Dos and Don'ts in Mobile Phone Sensing Middleware: Learning from a Large-Scale Experiment

Valerie Issarny, Vivien Mallet, Kinh Nguyen, Pierre-Guillaume Raverdy, Fadwa Rebhi, Raphael Ventura

Dos and Don’ts in Mobile Phone Sensing Middleware: Learning from a Large-Scale Experiment

Valerie Issarny
Inria
Paris Research Center, France
Valerie.Issarny@inria.fr

Vivien Mallet
Inria
Paris Research Center, France
Vivien.Mallet@inria.fr

Kinh Nguyen
Inria
Paris Research Center, France
Cong-Kinh.Nguyen@inria.fr

PG Raverdy
Inria
Paris Research Center, France
Pierre-Guillaume.Raverdy@inria.fr

Fadwa Rebhi
Inria
Paris Research Center, France
Fadwa.Rebhi@inria.fr

Raphael Ventura
Inria
Paris Research Center, France
Raphael.Ventura@inria.fr

ABSTRACT

Mobile phone sensing contributes to changing the way we approach science: massive amount of data is being contributed across places and time, and paves the way for advanced analyses of numerous phenomena at an unprecedented scale. Still, despite the extensive research work on enabling resource-efficient mobile phone sensing with a very-large crowd, key challenges remain. One challenge is facing the introduction of a new heterogeneity dimension in the traditional middleware research landscape. The middleware must deal with the heterogeneity of the contributing crowd in addition to the system’s technical heterogeneities. In order to tackle these two heterogeneity dimensions together, we have been conducting a large-scale empirical study in cooperation with the city of Paris. Our experiment revolves around the public release of a mobile app for urban pollution monitoring that builds upon a dedicated mobile crowd-sensing middleware. In this paper, we report on the empirical analysis of the resulting mobile phone sensing efficiency from both technical and social perspectives, in face of a large and highly heterogeneous population of participants. We concentrate on the data originating from the 20 most popular phone models of our user base, which represent contributions from over 2,000 users with 23 million observations collected over 10 months. Following our analysis, we introduce a few recommendations to overcome -technical and crowd- heterogeneities in the implementation of mobile phone sensing applications and supporting middleware.

CCS Concepts

•Information systems → Mobile information processing systems; *Sensor networks;* **•Human-centered computing → Ubiquitous and mobile computing systems and tools; Empirical studies in ubiquitous and mobile computing; •Software and its engineering → Middleware; Client-server architectures;** *Software architectures;*

Keywords

Mobile phone sensing; Urban sensing; Crowd-sensing; Sensors heterogeneity; Sensing accuracy.

Middleware’16, December 12 - 16, 2016, Trento, Italy

ISBN 978-1-4503-4300-8/16/12…$15.00

DOI: http://dx.doi.org/10.1145/2988336.2988353

1. INTRODUCTION

Mobile Phone Sensing (*MPS*) [23, 25, 19] is a powerful solution for massive-scale sensing at low cost. The ubiquity of phones together with the rich set of sensors that they increasingly embed make mobile phones the devices of choice to sense our environment. Further, thanks to the –even sometimes unconscious– participation of people, MPS allows for leveraging both quantitative and qualitative sensing. And, still thanks to the participation of people who are moving across space, mobile phones may conveniently act as opportunistic proxies for the sensors in their communication range, which includes the fast developing wearables.

The many applications that have emerged over the years illustrate the potential of MPS, such as micro-blogging [13], mobile social networking [31], quantified selves [6], urban tomography [20], environmental monitoring [33], transportation [18], or dynamic indoor map construction [35]. As a matter of fact, applications revolving around MPS are now part of our daily activities –e.g., mobile social networking, quantified selves, intelligent transportation systems– and supported by major ICT players.

In general, we anticipate that MPS is going to generate drastic changes in the way we approach science in the years to come. A major driver in that direction is the massive amount of data that is being collected and that paves the way for advanced analyses of numerous phenomena at an unprecedented scale. The development of citizen science well illustrates this trend [40].

However, despite the numerous research work since the end 2000s, MPS keeps raising key challenges:

- *How to make MPS resource-efficient*? MPS involves power-hungry functions and especially location management and communication. MPS also often implies continuous monitoring. As a result, resource-efficiency and especially energy-efficiency require special care in the development of MPS applications [37, 26, 32, 34]. Although resource-efficiency in mobile computing has been on the research agenda for decades, MPS poses new questions. This is due to the combined effect of continuous monitoring and resource consumption that become dependent on overall user activities [8, 32].

- *How to mitigate mobile sensing heterogeneities*? While mobile phones increasingly embed sensors, the intrinsic performance of the sensors varies across mobile devices [41]. In addition, the position of the phone [21] as well as the trustworthiness of the contributing user [28] significantly affect the quality of the sensing. It is then important to know how to reconcile this heterogeneity; this includes the calibration of sensors to infer meaningful data [27, 29].

- *How to involve and leverage the crowd*? It is easy to anticipate that the above challenges compromise the participation of the crowd. Users get involved in MPS only if this brings them obvious benefits and is not detrimental to their habits (including the battery lifetime of their phone) and their privacy [5, 9]. In general, MPS applications should come along with the right incentive [46].

- *How to leverage prior experiences*? MPS applications obviously differ according to their application domain; however, they share numerous functionalities from the access to the sensors to the communication of their observations to some cloud- or fog-based crowdsourcing/sensing server. This calls for dedicated programming frameworks [36] and middleware [3, 16]. Still, the supporting software infrastructure must sustain the evolution of the underlying operating systems, which keep integrating new resource management and sensing features over time.

Addressing the above MPS challenges primarily lies in taming the high heterogeneity not only of the computing system but also the crowd. The latter introduces a new dimension compared to traditional middleware research that has been concentrating on overcoming the heterogeneities of the computing infrastructure. In order to tackle these two dimensions together, we have been conducting a large-scale empirical study in cooperation with the city of Paris – http://tinyurl.com/soundcity-paris. Our experiment revolves around the public release of a MPS app for urban pollution monitoring that is built upon a dedicated mobile crowd-sensing middleware [14]. In this paper, we build on this experiment to report on the empirical analysis of the MPS efficiency from both a technical and a social perspectives in face of a large and highly heterogeneous population of participants. We more specifically concentrate on the data originating from the 20 most popular phone models of our user base, which represent contributions from 2091 users with 23M observations collected over a period of 10 months. Further to our analysis, we introduce a few recommendations to overcome –technical and crowd– heterogeneity in the implementation of MPS applications and supporting middleware.

After providing an overview of the background and motivation for our work (Section 2), we make the following contributions:

- We present a MPS system (Section 3) and a specific application instance called SoundCity for noise pollution monitoring (Section 4). They both integrate state-of-the-art software solutions for large scale mobile crowd sensing. They are customized for the sake of user participation.

- We systematically study the influence of resource-efficiency and sensing accuracy on the effectiveness of the crowd participation (Section 5). In a complementary way, we analyze user participation across time, so as to derive participation patterns that MPS middleware and application design may leverage (Section 6).

- Further to the above analyses, we draw recommendations for the implementation of MPS middleware and applications (Section 7).

2. BACKGROUND & MOTIVATION

The design space for MPS systems and related research questions have been articulated since the early surveys of [25, 12, 19]. In a nutshell, they are concerned with the main functions associated with MPS:

- *Sensing* – MPS must allow leveraging the many -yet highly heterogeneous- sensors embedded in our phones that various applications use. Sensing must also adapt to the activity of the users since it greatly impacts the resource-efficiency [32] and the quality of the observations [21]. In that direction, piggybacking crowd-sensing is an effective solution because it coordinates with the relevant application activities [22, 45]. Similarly, energy-delay tradeoffs may be adequately managed so as to schedule sensing activities when the most energy-efficient [37, 34]. Another challenge for sensing on the phone is to reliably infer user behavior and context from noisy and complex sensor data that are collected under mobile device constraints. Increasing the sensing quality may possibly benefit from advanced learning techniques [24] together with collaborative inference [30]. Yet, MPS applications are successful only if an appropriate crowd gets involved, which requires adequate incentive mechanisms. Mechanisms may be either platform-centric or user-centric for which theoretical properties have been studied in [46]. A complementary pathway to attract sufficiently meaningful participation is to leverage the built-in incentives of location-based gaming and social applications by pairing crowd-sensing with such applications [39]. Obviously, privacy guarantees should be offered to participants, which may in particular be handled at the time of data collection and aggregation [9, 43, 17, 29]. In general, MPS applications must match the users' interests, which relates to the user experience and interface design.

- *Analyzing* – The power of MPS comes from that of its crowd; MPS allows collecting data at a massive scale to inform our knowledge of various phenomena. However, the analysis of data coming from MPS is challenged by the quality of the observations that is overall considered to be low [1]. Machine learning techniques may allow enhancing the quality of the observations at the time of sensing on the phone [24]. Still, data analysis greatly benefits from processing at the server level, where it is possible to correlate data at a larger scale [27, 28]. In that direction, data assimilation techniques are a powerful tool. Those techniques have been widely exploited to integrate observations from various data sources with mathematical models to simulate the state of a system or an urban phenomenon [42].

- *Informing, sharing & persuading* – MPS applications must create a feedback loop with their users. The knowledge acquired thanks to the participation of people must be made available to them and possibly be at the source of community awareness as illustrated by citizen science [40]. Further, the communicated knowledge may possibly influence the users' behavior, which is coined as *persuasive technology* [11]. This dimension is primarily application-specific. Still, closing the feedback loop involves two-way communication, which must be supported by the underlying middleware.

Our motivation for undertaking middleware research on MPS is to contribute to engaging citizens toward smarter cities - cf. http://urbancivics.com. Indeed, citizens often have important data or insights about a problem or its solution, which is not available to technocracies [10]. For instance, in the environmental health arena, citizen-led collection of environmental data has challenged established governmental expertise about what is hazardous or not [7]. Moreover, communities may be more likely to view issues holistically and thus be better positioned than technocracies to propose solutions that transcend domain-specific objectives.

Following, we have developed an application and supporting middleware to engage people in the understanding of urban pollution –from sensing to aggregation and mapping. In a first step, we have concentrated on noise pollution, which can be sensed using the phone's microphone and further analyzed using data assimilation techniques [15, 14]. Our middleware solution leverages state-of-the-art messaging (RabbitMQ - http://www.rabbitmq.com) and database (MongoDB - http://www.mongodb.com) systems for the implementation of scalable crowd-sensing, while the data assimilation engine relies on the Verdandi library (http://verdandi.gforge.inria.fr/). However, despite the broad literature on MPS and well-known guidelines for developing large-scale, resource-efficient mobile systems, the deployment of our system at the urban-scale allowed us to identify a number of pitfalls. We thus revisited the middleware and application design as our user base grew. The two next sections outline the resulting middleware and application design, highlighting the key features that allow engaging citizens. We then focus on the empirical analysis of mobile phone sensing efficiency, from which we derive guidelines for the development of MPS systems that deal with the systems' intrinsic –technical & social– heterogeneity.

We highlight that our study distinguishes itself from studies involving an a priori group / community of users as in [4] since our user base is that of users who have downloaded the app from the Google Play store due to their interest. Although this prevents the precise characterization of our user population from a social perspective, it allows reaching a large user base and directly confronting our system with our target end-users, i.e., people concerned with environmental pollution and its impact on health and the environment.

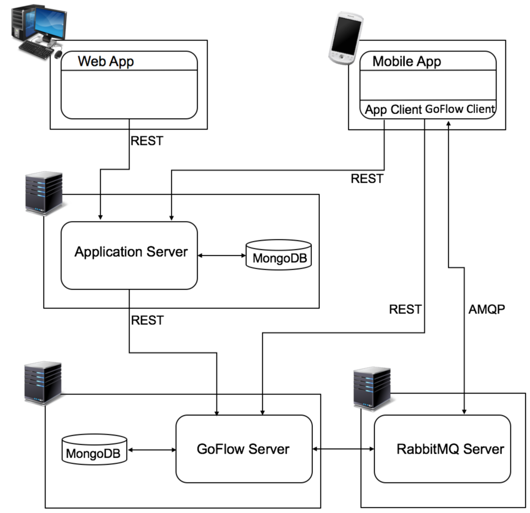

Figure 1: Crowd-sensing system deployment.

3. CROWD-SENSING SYSTEM

Figure 1 depicts the deployment architecture of our crowd-sensing system that is structured around: (i) the application and (ii) the *GoFlow* middleware and associated *RabbitMQ* server. Most of the system components interact through a REST-based API, except for the interaction with the RabbitMQ server that is through the Advanced Message Queuing Protocol (AMQP).

The application features Web and mobile instances that customize information access and display as needed; sensing is restricted to the mobile app. The Web application server maintains data about the contributing users in an anonymized way, so that specific contributions may be retrieved provided the user's credentials. In the experiment reported in this paper, our app focuses on urban noise monitoring (see Section 4).

3.1 The GoFlow Crowd-sensing Middleware

The GoFlow middleware implements the crowd-sensing server and associated messaging system. The GoFlow server stores the crowd's contributions; it builds upon the MongoDB database that provides high availability and scalability. GoFlow implements the privacy policy set by the French CNIL –The National Commission on Informatics and Liberty (http://www.cnil.fr)– that is the administrative regulatory body whose mission is to ensure that the French data privacy law is applied to the collection, storage, and use of personal data. The GoFlow server may host contributions by multiple MPS applications. GoFlow is built with open data in mind; contributing applications specify the data that they want to keep private and those that they agree to share with other applications.

Figure 2 depicts the main components of the GoFlow server, where –from left to right –top to bottom:

- *REST-based GoFlow API* is for clients and administrators to: authenticate and register subscribers and publishers, retrieve crowd-sensed data based on various filtering parameters, manage user accounts for an app, and submit and manage background jobs.

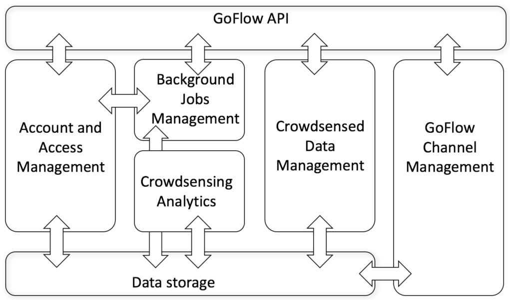

Figure 2: GoFlow Server.

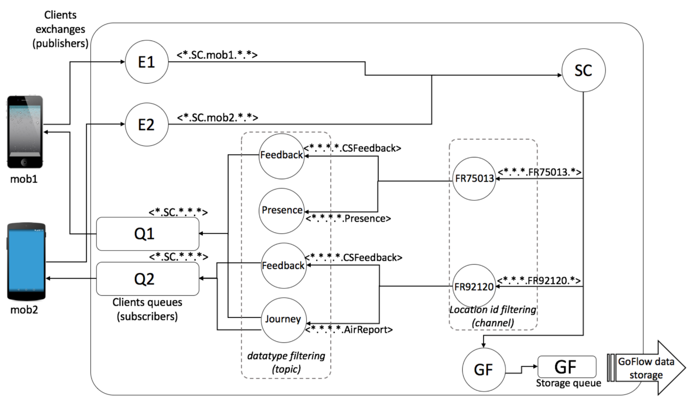

Figure 3: Messaging using RabbitMQ Exchanges (circles) and Queues (rectangles).

- *Account and access management* is to add/remove users with different roles for the registered apps.
- *Background jobs* manages scripts which are submitted by the application's managers and perform various operations on the crowd-sensed data stored on behalf of the application.
- *Crowd-sensing analytics* generates statistics about the app/clients operations.
- *Crowd-sensed data* management allows the retrieval of crowd-sensed information based on various filtering parameters, and various packaging solutions (file, json stream, ...).
- *Data storage* stores/deletes individual crowd-sensed messages as well as accounts, jobs and analytics information. It in particular builds upon MongoDB.
- *Channel management* creates and terminates the relevant RabbitMQ queues and message exchanges on behalf of clients to collect crowd-sensed data that GoFlow needs to store, as detailed next.

3.2 Message Routing

As depicted in Figure 1, messaging from the app to the GoFlow crowd-sensing server is routed through the RabbitMQ message broker that implements the Advanced Message Queuing Protocol (AMQP). RabbitMQ supports a very large number of connections and manages buffering for mobile sessions. The RabbitMQ messaging model is structured around *exchanges* and *queues* where exchanges forward messages to other exchanges or queues depending on their type (i.e., direct, fanout, topic).

Figure 3 illustrates the RabbitMQ exchanges and queues created by GoFlow for the routing of crowd-sensing messages that rely on topic-based exchanges. For each application, an exchange is created that forwards all the crowd-sensed messages to a GoFlow exchange and queue (named *GF* on the figure) for further processing. Then, when a mobile client (e.g., *mob1*, *mob2*) logs in on the GoFlow server, a client exchange is created (e.g., *E1*, *E2*); the exchange forwards the client's messages to the exchange of the application (e.g., *SC* that stands for our SoundCity App, see § 4). A queue (e.g., *Q1*, *Q2*) is also created for the client so that it can receive incoming crowd-sensed messages. When a client registers a subscriber for a given crowd-sensed data type at a location, the GoFlow server creates, if not available yet, the relevant exchanges for the location and datatype using their respective ids (e.g., *FR75013*, *Feedback*). The server also sets the bindings using the location and datatype ids as filtering parameters. In Figure 3, the client *mob1* registers to retrieve feedback reports of other mobile clients in its current neighborhood (e.g., location id *FR75013* based on the country and zip codes for the current location) as well as new public *Journeys* (i.e., collaborative noise maps) notifications from other users at his home location (i.e., *FR92120*). The *mob1* client may at some point publish some feedback for noisy events at its current location, as well as advertise its presence if other users in the vicinity want to proceed with a new collaborative map of the area.

The creation of the various exchanges and queues as well as the bindings is performed by the GoFlow server (i.e., the *GoFlow Channel management*) on behalf of the mobile users. The server then returns the unique ids of the relevant exchange and queue to the mobile client for connection. For security, the binding for the exchange of the client uses the client id (shared secret between the *GoFlow* client and server) as one of its filtering parameter.

The crowd-sensing middleware architecture builds upon state-of-the-art solutions, which provide the necessary guarantees in terms of scalability and availability. However, the effectiveness of MPS regarding crowd participation remains an open issue. We have investigated it in the specific context of noise pollution monitoring.

4. THE SOUNDCITY USE CASE

People engage into an activity -including MPS- if they care about its purpose. In our work, we concentrate on leveraging MPS for environmental pollution monitoring, which is of increasing concern for the urban population. Existing approaches to pollution monitoring primarily rely on simulation and the deployment of fixed, expensive sensors in relatively few locations. MPS is a promising complement to sensing across urban areas and spaces, thanks to the mobility of people. However, the MPS challenges that we have presented in the introduction question the significance of MPS-based pollution sensing, which we have been empirically investigating for more than 10 months (as of May 2016).

4.1 Noise Matters

So far, we have concentrated on noise pollution sensing. From a technical perspective, the sensors embedded in the phones (i.e., microphone and context-related sensors) are

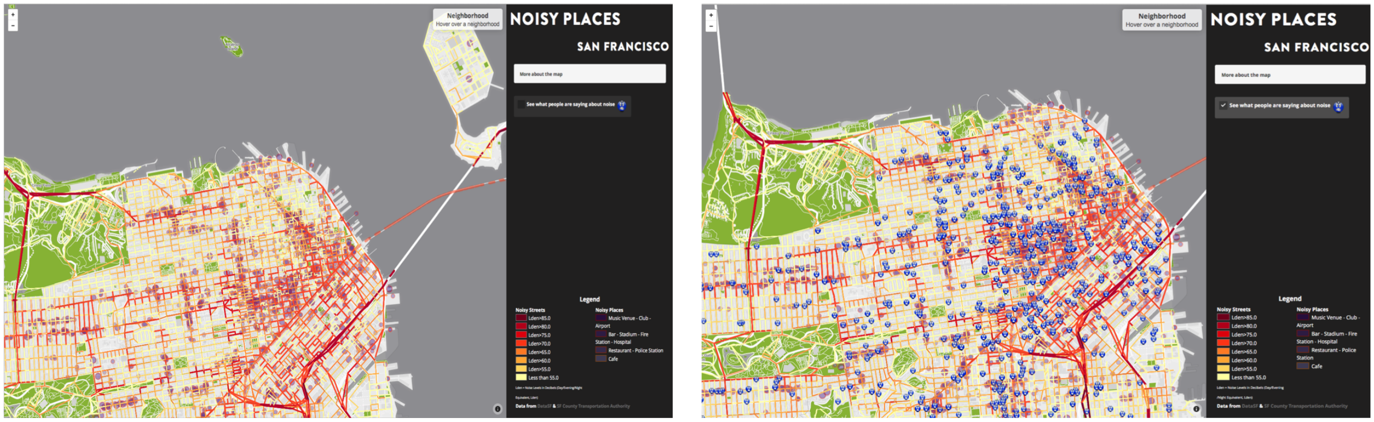

Figure 4: Noise matters - Left: Example noise levels simulated in San Francisco streets - Right: Locations of reported noise complaints.

sufficient to achieve MPS-based noise sensing [15, 14, 38]. Further, noise pollution, which lowers quality of life and harms health, is a serious environmental challenge in almost every major city. The noise levels found in most cities today can interfere with memory and learning, disturb sleep, and increase the risk of heart disease [44]. In addition, people are sensitive to noise pollution. As an illustration, Figure 4 (left) shows a noise map of San Francisco that we have built from the city's open data. The provided map aggregates noise due to traffic and places that are subject to noise (bars, restaurants, ...), high noise is shown in the reddish color on the map. Then, Figure 4 (right) adds to the map the complaints (the blue circles) due to noise that have been received at the city's 311 call number. We see that there is a strong correlation, highlighting the noise sensitivity of people. Hence, noise pollution sensing is a priori a good use case for MPS: from a technical perspective, the relevant sensors are available in any phone; from a social perspective, people care about it and some of them are likely to be willing to contribute to increasing the collective knowledge about noise pollution exposure.

4.2 The SoundCity MPS

We have developed the SoundCity application (http://tiny.cc/soundcity) on top of our crowd-sensing system. SoundCity is available on the Google Play Store since July 2015.

The SoundCity crowd-sensing system (Figure 5) introduces a new component, the *Data Assimilation Engine*, to overcome the high heterogeneity of the contributing sensors. The engine integrates and aggregates highly heterogeneous simulation and observational data to produce comprehensive representations about urban phenomena. Various numerical models exist to simulate urban phenomena or processes like traffic, air pollution, outdoor noise, etc. These models solve physical, sometimes also biological or chemical, equations that describe the main urban phenomena. The models often compute maps of variables that represent the city state at a given date. They provide spatialized information whereas observations tend to be only available at few locations, or along given trajectories in the case of a mobile sensor. The models may however show large errors which originate from the shortcomings of their formulations and their uncertain input data. The data assimilation engine reduces these uncertainties [2, e.g.,]. The effectiveness of data assimilation is beyond the scope of this paper and we refer the interested reader to [42] for an application at urban scale.

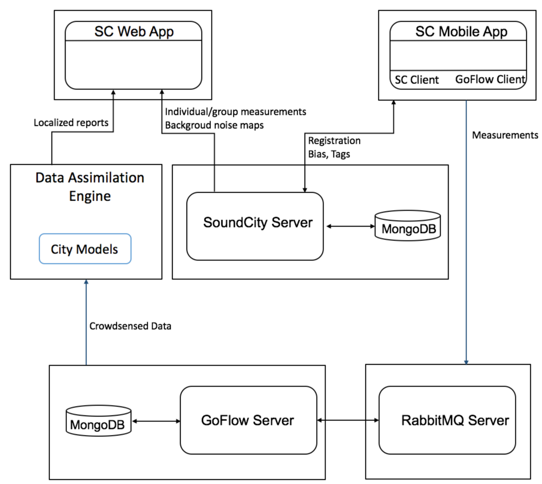

Figure 5: The SoundCity crowd-sensing system.

Although noise matters, this is not sufficient to ensure the traction of a large user base. We thus promote 3 complementary experiences to SoundCity users:

1. *Quantified self:* SoundCity shows the individual's daily and monthly exposure to noise in relation with its impact on health (Figure 6 (left & middle)).
2. *Engage:* By default, SoundCity implements an *opportunistic sensing*; it periodically measures, in the background, the sound levels with the microphone of the device. Obviously, this may generate many erroneous measurements depending on the situation of the phone. The user may also engage pro-actively in noise pollution monitoring by requesting for a measurement ("*sense now*" button on the home page). We have further introduced a new mode, called *Journey*, for *participatory sensing* (Figure 6 (right)). In this mode,

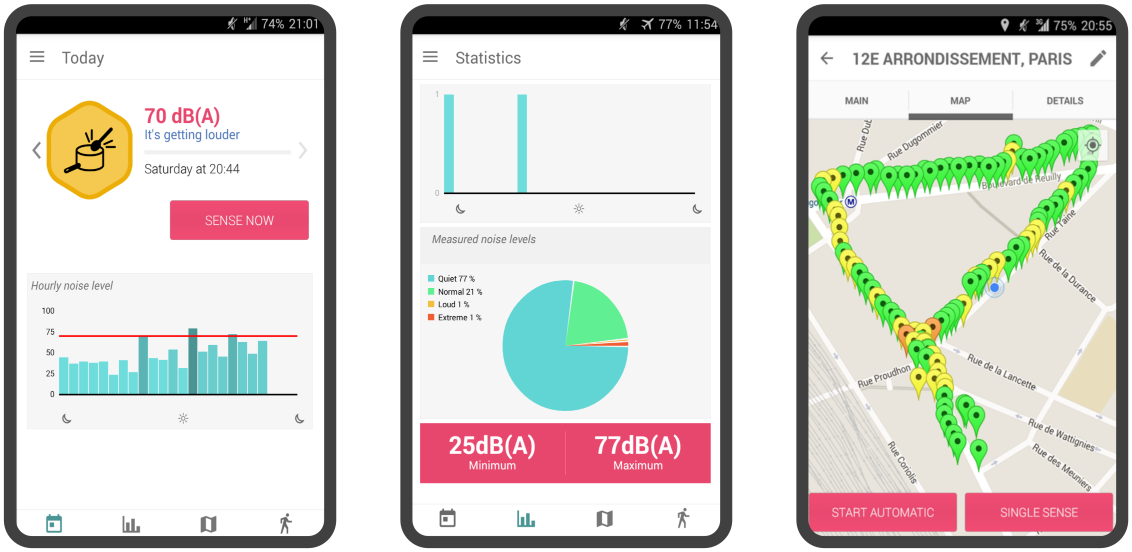

Figure 6: The SoundCity mobile app - Left: Home - Middle: Statistics - Right: Journey mode.

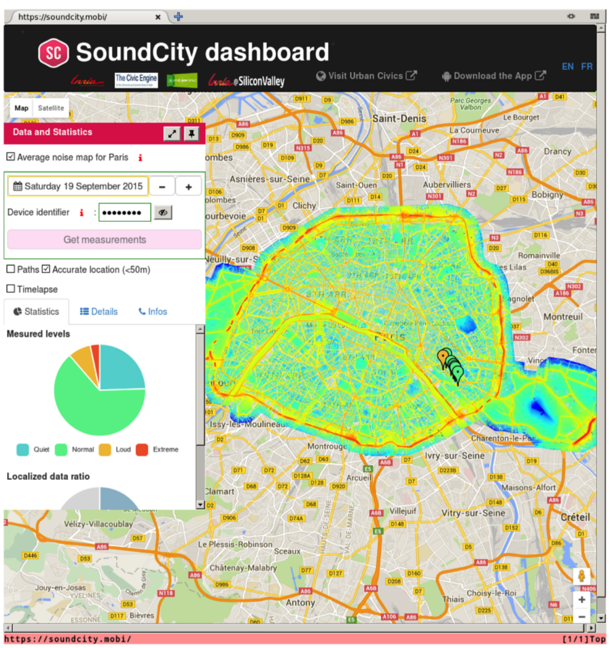

Figure 7: The SoundCity web app.

the user engages in the measurement of noise across a journey and defines the sensing frequency.

3. *Share:* By default, the observations collected by a user are made available to the user only (Figure 7). If the user accepts, the observations are communicated to the GoFlow server for assimilation. With the Journey mode, users may further share their observations publicly or within a community.

4.3 SoundCity in Paris & User Participation

SoundCity benefits from the support of the city of Paris that promotes actions oriented toward participatory democracy and is also engaged into pollution prevention. We launched the application with the City of Paris smart city initiative and Bernard Jomier, deputy mayor responsible for health, disability, and relations with Paris public hospital system in July 2015. This generated different citations in the media (see http://tinyurl.com/soundcity-paris). There is no doubt that this was key to attracting a large user base. As of May 2016, we had collected more than 45M observations (Figure 8); however, in this paper we concentrate on the analysis of the 23M observations provided by the 20 most popular phone models among our user base from July 2015 to May 2016 (Figure 9).

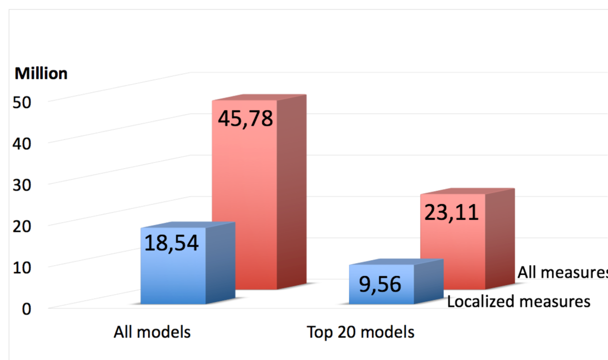

Figure 8: Contributed observations.

5. ANALYZING MPS EFFICIENCY FROM A TECHNICAL PERSPECTIVE

From a technical perspective, the contributing devices are highly heterogeneous regarding the quality of the embedded sensors. Using SoundCity contributions, we analyze more precisely the accuracy of location and noise sensing. Another technical challenge for the MPS system is the energy efficiency so that the supporting app does not drain too

Device model	Number		
	Devices	Measurements	Localized measurements
SAMSUNG GT-I9505	253	2 346 755	1 014 261
SAMSUNG SM-G900F	211	2 048 523	847 591
SONY D5803	112	1 097 018	778 732
LGE LG-D855	87	1 098 479	669 446
ONEPLUS A0001	84	1 177 343	657 992
LGE NEXUS 5	129	843 472	530 597
SAMSUNG GT-I9300	185	1 432 594	528 950
SAMSUNG SM-G901F	73	1 113 082	524 761
SONY D6603	51	815 239	524 287
SAMSUNG SM-N9005	134	1 448 701	503 379
SAMSUNG GT-I9195	174	2 192 925	464 916
SAMSUNG SM-G800F	66	989 210	393 045
HTC HTCONE_M8	76	854 593	177 342
LGE NEXUS 4	67	702 895	380 751
SONY D 6503	52	716 627	200 360
SAMSUNG SM-N910F	116	812 207	344 337
SAMSUNG GT-I9305	39	692 420	209 917
LGE LG-D802	46	728 469	278 089
SONY D2303	40	585 396	221 686
SAMSUNG GT-P5210	96	1 412 188	305 735
Total	**2 091**	**23 108 136**	**9 556 174**

Figure 9: Top 20 models.

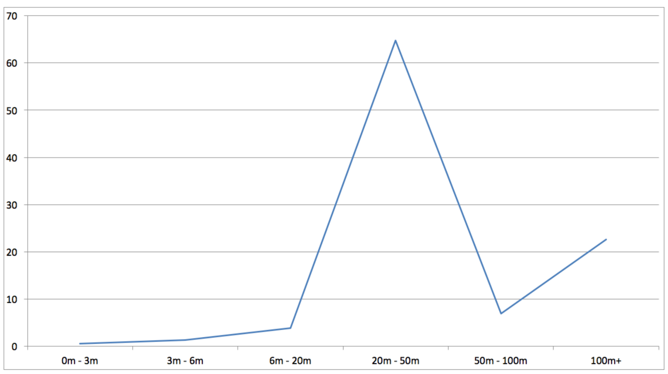

Figure 10: Distribution (%) of location accuracy (all) for the top 20 models.

much battery.

5.1 About Location Accuracy

Location is critical in most MPS systems. Today's OSes (Android in our study) offer the following location sources: GPS, network, and fused that leverages GPS and network while optimizing energy efficiency. As shown in Figure 9, about 40% of the observations contributed by the top 20 models are localized. Interestingly, this percentage is the same for the observations contributed by our overall user base (see Figure 8).

Figure 10 depicts the distribution of the location accuracy estimates reported by Android for all the analyzed observations that are localized. The (estimated) accuracy of most of the observations is in the $[20-50]$ meters range. There is then a peak at accuracies lower than 100 meters.

Looking at the observations per type of location source, Figure 11 shows, as expected, that GPS delivers the highest accuracy with most of the observations in the $[6-20]$ meters range. However, GPS is not the source of choice since only 7% of the localized observations are provided with GPS location.

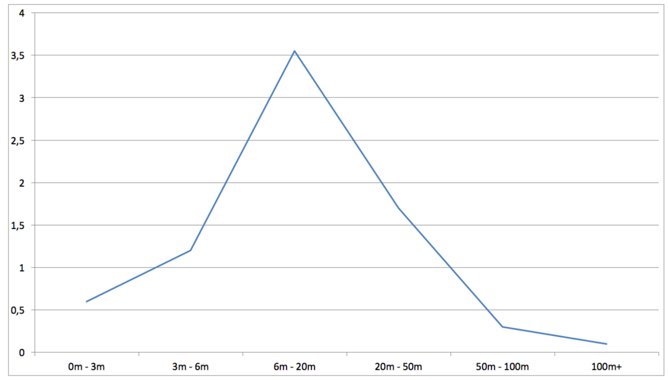

Figure 11: Distribution (%) of location accuracy (GPS) for the top 20 models.

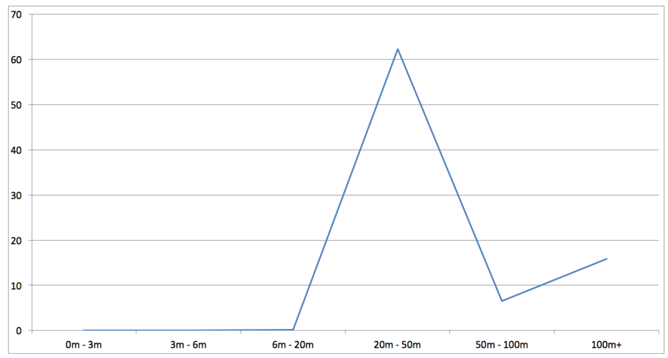

Figure 12: Distribution (%) of location accuracy (network) for the top 20 models.

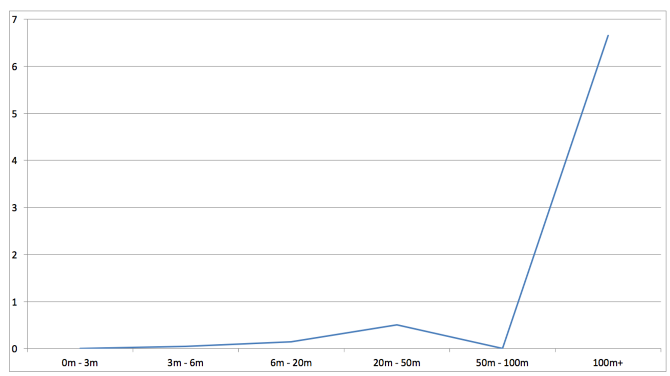

Figure 13: Distribution (%) of location accuracy (fused) for the top 20 models.

Network-based location is the most common and accounts for 86% of the localized observations (Figure 12). Hence, the distribution of location accuracy matches that of Figure 10; most of the localized observations are in the $[20-50]$ meters range accuracy.

Finally, the remaining 7% of the localized observations use fused location (Figure 13). We note that few models provide "fused" data. And the location accuracy is rather low.

In conclusion, when location matters, about 40% of the

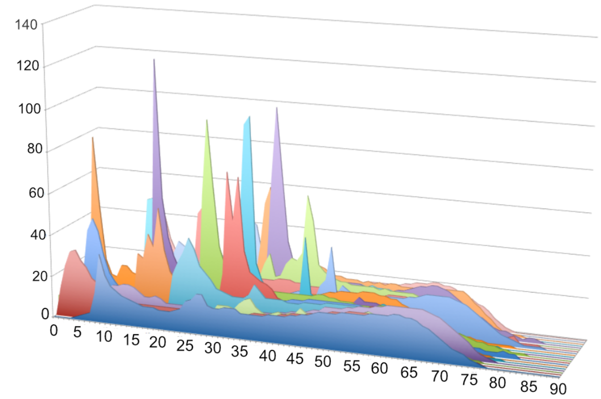

Figure 14: Distribution (‱) of raw SPL measurements (dB(A)) for the top 20 models.

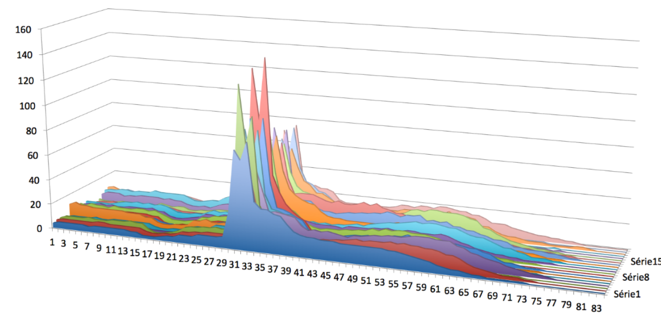

Figure 15: Distribution (‱) of raw SPL measurements (dB(A)) for the top 20 users owning a Samsung SM-G901F.

collected observations remain relevant with MPS, while the use of GPS needs to be encouraged as far as possible by the MPS app.

5.2 About Sensing Heterogeneities

Figure 14 shows the distribution of the raw Sound Pressure Levels (SPLs) for our top 20 contributing models, where each model distinguishes itself using a specific color. The measurements are provided in per-thousand of the number of measurements for the given dB(A) level. We observe the same pattern for all the models: a first peak at the low noise levels and then a small bump for active environments. However, the dB(A) values at which the peak occurs varies significantly across device models, which shows the heterogeneity of the (noise) sensors across models. Nonetheless, if we concentrate on the observations for a single model (e.g., see Figure 15), we see that the measurements follow much similar patterns, including with respect to the specific dB(A) measurements. Hence, the heterogeneity of sensors may be tamed at the model level. At least, this has so far been confirmed in our experiment for the case of noise sensing. We are thus maintaining a calibration database where we assess the bias of a particular model compared to a reference sound level meter. Obviously, the number of Android phones is very large and we therefore organize "calibration parties" to meet with our users and calibrate their phones.

In general, the calibration of sensors across devices is an open research question. However, our experience so far suggests that combining data assimilation with model-specific calibration allows gathering valuable data from MPS. Hence, MPS stands as a relevant complement –*not* an alternative– to observations using highly accurate -and expensive- fixed sensors.

5.3 About Energy Efficiency and Timeliness

MPS is energy-hungry due to continuous sensing and communication. As evidenced in the literature [37, 34], a way to reduce energy consumption is to implement energy-delay tradeoffs. We have implemented two versions of the GoFlow client: one sends the measurements after each observation (every 5 min by default); the other buffers a series of 10 measurements before sending them (hence every 50 min by default). In both cases, if there is no network connection at the time of emission, the measurements are sent at the next cycle.

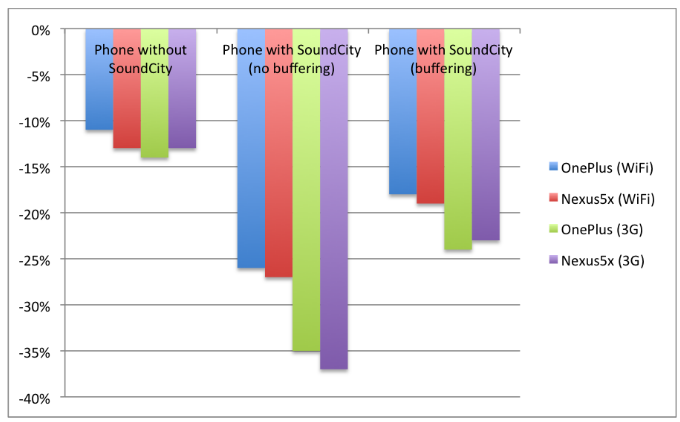

Figure 16: Battery depletion comparison per version.

Figure 16 compares the battery depletion of both versions and also in the absence of MPS for the OnePlus and Nexus5x models. We also compare the energy consumption of the 3G with WiFi network connection. For all the experiments, the phones were all initially charged at 80% (since the battery usage over the first 20% is not linear) and ran the application over the day from 10AM to 5PM. The phones were activated on a regular basis to avoid the deep sleep mode of the OS. The phones were set close to a window to limit 3G connection issues, and they were only running SoundCity (in addition to the default system functions). We also ran intensive measurements: measurements were taken every minute and thus sent every 1 min or 5 min, depending on the version. We see that in the absence of buffering, the MPS app consumes twice as much battery as in the absence of the app when the network is the WiFi. Using 3G network increases the battery depletion rate by 50%. Buffering then allows significant energy saving since it increases by less than 50% the battery depletion with the WiFi connection. It is important to note that the experiment focuses on battery depletion due to SoundCity only and further delivers measurements to the server at a frequency that is 10 times that of the default setting of the app. Hence, in practice, the energy consumption due to SoundCity is much lower, while the experiment evidences the energy gain due to buffering.

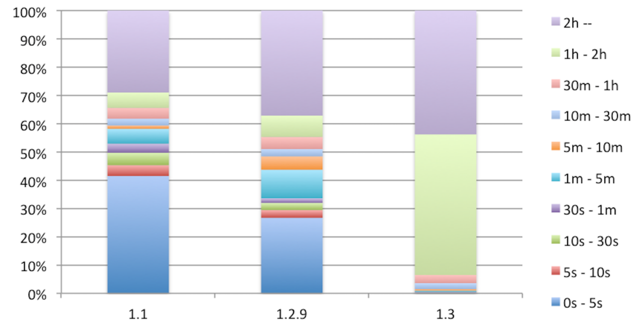

Figure 17: Transmission delay vs energy efficiency.

Over our 10 months experiment, we have released 3 versions of the MPS app: v1.1 in July 2015 without buffering, v1.2.9 in November 2015 still without buffering, but with optimized use of RabbitMQ, and v1.3 in April 2016 with buffering. Figure 17 analyzes the resulting energy versus delay tradeoff; it provides the distribution of the transmission delays for the set of collected observations. We recall that in the non-buffering case, the client sends observations every 5 min. Focusing on v1.2(.9) for the non-buffering version, we note that 35% of the measurements reaches the server after 2 hours, which stresses the disconnection of devices. We also note that nearly 30% of the measurements reaches the server within 10 s. In the case of the buffering version, 45% of the measurements reaches the server after 2 hours and most of the rest within one hour, which is the default frequency set for the app. Hence, the buffering version moderately alters the worst case that is 2-hour delay or more. Then, the 1-hour delay is due to the default buffering value, which fits well our target application. Concluding, the buffering duration may be tuned according to the application, again regarding the necessary trading of energy versus timeliness.

6. ANALYZING MPS EFFICIENCY FROM A SOCIAL PERSPECTIVE

From a technical perspective, MPS confirms to be a relevant tool to sense the physical world: 40% of the contributed observations are sufficiently localized and may be calibrated toward accuracy. In addition, the MPS activity does not introduce significant battery drain. However, the value of the MPS system highly depends on the involved crowd.

6.1 Analyzing User Participation across Time

Figure 18 shows the daily distribution of the crowd contributions, still distinguishing the contributions associated with each of the top 20 models using different colors. We notice an overall pattern with the highest participation from 10AM to 9PM. However, analysis at a finer grain shows high heterogeneity across users. For instance, Figure 19 shows the daily usage pattern of users owning a One Plus One phone;

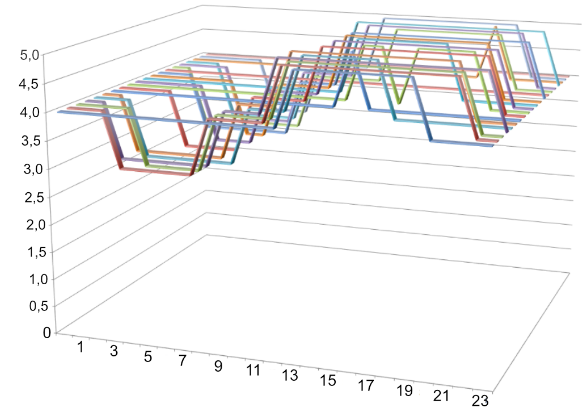

Figure 18: Daily distribution (%) of measurements for the top 20 models.

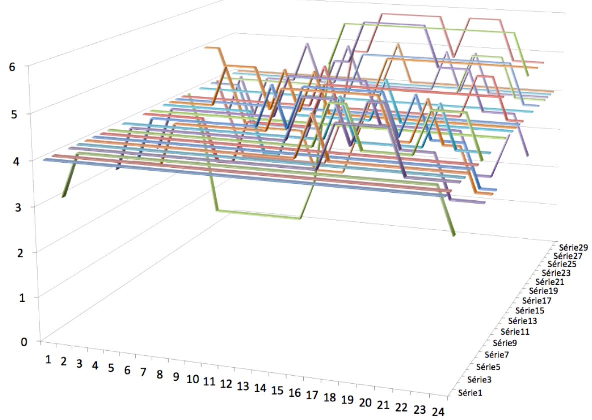

Figure 19: Diversity across users: Illustration with daily distributions (%) of measurements from One Plus One users.

we see a quite large diversity. We conclude that crowdsensing enables collecting contributions over the 24 hours range, thanks to the high heterogeneity of the crowd.

6.2 Opportunistic vs Participatory Sensing

SoundCity enables both opportunistic and participatory sensing: the former enables to collect a high number of contributions due to its background activity, while the latter promotes higher quality contributions.

Figure 20 (left) shows the the distribution of the location providers in the case of opportunistic sensing. Figure 20 (middle) - resp. 20 (right) - provides the distribution of the location providers for the participatory case in the manual mode (the user asks for a measurement on the home page) –resp. the journey mode (the user engages in the collection of observations along a path).

We notice that participatory sensing enables collecting a larger set of GPS-based location by more than 20% in the manual mode and by 40% in the journey mode. Since the Journey mode was released only recently, it is difficult to draw definite conclusion since the number of measurements

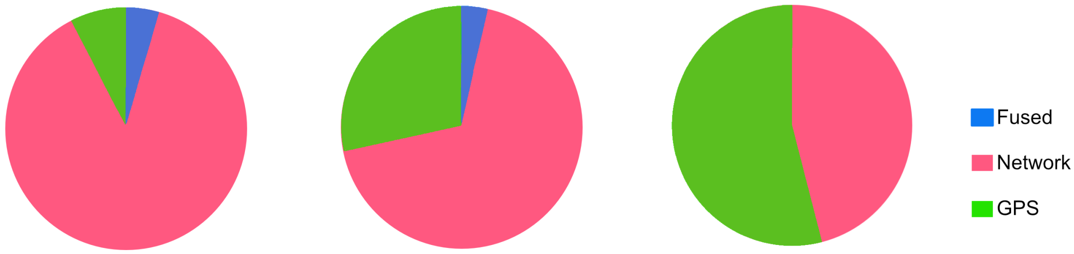

Figure 20: Location providers & opportunistic (left) - manual (middle) - journey (right) - sensing.

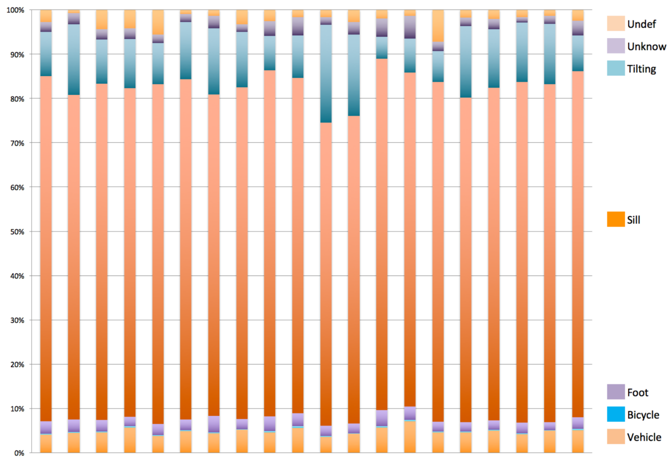

Figure 21: Distribution (%) of user activities for the top 20 models.

is comparatively much smaller. However, it is very likely that the opportunistic sensing is producing better quality observations. Still, this is at the expense of the number of contributions in space and time. Our ongoing work is about assessing the respective values of each mode in the context of data assimilation, i.e., assessing which contributed observation are the most significant to correct pollution maps using data assimilation techniques.

6.3 Analyzing User Activities across Time

The activity of the user –still vs moving– is also an important information for MPS systems. Figure 21 shows the distribution of the activities of the SoundCity users owning the top 20 phone models. When enlarged, the picture shows the distribution of the following activities: undefined, unknown, tilting, still, foot, bicycle, vehicle. The activity cannot be characterized for 20% of the time (i.e., the accuracy confidence is less than 80%), which indicates that activity tracking is rather efficient. We see that the population is moving for less than 10% of the time and is therefore remaining still for 70% of the time. Overall, this suggests that attracting a large crowd is necessary to be able to cover a large area.

7. TAKE-AWAY FOR MPS MIDDLEWARE & APP DESIGN

MPS is a promising approach to sensing the physical world on a very large scale over space and time. However, its efficiency is challenged from both a technical and social perspectives, the highest challenge being to overcome the a priori heterogeneity of the system in both dimensions. We have analyzed such heterogeneity in practice through the deployment of a MPS system dedicated to noise pollution monitoring.

Key outcomes are as follows:

- While contributors exhibit high heterogeneity regarding the accuracy of their sensors, they overall exhibit similar patterns. Location accuracy leads to discard about 60% of the observations and most observations are in the $[20 - 50]$ meters accuracy range. Noise sensing accuracy varies but calibration may be achieved per model rather than per device; calibration may then combine a number of techniques from comparison using a high-quality reference sensor to automated techniques leveraging assimilation and machine learning. Although our experiment is focused on noise sensing, we may expect similar results for other physical sensors. Overall, MPS allows collecting and assimilating relevant observations/measures. Still, the number of contributed measures by the MPS system needs to be high enough to overcome the low accuracy of the phone sensors.
- Although not specifically related to heterogeneity, energy efficiency is critical for the adoption of MPS. Our study confirms that energy-delay tradeoffs is a valuable approach; hence, the middleware must enable the buffering of the observations while the frequency of the transfers must be tuned by the application. Still, we notice that 30% of the observations reach the server after 2 hours even when observations are not buffered and are sent every 5mns, which indicates long periods of disconnection. Hence, if the timeliness of the observation is critical, then participatory sensing is most likely the approach to follow to ensure that the user is conscious about the sensing and activates appropriate network connection.
- The heterogeneity of the contributing crowd is obvious. However, it turns out to be an asset rather than a shortcoming of MPS. Indeed, the crowd overall exhibits similar contribution patterns across time. However, in the detail, each individual has different contribution patterns. This allows for the collection of complementary contributions over the whole day.
- The users appear to be still most of the time, while the user's activity cannot be qualified for 20% of the observations. This should be accounted for in the design of mobility-dependent MPS.

- One design issue that arises for MPS is whether to promote participatory or opportunistic sensing. It is our belief that a system (and thus supporting app) must support both. This enables to collect as many observations as possible from a large diversity of people, while participatory sensing guarantees contributions of higher quality.

Concluding, our empirical analysis so far shows that the heterogeneity of the crowd, both technical and social, is more a strength rather than a weakness of MPS; the crowd heterogeneity allows collecting relevant data despite the inherent shortcomings of low cost sensors.

8. CONCLUSIONS

MPS has been on the research agenda for more than a decade, leading to the introduction of a number of techniques and platforms to ensure resource-efficiency and also promote crowd participation. However, it remains difficult to assess the significance of the various approaches in practice. Also, most middleware-related works focus on the technical challenges of MPS, while social challenges are equally important. Indeed, MPS is successful only if it attracts a relevant crowd and if the crowd contributes with high quality values. To better understand the overall –both technical and social– design space of MPS, we have undertaken an urban-scale experiment related to noise pollution awareness, with the support of the city of Paris.

This paper has introduced the supporting MPS system and related SoundCity app for noise pollution monitoring. We have then reported the analysis of our empirical study, which accounts for observations collected over a period of 10 months. In total, we have gathered over 45M observations, while this paper concentrated on the 23M observations contributed by the 20 most popular phone models. Results show that the crowd is heterogeneous but that is beneficial rather than detrimental to the performance of the MPS system.

Future work should address the collection of qualitative data inputted by the user. It can be challenging to engage the users to the point where they would willingly provide qualitative feedback, e.g., on their individual perceptions of their environment at certain moments. The feedback mechanism should be easily accessible and yet not invasive. Also, it might be beneficial to trigger it at some proper times, to be determined by the available quantitative information. In the case of SoundCity, user feedback at locations where the noise is accurately measured would be helpful to build an individual profile of sensitivity to noise.

The quantitative data should also receive further attention because its quality may be significantly improved with adequate analysis. We expect crowd-sensing to be accompanied with crowd-calibration which calibrates individual devices based on each other's devices. Some missing data for one individual user may also be inferred from the crowd measurements, and the sensing times and locations could be chosen accordingly, with the objective of collecting the most informative data while limiting energy consumption.

Besides longitudinal analysis, advanced spatial-temporal processing of all the data can produce unique information about the entire environment, especially in urban areas where complex, fast varying (in time and space) phenomena continuously occur. One research direction is the development of adapted data assimilation algorithms that merge traditional simulations of the urban area, preferably down to street-level, with fixed and mobile observations of various accuracies. The amount of observations to assimilate, the moving sensors and the lack of measurement protocol raise a number of issues that classical algorithms do not take into account.

Acknowledgment. The authors would like to thank Rajiv Bhatia and Estelle Hallaert for their contribution to the production of the noise-related maps of San Francisco. The authors further acknowledge the support of: the Inria@SiliconValley program (project.inria.fr/siliconvalley), the CityLab Inria Project Lab (citylab.inria.fr), the Env&You / Ambiciti innovation activity (ambiciti.io) funded by EIT Digital (eitdigital.eu), and the European project FIESTA-IoT (http://www.fiesta-iot.eu) funded by the European Union's Horizon 2020 Programme with the Grant Agreement No. CNECT-ICT-643943.

9. REFERENCES

[1] H. Blunck, N. O. Bouvin, T. Franke, K. Grønbæk, M. B. Kjaergaard, P. Lukowicz, and M. Wüstenberg. On heterogeneity in mobile sensing applications aiming at representative data collection. In *Proceedings of the 2013 ACM Conference on Pervasive and Ubiquitous Computing Adjunct Publication*, UbiComp '13 Adjunct, pages 1087–1098, New York, NY, USA, 2013. ACM.

[2] F. Bouttier and P. Courtier. Data assimilation concepts and methods. Meteorological training course lecture series, ECMWF, 1999.

[3] N. Brouwers and K. Langendoen. Pogo, a middleware for mobile phone sensing. In *Proceedings of the 13th International Middleware Conference*, Middleware '12, pages 21–40, New York, NY, USA, 2012. Springer-Verlag New York, Inc.

[4] G. Cardone, A. Corradi, L. Foschini, and R. Ianniello. Participact: A large-scale crowdsensing platform. *IEEE Transactions on Emerging Topics in Computing*, 4(1):21–32, Jan 2016.

[5] D. Christin, A. Reinhardt, S. S. Kanhere, and M. Hollick. A survey on privacy in mobile participatory sensing applications. *J. Syst. Softw.*, 84(11):1928–1946, Nov. 2011.

[6] S. Consolvo, D. W. McDonald, T. Toscos, M. Y. Chen, J. Froehlich, B. Harrison, P. Klasnja, A. LaMarca, L. LeGrand, R. Libby, I. Smith, and J. A. Landay. Activity sensing in the wild: A field trial of ubifit garden. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, pages 1797–1806, New York, NY, USA, 2008. ACM.

[7] J. Corburn. *Street science: Community knowledge and environmental health justice*. The MIT Press, 2005.

[8] T. Dao, I. Singh, H. V. Madhyastha, S. V. Krishnamurthy, G. Cao, and P. Mohapatra. Tide: A user-centric tool for identifying energy hungry applications on smartphones. In *Distributed Computing Systems (ICDCS), 2015 IEEE 35th International Conference on*, pages 123–132, June 2015.

[9] S. M. Erfani, S. Karunasekera, C. Leckie, and

U. Parampalli. Privacy-preserving data aggregation in participatory sensing networks. In *Intelligent Sensors, Sensor Networks and Information Processing, 2013 IEEE Eighth International Conference on*, pages 165–170, April 2013.

[10] F. Fischer. *Citizens, experts, and the environment: The politics of local knowledge*. Duke University Press, 2000.

[11] B. J. Fogg. *Persuasive Technology: Using Computers to Change What We Think and Do*. Science & Technology Books, 1 edition, 2002.

[12] R. K. Ganti, F. Ye, and H. Lei. Mobile crowdsensing: current state and future challenges. *IEEE Communications Magazine*, 49(11):32–39, November 2011.

[13] S. Gaonkar, J. Li, R. R. Choudhury, L. Cox, and A. Schmidt. Micro-blog: Sharing and querying content through mobile phones and social participation. In *Proceedings of the 6th International Conference on Mobile Systems, Applications, and Services*, MobiSys '08, pages 174–186, New York, NY, USA, 2008. ACM.

[14] S. Hachem, V. Mallet, R. Ventura, A. Pathak, V. Issarny, P. G. Raverdy, and R. Bhatia. Monitoring noise pollution using the urban civics middleware. In *Big Data Computing Service and Applications (BigDataService), 2015 IEEE First International Conference on*, pages 52–61, March 2015.

[15] S. Hachem, G. Mathioudakis, A. Pathak, V. Issarny, and R. Bhatia. Sense2Health: A Quantified Self Application for Monitoring Personal Exposure to Environmental Pollution. In *SENSORNETS 2015*, Angers, France, Feb. 2015.

[16] S. Hachem, A. Pathak, and V. Issarny. Service-oriented middleware for large-scale mobile participatory sensing. *Pervasive Mob. Comput.*, 10:66–82, Feb. 2014.

[17] J. Hamm, A. C. Champion, G. Chen, M. Belkin, and D. Xuan. Crowd-ml: A privacy-preserving learning framework for a crowd of smart devices. In *Distributed Computing Systems (ICDCS), 2015 IEEE 35th International Conference on*, pages 11–20, June 2015.

[18] J. C. Herrera, D. B. Work, R. Herring, X. J. Ban, Q. Jacobson, and A. M. Bayen. Evaluation of traffic data obtained via gps-enabled mobile phones: The mobile century field experiment. *Transportation Research Part C: Emerging Technologies*, 18(4):568 – 583, 2010.

[19] W. Z. Khan, Y. Xiang, M. Y. Aalsalem, and Q. Arshad. Mobile phone sensing systems: A survey. *IEEE Communications Surveys Tutorials*, 15(1):402–427, First 2013.

[20] M. H. Krieger, R. Govindan, M.-R. Ra, and J. Paek. Commentary: Pervasive urban media documentation. *Journal of Planning Education and Research (JPER)*, 29(1):114–116, September 2009.

[21] K. Kunze, P. Lukowicz, H. Junker, and G. Troster. Where am i: Recognizing on-body positions of wearable sensors. In *LOCA'04: International Workshop on Location and Context-Awareness*, pages 264–275. Springer-Verlag, 2005.

[22] N. D. Lane, Y. Chon, L. Zhou, Y. Zhang, F. Li, D. Kim, G. Ding, F. Zhao, and H. Cha. Piggyback crowdsensing (pcs): Energy efficient crowdsourcing of mobile sensor data by exploiting smartphone app opportunities. In *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems*, SenSys '13, pages 7:1–7:14, New York, NY, USA, 2013. ACM.

[23] N. D. Lane, S. B. Eisenman, M. Musolesi, E. Miluzzo, and A. T. Campbell. Urban sensing systems: Opportunistic or participatory? In *Proceedings of the 9th Workshop on Mobile Computing Systems and Applications*, HotMobile '08, pages 11–16, New York, NY, USA, 2008. ACM.

[24] N. D. Lane and P. Georgiev. Can deep learning revolutionize mobile sensing? In *Proceedings of the 16th International Workshop on Mobile Computing Systems and Applications*, HotMobile '15, pages 117–122, New York, NY, USA, 2015. ACM.

[25] N. D. Lane, E. Miluzzo, H. Lu, D. Peebles, T. Choudhury, and A. T. Campbell. A survey of mobile phone sensing. *IEEE Communications Magazine*, 48(9):140–150, Sept 2010.

[26] D. Li, S. Hao, W. G. J. Halfond, and R. Govindan. Calculating source line level energy information for android applications. In *Proceedings of the 2013 International Symposium on Software Testing and Analysis*, ISSTA 2013, pages 78–89, New York, NY, USA, 2013. ACM.

[27] Y. Li, Q. Li, J. Gao, L. Su, B. Zhao, W. Fan, and J. Han. On the discovery of evolving truth. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '15, pages 675–684, New York, NY, USA, 2015. ACM.

[28] C. Meng, W. Jiang, Y. Li, J. Gao, L. Su, H. Ding, and Y. Cheng. Truth discovery on crowd sensing of correlated entities. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*, SenSys '15, pages 169–182, New York, NY, USA, 2015. ACM.

[29] C. Miao, W. Jiang, L. Su, Y. Li, S. Guo, Z. Qin, H. Xiao, J. Gao, and K. Ren. Cloud-enabled privacy-preserving truth discovery in crowd sensing systems. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*, SenSys '15, pages 183–196, New York, NY, USA, 2015. ACM.

[30] E. Miluzzo, C. T. Cornelius, A. Ramaswamy, T. Choudhury, Z. Liu, and A. T. Campbell. Darwin phones: The evolution of sensing and inference on mobile phones. In *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services*, MobiSys '10, pages 5–20, New York, NY, USA, 2010. ACM.

[31] E. Miluzzo, N. D. Lane, K. Fodor, R. Peterson, H. Lu, M. Musolesi, S. B. Eisenman, X. Zheng, and A. T. Campbell. Sensing meets mobile social networks: The design, implementation and evaluation of the cenceme application. In *Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems*, SenSys '08, pages 337–350, New York, NY, USA, 2008. ACM.

[32] C. Min, Y. Lee, C. Yoo, S. Kang, S. Choi, P. Park, I. Hwang, Y. Ju, S. Choi, and J. Song. Powerforecaster: Predicting smartphone power impact

of continuous sensing applications at pre-installation time. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*, SenSys '15, pages 31–44, New York, NY, USA, 2015. ACM.

[33] M. Mun, S. Reddy, K. Shilton, N. Yau, J. Burke, D. Estrin, M. Hansen, E. Howard, R. West, and P. Boda. Peir: the personal environmental impact report, as a platform for participatory sensing systems research. In *in Proc. ACM/USENIX Int. Conf. Mobile Systems, Applications, and Services (MobiSys) Krakow*, 2009.

[34] N. Nikzad, M. Radi, O. Chipara, and W. G. Griswold. Managing the energy-delay tradeoff in mobile applications with tempus. In *Proceedings of the 16th Annual Middleware Conference*, Middleware '15, pages 259–270, New York, NY, USA, 2015. ACM.

[35] C. Qiu and M. W. Mutka. iframe: Dynamic indoor map construction through automatic mobile sensing. In *2016 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 1–9, March 2016.

[36] M.-R. Ra, B. Liu, T. F. La Porta, and R. Govindan. Medusa: A programming framework for crowd-sensing applications. In *Proceedings of the 10th International Conference on Mobile Systems, Applications, and Services*, MobiSys '12, pages 337–350, New York, NY, USA, 2012. ACM.

[37] M.-R. Ra, J. Paek, A. B. Sharma, R. Govindan, M. H. Krieger, and M. J. Neely. Energy-delay tradeoffs in smartphone applications. In *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services*, MobiSys '10, pages 255–270, New York, NY, USA, 2010. ACM.

[38] R. K. Rana, C. T. Chou, N. Bulusu, S. S. Kanhere, and W. Hu. Ear-phone: A context-aware noise mapping using smart phones. *Pervasive and Mobile Computing*, 2015.

[39] J. P. Rula and F. E. Bustamante. Crowdsensing under (soft) control. In *2015 IEEE Conference on Computer Communications, INFOCOM 2015, Kowloon, Hong Kong, April 26 - May 1, 2015*, pages 2236–2244, 2015.

[40] S. A. Russel. *Diary of a Citizen Scientist*. Oregon State University Press, 2014.

[41] A. Stisen, H. Blunck, S. Bhattacharya, T. S. Prentow, M. B. Kjærgaard, A. Dey, T. Sonne, and M. M. Jensen. Smart devices are different: Assessing and mitigating mobile sensing heterogeneities for activity recognition. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*, SenSys '15, pages 127–140, New York, NY, USA, 2015. ACM.

[42] A. Tilloy, V. Mallet, D. Poulet, C. Pesin, and F. Brocheton. BLUE-based NO_2 data assimilation at urban scale. *Journal of Geophysical Research*, 118(4):2,031–2,040, 2013.

[43] I. J. Vergara-Laurens, D. Mendez, and M. A. Labrador. Privacy, quality of information, and energy consumption in participatory sensing systems. In *Pervasive Computing and Communications (PerCom), 2014 IEEE International Conference on*, pages 199–207, March 2014.

[44] WHO. Guidelines for community noise. Technical report, WHO, 1999.

[45] H. Xiong, D. Zhang, L. Wang, J. P. Gibson, and J. Zhu. Eemc: Enabling energy-efficient mobile crowdsensing with anonymous participants. *ACM Trans. Intell. Syst. Technol.*, 6(3):39:1–39:26, Apr. 2015.

[46] D. Yang, G. Xue, X. Fang, and J. Tang. Crowdsourcing to smartphones: Incentive mechanism design for mobile phone sensing. In *Proceedings of the 18th Annual International Conference on Mobile Computing and Networking*, Mobicom '12, pages 173–184, New York, NY, USA, 2012. ACM.