of their urban counterparts. One of the key limiting factors is the lack of adequate broadband connectivity, which limits agencies' capabilities to (i) disseminate emergency preparedness and response information to residents and (ii) efficiently coordinate in the face of a disaster. In this demo, we will present the EApp; a smartphone application that strives to improve the information access regarding emergencies for rural residents and first responders.**

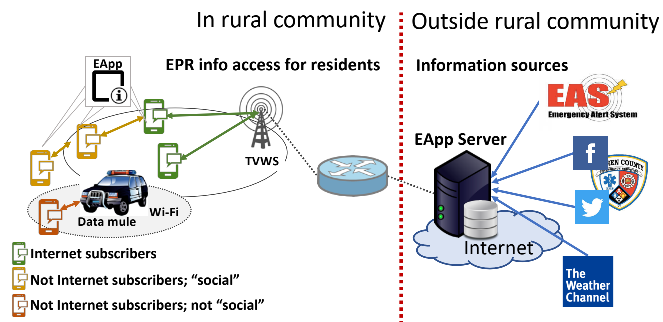

Fig. 1. EApp System Architecture

I. Introduction

Emergency preparedness and response (EPR) in rural areas lags behind urban communities. EPR services increasingly rely on mobile broadband connectivity for timely access to emergency information [1], [2], [3]. Although there are an abundance of these technologies in urban settings, they are lacking in the rural communities. 60 million (19.3%) of the U.S. population resides in rural areas [4] and 30% of these residents lack broadband access [5]. There is little expectation that these areas will be reached by commercial technologies in the foreseeable future, as mobile operators find it hard to justify the low return on investment of building infrastructure in areas with sparse or impoverished populations [6], [7].

We aim to improve EPR services in rural communities by maximizing the reach of EPR information to first responders and residents using heterogeneous networks. The goal is to create a usable interface through a mobile Emergency App (EApp), to provide timely emergency information to local residents, and to observe the characteristics and limitations of rural communities so that we may better understand how to optimize information exchange. Since most rural communities are technically challenged, EApp must make use of heterogeneous networks including traditional WiFi, peer to peer, bluetooth and cellular to maximize the reach of emergency information. To validate our approach, we collaborate with a rural community in Warren County, New York. The town, as with many rural communities in the U.S., has limited connectivity which is further impeded by mountainous terrain. The town currently has a TV White Space (TVWS) network setup to provide Internet access to a subset of local residents.

II. EApp Architecture and Implementation

The architecture of our system is illustrated in Figure 1. EApp is a smartphone application that supports the collection of information from various sources and its exchange among first responders, government agencies and residents. EApp is connected to a server that hosts the EPR information from various sources including local government agencies, first responders, and state agencies. Residents of the community can access the information directly while connected to WiFi, a cellular network or by a peer to peer exchange when coming into the vicinity of another resident that has the same mobile app. We define 3 types of users that will use the app i) an Internet subscriber, ii) not an Internet subscriber but "social" (i.e. these who typically meet with other members in the community), and iii) not an Internet subscriber and "not social" (i.e. these who do not tend to meet with other members of the community).

The app is designed for two main purposes: (i) provide timely EPR information to as many residents and first responders as possible, and (ii) collect research data to model information exchange in rural socio-physical networks. Residents can access EPR information through an interface on the app. This information is updated periodically, unbeknownst to the resident. The app will periodically connect, when WiFi is available, to the EApp server to check for new information. The challenge with this is that only about 10 percent of the residents are actual subscribers to the local area network, which promotes a massive challenge to reach all the residents in the community. To overcome this challenge, we use WiFi Direct (P2P) technology [8], to leverage phone-to-phone (P2P) information exchange and extend the reach of emergency information in the community by leveraging residents' social interactions. Local residents, that are TVWS subscribers or that travel outside of town, can access information directly from the server. For residents that are neither a TVWS subscriber nor a traveler, we seek to extend the information access via P2P information exchange.

We implement the EApp using Android Platform 3.3.1 with an API level 28. This API is compatible with Android 8, Oreo. On the server-side, we use MySQL for the database and Java framework Hibernate and Spring as an interface between EApp and the server database. Additionally, we use several python scripts to scrape information from various sources.

Challenges in efficient P2P information exchange. Maximizing P2P information exchange critically depends on peers' ability to discover each other, which is affected by several factors. First, the synchronicity of scan attempts across peers naturally affects how likely these peers are to find each other and establish a connection. Second, even though scan attempts are scheduled in the app at regular intervals, the Android operating system may de-prioritize scans, thus, interfering with the scan periodicity and further aiding to the dis-synchronization across peers. These factors can be mitigated by careful selection of the intensity and duration of scan attempts, which creates a key tradeoff between connection success and battery lifetime we will seek to explore.

III. Evaluation

In our experiments, we seek to evaluate the potential of WiFi Direct technology to cater to EPR information exchange in rural disconnected areas. Of utmost importance are that both devices are scanning within the same time interval and successfully establishing a connection with each other. In this section, we analyze factors that influence P2P encounter in a small scale experiment. We use two phones that scan for a duration of time and we observe the intervals in which each phone is scanning and establishing a connection with each other. We begin by describing our experimental setup and then present results.

A. Experimental Setup

For each experiment, we use two Motorola G6 smartphones with Android 8 operating system and the EApp installed on the phones. Two metrics are important to evaluate: (i) the achieved scan periodicity on each phone and (ii) the scan offset between the two phones. We define the scan period as the time difference between consecutive scans on the same phone and note that while the EApp is programmed with a fixed scan duration, the actual scan time is mandated by the Android operating system. Additionally, we define the scan offset Δt_O^i as the time difference in scan attempts between the two phones $\Delta t_O^i = |t_i^1 - t_i^2|$ for the i-th scan attempt. As long as Δt_O^i remains smaller than the duration of scan there is a good probability that the devices will discover each other and attempt to connect (form a group). If that is not case then there will be little chance of discovery. We run 3 experiments using two smartphones all with a periodicity of 60 second intervals, for a duration of approximately 2 hours. We synchronize the phones using NTP [9] and then start the app on both phones simultaneously. In the first and second experiment both phones are offline with WiFi enabled. In the third experiment one phone is offline with WiFi enabled, and the other phone is connected to a legacy WiFi access point (AP).

B. Results

The results of our experiment are presented in Figure 2, Figure 3, and Figure 4. In all cases, we seek to understand the scan offset and scan periodicity trends. We then analyze the likelihood that the phones can discover each other and connect (form a group). On the x-axis of all graphs, we have the relative time (in seconds) from the beginning of each experiment. The top graphs plot the scan offset between the two phones, whereas the bottom graphs plot scan periodicity on each of the phones. In the top graphs, the red line represents the scan offset, whereas the blue line designates scans that resulted in a successful connection. In the bottom graphs the red line represents the scan periodicity for Phone 1, whereas, the green line represents the scan periodicity for Phone 2.

Considering the scan periodicity across all experiments (bottom graphs), we see that many of the time intervals maintained their expected periodicity of 60 seconds, however, some scan attempts saw a greatly increased scan interval of up to 85 seconds. Although the app was started at the same time, we observe that each phone did not start to scan at the same time. For example in Figure 2, Phone 2's first interval is at the expected 60 seconds but Phone 1's first interval is around 75 seconds. This initial offset was further adjusted by a few rapid changes in the periodicity of the following scan attempts, which ultimately resulted in majority of the scan attempts being offset by about 10 seconds (top graph). The blue markers on this graph show that only a subset of all attempts resulted in a successful connection, while a majority of them failed. These results show that even though phones are time-synchronized and apps started simultaneously, they might experience a persistent offset due to the Android OS behavior, which in the long run, results in peers not being able to establish a successful connection.

We see that the success rate across the three tests varies: in the first test we see 9.5% successful attempts, followed by 13.4% in the second and 13.5% in the third. We hypothesize that the first scan was the least successful, because of the large variation of scan intervals experienced on both phones early in the test. This ended up creating a persistent and large offset, which hampered further connection success. At the same time, the second and third test saw less variation in scan intervals at the onset of the experiment, which, in the long run, gained more successful connections. We will investigate these trends across multiple repetitions and phone behaviors to understand how we can model, predict, control and incorporate them in the EApp behavior to maximize the P2P connections.

IV. Discussion and Future Work

While infrastructureless phone-to-phone communication brings promise for improved information dissemination

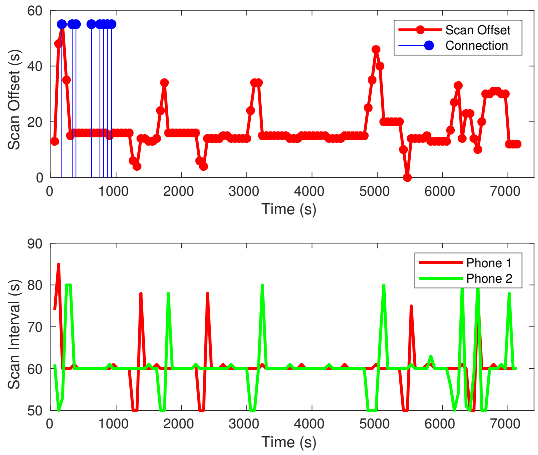

Fig. 2. (Exp 1) 2 Hour Duration with 60 (s) Intervals

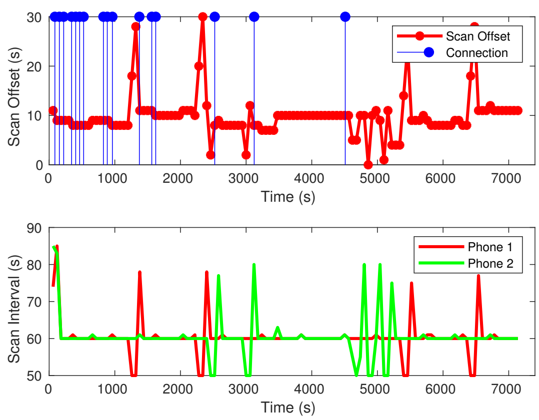

Fig. 3. (Exp 2) 2 Hour Duration with 60 (s) Intervals

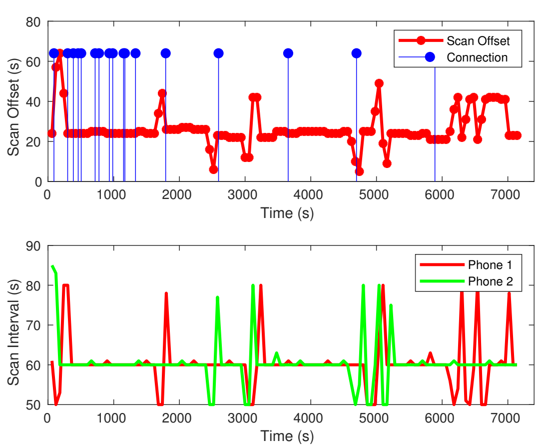

Fig. 4. (Exp 3) 2 Hour Duration with 60 (s) Intervals

in various application scenarios, P2P communication was conceived with the assumption that users will be actively engaging with their phones to establish a connection. Repeated, programmatic peer-to-peer exchange, while highly-desirable for applications such as the EApp, is very challenging due to the synchronization issues discussed in this paper. Our initial exploration of WiFi Direct for P2P exchange of emergency information shows that the success of a connection critically hinges on the synchronicity of scan attempts and the interference inflicted by the Android OS scheduler. In our future work we will explore ways to account for the OS behavior and perform runtime adjustments of the scanning configuration to maximize the P2P connections. In addition, we will also investigate how does the interplay between legacy WiFi APs and WiFi Direct affect the connection success.

V. EApp Demonstration

We will be demonstrating our smartphone application, the EApp. The presentation will focus on the peer-to-peer exchange functionality between two phones. A laptop will be used to post new tweets (mock emergency information) to a twitter account. These tweets will be posted to our database on the remote server. The first phone (online phone with internet connection) will demonstrate the newest information received by the app and will exchange this new information to the second phone (offline phone no internet connection). For our demo, we will require internet access. A setup time of 1 hour is sufficient and we do require any additional space beyond the default setup.

References

[1] "Department of Homeland Security. System Assessment and Validation for Emergency Responders (SAVER) ." https://www.dhs.gov/sites/default/files/publications/CAD_TN_0911-508.pdf, 2011.
[2] "FEMA Smartphone App." https://www.fema.gov/mobile-app.
[3] "Facebook Safety Check." https://www.facebook.com/help/695378390556779.
[4] "U.S. Census Beureu. New Census Data Show Differences Between Urban and Rural Populations." https://www.census.gov/newsroom/press-releases/2016/cb16-210.html.
[5] "Federal Communications Commission 2018 Broadband Deployment Report." https://www.fcc.gov/reports-research/reports/broadband-progress-reports/2018-broadband-deployment-report.
[6] M. Zheleva, A. Paul, D. L. Johnson, and E. Belding, "Kwiizya: Local Cellular Network Services in Remote Areas," MobiSys'13, (Taipei, Taiwan), 2013.
[7] "Rural America: How Wireless Technologies Could Impact America's Heartland." https://wia.org/wp-content/uploads/WIA_RuralAmerica-2.pdf.
[8] "WiFi Alliance Technical Specification." https://www.wi-fi.org/discover-wi-fi/specifications.
[9] "Network Time Foundation." https://www.ntp.org.