Towards Commoditized Real-time Spectrum Monitoring

Ana Nika, Zengbin Zhang, Xia Zhou†, Ben Y. Zhao and Haitao Zheng
University of California, Santa Barbara †Dartmouth College
{anika, zengbin, ravenben, htzheng}@cs.ucsb.edu, †xia@cs.dartmouth.edu

ABSTRACT

We are facing an increasingly difficult challenge in spectrum management: how to perform real-time spectrum monitoring with strong coverage of deployed regions. Today's spectrum measurements are carried out by government employees driving around with specialized hardware that is usually bulky and expensive, making the task of gathering real-time, large-scale spectrum monitoring data extremely difficult and cost prohibitive. In this paper, we propose a solution to the spectrum monitoring problem by leveraging the power of the masses, *i.e.* millions of wireless users, using low-cost, commoditized spectrum monitoring hardware. We envision an ecosystem where crowdsourced smartphone users perform automated and continuous spectrum measurements using their mobile devices, and report the results to a monitoring agency in real-time. We perform an initial feasibility study to verify the efficacy of our mobile monitoring platform compared to that of conventional monitoring devices like USRP GNU radios. Results indicate that commoditized real-time spectrum monitoring is indeed feasible in the near future. We conclude by presenting a set of open challenges and potential directions for follow-up research.

Categories and Subject Descriptors

C.2.3 [**Network Operations**]: Network monitoring

Keywords

Spectrum monitoring; Crowdsourcing

1. INTRODUCTION

Radio spectrum is one of the most sought-after resources in the world. In the US, public spectrum auctions by the government generate billions of dollars for the right to utilize spectrum bands. Outside of spectrum auctions, spectrum rights are viewed as commodity valuable enough to motivate the outright purchase of a company, as was the case for Sprint's purchase of Clearwire Communications, and the ongoing rumors of a takeover of Dish Networks.

But despite the value placed on wireless spectrum, there is surprisingly little attention paid to an increasingly difficult challenge: real-time spectrum management in deployed settings, *e.g.* measurements, fault detection and diagnosis, and attack and anomaly detection. In next generation wireless devices, the density of spectrum usage will continue to grow over both the geographic and frequency domains. The biggest challenge will be coverage, as current monitoring tools and mechanisms will not scale to cover the deployed physical networks, as well as the increasing range of spectrum frequencies being used.

As a compelling example of spectrum monitoring applications, consider the problem of *spectrum enforcement*. As context, the US government has opened up TV whitespaces to accommodate wireless broadband across the country, and could allocate more frequency bands for the same purpose [22]. Yet despite advances in algorithms, hardware and software platforms, research on whitespaces has not addressed spectrum enforcement, *i.e.* how do we detect and locate unauthorized users whose transmissions may interfere and disrupt transmissions from authorized spectrum users?

An effective spectrum enforcement system is indispensable to an efficient spectrum system. Without it, malicious users can freely "misuse" spectrum without authorization. For example, an unauthorized usage of Verizon's spectrum band can easily disrupt its normal cellular service, leading to dissatisfied customers and revenue losses [1]. Similarly, misuse of TV whitespaces can disrupt existing users and violate basic assumptions and guarantees fundamental to the operation of whitespace systems.

The challenges facing spectrum enforcement are typical of those faced by spectrum monitoring applications. First, perhaps the biggest challenge is how to gather detailed spectrum measurements with strong coverage of deployed regions [15]. Second, given the dynamics of wireless performance and mobility of users, these measurements should be "real-time," *i.e.* either on-demand or periodic with a high frequency. In contrast, today's spectrum measurements are carried out by government employees driving around with spectrum analyzers and specialized hardware that is usually bulky, expensive, and difficult to operate[1]. Add to this the budgetary and resource constraints of governments around the world, and it is clear that gathering real-time, large-scale spectrum measurements requires a new approach.

Crowdsourcing and Commoditized Spectrum Monitoring. Our solution to the spectrum monitoring problem is to leverage the power of the masses, *i.e.* millions of wireless users, by commoditizing spectrum monitoring hardware. Instead of bulky, specialized hardware sensors, we want to explore the use of cheap hardware extensions to millions of smartphones already deployed in the wild. Our goal is to evaluate the feasibility of commodity smartphone sensors for spectrum measurement, in a possible ecosystem where crowdsourced smartphone users perform automated and continu-

HotWireless'14, September 11, 2014, Maui, Hawaii, USA.

http://dx.doi.org/10.1145/2643614.2643615.

---

[1] Based on the NTIA presentation at the 2014 WSDR workshop.

ous spectrum measurements using their mobile devices, and report the results to a monitoring agency in real-time. Unlike traditional spectrum monitoring tools, our crowdsourced mobile devices can provide improved coverage over wide areas with minimum infrastructure cost.

Our current prototype leverages commodity mobile devices (Android smartphones and laptops) and portable RTL-SDR devices as spectrum sensors. With a price tag less than $20, the RTL-SDR device is easy to replace and effectively disposable. The device can scan spectrum activities between 52-2200MHz, covering a wide range of today's wireless networks. Finally, the device is compatible with open-source GNU radio software suites and can take advantage of a large variety of signal processing blocks.

Initial Feasibility Study. We performed initial measurements to compare the efficacy of our low-cost monitoring platform to that of conventional monitoring devices like USRP GNU radios. Our findings are encouraging. First, RTL-SDR radios on smartphones only have around 10dB sensitivity loss compared to USRP radios with laptops. In practice, this means that an RTL-SDR device might not pick up weak transmission signals, which can be easily compensated by the significant increase in coverage offered by our crowdsourcing approach. Second, while RTL-SDR radios have limited sensing bandwidth, *i.e.* 2.4MHz compared to USRP's 20MHz, they can still scan a 240MHz frequency band in less than $2s$, or cover the entire TV Whitespace (408MHz) within $2.9s$. And we can further reduce sensing time and improve its accuracy by partitioning sensing frequency among neighboring crowdsourcing users or aggregating their monitoring results in time and frequency.

Based on these results, we believe a real-time spectrum monitoring system via crowdsourcing and commoditized hardware is indeed feasible in the near future. However, deploying a practical system requires significant efforts to overcome a number of technical challenges. We conclude by identifying a set of open challenges and potential directions for follow-up research. To the best of our knowledge, we are the first to propose a real-time spectrum monitoring system using crowdsourcing and commodity mobile devices.

2. REAL-TIME SPECTRUM MONITORING

An effective spectrum monitoring system requires significant amounts of measurement data that is comprehensive in coverage area, accurate, and up to date. We propose to achieve this using crowdsourced spectrum monitoring and measurements. In the following, we first discuss the specific challenges of spectrum monitoring, and then present the concept and key components of our proposed design.

2.1 Key Challenges

While a variety of designs and tools can address part of the spectrum monitoring problem, many are impractical in realistic settings. These present significant challenges to our target design.

- *Cost* – we cannot rely on fixed or dedicated monitoring hardware such as spectrum analyzers or USRP GNU radios, because providing adequate coverage of wide areas would require unacceptably high infrastructure costs.
- *Responsiveness* – we cannot rely on periodic spectrum scans or offline processing of third party measurements. Such systems would be slow to react to changing transmissions, and misbehaving transmitters can easily detect and evade them.
- *Coverage* – we cannot simply use systems that extend a small set of measurements using abstract models. Those results would easily breakdown in real outdoor environments, where fading, obstacle blocking and changing physical conditions would render most propagation models highly inaccurate for our purposes.

(a) RTL-SDR/Laptop (b) RTL-SDR/smartphone

Figure 1: RTL-SDR connected to a laptop or a smartphone

2.2 System Overview

Our proposed system addresses the above challenges by integrating two components: *a crowdsourcing measurement framework* that gathers spectrum measurement data in wide areas, and *a low-cost mobile platform* that allows crowdsourced users to perform spectrum measurements automatically in real-time. Next, we introduce the high-level concepts of the two components.

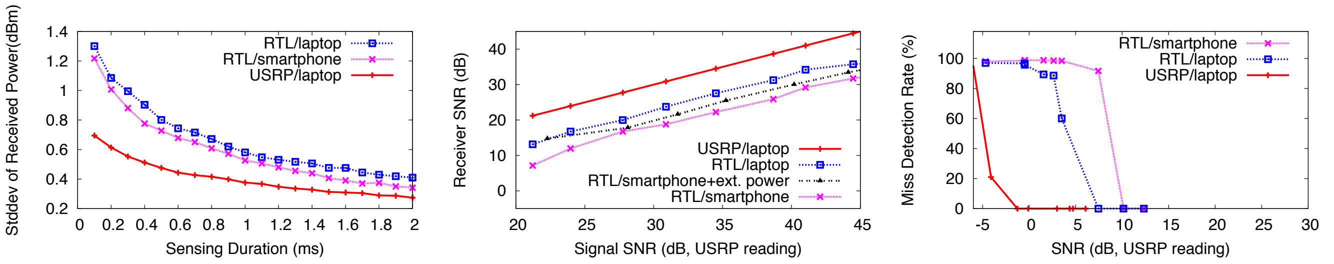
Spectrum Measurement via Crowdsourcing. To obtain adequate coverage for our real-time spectrum monitoring system, we explore a scalable and systematic approach of aggregating individual user effort, *i.e.* crowdsourcing. Today, crowdsourcing services have been widely used to achieve complex measurement tasks at a small fee. For example, Facebook crowdsources content moderation tasks to filter out pornographic and violent media posts; TaskRabbit outsources a family's household errands or a company's long-term and short-term projects to others in the neighborhood. Recently, researchers have successfully used crowdsourcing measurements to collect a large-scale human mobility GPS trace from users around the globe [28].

In our proposed framework, we break the large-scale spectrum monitoring task into simple measurement tasks that can be accomplished by individual users using their mobile devices. Specifically, individual users monitor and collect spectrum activities in their local neighborhood and submit their results in real-time to a spectrum monitoring agency. The agency then aggregates these monitoring results to produce a more complete view of the spectrum usage in a wide area. Furthermore, as these crowdsourcing users move (and possibly change) dynamically, our system can also obtain elasticity required for monitoring spectrum usage across wide areas.

One practical challenge we face in this component is how to ensure adequate coverage during the early deployment phases, when the number of active transmitters and users are both low. We will discuss potential solutions to this challenge in Section 5.

Commoditized Measurement Platform. Another key feature of our design is a low-cost, lightweight mobile measurement platform that enables crowdsourcing users to automatically collect spectrum measurement data. The platform has just two hardware components: a commodity mobile device, *e.g.* smartphones/tablets/laptops, and a cheap and portable Realtek Software Defined Radio (RTL-SDR) that connects to smartphones/laptops via a USB cable. The RTL-SDR behaves as a "spectrum analyzer" and collects raw spectrum usage signal in the wild, while the mobile host behaves as a "data processor" and translates the raw data into a data stream that is more compact (for easy storage and transmission) and meaningful for the monitoring system. Figure 1 illustrates two prototypes: the RTL-SDR connected to a laptop and a smartphone.

More specifically, the RTL-SDR device [2] is a DVB-T dongle that operates in the frequency range of 52-2200MHz and supports a maximum sample rate of 2.4MHz. The portable device can transfer on the fly raw I/Q samples to the host it is connected to. We



(a) Noise variation

(b) Signal measurements

(c) Impact on spectrum monitoring

Figure 2: Impact of sensing sensitivity on noise and signal measurements, as well as signal detection accuracy, using USRP/laptop, RTL-SDR/laptop, and RTL-SDR/smartphone.

chose this device for our initial implementation because of its cost, availability, portability, and its coverage in radio frequency. The device is widely available and costs <$20, and thus can be mass-distributed to crowdsourcing users at a marginal cost.

We also built necessary software to interconnect the two hardware components. For smartphones, we built an Android app on top of the existing RTL-SDR code [2], which enables the smartphone to command the RTL-SDR device in real-time, *i.e.* scanning a specific frequency range at a specific sampling rate for a specific time duration. The app does not require any special driver-level support or root access. For the laptop version, we leverage the open-source project PyRTLSDR [3]. After obtaining the raw I/Q samples from the RTL-SDR device, the mobile host performs FFT to produce power spectrum density map of the collected signal. These can be used to identify active transmissions or detect useful features related to spectrum misuse detection [23].

One key concern is whether the spectrum measurements collected by this low-cost hardware can provide the same level of fidelity and accuracy like those obtained from sophisticated hardware like USRP GNU radios. We have performed initial feasibility study on this issue, which we will discuss next.

3. INITIAL FEASIBILITY RESULTS

In this section, we evaluate the feasibility of using low-cost mobile platform (RTL-SDR) for spectrum monitoring. Compared to sophisticated hardware like USRP GNU radios, the RTL-SDR device has two key limitations:

- *Limited Sensing Sensitivity:* While USRP outputs 14-bit[4] quadrature samples, RTL-SDR outputs 8-bit quadrature signal samples. Because of this resolution difference, RTL-SDR is less sensitive to weak signals and can fail to detect them.
- *Limited Sensing Bandwidth:* While USRP supports up to 20MHz bandwidth, RTL-SDR can only support up to 2.4MHz. In order to monitor a frequency band wider than 2.4MHz, RTL-SDR needs to sweep the band sequentially. This means that it can fail to detect certain short-term (or on/off) transmissions that occupy only a portion of the frequency band.

Next, we perform measurement studies to understand the implications of these limitations on spectrum monitoring. Specifically, we evaluate and compare three monitoring platforms: a USRP N210 radio connected to a laptop, a RTL-SDR radio connected to a laptop (Figure 1(a)), and a RTL-SDR radio connected to a smartphone (Figure 1(b)). All three platforms use the same antenna model and all three antennas are co-located. Note that the comparison between the last two platforms allows us to understand the potential artifact of using smartphones as mobile hosts for spectrum monitoring.

We performed experiments on two frequency bands: an aeronautical telemetry band centered on 1512MHz and a TV Whitespace band centered on 690MHz. We have confirmed via a spectrum analyzer that both bands are vacant at our test area. For the TV Whitespace band, we have also confirmed its availability using the Google Spectrum Database Tool[5]. On each of the two bands, we used another USRP radio as the transmitter to be detected and used the three monitoring devices to detect its presence. To create signals of different strength, we either vary the USRP's transmission power or the distance between the USRP transmitter and the three monitoring devices.

3.1 Impact of Sensing Sensitivity

We start from quantifying the sensing sensitivity difference between USRP and RTL-SDR using noise and signal measurements. For a fair comparison, we configure all three monitoring platforms to operate on a 2.4MHz band. We then translate such difference in the context of spectrum monitoring as the probability of miss detecting the target transmission. Since our experiments on 1512MHz and 690MHz lead to the same conclusions, we only show the results for 1512MHz due to the space limitations.

Signal and Noise Measurements. We first look at the noise power reported by each platform when there is no active transmission, *i.e.* the USRP transmitter is off. Since each platform has a different noise floor, we focus on examining the variance of the noise power as a function of the sensing duration. Figure 2(a) plots the standard deviation of the noise power as we increase the sensing duration from $0.1ms$ to $2ms$. We make two key observations. First, compared to USRP, the two RTL-SDR based platforms report higher noise variance. This is mostly due to the radio, since using laptop rather than smartphone as the mobile host only leads to marginal improvements. Second, the increase in noise variance can be compensated by increasing the sensing duration. Once the sensing duration increases beyond $1ms$, the RTL-SDR based platforms perform similarly to the USRP platform. In the rest of the experiments, we set the sensing time to $1ms$ on each 2.4MHz channel.

Next, we perform signal measurements by turning on the USRP transmitter to emit OFDM signals continuously. We vary the transmit power to create signals of different signal-to-noise-ratio (SNR). Figure 2(b) plots the measured SNR values as we vary the transmit power. Due to RTL-SDR's limited sensitivity, the corresponding two monitoring platforms report lower SNR values ($\approx$13dB lower for RTL-SDR/smartphone, and $\approx$8dB lower for RTL-SDR/laptop) compared to the USRP platform. The difference is pretty consistent across a wide range (20-45dB) of SNR values.

---

[2] https://github.com/keesj/librtlsdr-android

[3] https://github.com/roger-/pyrtlsdr

[4] USRP can output 32-bit float quadrature samples, but its accuracy is fundamentally limited by its 14-bit DAC sampling.

[5] https://www.google.com/get/spectrumdatabase/

We found that one cause for the additional 5dB SNR loss of RTL-SDR/smartphone over RTL-SDR/laptop is that the smartphone's microUSB interface does not provide enough power to the RTL-SDR radio. After connecting the radio to an external power source, the SNR value increases by 3dB. We plan to explore the cause for the leftover 2dB loss in a future work.

Impact on Spectrum Monitoring. Our above results show that the limited sensitivity of RTL-SDR leads to 8-13dB loss in SNR reports, which means that it cannot capture weak signals reliably. In the context of spectrum monitoring, such limitation translates into the need for stronger coverage – monitoring devices "close" to a transmitter can detect its presence.

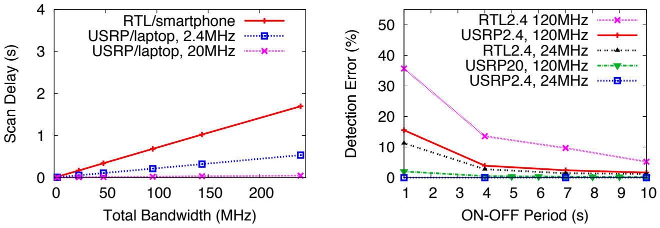
To further understand this impact, we consider a simple monitoring task for a single monitoring device. It needs to detect all signals with SNR values higher than 0dB while maintaining a false alarm rate below 1%. Again we use a separate USRP transmitter to create two types of signal events: "active" or "silent." During "active", the USRP transmitter emits signals continuously at a fixed transmit power and gain, while during "silent" it does not transmit anything. We then use the three platforms to capture the signal for $1ms$ and detect the type of the current event based on the captured signal strength. For each platform we apply a signal strength threshold to ensure 1% false alarm rate, and report the resulting miss detection rate in Figure 2(c) as a function of the signal SNR measured by the USRP platform. We see that the USRP platform can reliably detect signals with SNR$\geq$ -2dB, which increases to 7dB for RTL-SDR/laptop and 10dB for RTL-SDR/smartphone. For the 1512MHz band, such 12dB difference translates into roughly 50% loss in distance [27], which means that the coverage requirement for a monitoring system using RTL-SDR/smartphone devices needs to be 50% denser than that using USRP/laptop. This should be easily achievable using crowdsourcing.

Addressing the Sensitivity Limitation. There are two potential approaches. First, with crowdsourcing, we can deploy many monitoring devices in an area to reduce the sensitivity requirement on each individual device. Second, certain signal features, *e.g.* pilot tones [13] or cyclostationary features [6], are much more reliable signal indicators than energy, which can potentially relax the per-device sensitivity requirement. However, these methods are more complex than energy detection. We plan to investigate the feasibility of realizing them on commodity smartphones in a future work.

3.2 Impact of Sensing Bandwidth

Next, we investigate the impact of RTL-SDR's limited sensing bandwidth, *i.e.* 2.4MHz compared to USRP's 20MHz. To monitor a frequency band wider than 2.4MHz, the monitor device must scan the band sequentially in segments of 2.4MHz. Intuitively, the overall scan delay is the product of the number of segments and the sum of the scan time per segment ($1ms$ based on our earlier result) and the time required to switch between frequency segments.

To examine the scan delay, we configure the two RTL-SDR devices to monitor a wideband of bandwidth between 24MHz and 240MHz, with a sensing duration of $1ms$ per 2.4MHz segment. We configure the USRP device to monitor the same band. For reference, we also configure an extra USRP monitoring device with a sensing bandwidth of 2.4MHz, in order to compare the frequency switching time of RTL-SDR and USRP radios. Figure 3(a) shows the scan delay for RTL-SDR/smartphone and USRP/laptop. The result of RTL-SDR/laptop is the same as that of RTL-SDR/smartphone and thus omitted for clarity. We see that for both platforms, the scan delay increases linearly with the total bandwidth, which is as expected. Yet the scan delay of RTL-SDR is two times higher than USRP (2.4MHz) because its frequency switching delay is higher.



(a) Scanning time (b) Impact on spectrum monitoring

Figure 3: Impact of sensing bandwidth in terms of (a) scanning delay and (b) signal detection error rate, using RTL-SDR/smartphone and USRP/laptop platforms.

Our experiments show that the switching delay of RTL-SDR is upper bounded by $50ms$ with a median of $16ms$ while USRP takes a stable value of $3ms$. Overall, RTL-SDR radios can finish scanning a band of 240MHz bandwidth within $2s$. Consider today's TV Whitespace (68 channels of 6MHz each), it would only take a RTL-SDR device $2.9s$ to scan the entire band (68×6=408MHz).

Impact on Spectrum Monitoring. We quantify the impact of RTL-SDR's sensing bandwidth on spectrum monitoring by the amount of signal detection errors it can lead to. We assume the frequency band to be monitored can be divided into multiple segments of 2.4MHz. For simplicity, we assume each segment experiences independent signal events, modeled by a random ON/OFF process. We then compare the accuracy of RTL-SDR/smartphone, USRP/laptop (2.4MHz), USRP/laptop (20MHz) for detecting the status of these segments. A segment is occupied if the energy detected is above 15dB. To eliminate the impact of RTL-SDR's sensitivity loss, we configure the transmit power such that the signal SNR is 30dB at the measurement locations.

Figure 3(b) plots the detection error rate for monitoring a 24MHz band and a 120MHz band. We vary the mean ON-OFF period between $1s$ and $10s$, mapping to different signal occurrence frequency. For the 24MHz band, RTL-SDR/smartphone achieves $<$10% detection error even when detecting highly dynamic signal events ($1s$ average ON-OFF period). As the band becomes wider (120MHz), the error rate can reach 35% if the signal is highly dynamic.

Overcoming the Bandwidth Limitation. Our results show that the bandwidth limitation of RTL-SDR/smartphone can lead to moderate detection error when monitoring a wideband with highly dynamic signals. There are two potential solutions to this problem. First, leveraging crowdsourcing, we can either divide each wideband into several narrowbands and assign users to specific narrowbands, or aggregate results from multiple users with asynchronous scans. Second, we can leverage novel sensing techniques, *e.g.* QuickSense [25] or BigBand [10], which apply efficient signal search algorithms to perform wideband sensing using narrowband radios. The challenge here is how to realize these sophisticated algorithms on RTL-SDR/smartphone devices, which is our ongoing work.

4. RELATED WORK

Spectrum Sensing & Monitoring. Existing efforts have produced advanced spectrum sensing mechanisms on both narrowband [3, 18, 26] and wideband signals [10, 14, 20, 25] at individual nodes. Furthermore, researchers have also developed compressive sensing (*e.g.* [14, 20]) and collaborative sensing techniques (*e.g.* [8]) to improve sensing robustness and scale. Our work differs from these efforts by using crowdsourced low-cost mobile hardware to collect large-scale spectrum measurements.

Our work also differs from existing spectrum measurement platforms [8, 12] that require specialized and costly spectrum analyzers. The use of low-cost mobile platform is a key factor for attracting a large volume of crowdsourcing users without significant infrastructure cost. Finally, our work differs from WiSense on WiFi bands [17] since our system does not require any smartphone kernel modification and can monitor a wider range of spectrum.

Crowdsourcing and Wireless Measurements. Recent efforts have leveraged crowdsourcing to collect large-scale wireless measurements, enabling them to characterize wireless signal propagation and user mobility [7, 16, 28], to understand network performance and coverage [9, 11, 21], and to improve indoor localization accuracy [19, 24]. Our work adopts a similar crowdsourcing approach but focuses on achieving real-time spectrum monitoring using low-cost mobile measurement platforms. In addition, a number of recent works in the HCI community are relevant to our work, in that they provide the proof of concept for fast work and responses in crowdsourcing systems. Two recent projects studied how workers can respond quickly by either preemptively scheduling tasks [5], or by keeping users on retainer [4]. Our work can leverage a similar methodology to enable real-time monitoring of spectrum usage.

5. ADDITIONAL CHALLENGES

Our initial results indicate that a real-time spectrum monitoring system is indeed feasible in the near future. Transforming this concept into a practical system, however, faces several challenges and requires large research efforts. Next, we discuss several key challenges and potential research directions to address them.

Achieving Adequate Coverage. With crowdsourcing, a practical challenge is how to ensure adequate coverage, especially during the early deployment phases, when the number of active transmitters and users are both low. One potential solution is to adopt a combined in-network and out-of-network mechanism where spectrum measurements come from two distinctive groups of users. First, passive measurements will be collected from each wireless service provider's own user population, by energy-efficient background software running on mobile devices. These providers are active spectrum users who seek reliable spectrum usage to support-/augment their services, and thus are incentivized to participate in spectrum monitoring and enforcement. Second, on-demand measurements from users of other networks can be requested as necessary to augment passive data. Here, a local network entity estimates the coverage from in-network users. If the coverage is below our desired coverage density, it preemptively generates crowdsourcing requests to all out-of-network users in the region.

Minimizing Measurement Overhead. By performing spectrum measurements on mobile devices with limited battery resources, a practical design must account for energy consumption to attract crowdsourcing users. One approach is to schedule measurements based on user context, *e.g.* location, device placement (hand, pocket), user movement speed/direction, statistics of observed signals, and density of nearby transmitters. The design must also consider the overhead (in energy and bandwidth) required to upload measurements to the monitoring agency (in real-time). This means that we must identify the minimum form of measurement data per user that can be aggregated to produce a real-time map of spectrum usage. One can also explore novel data compression algorithms that compress spectrum reports on the fly without missing significant events.

Handling Measurement Noise. Our initial results show that the use of mobile monitoring devices introduces noises into monitoring data. This, combined with potential human operation errors, can affect the accuracy of spectrum usage characterization. One potential solution for handling measurement noise is to take into account noisy data as part of the signal modeling and estimate the actual measurement data. Existing models such as Gaussian process, Bayesian and Kalman filters can take noisy data as input, and produce confidence levels on signal estimations.

6. REFERENCES

[1] `http://www.networkworld.com/article/2174194/smb/la-building--39-s-lights-interfere-with-cellular-network--fcc-says.html`.

[2] `http://sdr.osmocom.org/trac/wiki/rtl-sdr`.

[3] BAHL, P., ET AL. White space networking with Wi-Fi like connectivity. In *SIGCOMM* (2009).

[4] BERNSTEIN, M., BRANDT, J., MILLER, R., AND KARGER, D. Crowds in two seconds: Enabling realtime crowd-powered interfaces. In *UIST* (2011).

[5] BIGHAM, J., ET AL. Vizwiz: nearly real-time answers to visual questions. In *UIST* (2010).

[6] CHEN, H.-S., GAO, W., AND DAUT, D. G. Spectrum sensing using cyclostationary properties and application to ieee 802.22 wran. In *GLOBECOM* (2007).

[7] FAGGIANI, A. A. O. Network sensing through smartphone-based crowdsourcing. In *SenSys* (2013).

[8] FATEMIEH, O., CHANDRA, R., AND GUNTER, C. Secure collaborative sensing for crowdsourcing spectrum data in white space networks. In *DySPAN* (2010).

[9] GEMBER, A., ET AL. Obtaining in-context measurements of cellular network performance. In *IMC* (2012).

[10] HASSANIEH, H., ET AL. Ghz-wide sensing and decoding using the sparse fourier transform. In *INFOCOM* (2014).

[11] HUANG, J., ET AL. Anatomizing application performance differences on smartphones. In *MobiSys* (2010).

[12] IYER, A. P., ET AL. SpecNet: Spectrum sensing sans frontières. In *NSDI* (2011).

[13] KIM, H., AND SHIN, K. G. In-band spectrum sensing in cognitive radio networks: energy detection or feature detection? In *MobiCom* (2008).

[14] LASKA, J., ET AL. Compressive sensing for dynamic spectrum access networks: Techniques and tradeoffs. In *DySPAN* (2011).

[15] LITTMAN, L., AND REVARE, B. New times, new methods: Upgrading spectrum enforcement. Silicon Flatirons Roundtable Series on Entrepreneurship, Innovation, and Public Policy, Feb. 2014.

[16] MEIKLE, R., AND CAMP, J. A global measurement study of context-based propagation and user mobility. In *HotPlanet* (2012).

[17] PATRO, A., ET AL. WiSense: A client based framework for wireless diagnosis (Poster). In *NSDI* (2014).

[18] RAHUL, H., ET AL. Learning to share: Narrowband-friendly wideband networks. In *SIGCOMM* (2008).

[19] RAI, A., ET AL. Zee: Zero-effort crowdsourcing for indoor localization. In *MobiCom* (2012).

[20] RASHIDI, M., ET AL. A NLLS based sub-nyquist rate spectrum sensing for wideband cognitive radio. In *DySPAN* (2011).

[21] SEN, S., ET AL. Can they hear me now?: A case for a client-assisted approach to monitoring wide-area wireless networks. In *IMC* (2011).

[22] THE NATIONAL TELECOMMUNICATIONS AND INFORMATION ADMINISTRATION. An Assessment of the Near-Term Viability of Accommodating Wireless Broadband Systems in the 1675-1710 MHz, 1755-1780 MHz, 3500-3650 MHz, and 4200-4220 MHz, 4380-4400 MHz Bands, October 2010.

[23] YANG, L., ET AL. Enforcing dynamic spectrum access with spectrum permits. In *MobiHoc* (2012).

[24] YANG, Z., ET AL. Locating in fingerprint space: Wireless indoor localization with little human intervention. In *MobiCom* (2012).

[25] YOON, S., ET AL. Quicksense: Fast and energy-efficient channel sensing for dynamic spectrum access networks. In *INFOCOM* (2013).

[26] YUCEK, T., AND ARSLAN, H. A survey of spectrum sensing algorithms for cognitive radio applications. *Commun. Surveys Tuts. 11*, 1 (2009), 116–130.

[27] ZHANG, X., ET AL. Propagation comparisons at vhf and uhf frequencies. In *Radio and Wireless Symposium* (2009).

[28] ZHANG, Z., ET AL. On the validity of geosocial mobility traces. In *HotNets* (2013).