CrowdLoc: Cellular Fingerprinting for Crowds by Crowds

RAVI BHANDARI and BHASKARAN RAMAN, Indian Institute of Technology Bombay
K. K. RAMAKRISHNAN, University of California, Riverside
DEEPTHI CHANDER, Indian Institute of Technology Bombay
NAVEEN AGGARWAL, UIET, India
DIVYA BANSAL, PEC, India
MAHIMA CHOUDHARY and NISHA MOOND, Indian Institute of Technology Bombay
ANEESH BANSAL and MEGHA CHAUDHARY, PEC, India

Determining the location of a mobile user is central to several crowd-sensing applications. Using a Global Positioning System is not only power-hungry, but also unavailable in many locations. While there has been work on cellular-based localization, we consider an unexplored opportunity *to improve location accuracy by combining cellular information across multiple mobile devices located near each other.* For instance, this opportunity may arise in the context of public transport units having multiple travelers.

Based on theoretical analysis and an extensive experimental study on several public transportation routes in two cities, we show that combining cellular information across nearby phones considerably improves location accuracy. Combining information across phones is especially useful when a phone has to use another phone's fingerprint database, in a fingerprinting-based localization scheme. Both the median and 90 percentile errors reduce significantly. The location accuracy also improves irrespective of whether we combine information across phones connected to the same or different cellular operators.

Sharing information across phones can raise privacy concerns. To address this, we have developed an id-free broadcast mechanism, using audio as a medium, to share information among mobile phones. We show that such communication can work effectively on smartphones, even in real-life, noisy-road conditions.

CCS Concepts: • **Human-centered computing → Empirical studies in ubiquitous and mobile computing**;

Additional Key Words and Phrases: Localization, GSM, android, cellular fingerprinting, audio communications

4

The authors thank Information Technology Research Academy (ITRA) for their partial financial support. Other support for this work comes from Prime Minister's Fellowship provided by Science & Engineering Research Board (SERB), Department of Science & Technology, Government of India and Confederation of Indian Industry (CII).
Authors' addresses: R. Bhandari (corresponding author), B. Raman, D. Chander, M. Choudhary, and N. Moond, Department of Computer Science and Engineering, Indian Institute of Technology Bombay, Mumbai 400076, India; emails: {ravib, br, deepthib}@cse.iitb.ac.in, {mmahima09, nisha.moond412}@gmail.com; K. K. Ramakrishnan, Department of Computer Science and Engineering, University of California, Riverside, CA 92521, USA; email: kk@cs.ucr.edu; N. Aggarwal, Computer Science Department, UIET, Panjab University, Chandigarh, India; email: navagg@gmail.com; D. Bansal, A. Bansal, and M. Chaudhary, Department of Computer Science and Engineering, Punjab Engineering College, Chandigarh 160012, India; emails: {divya}@pec.edu.in, {aneeshbansal.0014, meghaa91}@gmail.com.

https://doi.org/10.1145/3155326

ACM Reference format:
Ravi Bhandari, Bhaskaran Raman, K. K. Ramakrishnan, Deepthi Chander, Naveen Aggarwal, Divya Bansal, Mahima Choudhary, Nisha Moond, Aneesh Bansal, and Megha Chaudhary. 2018. CrowdLoc: Cellular Fingerprinting for Crowds by Crowds. *ACM Trans. Sen. Netw.* 14, 1, Article 4 (January 2018), 36 pages.
https://doi.org/10.1145/3155326

1 INTRODUCTION

The location of a mobile user is important for many crowd-sensing applications, and the location information is typically tagged along with the crowd-sourced data (Mohan et al. 2008). It is well known that there is a location accuracy versus energy trade-off in using GPS (higher accuracy but high energy) versus the cellular network (low-power but inaccurate). Turning on the GPS on a phone can drain the battery quickly (Constandache et al. 2009; Paek et al. 2011), since it requires acquiring and maintaining a fix on GPS satellites. When using the alternative of positioning smartphones with cellular networks, the location error can be as high as several hundred meters in the median and several kilometers in many cases (Chen et al. 2006). This may not be useful for most scenarios. But, the use of cellular network information is power efficient compared to both GPS and WiFi (Constandache et al. 2009; Thiagarajan et al. 2011). In terms of battery usage, GPS will drain the battery in just 9h at a sampling period of 30s (Constandache et al. 2009), whereas WiFi and GSM can last up to 40 and 60h, respectively. The work in Thiagarajan et al. (2011) reports that at a sampling period of 1s, GSM is 10× more power efficient than GPS and 6× more power efficient than WiFi. While A-GPS can reduce GPS's time-to-first-fix, its steady state power consumption is high as well (Wang et al. 2009).

Another problem with GPS is that it does not always work in the vicinity of tall buildings, crowded buses or trains, inside tunnels, or sometimes in users' pockets (Thiagarajan et al. 2009). In our own experiments on crowded trains (Mumbai 2012; BBC 2015), GPS availability was found to be as low as 14%. Further, most smartphone users in the region of our experimental study do not possess high-end smartphones, which are typically equipped with high-accuracy GPS chips. WiFi-based location can be an alternative in some urban scenarios, but this too has significant coverage holes. In our data, we observed a lack of WiFi 20–26% of the time on roads, and 51% of the time when aboard city trains in Mumbai, India.

In our research, we have focused on crowd-sourcing information, specifically for public transportation commuters. A significant opportunity that arises in this context is of improving the location accuracy, by combining information (crowd-sourcing) across "nearby" mobile devices. For instance, if we could combine the cellular network information across phones of users in a bus, the location error could potentially reduce. This idea is explored in depth in this paper (Section 3) and is depicted in Figure 1. The intuition is that location errors arise due to fading and shadowing of cellular network transmissions. The errors would potentially "cancel out," if we combine information across "nearby" phones. While the original motivation for combining information to improve location accuracy arose in the context of public transportation, the idea could be useful in other settings as well.

While prior work has looked at various aspects of cellular network based positioning (Chen et al. 2006; Thiagarajan et al. 2009; Paek et al. 2011), we believe that the notion of combining cellular information across phones has not yet been explored. Combining signals received by a mobile across multiple cellular towers has indeed been explored previously in Chen et al. (2006) and found to be useful in reducing errors. However, the direction of our work is different: we seek to combine cellular information across multiple nearby phones. This is potentially complementary to combining information at individual mobiles across multiple cellular towers. Further, combining information across phones can potentially smooth out location errors that can arise due to

Fig. 1. CrowdLoc: combining information across multiple nearby phones.

client-side variability. We observe that client-side variability is significant, given the ever-growing variety of smart devices today.

When attempting to combine information across multiple smartphones to improve location accuracy, several questions arise. (1) What exactly is the information to be combined across phones, and how should this be done? What are the privacy concerns to be addressed? (2) What is the location accuracy improvement, if any, and how does this depend on the number of phones across which we combine information? (3) Can we combine information across phones of different cellular operators? How does this compare with combining information across phones of the same operator? (4) How do we go about detecting which phones are "nearby" in the first place, so that we can potentially combine their location information?

In this article, we answer the above questions (Section 4) through extensive analysis of experimental data collected using several smartphones, in two different Indian cities, Mumbai (a large metropolitan city) and Chandigarh (a smaller city with a well planned road network). In this article, we refer to the two cities, Mumbai and Chandigarh as city-A and city-B, respectively. We have collected data in a variety of scenarios from these cities: different types of routes (internal road, highway, suburban train), different times of day, different phones, across different cellular operators, with/without an active data connection. Prior to exploring the combining of information across phones, we first experimentally determined that the accuracy of the inbuilt "network"-based localization mechanism in the Android operating system for smartphones (referred to as NET in this work). It was poor. The fingerprinting mechanism (Chen et al. 2006) performed better in comparison to other localization methods (Haeberlen et al. 2004; Ladd et al. 2005); hence, we explore the combining of fingerprint information across phones.

Our experimental data shows that combining of information across phones is beneficial in reducing the location estimation error: taken across all of our experimental data, the median location error reduces by a factor of 1.3× while the 90%-ile error reduces by 1.9×. We see a reduction in location error by the process of combining information, irrespective of whether the mobiles have a data connection or not.

Any fingerprinting mechanism has to be concerned with the overhead of fingerprinting. To this end, we show that the training overhead in our mechanism is small. More importantly, we observe that there is benefit in combining location information across phones, whether they belong to the same cellular operator, or to different cellular operators. Further, we also explore the effect of training versus testing on different phones. This is important, as in practice, it may not be feasible to collect fingerprints on all different types of client devices. We find that the combining information across phones is even more beneficial for such a scenario, with the overall median location error reducing by a factor of 1.6 and overall 90 percentile error reducing by a factor of 3.5.

This article explores next (Section 5) the related aspect of determining which mobile devices are in the vicinity of one another, which is an important and required step prior to combining information across phones. We design a distributed, client-based approach using audio communication to share information across nearby phones. This mechanism uses a smart-phone's audio hardware (microphone and speaker) to perform *id-free broadcast* of cellular network fingerprint information. That is, each client simply broadcasts "this is (my) fingerprint," without revealing its identity. Since audio is used, there is no identity at the PHY/MAC layers transmitted either, thus retaining the anonymity of the users that are cooperating to achieve this improved accuracy. The broadcast is inherently range-restricted, to about 5m, which acts as a working definition of "nearby" for sharing location information. We carefully design the PHY/MAC layers for the audio communication protocol, so as to achieve reliable and efficient communication even in real-world noisy outdoor conditions. We also designed a simple MAC protocol to enable phones to exchange this audio information.

As in all crowdsourcing solutions, an important concern can be about the feasibility of large-scale participation of users. An altruistic scenario where the user has to keep sharing cellular information continually, incurring some energy costs, may not be sustainable or even realistic. Therefore, in this work, we assume that there is necessarily an engaging incentive mechanism that may have to be tied to tangible/monetary rewards. To illustrate the feasibility of our solution, consider the use case of a commuter standing at a bus stop, wanting to know the Expected Time of Arrival (ETA) of buses on a particular route. In a crowdsensing solution, users inside a bus on the route must participate by sharing their locations. To motivate users to participate, we assume that an incentive mechanism such as described in Yang et al. (2012), Jaimes et al. (2012), Biswas et al. (2015), and Gao et al. (2015) is in place. The incentive by itself could be linked to a subsidy on a metro travel card of the public transportation system, as in INSINC (2012). With a fingerprinting database that can be built over a few weeks; and with CrowdLoc potentially being integrated with popular transportation apps such as Moovit (Moovit 2017), m-Indicator (m Indicator 2017), Ridlr (Ridlr 2017), MyTransport (MyTransport 2017) in cities such as Mumbai, Singapore, New York, we believe that it is possible to scale our solution to city-wide deployments.

To summarize, this article develops, implements, and evaluates a real-world system that demonstrates the benefits of combining cellular information across nearby phones. Our key contributions are: (1) We propose a crowd-sourced framework for combining cellular information provided by co-located mobile phones, to improve location accuracy. (2) We extensively study the effect of heterogeneity (arising from different phone models, operators, data connections) on fingerprinting mechanisms using real-world data. (3) We evaluate the feasibility of sharing cellular fingerprints among co-located mobile phones, in an anonymous fashion, using an audio communication-based mobile application.

2 RELATED WORK

Cooperative localization is well studied in the context of sensor networks (Patwari et al. 2005). In cooperative localization, different nodes find out their locations with respect to one another, e.g., using RF propagation models. In our case, we are not concerned about internode distances or relative positions, but in improving accuracy of absolute positioning by combining information across nearby crowd sensing nodes. Wymeersch et al. (2009) has demonstrated indoor localization using a cooperative approach, in the absence of GPS. It, however, uses UWB radios for internode ranging as well as extensive inter-node communication. Similar cooperation for indoor location is considered in robot networks as well (Howard et al. 2003). The work by Hemmes et al. (2010) uses cooperation to improve dead reckoning errors. Another work by Liang et al. (2016) uses cooperation to improve GPS only and GPS plus NET localization errors. None of these works have considered combining cellular network information for localization.

GSM-based outdoor localization has been studied in-depth in past work. Chen et al. (2006) have examined the feasibility of applying methods from WiFi-based positioning literature to cellular phones, in a metropolitan setting. They use a radio fingerprint consisting of a set of cellular towers and the signal strength from them as seen by the mobile. Using traces collected in Seattle's metropolitan area, they found the fingerprinting mechanism, with a median error of 94m, to be the best among three positioning algorithms, in an area where cell tower density is high. This they attribute to the presence of buildings and other obstacles in an urban environment that leads to formation of unique fingerprints, which in turn improves accuracy. They report that if information is combined across seven cellular towers, then median error lies in the range of 65–134m, and the 95 percentile error goes down to 163m. The work also reports the effect of training using one device and testing on another, a dimension that we also stress upon in our work.

The main difference between Chen et al. (2006) and our work is that we explore combining of information across "nearby" phones, an aspect not explored in Chen et al. (2006). Combining of information across phones could potentially smooth out not only fading- and shadowing-related location errors but also client-side variability: antenna type, receiver calibration, and so on. This is complementary to combining information from multiple cellular towers, as explored in Chen et al. (2006). In our experiments, we sought to obtain information about multiple cellular towers but ran into a practical limitation: we observed that this information is not available on many current smart-phones (Paek et al. 2010).

Nericell (Mohan et al. 2008) also studied the use of GSM for localization, switching to GPS only if an event of interest is encountered. They have also used GSM fingerprinting and hence collected cell-id, signal strength, and GPS values (latitude, longitude) as the training data. A mapping was created between cell-id (with the strongest signal strength) and GPS data. The median localization error and 90%ile error reported in Mohan et al. (2008) were 117m and 660m, respectively, in the city of Bangalore, India. Nericell too does not explore combining of information across multiple phones, which we explore.

Paek et al. (2011) have developed a Cell-ID Aided Positioning System (CAPS), which leverages spatio-temporal consistency in user mobility to aid position estimation. It was observed along four routes that Android's network-based location API (which we refer to as NET, in our work) gave a median error of around 400m and 95 percentile error as large as a few kilometers, while CAPS achieved a median error of less than 75m, while using GPS 4% of the time. While CAPS improves location accuracy by turning on GPS for some intervals of time, we improve location accuracy through combining information across multiple phones.

Transfer of small amounts of data between smartphones over short ranges, using audio communication, has been explored in the recent past. Dhwani (Nandakumar et al. 2013), PriWhisper (Zhang et al. 2014), ANT (Patro et al. 2011) use audio communication in the audible frequency range. Dhwani (Nandakumar et al. 2013) implemented OFDM-based audio communication in the range 6–7KHz and achieved a data rate of 2.4Kbps over less than 20cm range, while using a 24-bit CRC for error detection in noisy conditions. While PriWhisper (Zhang et al. 2014) used a 9KHz carrier frequency and MFSK modulation, to achieve a data rate of 1kbps in the range of less than 0.5cm, Patro et al. (2011) used FSK modulation over 1200–1300Hz frequencies and achieved a data rate up to 8bps within few centimetres range. Unlike the above, in our setting, to combine location information across phones of "nearby" commuters, we need a communication range of a few metres.

Communication over inaudible frequencies (16–22kHz) can span larger distances of a few meters. In this context, Madhavapeddy et al. (2005) used 21KHz as a carrier frequency to achieve a data rate of 8bps over a distance of 3.4m, while Lee et al. (2015) used a chirp BOK modulation scheme over 19.5–22KHz to achieve a data rate of 16bps over a distance of 25m. However, these existing works in communication over inaudible frequencies have used high-end speakers or microphones in their implementations. Moreover, they have considered relatively noise-free indoor/lab settings.

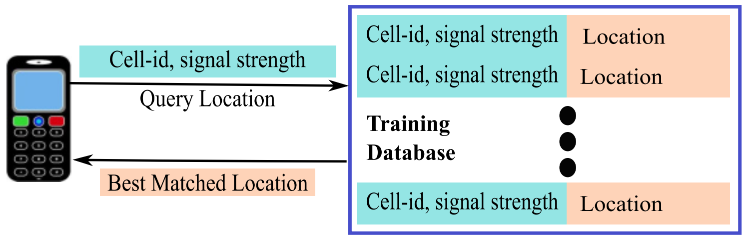

Fig. 2. Cellular Fingerprinting: Training and Testing.

Kannan et al. (2012) proposes an audio-tone counting technique in the 15–20kHz frequency range using MFSK modulation. The implementation is tested on smartphones for 5m in noisy environments such as a bus stop and running bus. However, unlike our work, Kannan et al. (2012) focuses only on detection of mobile phone presence and does not propose a smartphone system for audio communication.

In contrast to these previous works, we have designed and implemented audio communication on smart-phone hardware, in noisy outdoor conditions. Noise would include vehicular noise, people conversing, and so on. In some parts of the world such as India, road-noise has been reported to be very high and this includes vehicular honks (Sen et al. 2010; Mohan et al. 2008). The prior work in the inaudible frequency range have not addressed the issue of noise, e.g., none discuss a robust preamble detection scheme or PHY layer coding. Dhwani (Nandakumar et al. 2013) uses high-pass filtering to nullify noise, OFDM to counter frequency selectivity, a chirp-based preamble for frame synchronization, and basic error correcting codes. However, the communication range of Dhwani is limited to a few centimeters and it operates in the audible frequency range.

In summary, we believe our work is the first to explore the possibility of combining cellular information across phones for improving the location accuracy during outdoor localization, and to implement and demonstrate the same using audio communication in real-world settings.

3 CROWDLOC OVERVIEW

In *CrowdLoc*, we combine cellular network-based location information of "nearby" phones to improve their location accuracy. The motivation behind combining location information across phones is that cellular network-based location is inaccurate because of various fading and shadowing effects on cellular signals. It is likely that two different phones in the vicinity of one another would face different fading and shadowing effects. Thus, we intuitively expect that appropriately combining information across these devices would reduce location error.

What exactly is the information we should combine? This depends on the mechanism each device uses for localization. We initially considered the inbuilt Android mechanism for determining cellular network based location (termed NET (Paek et al. 2011)), to determine each individual phone's location. However, early on in our experiments, we realized that NET performs quite poorly, with a median error as high as 800m in some cases. Hence, we tried the *fingerprint* algorithm (Chen et al. 2006), which uses the tuple $< cell\,ID,\ signal\ strength >$ as a fingerprint to map to a location. This is depicted in Figure 2.

Cellular fingerprinting comprises two phases: a training phase and a testing phase. During the training phase a cellular fingerprint database, which is essentially a lookup table, is constructed. An Android application is used to collect the cellular fingerprint and to record the corresponding GPS coordinates at that location. The $< cell\,ID,\ signal\ strength >$ tuples, along with the corresponding GPS locations, are used to populate the lookup table, as shown in Figure 2. It is to be noted that the *cell ID* uniquely identifies the operator. The lookup table can be stored locally or hosted on a

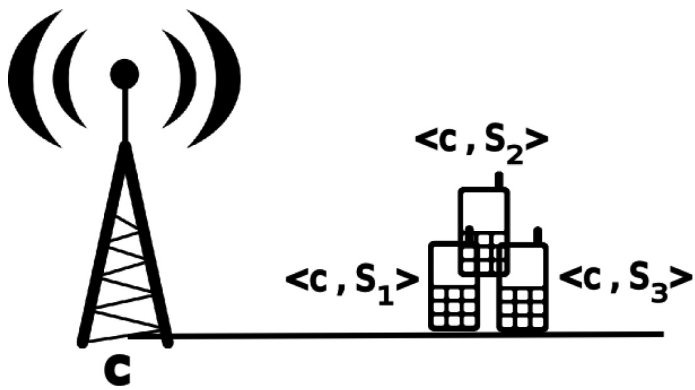

Fig. 3. Co-located phones with same cell-id.

server. During the testing phase, a reverse mapping approach is used. Here, a query comprising only a $< cell\,ID$, $signal\,strength >$ corresponding to an unknown location is sent from a mobile phone to the fingerprint database. In the classical fingerprinting approach (Chen et al. 2006), the server simply looks up (reverse maps) this tuple to the GPS location in the lookup table. If the cell ID does not match any of the cell IDs in the lookup table, then the server discards the query. If the cell ID does match any of the cell IDs in the lookup table, then it retrieves the GPS location corresponding to the closest value of $< signal\,strength >$ in the query and provides this location as the "best matched" location, as shown in Figure 2. The number of trips conducted during the training phase coupled with the density of cell towers will determine the richness of the fingerprint database, and consequently, the accuracy of the location estimated during the testing phase.

Cellular fingerprinting showed better results than NET (details in Section 4) in our experimental study. Hence, we considered this approach for combining information across phones. We now present a theoretical analysis that demonstrates the benefits of our proposed solution of combining cellular fingerprints, and illustrate the same for some simple scenarios.

3.1 Combining Cellular Fingerprints of Co-Located Phones with the Same Cell ID

For simplicity of the theoretical analysis and to focus on the intuition behind our idea, we first consider a scenario where N co-located mobile phones are associated with the same cell ID, c. Let $< c, s_1 >, < c, s_2 > \ldots, < c, s_N >$ represent their cellular fingerprints as shown in Figure 3. The GPS location $< lat_i, lon_i >$, where mobile phone i is co-located with a number of other mobile phones, can be estimated using a reverse mapping of the cellular fingerprint $< c, s_i >$ of *any* of the mobile phones, i. However, this signal strength s_i may vary significantly, not only across time instants but also across phones, primarily due to *independent shadowing conditions* experienced by each of the phones. The presence of shadowing in a wireless environment does not permit a 1:1 relationship between location and signal strength. Therefore, for a lookup table used for the reverse mapping process, the GPS location corresponding to the cellular fingerprint of any single phone, will also vary considerably depending on the signal strength of the chosen phone and can thus result in large errors in estimated location.

For the reverse mapping based approach, is it then possible to obtain a location estimate with a lower variance, in this scenario? We explore a solution where we make use of the fact that there may be multiple phones associated with the same cell ID that are concurrently located at the same location, as in Figure 3. We first define the sample mean ($\bar{s}$) of signal strengths of the N phones as

$$\bar{s} = \frac{s_1 + s_2 + \cdots + s_N}{N}.$$

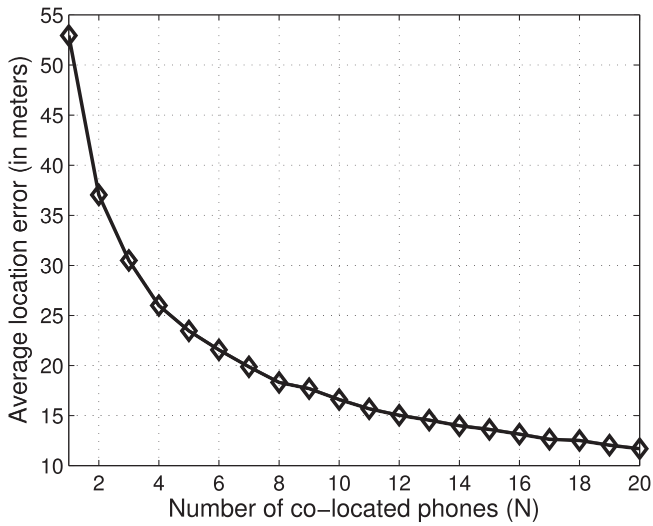

Fig. 4. Single-cell ID case: Average location error (in meters) with increasing number of co-located phones (N).

Since the signal strengths $s_1, s_2 \cdots s_N$ perceived by each of the N phones are caused by independent shadowing conditions, these can be considered to be independent random variables. Consequently, the variance $\sigma_{\bar{s}}^2$ of the sample mean, $\bar{s}$ of the N co-located mobile phones becomes (Papoulis and Pillai 2002)

$$\sigma_{\bar{s}}^2 = \frac{\sigma_1^2 + \sigma_2^2 + \cdots + \sigma_N^2}{N^2},$$

where, $\sigma_1^2, \sigma_2^2, \ldots, \sigma_N^2$, are the individual variances of the signal strengths of the N phones, respectively. In the simple case where $\sigma_1 = \sigma_2 = \cdots = \sigma_N = \sigma$,

$$\sigma_{\bar{s}}^2 = \frac{\sigma^2}{N}.$$

As can be seen from the above equation, the effect of averaging the independent signal strengths of the N "co-located" phones, is to reduce the overall variance of signal strength by a factor of N, when compared to the variance of the signal strength of any one of the N co-located phones. Therefore, to obtain a relatively consistent (across time and phones) representative location for the N co-located mobile phones, one can lookup the GPS location corresponding to $< c, \bar{s} >$ instead of looking up the locations corresponding to any $< c, s_n >$, where $n = 1 \cdots N$.

Figure 4 shows a MATLAB simulation of how the average location error (in meters), varies inversely with increasing N, for 10,000 iterations. The signal strengths in dBm for each phone has been drawn from a normal distribution with mean and standard deviation of −106 and 2dBm, respectively, based on observations made in outdoor wireless environments (Rappaport et al. 1996). A reduced variance in the combined signal strength at a particular location implies that a relatively consistent value for the signal strength is used to retrieve the location from the lookup table. The consistency in the signal strength value in turn translates to a consistency in the location estimate retrieved through the lookup process. Therefore, a direct consequence of the reduction in variance of the combined signal strength with increasing N is a reduction in the variance of the GPS location retrieved from the lookup table.

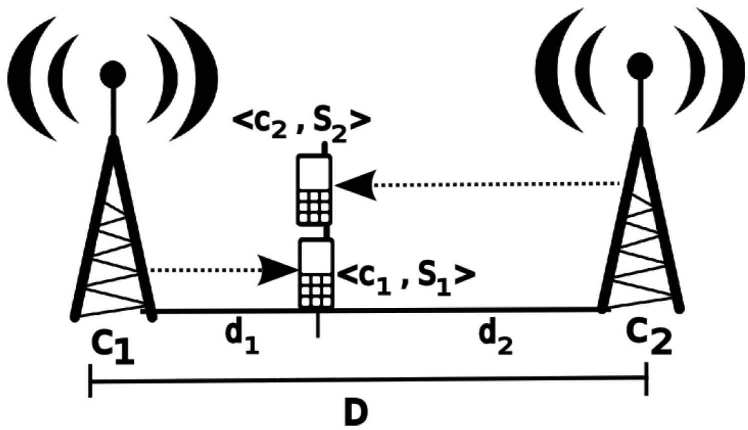

Fig. 5. Co-located phones with different cell IDs in their cellular fingerprints.

3.2 Combining Cellular Fingerprints of Phones with Different Cell IDs

The above analysis shows the implication of combining signal strengths for the single cell ID case. We now consider a more practical scenario, where co-located mobile phones may be associated with different cell IDs of the same cellular operator or of different operators. In this case, combining signal strengths (as in the single cell ID case described above) across mobile phones associated with different cell IDs does not have a logical meaning. We therefore explore an alternate approach, which is a two-step process. We first reverse map the cellular fingerprints to their corresponding locations in the lookup table. The retrieved locations are then combined using any of the algorithms described in Section 3.3. We now explain this approach in detail.

To illustrate the idea, consider a two-cell ID scenario where there are N co-located phones. For simplicity of the theoretical analysis, the N phones are assumed to lie on the line between the two cell towers, as shown in Figure 5. In practice, the mobile phones can be located anywhere in the two-dimensional coverage overlap region of the two cell towers. In the following discussion, this assumption simplifies the two-dimensional localization problem to a distance estimation problem. Out of the N co-located phones, let N_1 phones be associated with cell ID c_1; while N_2 phones are associated with cell ID c_2 and $N = N_1 + N_2$. Let the phones be located at a distance d_1 from cell tower 1 and d_2 from cell tower 2, and let $d_1 + d_2 = D$, the distance between the two towers, as shown in Figure 5.

Now, the cellular fingerprints corresponding to the N_1 phones associated with cell ID c_1 are tuples of the form $< c_1, s_{1,n_1} >$, where s_{1,n_1} is the signal strength measured by each phone n_1 in the group $n_1 = 1, 2 \cdots N_1$. Let the location (it is the distance here) corresponding to the measured signal strength s_{1,n_1} be recorded as r_{1,n_1} in the lookup table during the training phase, for each of the phones $n_1 = 1, 2 \cdots N_1$, respectively. Note that the recorded r_{1,n_1} maybe different from the actual distance d_1 from tower 1, due to shadowing effects. Similarly, the cellular fingerprints corresponding to the N_2 phones associated with cell ID c_2 are tuples of the form $< c_2, s_{2,n_2} >$, where s_{2,n_2} is the signal strength measured by phone n_2 for the phones in $n_2 = 1, 2 \cdots N_2$. Let the location (distance, here) corresponding to the measured signal strength s_{2,n_2} be recorded as r_{2,n_2} in the lookup table during the training phase, for phones $n_2 = 1, 2 \cdots N_2$, respectively. Due to shadowing effects, the distances r_{2,n_2} maybe different from the actual distance d_2 from cell tower 2.

To obtain a representative location estimate of low variance for the N co-located mobile phones, we consider the following approach. We first retrieve the N recorded locations (distances): r_{1,n_1} and

r_{2,n_2} for $n_1 = 1, 2 \cdots N_1$ and $n_2 = 1, 2 \cdots N_2$, respectively, from the lookup table. We then examine if combining these locations (distances) will yield any benefits in reducing the variance of the representative location of the N co-located mobile phones. In other words, will combining r_{1,n_1}, r_{2,n_2} for $n_1 = 1, 2 \cdots N_1$ and $n_2 = 1, 2 \cdots N_2$, using any algorithm, reduce the variance of the estimated distance of the N co-located phones, with respect to cell towers c_1 or c_2? Will the variance of the distance obtained after combining the locations be less than the variance of the distance obtained while considering only the location of each of the individual phones?

To answer these questions, we consider the signal strengths measured at each of the mobile phones. The signal strength measured at any location is primarily a function of distance from the cell tower and any shadowing effects. Consider the following log-distance path loss model with log-normal shadowing (Rappaport et al. 1996):

$$PL(r)[dB] = P\bar{L}(r_0) + 10\alpha \log\left(\frac{r}{r_0}\right) + X_\sigma.$$

Here, $PL(r)$ is the normally distributed path loss in dB measured at a distance r from the cell tower; $P\bar{L}(r_0)$ is the average path loss in dB at a reference distance r_0, α is the path loss exponent (ranging from 2.7 to 3.5dB in urban cellular environments) and X_σ is a zero-mean Gaussian random variable with standard deviation σ, which accounts for shadowing effects in dB. The actual signal measured at distance r from the cell tower is therefore log-normally distributed (normal in dBm) and is given by $P_t[dBm] - PL(r)[dB]$, where $P_t[dBm]$ is the transmit power (in dBm) at the cell tower. The model for the log-normally distributed signal measurement, $s(r)$ (in Watts), at the distance r, can be simplified to the linear scale as follows:

$$s(r) = \frac{K}{r^\alpha},$$

where the constant K accounts for cell tower transmit power, antenna gains, RF signal frequency and average channel attenuation. In the above equation, it can be shown that, if the signal strength $s(r)$ measured at a location follows a log-normal distribution $\log N(\mu, \sigma^2)$, the corresponding distance r from the cell tower also follows a log-normal distribution given by Papoulis and Pillai (2002):

$$f_r(r) = \frac{\alpha}{r\sigma\sqrt{2\pi}} \exp^{-\frac{\left(\log(r) + \frac{\mu}{\alpha} - \log(K)\right)^2}{2\frac{\sigma^2}{\alpha^2}}}. \tag{1}$$

From Equation (1), the mean $E(r)$ is given by $\frac{-\mu}{\alpha} + \log(K)$, and the variance $Var(r)$ is given by $\frac{\sigma^2}{\alpha^2}$.

In Figure 5, let the N_1 phones associated with c_1 have log-normally distributed signal strengths from the distribution $\log N(\mu_1, \sigma_1^2)$ and let the N_2 phones associated with c_2 have log-normally distributed signal strengths from the distribution $\log N(\mu_2, \sigma_2^2)$. Without loss of generality, if we assume the same transmit powers at the two cell towers, from equation 1 the locations of the N_1 phones with respect to cell tower c_1 follows a log-normal distribution with mean and variance given by

$$E(r_{1,n_1}) = \frac{-\mu_1}{\alpha} + \log(K)$$

and

$$\sigma^2_{r_{1,n_1}} = Var(r_{1,n_1}) = \frac{\sigma_1^2}{\alpha^2},$$

where signal strengths s_{1,n_1} follow a log-normal distribution $\log N(\mu_1, \sigma_1^2)$ for $n_1 = 1, 2 \cdots N_1$. For distance D between cell towers c_1 and c_2, assuming the towers are located on a line (Figure 5), the distance of the N_2 phones with respect to cell tower c_1 is given by $r_{1,n_2} = D - r_{2,n_2}$, for

$n_2 = 1, 2 \cdots N_2$. Using this relation, the locations of the N_2 phones with respect to cell tower c_1 then follows a log-normal distribution with mean and variance given by

$$E(r_{1,n_2}) = D - \left(\frac{-\mu_2}{\alpha} + \log(K)\right)$$

and

$$\sigma^2_{r_{1,n_2}} = Var(r_{1,n_2}) = \frac{\sigma_2^2}{\alpha^2},$$

respectively, where signal strengths s_{2,n_2} follow a log-normal distribution $\log N(\mu_2, \sigma_2^2)$ for $n_2 = 1, 2 \cdots N_2$.

CrowdLoc potentially combines these locations (distances, in the above discussion) from the lookup table to obtain a representative location for the N mobiles. In the illustrative scenario presented in Figure 5, we combine r_{1,n_1} and r_{1,n_2} for $n_1 = 1, 2 \cdots N_1$ and $n_2 = 1, 2 \cdots N_2$ and $N_1 + N_2 = N$, as follows:

$$\bar{r} = \frac{\sum_{n_1=1}^{n_1=N_1} r_{1,n_1} + \sum_{n_2=1}^{n_2=N_2} r_{1,n_2}}{N}.$$

Since r_{1,n_1} and r_{1,n_2} are independent random variables, the variance of their sum equals the sum of their individual variances. Therefore, if we consider a simple combining algorithm that computes the final location estimate as an average (centroid) of the looked-up locations, then the variance of the estimated location becomes (Papoulis and Pillai 2002):

$$\sigma^2_{\bar{r}} = \frac{\sum_{n_1=1}^{n_1=N_1} \sigma^2_{r_{1,n_1}} + \sum_{n_2=1}^{n_2=N_2} \sigma^2_{r_{1,n_2}}}{N^2}.$$

In the above equation, if we set $\sigma^2_{r_{1,n_1}} = \frac{\sigma_1^2}{\alpha^2}$ to be the same for all $n_1 = 1, 2 \cdots N_1$, and $\sigma^2_{r_{1,n_2}} = \frac{\sigma_2^2}{\alpha^2}$ to be the same for all $n_2 = 1, 2 \cdots N_2$, then the variance of the combined location $\bar{r}$ becomes

$$\sigma^2_{\bar{r}} = \frac{N_1 \frac{\sigma_1^2}{\alpha^2} + N_2 \frac{\sigma_2^2}{\alpha^2}}{N^2}. \tag{2}$$

As can be seen from Equation (2), $\sigma^2_{\bar{r}}$ decreases with increasing N. We perform a MATLAB simulation to study the error in the estimated combined location with varying number of phones N for a case of $N_1 = N_2 = N/2$ assuming a log-normal path loss model with $\alpha = 3.5$, and for the mobiles located at distances $d_1 = d_2 = 500$m (as in Figure 5) from towers c_1 and c_2, respectively. We perform our simulation over 10,000 iterations, for a 950MHz RF signal, for $P_t = 20W, \sigma_1 = \sigma_2 = 2dB$ and with μ_1, μ_2 determined using the log-distance path loss model. The standard deviation of estimated location decreases as N increases, and this in turn translates to reduced average location error for the same scenario, as depicted in Figure 6. As expected, the average location error is higher and remains nearly constant, if we do not combine cellular fingerprints across phones.

Note that although we have illustrated the advantage of combining cellular fingerprints across multiple phones using the log-normal shadowing model due to its practical suitability and analytical simplicity, the results can be generalized for any propagation model. This is because the advantage of combining follows directly from the sole assumption that the signal strengths measured by each of the phones at a location are independent of each other, as observed in practice. Other factors affecting signal measurements, such as heterogeneity across mobile phones, variations in path loss exponents will further cause the signal strengths measured by different mobile phones at a location to be independent of each other, causing the analysis to be valid in such conditions as well.

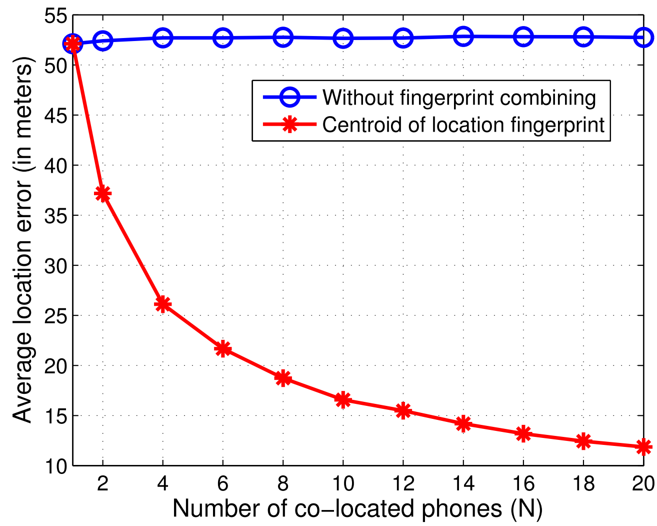

Fig. 6. Two cell ID case: Average location error (in meters) versus number of co-located phones (N).

3.3 Approaches to Combine Cellular Fingerprints

Having analysed the benefits of combining fingerprint information in Sections 3.1 and 3.2, we now present a few approaches that can be used to combine cellular fingerprint information across co-located phones. The algorithms presented here can be implemented easily and yield location estimates of comparable accuracies.

Centroid: Given the individual location estimates L_i of N devices, a straightforward approach to combine their location information is to take the centroid of the set $\{L_i\}$. This gives equal weightage to all the location estimates.

In Equation (2), the variance of the centroid $\sigma^2_{\bar{r}}$, may not always be less than $\sigma^2_{r_{1,n_1}}$ or $\sigma^2_{r_{1,n_2}}$ for $n_1 = 1, 2 \cdots N_1$ and $n_2 = 1, 2 \cdots N_2$. For instance, consider the case where $\sigma^2_{r_{1,n_1}} \approx 0$, $N_1 = 1$ and $N_2 >> N_1$. For high $\sigma^2_{r_{1,n_2}}$ values, $\sigma^2_{\bar{r}} \approx \frac{\sigma^2_{r_2,n_2}}{N_2}$, which may still be a high value compared to $\sigma^2_{r_{1,n_1}}$. Therefore, even though averaging of locations brings down the location variance with increasing N, to further reduce the location variance (and hence the location error) of the centroid location, we may need to suitably combine only the locations of phones, which by themselves have low location variances. However, due to the mobility inherent to the practical setting, we may not be able to estimate the location variance of a phone at a particular location. Therefore, we decided to instead use signal strength as an indication of the location variance (and therefore of the location error) as follows: a mobile measuring the highest signal strength, possibly has the lowest location error. This is based on the intuition that: in a log-distance path loss model, a high (low) signal strength reflects a greater (lesser) proximity to the cell tower, and therefore lower (greater) location uncertainty (and error), with respect to the cell tower. Therefore, based on this intuition, we implemented the following alternate approaches.

BestSig: In *BestSig*, we take the location estimate to be the location corresponding to the phone that has the best signal strength to its cellular tower. In case of a tie between phones, we take the centroid of their locations as the final location estimate.

GoodSig: In our data, we found that there was correlation between signal strength and error in fingerprinting-based positioning. However the correlation was weak (Pearson correlation coefficient around −0.3). So, we considered a third approach *GoodSig*, where we take the centroid location of all the phones that have a signal strength within a threshold T of the phone with the best signal strength. We experimentally arrived at a threshold T=10dB, and discovered that varying this threshold had a marginal effect, as algorithms performed similarly.

The *opportunity* for combining cellular information across phones is available in plenty in a number of scenarios. The specific scenario that motivated us to consider the notion of combining information, is that there are a number of devices in a public transportation unit such as a bus. But combining of information need not be restricted to this scenario and is more generally applicable.

An important point to note here is that during the process of combining of cellular network information, a phone need not reveal its identity, as this identity information is irrelevant to the functioning of the mechanism. Our mechanism also does not require any per-phone history information.

4 MAKING CROWDLOC WORK IN PRACTICE

We now present our experiment-driven design of CrowdLoc. The primary metric we are interested in is the error in reported location. For each data point in our experimental run, we compare the location as determined by an algorithm with the ground truth of GPS-reported location.

We start with a description of our experimental data (Section 4.1). Prior to evaluating the effectiveness of combining information from multiple phones, we first present results (Section 4.2) comparing GSM fingerprinting based location determination with the location given by the inbuilt Android API (referred to as NET in this article). We find that the inbuilt mechanism on Android phones (NET) performs much worse than GSM fingerprinting. Next, we characterize the amount of training required for fingerprinting to be effective (Section 4.3). We then evaluate the effectiveness of combining information across phones, with training and testing on the same phone (Section 4.4), as well as on different phones (Section 4.5). We also explore the effectiveness of combining information across phones of the same versus different cellular operator (Section 4.6).

4.1 Experimental Setup and Data Collection

Since our original motivation for location determination was for crowdsourcing information from public transportation commuters, we collected our data while traveling on buses. We collected data along two bus routes in city-A and along one route in city-B. All the phones used in the experiment had Android 4.0+ as their operating system and had SIMs fpr GSM cellular networks. We developed an Android app that collected both GPS and GSM cellular information at the same time.

GPS-based position information was collected as the ground truth. For each data point, note that we have GPS-reported ground truth from more than one phone (as many as four). The GPS location as reported by each of the phones may not be exactly the same, due to error in GPS itself. Hence, we take the centroid of the available GPS location readings at a given data point, as the ground truth location for that data point.

For GSM, Android has a network-based localization option, which we refer to as NET. The work in Paek et al. (2011) reports that NET uses cell-tower triangulation to determine the phone's location. Alongside the latitude and longitude obtained from NET, we have logged the GSM network parameters of cell-id and signal strength. Here cell-id refers to the current cell tower to which phone is attached at that time instant. We also attempted to record information about neighbouring cell towers, but it turned out that this information is not available universally (Paek et al. 2010); we found that its availability appears to depend on both the phone in use as well as the operator. Among the set of phones available with us, only Micromax A63 and Google Nexus 4 were able to provide us neighbouring cell information, and even with these not for all cellular operators.

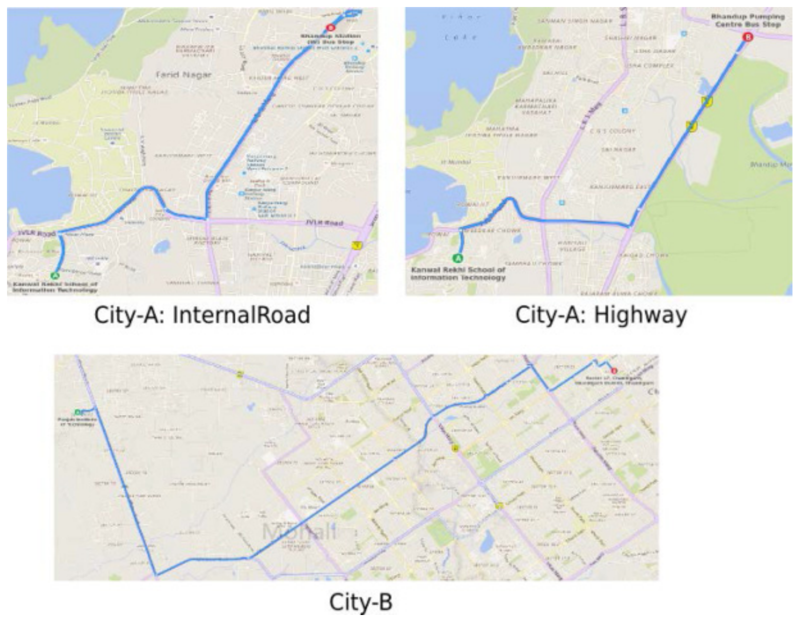

Fig. 7. Bus routes in City-A and City-B for collecting data.

So, we decided to not rely on neighbouring cell towers for our calculation of fingerprinting based location.

In each data collection trip, the GPS and GSM data were logged once every 2s. In each trip, we collect data simultaneously on more than one phone, since our primary purpose is to explore the combination of location information across devices. Data collection in the app was started by manual press of a "start" button, near simultaneously on the group of phones used for that data collection run. Thus the data collected on the phones was time-synchronized. Due to the manual step of button press, there could be a synchronization error of 1–2s, but this is negligible for our purpose of location determination as the bus is not expected to move much in this short time. The above time synchronization step is useful for the following reason. In the initial test runs, we observed that there were many instances where one or more phones in the group lost GPS connectivity. In such scenarios, the fact that the phones were roughly time-synchronized enabled us to use the GPS reading from other phones in the group within the same data collection run.

4.1.1 City-A Data. Four phones were used to collect data on city-A's roads. The phones used were Micromax A94, Samsung S3 Neo, Samsung Note3 and Samsung S3. They are referred to as Phone 1, 2, 3, and 4, respectively, in this article. We denote the SIM cards used in these phones as SIM1x, SIM2y, SIM3y, and SIM4z, respectively. SIM2y and SIM3y belong to the same cellular network provider while the other two were different: we thus use three different network providers for our data collection. The data collection spanned about 9 months. Two routes were selected for data collection in city-A. One route was along a busy internal city road with office buildings and malls along its route. Another was along an expressway and had a smaller concentration of buildings along the road. The two routes are labelled "InternalRoad" and "Highway" in the rest of the article and are shown in Figure 7.

The round-trip distance of the InternalRoad route is 9km and that of the Highway route is 13.5km. For both routes, we collected experimental data both with and without cellular data (3G)

Table 1. Number of Data Collection Trips in City-A

Data	InternalRoad	Highway	Suburban Train
off	13	11	-
3G	30	9	7
2G	12	-	-

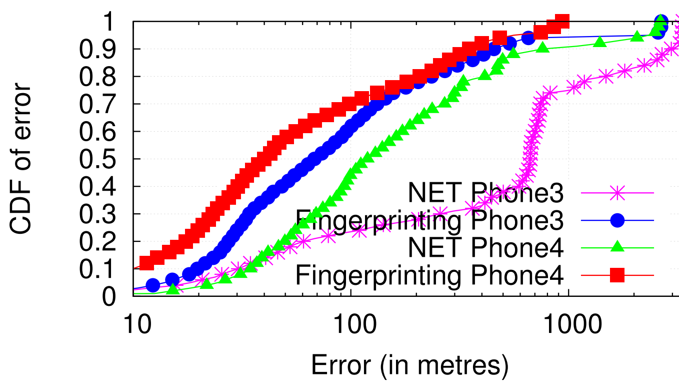

Fig. 8. InternalRoad, data turned off: NET vs GSM FP.

turned on. We considered these variations of cellular data connection for the following reasons. First, users do not always turn on cellular data, as this consumes battery and could also eat into the user's data plan. Further, presence of cellular data could potentially affect the error in location as determined by NET. For the InternalRoad route, we additionally collected data with only a 2G cellular data connection, to examine if there is a difference due to the type of mobile data connection.

City-A has another popular mode of public transportation, viz. suburban train network. We have collected over 200km of data in suburban trains from one stretch of the city to another. For this purpose, we used eight phones with five different operators and kept cellular data (3G) turned on. Table 1 summarizes the number of trips undertaken for each of the above cases.

4.1.2 City-B Data. A round-trip bus route with a length of 14 kilometres was selected for data collection in city-B. Primarily two phones, namely Moto E and Moto G were used to collect data from 15 trips. Apart from this, we also had four data collection trips with three phones: Samsung Grand, Samsung Prime, and Samsung S4. We used this data to examine the effect of one phone using another phone's fingerprint database to determine its location.

4.2 NET vs Fingerprinting (FP)

Prior to exploring combining of information across phones, we first explore as to which mechanism is effective for location determination within a single phone. Given the importance of location determination and the maturity and ubiquity of Android phones, we expected that the inbuilt mechanism in Android, called NET, would be effective. We compare this with a standard GSM fingerprinting (FP) mechanism for location determination. As mentioned earlier, a location fingerprint consists only of one cell tower's id and signal strength.

Figure 8 depicts CDF of errors for two phones (the results for the other two phones were similar and are omitted for clarity). We calculate the CDF as follows. First, we run our algorithms on all the test trips. We then find the error in location obtained through FP and groundtruth location

Table 2. Improvement = Err_{NET}/Err_{FP}

Route	Data	Impr. in median error	Impr. in 90%ile error
InternalRoad	off	3.8	2.3
Highway	off	7	11.5
InternalRoad	3G	1.9	1.2
Highway	3G	3.8	2.9
InternalRoad	2G	2.4	0.6
Suburban Train	3G	0.9	2.3

Table 3. NET vs GSM FP in City-B

	Median error(m)	90%ile error(m)
NET	4,663	10,471
Fingerprinting	216	1,230

(obtained through GPS). These results are for the *InternalRoad* route in the scenario where mobile data was turned off. It can be observed that FP performs far better than NET. For example, for phone-3, the median error is 9.8 times better and the 90 percentile error is 6.5 times better, for fingerprinting compared to NET.

Similar results were obtained for other cases: the *Highway* route, and with mobile data connection turned on. The overall improvement factor in the location error, by using fingerprinting over NET is summarized in Table 2. The maximum improvement can be seen in the cases where data is turned off, as NET tends to perform poorly in these cases.

While the above results are for our city-A data, for city-B we have collected 15 trips each on Moto E and Moto G. Since the signal strength of Moto G remained constant, we discarded its data during the fingerprinting process. Table 3 shows that fingerprinting performs significantly better than NET in this case as well. In this data, in fact, we observed that the location reported by NET changed only twice during the whole journey, thus resulting in high location error.

It is unclear as to why NET performs so poorly as compared to fingerprinting, across so many different scenarios. The implication of this is that the inbuilt Android cellular network location mechanism has a significant scope for improvement. The implication of this for us is that, going forward, we use the fingerprinting mechanism as the location scheme for a single phone, and look to improve it further by combining information across phones.

We have also compared the efficacy of fingerprinting across the scenarios: without mobile data, 2G connection, and 3G connection. This comparison for InternalRoad route is shown in Figure 9. Here, only data from two phones is shown, for clarity. We observe here that there is no definite pattern or correlation between fingerprinting and presence/type of data connection. Similar results were found for the Highway route. Thus the performance of fingerprinting is independent of whether mobile data is turned on or off. This is intuitive, as the fingerprinting mechanism takes into consideration the cell tower id and signal strength, and presence or type of data connection has little effect on these.

4.3 Fingerprinting: Training Overhead

In any fingerprint-based mechanism, the training overhead is a concern. How much training is required in our case? We explore this now. The training used for the fingerprinting was varied from one trip to a maximum of up to four trips. The CDF of the error for InternalRoad, for different

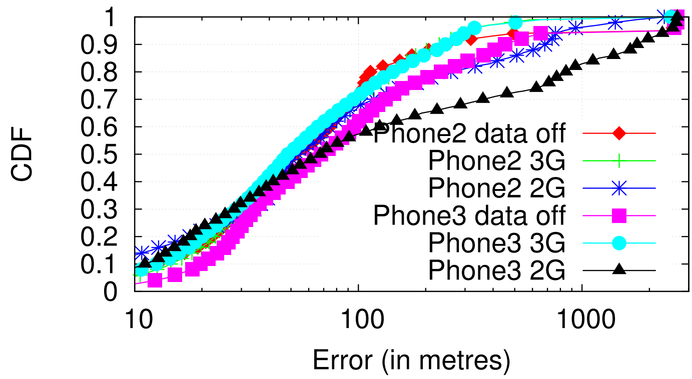

Fig. 9. InternalRoad: data off vs 2G vs 3G.

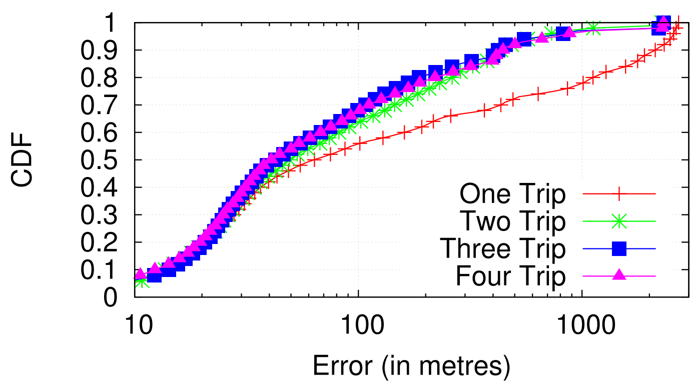

Fig. 10. Effect of Training: InternalRoad: Phone 1.

Table 4. Cell Towers, Fingerprint Density (InternalRoad)

Training	Avg. Cell towers/km	Avg. Fingerprints/km
One Trip	0.9	6.7
Two Trip	2.1	14
Three Trip	2.7	20.3
Four Trip	2.9	23.4

amounts of fingerprinting training, are shown in Figure 10 for phone-1 (the results for the other three phones are similar). It can be observed that there is a significant improvement in moving from one-trip training to two-trip training, but lesser improvement with further training data. For our experiments on both InternalRoad and Highway, we found training using two trips to be sufficient for all practical purposes: we use two-trip training in our results in the rest of the article.

The change in accuracy with varying number of trips used for training, depends upon the number of cell towers present in the training set. We extract this information from our data: Table 4 shows the density of unique cell towers seen, as well as the fingerprint density, averaged across all four phones, as a function of the training set size. We see that there is significant improvement in these metrics as we go from one-trip training to two-trip training, but lesser improvement with additional training trips. This is consistent with Figure 10 where the fingerprinting-based location

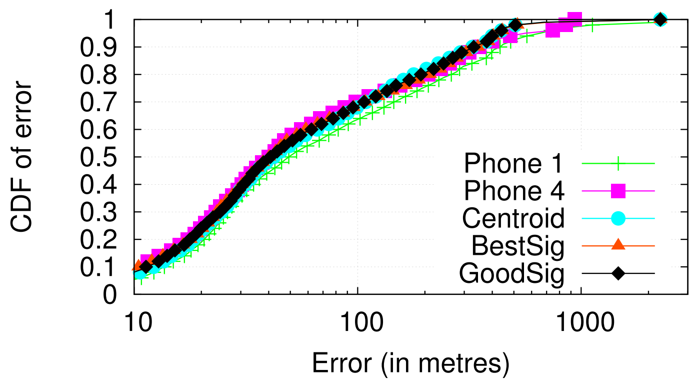

Fig. 11. Combining phones 1, 4: InternalRoad, data turned off.

accuracy increases significantly from one-trip training to two-trip training, but not significantly with additional training data.

Now, is a requirement of two-trip training acceptable? If each user or phone-model is required to collect fingerprints before using it, then this is a significant overhead. However, as we shall see in Section 4.5, we can effectively use training databases across different phones. Hence, a two-trip training requirement is not much, given that thousands of commuters traverse a city's routes everyday.

4.4 Single Phone Versus Multiple Phones

We now explore the effectiveness of combining location information across nearby phones, to potentially get a more accurate location estimate. We have explored the three algorithms presented earlier: Centroid, BestSig, and GoodSig. We first look at the effectiveness of combining information across a pair of phones. Figure 11 compares the CDF of location errors, of single phone fingerprinting with the combination of phones 1 and 4. We see that there is a noticeable improvement in the median location error, as well as the 90 percentile location error, especially for Phone-1. We also see that the three algorithms perform more or less similarly. On closer look at the data, we observed that the *GoodSig* algorithm performs slightly better than the other two, especially towards the tail of the distribution.

We have observed the benefits of combining information across pairs of phones is not universal in all situations. Figure 12 compares the CDF of location errors for the combination possibilities of phones 1 and 2, as well as phones 3 and 4. We see that when combining information across phones 1 and 2, there is little benefit. However, the performance does not degrade either, due to combining information across these phones.

Now, in several situations such as in a bus, there will be many more than two phones whose location information could be combined. We next look at the benefit of combining information across more than two phones. Figure 13 shows the CDF of location errors when we combine information across three and four phones. We see that median location error as well as 90 percentile error improve further, with increase in number of phones across which we combine information. It is thus beneficial to combine information across a large set of phones.

Figures 14 and 15 show the benefit of combining information across phones with 3G data connection and a 2G data connection, respectively. We see that we continue to benefit from combining information across phones, in terms of reduction in median as well as 90 percentile location error.

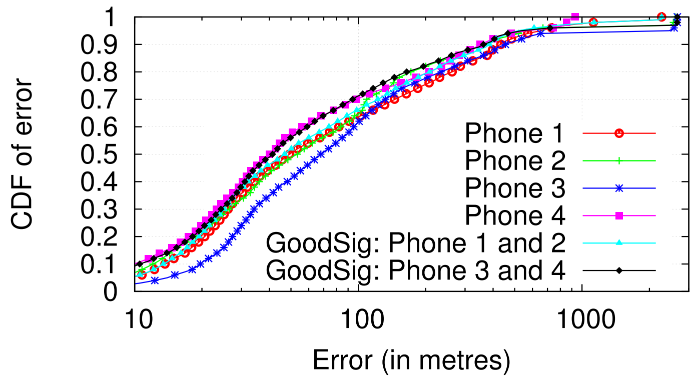

Fig. 12. Combining phones 1, 2; phones 3, 4: InternalRoad, data turned off.

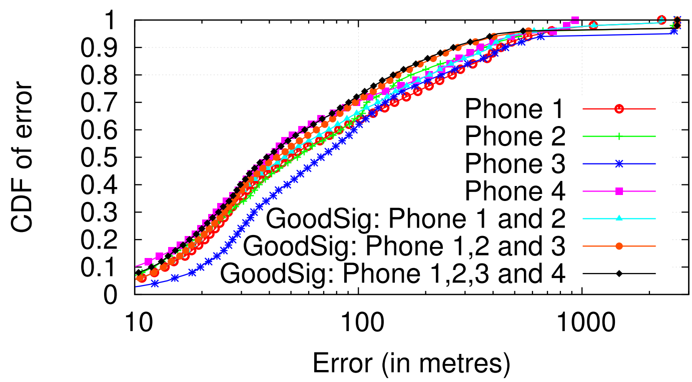

Fig. 13. Combining info across more phones: InternalRoad, data turned off.

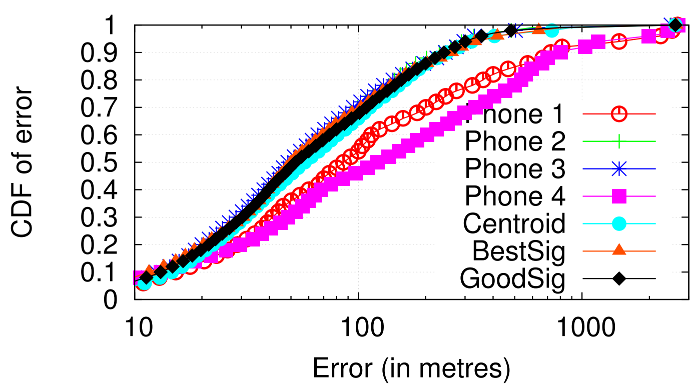

Fig. 14. Combining all phones: InternalRoad, 3G.

In the case of 3G, we see that phones 1 and 4 have relatively large errors when using individual phone information. Likewise, phone 3 has relatively large location errors in our 2G data. Note that in such scenarios, combining information across phones is especially beneficial: the reduction in 90% percentile errors in such cases is over a factor of 5.

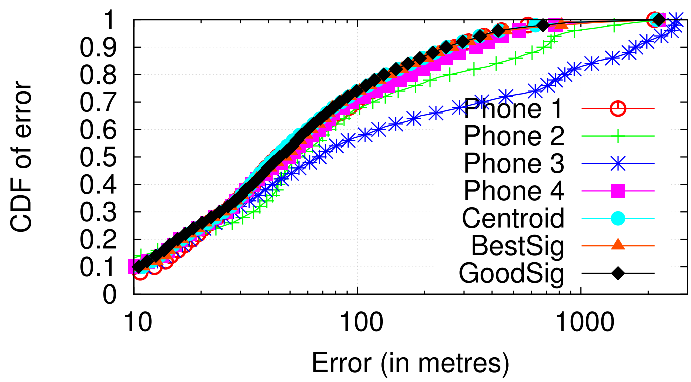

Fig. 15. Combining all phones: InternalRoad, 2G.

Table 5. Improvement = $Err_{1\,phone\,FP}/Err_{4\,phone\,FP}$

Route	Data	Impr. in median error	Impr. in 90%ile error
InternalRoad	off	1.3	1.5
Highway	off	1.4	1.3
InternalRoad	3G	1.3	2.3
Highway	3G	1.3	1.9
InternalRoad	2G	1.3	2.4
Suburban Train	3G	1.8	1.7

Table 6. Improvement = $Err_{NET}/Err_{4\,phone\,FP}$

Route	Data	Impr. in median error	Impr. in 90%ile error
InternalRoad	off	5.0	3.5
Highway	off	9.9	15.3
InternalRoad	3G	2.5	2.8
Highway	3G	4.9	5.6
InternalRoad	2G	3.1	1.4
Suburban Train	3G	1.7	3.9

Table 5 summarizes the benefit of combining information across phones; the performance improvement is measured as a factor of reduction in the median error and the 90 percentile error. We see that factor of improvement is especially significant in the 90 percentile error: combining information across phones is effective in reducing location error in the tail of the distribution.

To get an idea of the overall gain, Table 6 summarizes the error reduction ratios comparing the location error in NET versus the location error in the case of combining fingerprinting-based location information across four phones. As can be seen from the table, the factor of improvement in error is over 4 in most cases, and as high as 15!

Combining all our data, across InternalRoad and Highway, and across all four phones, Figure 16 plots the CDF of the location error in three cases: NET, fingerprinting using only single phone information, and combining information across multiple phones. Comparing single phone versus

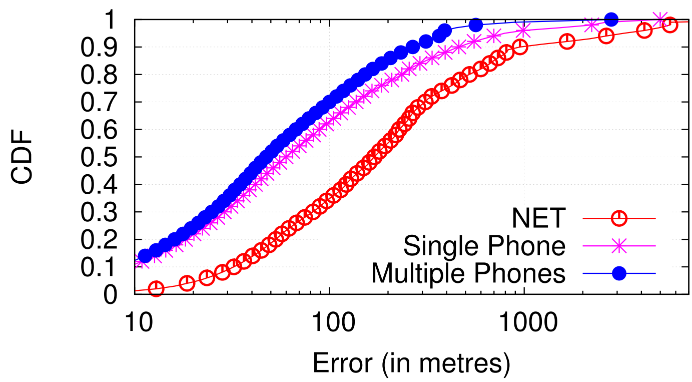

Fig. 16. Overall comparison: NET vs single-phone vs four-phones.

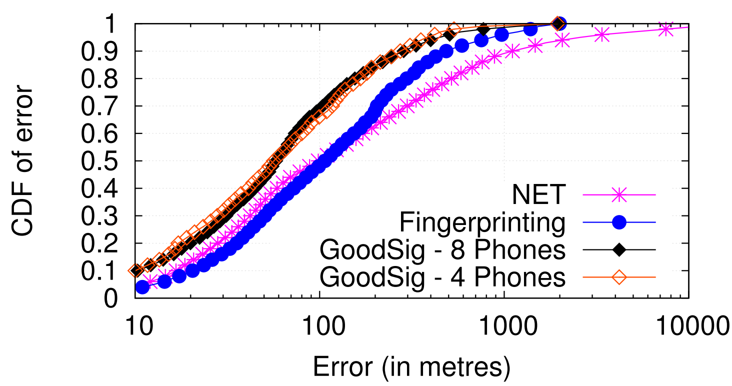

Fig. 17. City-A suburban train: overall improvement.

multiple phones, the overall median error improvement factor is 1.3 while the 90 percentile error improves by a factor of 1.6.

We saw similar benefits of combining information in case of City-A's suburban train network. We show in Figure 17 the overall improvement across all the phones, when two trips are used for training. The reduction in error when using multiple phones can be clearly observed over NET. The factor of improvement in median error is 1.7×, and 3.9× for the 90 percentile error, when combining information across four phones. We also note from the figure that combining information across all eight phones does not significantly reduce the error further, compared to combining across four phones; i.e., most of the gains are achieved by combining across just four phones.

It is to be noted here that by combining information, we have been able to bring the median error below 50m for bus and below 60m for suburban train. For the sake of completeness, the ground-truth GPS traces that we had collected had a median error of 13.6m as deduced from the accuracy values reported by Android API (Google 2017). Android reports these values based on the assumption that errors due to GPS follow normal distribution, which we have also used in deriving the median error.

An alternative view of the benefit of combining information across phones is the following. In the fingerprinting-based location scheme, we could have "outages" in location information: situations where there is no cell-id match in the fingerprint database. In such situations, the best we can do (in the case of single phone) is to use the last known location. In our InternalRoad and

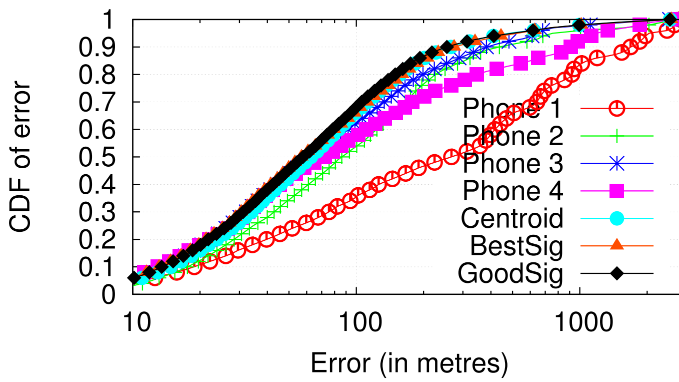

Fig. 18. Training & testing on different phones: InternalRoad.

Highway routes, we found that we had 12% and 11% outages respectively. Such outages contribute to location errors. When combining information across phones, it is unlikely that all the phones will have an outage simultaneously. In fact, in our data, there was no such situation: i.e., there was 0% outage when combining information across four phones. Thus in terms of the outage metric too, combining information across phones has significant benefits.

Note that all the evaluations carried out span large geographical areas (around 9–13.5km for bus routes and around 200km for train routes, as mentioned earlier). We would also like to mention that, prior to working on bus/train route fingerprints, we had validated the efficacy of combining cellular fingerprints by traversing the area of our campus. The performance of CrowdLoc was superior to NET and single-phone fingerprinting, and results similar to those in Figure 16 were obtained. This in fact motivated us to further validate our approach city-wide, leading us to conduct experiments on bus/train routes due to their ease of accessibility and scale.

4.5 Training and Testing on Different Phones

An important aspect of the fingerprinting mechanism is the fingerprint database. Thus far, we have assumed that there is a phone-specific fingerprint database. In practice, assuming that we have a per-phone fingerprint database for all locations is unrealistic. This is especially true given the wide and growing variety of smart-phones in the market. It is thus important to explore the scenario where the fingerprinting mechanism uses a training database from another phone. What is the benefit of combining information across phones in such a setting? We explore this now.

To examine the possibility of phone-1 using another phone's training database, note that we cannot simply use the data described in the prior sections. This is because phone-1 has a unique cellular operator: its cell-ids will not match with any cell-id of another phone's training database. Thus, we collected further data for six trips on the InternalRoad route, where we swapped the SIM cards of phone-1 and phone-2, and likewise swapped the SIM cards of phone-3 and phone-4. We use these six trips for testing the fingerprint mechanism, and the earlier data for training. Note that, in such a scheme, phone-1 would end up using phone-2's training database (collected with the same SIM/operator though) from the earlier data, and vice versa. Likewise, phone-3 would end up using phone-4's training database and vice versa.

Figure 18 shows the CDF of location errors for the four individual phones, as well as the three algorithms for combining information across (all four) phones. In all cases, the fingerprinting database was of a different phone as the phone trying to determine its location, as described above. We see that the benefits of combining information across phones is more significantly visible now.

Table 7. Testing & Training on Diff. Devices: City-B

Testing phone	Training phone	Median(m)	90%ile(m)
Samsung S4	Samsung Grand	168	5,182
Moto E	Samsung Prime	1,717	1,727
Moto G	Samsung S4	57	1,844
All phones	(as above)	106	1,834

We can see significant reductions in the median location error as well as the 90 percentile location error. The latter especially comes down from around a kilo-metre or more in the case of phones 1 and 4, to under 200m; this would be a significant improvement for a variety of location-based applications.

Taken across all the phones of Figure 18, we observed that the combining of information across phones improves the median error by a factor of 1.6, compared to the single-phone fingerprinting case. The improvement in the 90 percentile error was even higher: a factor of 3.5.

In City-B too, two trips each were undertaken for training and testing. Table 7 shows the results for single phone evaluation as well as combination of all the three phones, with the given mapping of testing and training phones. We see significant improvement in the median error in the case of Moto-E, and in the 90 percentile error for Samsung-S4. Moto-E had a high location errors in the single phone case due to very few fingerprint updates during the journey; however, the combination of information across phones is able to reduce the location error significantly.

4.6 Same Operator Versus Different Operator

Is it beneficial to combine information across phones only when they belong to different cellular operators? Or is there benefit in combining information across phones of the same operator as well? We have used four SIMs (SIM1x, SIM2y, SIM3y, SIM4z) in our City-A's data collection, with two SIMs (SIM2y, SIM3y) belonging to the same operator. To compare the combination of information across phones of the same operator, we have considered the following two cases: (case-1) compares data from phone-2 and phone-3 (same operator: SIM2y, SIM3y) versus data from the same pair of phones (different operators: SIM1x, SIM4z), (case-2) compares data from phone-1 and phone-4 (same operator: SIM2y, SIM3y) versus data from the same pair of phones (different operators: SIM1x, SIM4z). In each case, each phone used its own training data set. On plotting the CDF of the location errors for the two pairs of phones, we observed that the cases of same operator or different operator were not distinguishable in terms of location error. This is shown for Internal-Road in Figure 19. Thus, the benefits of combining information across phones will accrue, whether the phones belong to the same operator or to different operators.

4.7 Fingerprinting Versus Probabilistic Localization

Outdoor localization is a well-researched topic with techniques ranging from deterministic methods such as fingerprinting on the one hand to probabilistic methods on the other. We compare two popular probabilistic localization methods with our *GoodSig* algorithm. The first method (Haeberlen et al. 2004) divides the geographical area into grids and models signal strength as a Gaussian distribution for each cell tower for each grid. It then uses Bayesian inference to predict the posterior probability of an observation belonging to a particular grid. We empirically found that a grid size of 50m × 50m works well for our target applications. Selecting a smaller grid size resulted in a statistically insignificant number of observations in each grid for Gaussian fitting. On the other hand, selecting a larger grid size directly translates to larger errors.

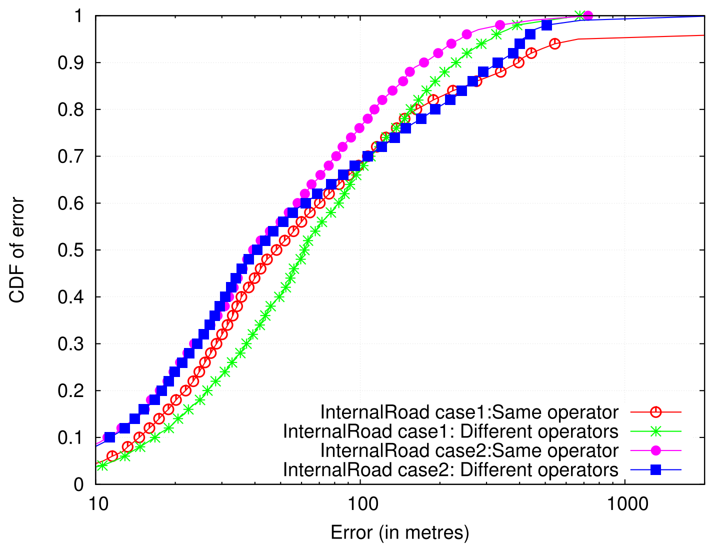

Fig. 19. Same vs Different operators: InternalRoad.

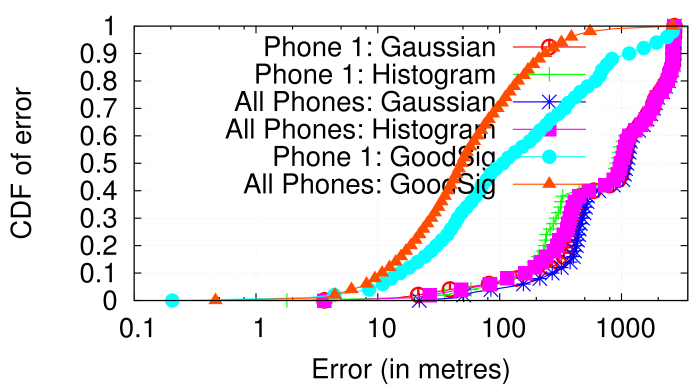

Fig. 20. Comparison of different localization algorithms.

The other method (Ladd et al. 2005) is that of building signal intensity histograms instead of Gaussian fitting. Here, unlike storing mean and standard deviation, we store the complete histogram of signal intensities for each cell tower and grid. During evaluation, if observations pertaining to a particular signal intensity are absent, then we simply consider the histogram bin corresponding to the nearest signal intensity for which observations are available.

We have implemented the above two methods as well and quantitatively compared them with the fingerprinting approach. We show these comparisons for InternalRoad route in Figure 20. The plot contains CDFs for three types of localization approaches viz. Gaussian-based, Histogram-based and fingerprinting-based. Since the focus of this article is on combining information from multiple phones, we also show comparison between using one phone versus using all phones for localization. We show this for each of the three localization approaches. Hence there are six lines in the plot. It can be clearly observed that fingerprinting outperforms other methods. Also, combining other phones' information (GoodSig) is found to be helpful in fingerprinting as opposed

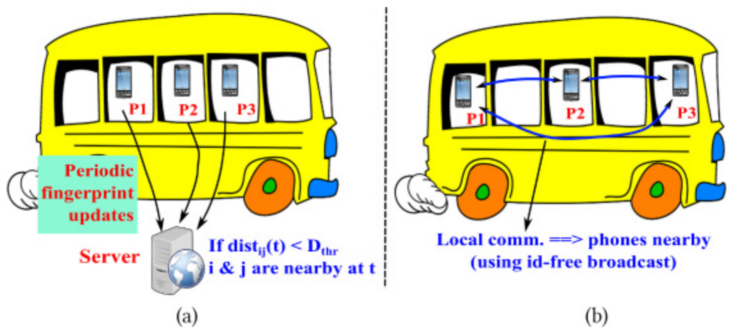

Fig. 21. (a) Server-side approach, (b) Client-side approach.

to probabilistic approaches. We believe that this is due to our dense urban setting, which makes the signal non-Gaussian due to multi-path effects, interference and absorption.

Computational complexity: Here we compare "GoodSig" with Probabilistic localization algorithms in terms of computational complexity. Let the number of cell towers observed during training phase be c and average number of signal strength observations per cell tower be k. For "GoodSig" algorithm, we just have to perform lookup for a particular cell-id signal strength pair and return the corresponding locations. Hence, both the space and time complexity here would be proportional to ck. This could be further improved by hashing. For probabilistic localization, we had divided the geographical area into grids. Let the number of grid cells be g, each having dimensions of 50m × 50m (empirically determined). The sensor model for each grid, for each cell tower has to be stored in memory. Hence, space and time complexity for Gaussian signal model is proportional to gc and g, respectively. For Histogram model, we need to additionally store the whole signal intensity histogram in memory. Due to this, space complexity increases in proportion to gck, for this case. Typical values of c and k are small ($c = 50$ and $k = 12$), while number of grids g, is large. We found "GoodSig" to be significantly more computationally efficient than other probabilistic localization approaches discussed in this article.

5 FINGERPRINT SHARING MECHANISMS

This section discusses the important aspect of determining the set of phones that are "nearby," whose location information can be combined. We have considered two approaches in this regard: server-side and client-side.

5.1 Centralized Server-Side Approach

We first considered a server-side approach, depicted in Figure 21(a). Each client (phone) periodically sends its fingerprint, i.e., (cell-id, signal-strength) tuple, to a central server. This is the same server that also maintains the fingerprint database, and answers location queries from client phones. We have considered a situation where the server knows the clients' fingerprints at a fine granularity: every t_u seconds; $t_u = 2$s in our evaluation below. Note that a client could bunch up its fingerprint updates to the server, say every 30s, to amortize the network traffic overhead.

Given the fingerprint of each client, the server first computes the fingerprint-based location (note that there is no combination of information across phones yet). For a client i, denote its location at time t by $L_i(t)$, and the distance between $L_i(t)$ and $L_j(t)$ by $dist_{i,j}(t)$. The server concludes i and j to be near one another at t if and only if $dist_{i,j}(t) < D_{thr}$.

Table 8. FNR, FPR for Server-Side Approach

D_{thr}(m)	100	200	300	400	500
FNR	0.45	0.28	0.17	0.12	0.06
FPR	0.09	0.18	0.31	0.44	0.50

Now, what threshold value D_{thr} to choose? To answer this, note that the threshold-based mechanism can result in false-positives (FP, phones not near each other being detected as being near each other) as well as false-negatives (FN). To evaluate the server-based approach in terms of FP and FN, we have used the same data as earlier. In the data, we had four phones traveling together: this thus becomes a natural test case for evaluating false negatives. To evaluate false-positives, we considered the location trace of two phones traveling together, and staggered one of them by 3min. This mimics a situation where one phone is traveling ahead of another, on the same bus route, but behind in time by 3min. We used this staggered trace to evaluate false-positives.

Table 8 summarizes the false-negative rate (FNR) and false-positive rate (FPR) for this mechanism, for different D_{thr}. With increasing D_{thr}, the FNR decreases but the FPR increases. We could potentially choose D_{thr} = 200m, with FNR=28% and FPR=18%. These values of FNR and FPR are not as low as one would ideally desire. This server-side approach also has potential privacy concerns, as the client has to share its real-time fingerprint (location) with the server. Hence, we now consider an alternate approach to determine if two phones are nearby.

5.2 Distributed Client-Side Approach

To combine cellular fingerprints across "nearby" phones, the phones need to share only their fingerprints; their identity is irrelevant. We now describe a distributed mechanism for phones "near" one another to share their cellular fingerprints anonymously.

In the proposed mechanism, smart-phones use a short-range communication technology (Figure 21(b)). Each device broadcasts its cellular fingerprint periodically (say, every few seconds). Phones that receive the broadcast are assumed to be *co-located* with the transmitting phones; this assumption introduces an error of at most the range of the communication link. Each phone can then simply combine fingerprint information (using the GoodSig algorithm) across all the broadcasts it has received, over a recent time interval. Note that in this mechanism, mobile phones *do not* need to share their identities while broadcasting their fingerprints in an ad hoc manner.

We initially considered WiFi and Bluetooth as possible modalities for the short-range communication among phones. WiFi suffers from higher power consumption and requires a setup for key-sharing among phones. Although Bluetooth communication consumes lower energy, most smartphones in market today, still require "pairing" for data sharing, and may not support advertisement broadcasts. Therefore, at present, not only will the handshaking process associated with key-sharing and pairing in WiFi or Bluetooth incur significant delay, but more importantly, these have significant privacy concerns. We therefore explore the possibility of using low-energy *audio communication* between smart-phones, for sharing fingerprint information in a manner that does not reveal the identity of the user.

Audio Communication

Audio communication has the advantage of consuming very little power (Madhavapeddy et al. 2003). It uses the acoustic channel in the *inaudible* frequency band of (16–22) kHz to transmit data.[1] In our setting, this technique uses the already available speakers and microphones of mobile

[1]Some children may be able to hear up to 20KHz; but we've noticed that in outdoor settings, its hard enough for anyone to hear in the audible frequency range!

Table 9. Design Parameters for Physical Layer

Audio sampling rate	44.1kHz
Frequency range	16-18.5kHz
Preamble	11-bit Barker Sequence
Frame Synchronization	Template matching
Modulation scheme	8-FSK
Error-correcting code	Viterbi code

phones, to send and receive data, respectively. Since it does not require any additional hardware, it is a completely software-based approach. Audio communication is not connection-oriented, and therefore does not require key exchange in any form. Data sharing thus becomes quick and energy-efficient. Furthermore, we do not associate any ID with a device; hence, such communication is anonymous and does not reveal the identity of the user. A disadvantage of audio communication, is the low data rate. However, note that cellular fingerprint sharing requires the periodic broadcast of only a few data bytes. Therefore, low-data rates associated with audio communication is not an impediment for our application. Even with a low data rate, a significant challenge for audio is the communication range.

Most prior work has reported range as short as a meter or less (Madhavapeddy et al. 2003; Nandakumar et al. 2013; Zhang et al. 2014). While some have reported a higher range (Lee et al. 2015), they have operated in otherwise silent, noise-free environments. In our setting, we expect a highly noisy environment: outdoors, on the road, inside a crowded bus/train. In some parts of the world such as India, road-noise has been reported to be very high and this includes vehicular honks (Sen et al. 2010; Mohan et al. 2008). For our application, we further require audio communication for distances of a few meters, to be able to share fingerprint data amongst phones within or in the vicinity of a bus/train. With these considerations, we have carefully designed the PHY, MAC, and Link layers of the audio communication, as detailed below.

Physical Layer: We conducted experiments using different mobile phones in lab (typically silent) environment, as well as noisy settings in crowded buses. These initial experiments used a Samsung S5 as transmitter and different receivers such as Xperia and Nexus 5. The following are the aspects we explored (as summarized in Table 9):

(a) Frequency range selection: We conducted preliminary tests on a crowded bus, to understand whether audio communication can work over these distances and in noisy conditions. We varied the distance between the transmitter and receiver from about 1 meter to about 5m, within the bus. In different experiments, we set the transmitter to transmit at 16kHz, 17kHz, 18kHz, and 19kHz. Our experiments revealed poor SNR performances for frequencies above 18.5kHz. Therefore, we confine data transmission to the 16–18.5kHz frequency band.

(b) Synchronization with preamble design: To detect the start of each frame, we explored the use of various Pseudo-Noise (PN) sequences, such as Barker codes, Kasami sequences, and Gold sequences, for the preamble. Owing to their favourable auto-correlation properties, PN sequences proved to be suitable candidates for frame synchronization over the audio channel. Amongst the explored PN sequences, a 11-bit Barker sequence ASK modulated at 16kHz, was found to be most suitable due to its lower auto-correlation values at side lobes compared to other PN sequences. We used a moving window-based *template matching* technique, to determine the start of the frame, at the receiver. Here, a preamble-length window of the received signal is correlated with the Barker sequence preamble template stored at the receiver. Correlation is calculated using the Pearson

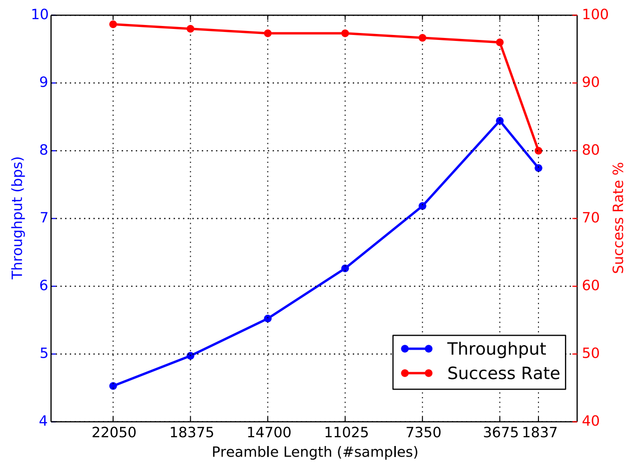

Fig. 22. Throughput, detection rate vs. preamble length.

correlation function. A peak detection algorithm that uses EWMA (Exponentially Weighted Moving Average), is then applied on the correlation values to find the maximum value, which corresponds to the start of a frame.

To reduce the computation overhead of the template matching algorithm, the moving window of the received signal is advanced by a larger number of samples (60 samples), until the correlation exceeds a threshold (experimentally estimated as 0.2), indicating the presence of the preamble. Thereafter, the window is advanced only one sample at a time, to determine the start of the frame with finer accuracy. This resulted in reduced correlation computation time—for a 2s window of received signal, and a sampling frequency of 44.1kHz, it took 83ms and 267ms, on an average, to calculate correlation when the preamble was absent and present, respectively. Choosing a suitable length for the Barker sequence preamble, in terms of number of samples was another challenge. We tested several preamble lengths and the one with good detection rate and throughput (in bits per second) was selected, for a payload of four data bits. Here, *throughput* is defined as the number of bits correctly received per second at the receiver. The graph in Figure 22 shows the detection rate of the preamble of varying lengths and the corresponding throughput.

Figure 22 shows that when preamble length reduces, detection rate decreases very slowly, and throughput increases because the preamble overhead is reduced by half each time. After the length of 3,675 samples, the detection rate reduced to less than 80% because of poor correlation for the small preamble. Hence, the throughput also dropped. Based on this behaviour, we use a preamble of length 3,675 (or 84ms) for the 11-bit Barker sequence.

(c) Modulation scheme selection: Multiple trials of our experiments showed that 8-FSK modulation scheme yielded highest throughput (in bits per second) in comparison with 4-FSK and 16-FSK. For an implementation of 8-FSK in noisy settings, the throughput with different data bits per frame is shown in the graph in Figure 23. Initially, the throughput increases with the number of data bits per frame. After 28 bits per frame, corresponding to a maximum throughput of 19.6bps in noisy settings, it drops as the error rate increased due to more bits per frame.

At 20, 24, and 28 data bits in payload, the throughput is almost same, but the highest throughput achieved is 19.6bps for 28 data bits per frame in noisy conditions. In lab environment without any noise, the throughput achieved for 28-bit payload is 23.2bps.

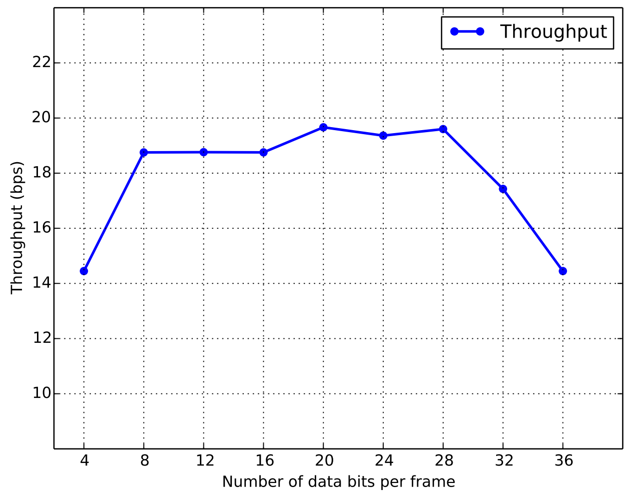

Fig. 23. 8-FSK: Throughput vs. Data bits per frame.

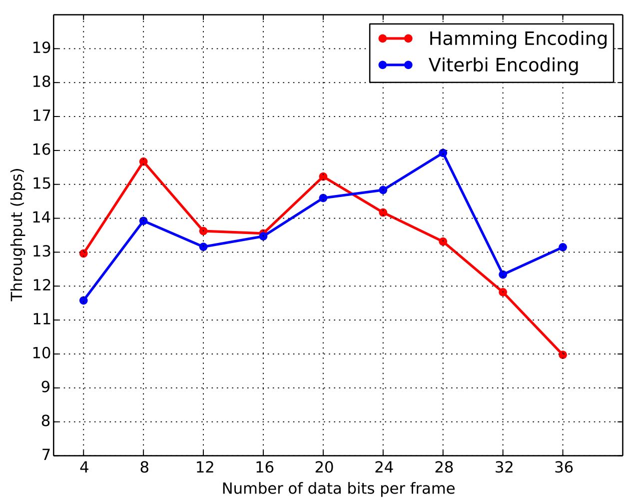

Fig. 24. Throughput vs frame len (4-FSK; Hamming/Viterbi).

(d) PHY layer coding: A-rate 1/2 Viterbi encoder yielded a higher throughput than a (7,4) Hamming code for 4-FSK modulated 28-bit frames, as shown in Figure 24. A similar observation was made for 8-FSK. The throughput shown in Figure 23 uses Viterbi encoding at the transmitter.

At the physical layer, the *packet* thus consists of the 84ms preamble, followed by the 56 encoded bits (corresponding to the rate 1/2 Viterbi encoded 28 data bits), transmitted using 8-FSK modulation. We now evaluate the overall PHY layer in real-life noisy conditions. Samsung S5 is used as a sender and Nexus 5 was the receiver for these experiments. The distance between the transmitter and receiver was varied from 1 to 5m for each of the settings. The *overall reception rate* is defined as the number of frames correctly decoded out of the total number of frames received at the receiver; and *preamble detection rate* is the rate at which the receiver is able to

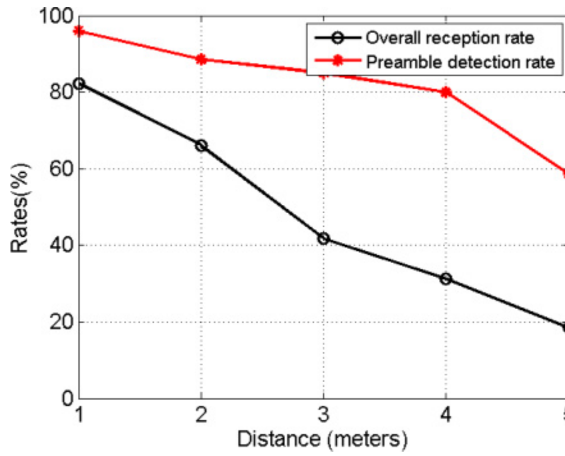

Fig. 25. Overall Reception and Preamble detection rate vs. dist. (within city bus).

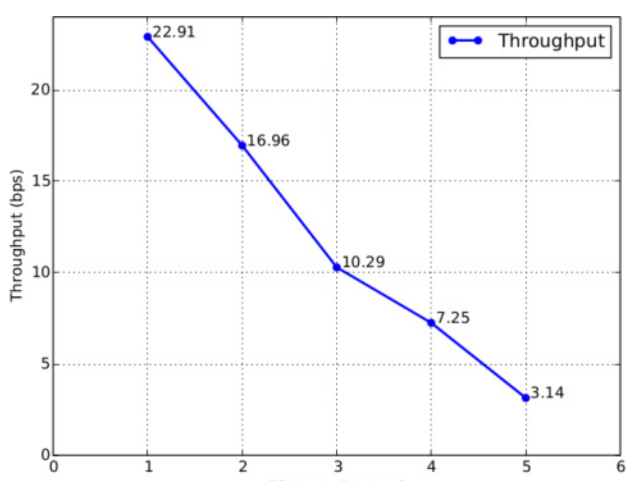

Fig. 26. Throughput vs. distance (within city bus).

detect a sent packet. Figure 25 shows the variation of preamble detection rate and overall reception rate with sender-receiver distance in a crowded bus. Figure 26 shows the variation of throughput (in bits per second) with sender-receiver distance for the same setting. As expected, the preamble detection rate, overall reception rate, and the throughput decrease as the sender-receiver distance increases. At 5m, the overall reception rate is 19% and the preamble detection rate is 59%, because of the higher signal distortion at these distances. We observed that the maximum throughputs achieved for a 28-bit payload, for varying transmitter-receiver distances were: 23.2 bps at 2m for lab settings; 19.6bps at 2m for noisy settings; and 22.9bps at 1m separation for city buses.

Data Link layer: To perform error detection, we reserve some of the data bits in the frame for CRC codes. Our experiments revealed that a suitable design choice for our settings was the 8-bit CRC $0x83$, with a theoretical 99.61% detection rate. The 28-bit PHY payload mentioned above now has,

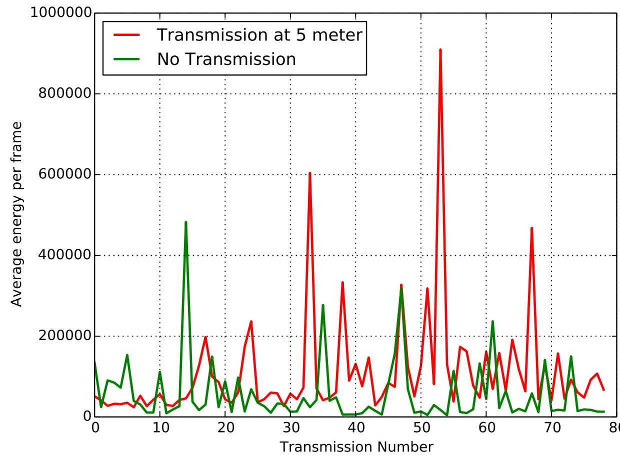

Fig. 27. Energy per frame at 5m: No-Tx, Tx.

20 data bits (to carry the cellular fingerprint) and an 8-bit CRC. Note that the data rates mentioned above considered all 28 bits in the payload to be data bits (as opposed to 20 bits with 8-bit CRC). We evaluate the data rate as 14bps for a city bus environment after incorporating CRC within the 28-bit data frame.

MAC layer: Our application requires seamless sharing of fingerprint data, without the need to transmit identity information or acknowledgement of broadcast receipts. A CDMA or synchronous TDMA scheme would require centralized coordination and synchronization: something unsuitable for our intended ad-hoc operation. We therefore explored a distributed CSMA (Carrier Sense Multiple Access) approach.

CSMA requires mobile phones to individually determine if the channel is busy or not, by measuring the signal energy in the channel. We conducted experiments in a crowded bus, where a frame was transmitted for 1s duration and the energy was measured by an Xperia Z3 phone. We noticed that the received energy was significantly different for *no transmission* and *transmission present*, at a distance of 1m. However, this difference diminished at a distance of 5m from the transmitter, as seen in Figure 27. Thus, for a given device, carrier sense threshold is distance-dependent. We further discovered that carrier sense thresholds vary across different mobile phones due to device heterogeneity. For a given frequency range, mobile phones transmit audio at different energy levels. Similarly, receptivity of microphones may vary across smart-phones, as shown in Figure 31. Here, for the Samsung S5 transmitter, the FFT of microphone reception of Xperia Z3 and LG Nexus are shown to vary. Thus, with a carrier sense threshold that varies across devices and distances, it becomes infeasible to have a CSMA-MAC scheme. Given the infeasibility of CSMA, we initially used an ALOHA scheme at the MAC layer. But that resulted in collisions, and hence was not scalable.

To minimize collisions, we designed a MAC protocol to facilitate sharing of fingerprints. This protocol is a distributed TDMA protocol: it uses slots to minimize collisions, while at the same time not requiring any central coordination. In this protocol, we assume a protocol parameter constant n as the maximum number of participating phones: we justify this shortly. A phone can be in either of two modes: transmitting or listening. A frame contains n time slots, one for each phone. We show a typical frame structure in Figure 28. Each phone transmits once in a frame for a time slot of size ks. When a new phone wishes to join a set of already transmitting phones, it

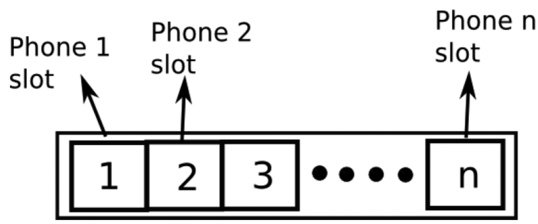

Fig. 28. Frame structure.

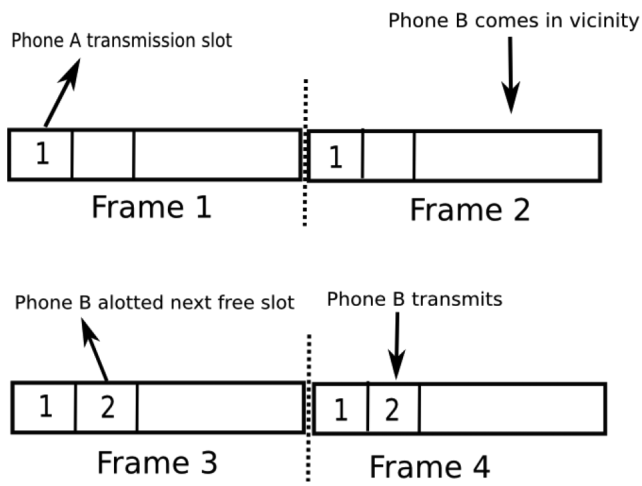

Fig. 29. New Phone arrival.

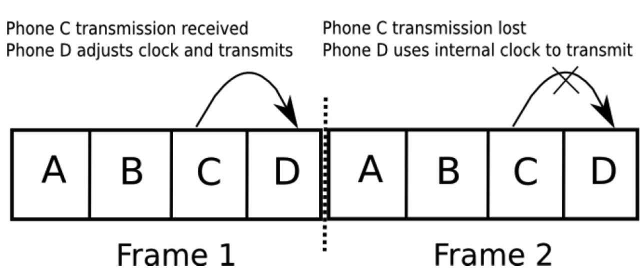

Fig. 30. Phones in steady state.

first listens for s frames. If in this duration it observes that all the n slots are already taken, then it does not transmit and only listens to other transmitters. If there is a free slot, then it uses the same to transmit in subsequent frames: this is shown in Figure 29. We experimentally tested this MAC protocol for the case of six phones.

As far as transmitting data is concerned, every phone maintains an internal clock. This clock is synchronized based on the transmission received from the phone in the preceding time slot. If by any chance transmission from preceding time slot is not received (packet loss or node detachment), then phone in the current time slot would simply use its internal clock to transmit. We show this clearly in Figure 30.

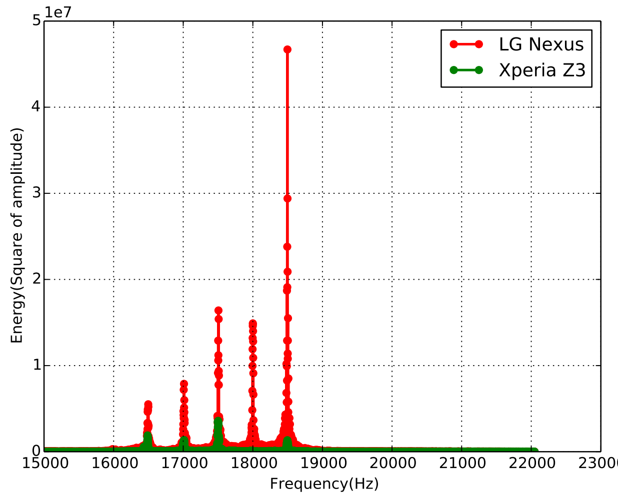

Fig. 31. Microphone receptivities in frequency domain.

Now, this MAC protocol imposes a limit on the maximum number of transmitters. This is suited to our application where the transmission is to share information, and in our application we found in Sections 3 and 4 of the article that combining information from more than four phones has marginal additional benefits. For this reason, we set $n = 4$ in our protocol implementation. Also, we found that setting $k = 4$s and $s = 3$ works well in practice.

Distributed sharing of cellular fingerprint (Application Layer): Having determined the PHY, MAC, and Link layers, we have put together a mobile app *SoundShare* to share cellular fingerprint information. The design at the application layer involves the choice of the fields in the data bits. Each cell ID (eg. 1721) is assigned 16 bits; while RSSI (e.g., −63dBm) is assigned 6 bits. To look-up the correct operator in the cell ID database, the frame should additionally carry operator information, which is assigned 4 bits. With 20 data bits per frame, we will therefore need two frames with the same sequence number (of length 4 bits) to carry fingerprint data about $< cell\ ID, RSSI >$ for a given operator. The first bit of the sequence indicates if the frame contains the 16-bit $< cell\ ID >$ or the $< RSSI, OperatorCode >$ tuple. Time taken to transmit the two frames was around 2s at the sender. Having received these frames, a receiver can potentially identify the operator and run the *GoodSig* algorithm to combine locations obtained using cellular fingerprints of different nearby phones. At no point is any information identifying the phone or the user included in the information being shared.

CPU time and Power analysis: We now evaluate the delay overhead and energy efficiency of the *SoundShare* mobile application. We have used new phones with Android 5.0 and above for all our measurements. The phone models used were LG Nexus 5x (Android 7.1), Samsung Note 3 (upgraded to Android 5.0), and Samsung Galaxy Grand Max (upgraded to Android 5.1). At the sender, the application takes 40ms on an average, to Viterbi encode and modulate a frame with a preamble of duration 84ms and 28 data-bit payload. The average power consumption associated with the transmission process is 8mW. At the receiver, the time taken for filtering, preamble detection, and decoding requires an average CPU time of 310ms. For our TDMA-based MAC protocol, we found an average power consumption of 171mW. In comparison, the power consumption for GPS is 435mW for the same setting. These values indicate that a distributed cellular fingerprint sharing mechanism incurs lesser energy and latency overhead in comparison with a centralized, GPS-based approach. The receiver side power consumption for *SoundShare* is high primarily due to the CPU intensive preamble matching process: we believe further improvements here are possible with optimized preamble detection mechanisms.

6 CONCLUSIONS AND DISCUSSION

Cellular network-based location determination is attractive for location-based applications, since the cellular interface is power efficient and is turned on anyway. However, such location determination suffers from inaccuracies. In this article, we have explored the idea of combining cellular network information across "nearby" phones. Using theoretical analysis and an extensive experimental study of data collected from two cities, we have shown that such combining improves the location accuracy significantly, especially when the fingerprinting database used is not the same as the phone trying to locate itself: in our data, the median error reduces by a factor of 1.6 and the 90 percentile error reduces by a factor of 3.5. Location accuracy improves even when just two phones' information is combined, with the benefits improving further with additional phones' information. Combining information across phones is beneficial irrespective of whether cellular data is turned on or off, and irrespective of whether the phones belong to the same operator or different operators. We have also designed, implemented, and evaluated an audio communication-based mechanism for nearby phones to share cellular fingerprint information, which does not reveal or use the identity of the user.

We believe that CrowdLoc has the potential for a real-world deployment. There are a number of popular public transportation apps in different metropolitan cities such as Mumbai, Singapore, New York, and many others. With an incentive mechanism in place, as well as a fingerprinting database that can be built over a few weeks, CrowdLoc can potentially be integrated with such apps to further their services in a city-wide deployment.

ACKNOWLEDGMENTS

We also thank Piyush Bhatore for his contributions on implementation of the SoundShare app.

REFERENCES

BBC. 2015. World's Busiest Railway. Retrieved from https://www.youtube.com/watch?v=D6nKHZwxr-c.

Arpita Biswas, Deepthi Chander, Koustuv Dasgupta, Koyel Mukherjee, Mridula Singh, and Tridib Mukherjee. 2015. PISCES: Participatory incentive strategies for effective community engagement in smart cities. In *Proceedings of the 3rd AAAI Conference on Human Computation and Crowdsourcing*.

Mike Y. Chen, Timothy Sohn, Dmitri Chmelev, Dirk Haehnel, Jeffrey Hightower, Jeff Hughes, Anthony LaMarca, Fred Potter, Ian Smith, and Alex Varshavsky. 2006. Practical metropolitan-scale positioning for gsm phones. In *UbiComp 2006: Ubiquitous Computing*. Springer, 225–242.

Ionut Constandache, Shravan Gaonkar, Matt Sayler, Romit Roy Choudhury, and Landon Cox. 2009. Enloc: Energy-efficient localization for mobile phones. In *Proceedings of the IEEE Conference on Computer Communications (INFOCOM'09)*. IEEE, 2716–2720.

Hui Gao, Chi Harold Liu, Wendong Wang, Jianxin Zhao, Zheng Song, Xin Su, Jon Crowcroft, and Kin K. Leung. 2015. A survey of incentive mechanisms for participatory sensing. *IEEE Commun. Surveys Tutor.* 17, 2 (2015), 918–943.

Google. 2017. Location | Android Developers. Retrieved from https://developer.android.com/reference/android/location/Location.html.

Andreas Haeberlen, Eliot Flannery, Andrew M. Ladd, Algis Rudys, Dan S. Wallach, and Lydia E. Kavraki. 2004. Practical robust localization over large-scale 802.11 wireless networks. In *Proceedings of the 10th Annual International Conference on Mobile Computing and Networking*. ACM, 70–84.

Jeffrey Hemmes, Douglas Thain, and Christian Poellabauer. 2010. Cooperative localization in GPS-limited urban environments. In *Ad Hoc Networks*, Vol. 1. Springer, 422.

Andrew Howard, Maja J. Mataric, and Gaurav S. Sukhatme. 2003. *Cooperative Relative Localization for Mobile Robot Teams: An Ego-Centric Approach*. Technical Report. DTIC Document.

INSINC. 2012. More Incentives to Encourage Off-Peak Travel on Trains. Retrieved from https://www.lta.gov.sg/apps/news/page.aspx?c=2&id=529b3055-b2fa-487b-88d9-0fb5ef7094ea.

Luis G. Jaimes, Idalides Vergara-Laurens, and Miguel A. Labrador. 2012. A location-based incentive mechanism for participatory sensing systems with budget constraints. In *Proceedings of the IEEE International Conference on Pervasive Computing and Communications (PerCom'12)*. IEEE, 103–108.

Pravein Govindan Kannan, Seshadri Padmanabha Venkatagiri, Mun Choon Chan, Akhihebbal L. Ananda, and Li-shiuan Peh. 2012. Low cost crowd counting using audio tones. *Proceedings of the ACM Conference on Embedded Networked Sensor Systems (SenSys'12)*. 155. DOI: http://dx.doi.org/10.1145/2426656.2426673

Andrew M. Ladd, Kostas E. Bekris, Algis Rudys, Lydia E. Kavraki, and Dan S. Wallach. 2005. Robotics-based location sensing using wireless ethernet. *Wireless Netw.* 11, 1–2 (2005), 189–204.

Hyewon Lee, Tae Hyun Kim, Jun Won Choi, and Sunghyun Choi. 2015. Chirp signal-based aerial acoustic communication for smart devices. In *Proceedings of the IEEE Conference on Computer Communications (INFOCOM'15)*.

Huiguang Liang, Hyong S. Kim, Hwee-Pink Tan, and Wai-Leong Yeow. 2016. Where am I? Characterizing and improving the localization performance of off-the-shelf mobile devices through cooperation. In *Proceedings of the Network Operations and Management Symposium (NOMS'16)*. IEEE, 375–382.

m Indicator. 2017. Mumbai Local train timetable. Retrieved from m-indicator.mobond.com/.

Anil Madhavapeddy, David Scott, and Richard Sharp. 2003. Context-aware computing with sound. In *UbiComp 2003: Ubiquitous Computing*. Springer, 315–332.

Anil Madhavapeddy, David Scott, Alastair Tse, and Richard Sharp. 2005. Audio Network.: The Forgotten Wireless Technol. 4 (2005), 55–60. DOI: http://dx.doi.org/10.1109/MPRV.2005.50

Prashanth Mohan, Venkata N. Padmanabhan, and Ramachandran Ramjee. 2008. Nericell: Rich monitoring of road and traffic conditions using mobile smartphones. In *Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems*. ACM, 323–336.

Moovit. 2017. Moovit: Your Public Transit Guide. Retrieved from https://www.moovitapp.com/.

Local Mumbai. 2012. Mega Block 31st December 2012. Retrieved from https://www.youtube.com/watch?v=NVeI-IeFg48.

MyTransport. 2017. Be empowered for an all new travelling experience. Retrieved from https://www.mytransport.sg/mobile/mytransport_mobile.html.

Rajalakshmi Nandakumar, Krishna Kant Chintalapudi, and Venkata N. Padmanabhan. 2013. Dhwani: Secure peer-to-peer acoustic NFC. In *ACM SIGCOMM*. 63–74. DOI: http://dx.doi.org/10.1145/2486001.2486037

Jeongyeup Paek, Joongheon Kim, and Ramesh Govindan. 2010. Energy-efficient rate-adaptive GPS-based positioning for smartphones. In *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services*. ACM, 299–314.

Jeongyeup Paek, Kyu-Han Kim, Jatinder P. Singh, and Ramesh Govindan. 2011. Energy-efficient positioning for smartphones using cell-id sequence matching. In *Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services*. ACM, 293–306.

Athanasios Papoulis and S. Unnikrishna Pillai. 2002. *Probability, Random Variables, and Stochastic Processes*. Tata McGraw-Hill Education.

Ashish Patro, Yadi Ma, Fatemah Panahi, Jordan Walker, and Suman Banerjee. 2011. A system for Audio signalling based NAT traversal. In *Proceedings of the 3rd International Conference on Communication Systems and Networks (COMSNETS'11)*. DOI: http://dx.doi.org/10.1109/COMSNETS.2011.5716432

Neal Patwari, Joshua N. Ash, Spyros Kyperountas, Alfred O. Hero, Randolph L. Moses, and Neiyer S. Correal. 2005. Locating the nodes: Cooperative localization in wireless sensor networks. *IEEE Signal Process. Mag.* 22, 4 (2005), 54–69.

Theodore S. Rappaport et al. 1996. *Wireless Communications: Principles and Practice*. Vol. 2. Prentice Hall PTR, New Jersey.

Ridlr. 2017. Ridlr - Real Time Traffic Alerts | Local Timetable | Traffic Updates in Delhi, Mumbai. Retrieved from http://ridlr.in/.

Rijurekha Sen, Bhaskaran Raman, and Prashima Sharma. 2010. Horn-ok-please. In *Proceedings of the 8th International Conference on Mobile systems, applications, and services*. ACM, 137–150.

Arvind Thiagarajan, Lenin Ravindranath, Hari Balakrishnan, Samuel Madden, Lewis Girod et al. 2011. Accurate, low-energy trajectory mapping for mobile devices. In *Proceedings of the Conference on Networked Systems Design and Implementation (NSDI'11)*.

Arvind Thiagarajan, Lenin Ravindranath, Katrina LaCurts, Samuel Madden, Hari Balakrishnan, Sivan Toledo, and Jakob Eriksson. 2009. VTrack: Accurate, energy-aware road traffic delay estimation using mobile phones. In *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems*. ACM, 85–98.

Yi Wang, Jialiu Lin, Murali Annavaram, Quinn A. Jacobson, Jason Hong, Bhaskar Krishnamachari, and Norman Sadeh. 2009. A framework of energy efficient mobile sensing for automatic user state recognition. In *Proceedings of the 7th International Conference on Mobile Systems, Applications, and Services*. ACM, 179–192.

Henk Wymeersch, Jaime Lien, and Moe Z. Win. 2009. Cooperative localization in wireless networks. *Proc. IEEE* 97, 2 (2009), 427–450.

Dejun Yang, Guoliang Xue, Xi Fang, and Jian Tang. 2012. Crowdsourcing to smartphones: Incentive mechanism design for mobile phone sensing. In *Proceedings of the 18th Annual International Conference on Mobile Computing and Networking*. ACM, 173–184.

Bingsheng Zhang, Qin Zhan, Si Chen, Muyuan Li, Kui Ren, Cong Wang, and Di Ma. 2014. PriWhisper:Enabling keyless secure acoustic communication for smartphones. *IEEE Internet Things J.* 1, 1 (2014), 33–45.

Received March 2017; revised October 2017; accepted October 2017