Sensing as a Service: Challenges, Solutions and Future Directions

Xiang Sheng, *Member, IEEE*, Jian Tang, *Member, IEEE*, Xuejie Xiao, and Guoliang Xue, *Fellow, IEEE*

***Abstract*— Sensors on (or attached to) mobile phones can enable attractive sensing applications in different domains, such as environmental monitoring, social networking, healthcare, transportation, etc. We introduce a new concept, sensing as a service (S^2aaS), i.e., providing sensing services using mobile phones via a cloud computing system. An S^2aaS cloud needs to meet the following requirements: 1) it must be able to support various mobile phone sensing applications on different smartphone platforms; 2) it must be energy-efficient; and 3) it must have effective incentive mechanisms that can be used to attract mobile users to participate in sensing activities. In this vision paper, we identify unique challenges of designing and implementing an S^2aaS cloud, review existing systems and methods, present viable solutions, and point out future research directions.**

***Index Terms*— Mobile phone sensing, sensing as a service, cloud computing, energy-efficiency, incentive mechanisms.**

I. Introduction

MOBILE phones have evolved as key electronic devices for communications, computing and entertainment, and have become an important part of people's daily life. Most current mobile phones (such as iPhone, Samsung's Android phones, etc.) are equipped with a rich set of embedded sensors such as camera, GPS, WiFi/3G/4G radios, accelerometer, digital compass, gyroscope, microphone and so on. Moreover, external sensors (such heartbeat sensor, air pollution sensor, etc) can also be connected to a mobile phone via its Bluetooth interface. These sensors can enable attractive sensing applications in various domains such as environmental monitoring, social networking, healthcare, transportation, etc. Mobile phone sensing has been studied by a few recent works [17], most of which, however, presented the design and implementation of a mobile phone sensing system for a particular application.

In this paper, we propose to leverage emerging cloud computing model to provide various sensing services using mobile phones for a large number of cloud users and introduce a new concept: *Sensing as a Service (S^2aaS)*. A typical S^2aaS cloud is illustrated in Fig. 1. In an S^2aaS cloud, multiple sensing servers (as shown in the figure) can be deployed to handle sensing requests from different locations. When a clouduser initiates a sensing request through a web front-end from either a mobile phone or a computer (desktop/laptop), the request will be forwarded to a sensing server which will then push the request to a subset of mobile phones that happen to be in the area of interest. The corresponding sensing task will be fulfilled by these mobile phones. The sensed data will then be collected by a sensing server, stored in the database and returned to the cloud user who requests the service.

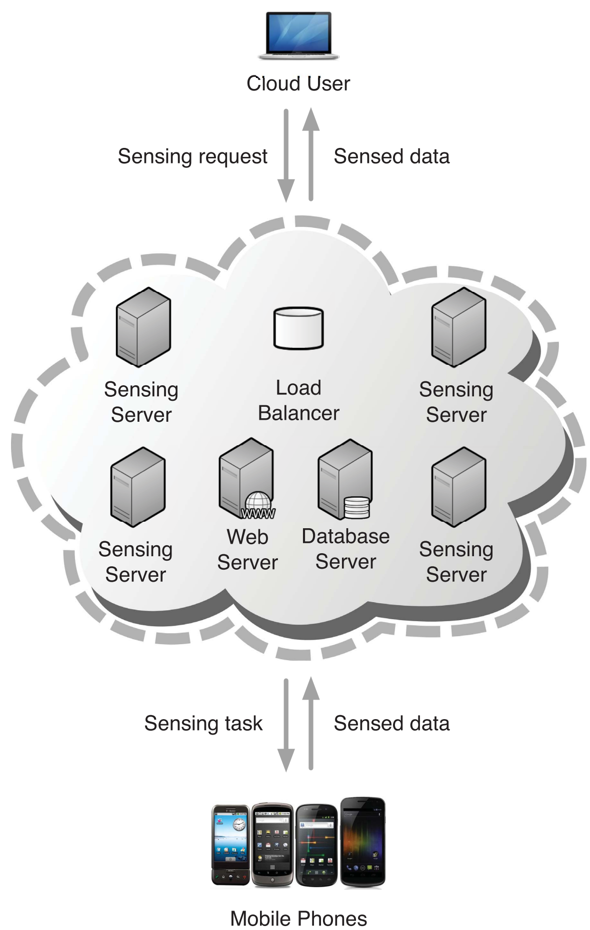


Fig. 1. An S^2aaS cloud.

Manuscript received February 8, 2013; revised April 30, 2013; accepted May 3, 2013. Date of publication May 13, 2013; date of current version August 28, 2013. This work was supported in part by NSF under Grants 1218203 and 1217611. The associate editor coordinating the review of this paper and approving it for publication was Dr. Honggang Wang.

X. Sheng, J. Tang, and X. Xiao are with the Department of Electrical Engineering and Computer Science, Syracuse University, Syracuse, NY 13244 USA (e-mail: xsheng@syr.edu; jtang02@syr.edu; xxiao04g@syr.edu).

G. Xue is with the School of Computing, Informatics, and Decision Systems Engineering, Arizona State University, Tempe, AZ 85281 USA (e-mail: xue@asu.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSEN.2013.2262677

An interesting feature of such a system is that a mobile phone user (or simply mobile user) can be not only a cloud (service) user who can request sensing services from the cloud but also a service provider who fulfills sensing tasks according to sensing requests from other mobile users.

There are primarily two mobile phone sensing paradigms [17]: *Participatory Sensing and Opportunistic Sensing*. In participatory sensing, mobile users actively engage in sensing activities by manually determining how, when, what, and where to sense. In opportunistic sensing, sensing activities are fully automated without the involvement of mobile users.

To provide sensing services for a large number of cloud users with different needs, an S^{2}aaS cloud must be able to support various participatory sensing and opportunistic sensing applications on different smartphone platforms.

Performing sensing tasks may consume a significant amount of energy of a mobile phone. Therefore, without carefully managing very limited energy resources on mobile phones, users may end up with an awkward situation after performing a few sensing tasks, in which phones run out of energy when they are needed to make phone calls. Moreover, most mobile phone sensing applications are location-dependent. If energy-hungry GPS is turned on during the whole sensing procedure, the battery may be drained very quickly. There is a large space for energy savings. However, fundamental energy-efficient resource management problems have not been well studied for mobile phone sensing. Furthermore, unlike a traditional sensor network which is usually operated by a single organization, mobile phones and their sensors are owned and controlled by different individual users. Hence, the mobility is totally uncontrollable and hard to predict.

In addition, an S^{2}aaS cloud is truly a crowdsourcing system that depends on mobile users to provide data. Without adequate user participation, it is impossible to achieve any level of service quality. However, by participating in sensing activities, mobile users will consume resources on their own phones, such as battery and computing power. More importantly, participating users will also expose themselves to potential privacy threats. Hence, a mobile user would not be interested in participating in mobile phone sensing, unless he/she receives a satisfying reward to compensate his/her resource consumption and potential privacy breach. A fundamental problem is how to provide incentives to attract these selfish mobile users to participate in sensing activities, which, to the best of our knowledge, has not been well addressed yet.

Developing a *unified, green and incentive* cloud computing system for mobile phone sensing is quite challenging. In this paper, we identify *unique challenges* in designing and implementing an S^{2}aaS system, review existing systems and methods, present viable solutions, and point out future research directions. Even though (sensed) data processing and analysis, and security and privacy are critical issues, they are out of scope of this paper since we believe they are common issues in sensor networks and mobile cloud computing systems. We aim to address research challenges unique to developing an S^{2}aaS cloud here.

The following important and special issues need to be carefully addressed for designing and implementing an S^{2}aaS cloud:

1) The system must be general enough such that it can support various opportunistic and participatory sensing applications (which may even involve a large variety of sensors) over different mobile phone platforms (Android, iOS, Blackberry, Windows Mobile), and there is very little overhead to launch a new sensing application/service.
2) The system can be easily and quickly reconfigured to replace old inefficient algorithms or policies with new ones.
3) Sensing energy consumption should be minimized such that mobile phones can undertake sensing tasks, and in the meanwhile, can still fulfill its regular duties, such as making phone calls, sending/receving emails, browsing webpages, etc.
4) The system must have effective incentive mechanisms to attract mobile phone users to participate in sensing activities.

The rest of the paper is organized as follows: In Section II, we discuss the design and implementation of the software architecture of an S^{2}aaS system. Section III is focused on energy-efficient sensing task management. The incentive mechanism design is discussed in Section IV. We point out future research directions in Section V and conclude the paper in Section VI.

II. Architecture Design and Implementation

Recently, research efforts have been made to develop systems to support mobile phone sensing in different areas such as environmental monitoring, social networking, healthcare, transportation, etc [17].

A mobile phone sensing application, called PEIR (Personal Environmental Impact Report) [30], was developed to use location data sampled from mobile phones everyday to calculate personalized estimates of environmental impact and exposure. In [35], Rena *et al.* proposed the design and implementation of the "ear-phone" system which is a mobile phone sensing system that can be used to monitor the background noises. Each mobile device in the ear-phone system has a signal processing module for analyzing the background sound level. A central server is used to manage data collected by mobile phones and create an urban noise map. In addition, Lu *et al.* proposed SoundSense in [22], a scalable framework for modeling sound events on mobile phones, which uses a combination of supervised and unsupervised learning techniques to classify both general sound types (e.g., music, voice) and discover novel sound events specific to individual users. SoundSense was implemented on the iPhone and represents the first sound sensing system specifically designed to work on resource limited mobile phones.

In [27], Miluzzo *et al.* presented the design, implementation, evaluation, and user experiences of a mobile phone sensing system called CenceMe. CenceMe represents the first system that combines the inference of the presence of individuals using sensor-enabled mobile phones with sharing of this information through social networking applications such as

Facebook and MySpace. Micro-blogs [13] is another mobile phone sensing system developed for social networking. Similar to other blog systems, users could upload a multimedia blog annotated with a geo-tag to a micro-blog server. Information could be queried and browsed via a digital map service, such as Google Map. This system is also able to dispatch sensing requests to a set of phones in the region of interest.

Mobile phone sensing has the potential to collect continuously sensed data for health and wellness analysis. UbiFit Garden [6] is a mobile phone sensing system jointly developed by Intel and University of Washington, which uses small inexpensive on-body sensors and machine learning techniques on activity modeling to infer people's activities throughout everyday life. This system captures levels of physical activity and relates this information to personal health goals when presenting feedback to the user for encouraging physical activity. In another project [18], Lee *et al.* connected the ZigBee-based built-in blood glucometer to mobile phones. The measured blood glucose could be transmitted directly to the web. Leijdekkers and Gay [20], developed a heart attack self-test system. In this system, electrocardiogram sensors are wirelessly connected with a mobile phone, which can collect a mobile user's symptoms and send them to a mobile phone application. The mobile application can then analyze the streaming data to detect the onset of a heart attack. If the application assesses that the user is at risk, it will urge him/her to call the emergency service immediately. If the user has a cardiac arrest, the application will automatically determine the current his/her location and alert the ambulance service.

In [29], a mobile phone sensing system, Nericell, was presented for monitoring road and traffic conditions in a city, which uses the accelerometer, microphone, GSM radio, and/or GPS sensors in a mobile phone to detect potholes, bumps, braking and honking. In [41], Thiagarajan *et al.* presented VTrack, a mobile sensing system which tracks the traffic delays and congestions. In VTrack, drivers' smartphones are used to provide spatio-temporal samples to monitor traffic delays. Specifically, VTrack uses WiFi signals to estimate the driver's locations, along with a hidden Markov model based map matching method, to identify the road segments and the time spent on these segments.

General-purpose mobile phone sensing systems have also been introduced by a few recent works. In [23], the Bubble-Sensing framework was proposed to bind sensing tasks to a specific physical locations of interest. This framework can be used for both opportunistic and participatory sensing. Mobile users are selected to collecting information, such as background noise and photo. A "bubble" task can be bound to the location of interest and remains active for a period of time. This sensing framework could be used to keep a living documentary of places of interest. Cornelius *et al.* introduced AnonySense in [5], which is a privacy-aware system for mobile phone sensing. Designed for community-oriented information services, AnonySense distributes sensing tasks among a set of anonymous mobile devices, and collect verified yet anonymous sensing results. The proposed system aims at addressing the privacy concerns in large-scale sensing applications. In [8], Das *et al.* presented a Platform for Remote Sensing using Smartphones (PRISM). PRISM enables third-party applications to be packaged as executable binaries and push them automatically to an appropriate set of phones. The mobile phone end of PRISM will then execute the received executable file.

However, these existing systems have the following problems: 1) Most existing mobile phone sensing systems, such as systems presented in [6], [13], [18], [20], [22], [27], [29], [30], [35], [41] target a particular sensing application. Therefore, there is a large overhead for developing and maintaining such a special-purpose system. 2) Even though PRISM [8] is a general-purpose mobile sensing platform. However the sensing tasks delivered to mobile phones are packaged as executable binaries, which are platform-dependent (Windows Mobile only) and may cause security issues. 3) AnonySense [5] uses a customized, yet very limitedly-used Lisp dialect for implementation. Therefore, its applicability is very limited. 4) Important issues, energy-efficiency and incentive mechanism design, have not been addressed by these related works.

The following functionalities should be supported by an S^2aaS cloud: (1) *Web Interface*: It needs to provide a web interface for collecting sensing request information from cloud users, which can be accessed via a mobile device or a regular computer. (2) *Generating Sensing Tasks*: It needs to generate new sensing tasks in a standard format based on request information (e.g., what sensors to use, what data to collect, what is the area of interest, how many readings to collect, etc) collected from the web interface. (3) *Tracking Mobile Phones*: It needs to maintain important information of a list of mobile phones that are available for participating in sensing tasks, including locations, available sensors, residual energy, etc. Moreover, it needs to provide an interface between a sensing server and mobile phones for pushing sensing tasks to mobile phones and collecting sensed data from them. (4) *Recruiting Mobile Users*: It needs to recruit a set of mobile phone users to participate in sensing activities for each incoming sensing task using an incentive mechanism (discussed in Section IV). (5) *Scheduling Sensing Activities*: It needs to schedule sensing activities of the set of mobile phones recruited for each sensing task using a scheduling algorithm or policy (discussed in Section III). (6) *Managing Sensors*: an application needs to be deployed on each mobile phone to operate its sensors to perform the requested sensing actions, collect sensed data and send them to a sensing server. (7) *Processing and Storing Data*: It should be able to obtain sensed data from mobile phones, store some useful information to the database (for future use) and/or returns data reports to users.

Essentially, currently available web servers (such as Apache HTTP server [1]) and emerging database systems (such as BigTable [2]) can be used to provide the web interface and to store sensed data respectively. A sensing server needs to be developed to support functions (2)–(5). A mobile phone application needs to be developed to implement function (6).

The biggest challenge for building an S^2aaS cloud is to support various sensing applications over different mobile phone platforms. To this end, using scripting language to describe sensing tasks is a viable solution. Scripts written in a scripting language (rather than binary codes [8]) can be employed to describe every sensing task in a standard format and can then be pushed to mobile phones on which they will be executed with the help of an interpreter. Scripting languages can bring portability to the system such that the population of the sensing crowd can be effectively increased because sensing tasks described using scripts can run on hardware platforms with different CPU architectures, such as ARM, MIPS, SPARC, x86. Scripting languages can eliminate potential security threats by running the scripts in a sandbox that only allows them to use a *white* list of APIs such that they only interact with the hardware in the ways we trust. In addition, scripting languages can enable dynamic and flexible loading of programs on mobile phones because using a scripting language, an interpreter can be integrated into mobile applications to download and interpret the scripts on the fly, while, all binaries (packed as APK applications) need to be signed by Google before being loaded to users' mobile phones on the Android platform and a similar method is used on the iOS platform too.

Modular design and clearly defined interfaces will play a key role in supporting *re-configurability*. Every major functionality should be implemented as an independent module with well-defined interfaces to interact with other components. In order to improve *energy-efficiency*, efficient and practical algorithms need to be developed for mobile phone scheduling on the server side as well as sensing task scheduling on the mobile client side with the objective of minimizing and balancing energy consumption. This will be discussed in greater details in Section III. Furthermore, game-theoretic incentive mechanisms need to be developed for attracting user participation, which will be discussed in Section IV.

Even though scalability is not listed as a unique challenge above (since we believe it is a common issue for all kinds of cloud computing systems), it is also very important for building such a cloud system since the system may need to support millions of mobile phones around the whole world. How to deploy sensing servers and balancing sensing workloads among them should be carefully addressed in system design and implementation.

III. Energy-Efficient Sensing Task Management

In this section, we focus on energy-efficient sensing task management.

Energy-efficient issues have been studied in the context of mobile phone sensing recently [24], [31], [33], [46]. In [24], the authors presented the Jigsaw continuous sensing engine for mobile phone sensing, which balances performance needs and resource demands. Jigsaw comprises a set of sensing pipelines for accelerometer, microphone and GPS sensors, which are built in a plug-and-play manner to support resilient accelerometer data processing, smart admission control and adaptive pipeline processing. The authors of [31] presented several techniques to optimize the information uploading process for continuous sensing on mobile phones. The authors focused on the energy consumption related to data transmission over the cellular network. Based on the Markov chain prediction model, the authors claimed that location information should be used to save the uploading energy consumption.

Most mobile sensing tasks need location information. GPS currently is available for most smartphones, which can provide very accurate positioning information. However, GPS is really power hungry and a mobile phone's battery can be drained very quickly when the GPS is turned on. Research efforts have been made to find an energy-efficient localization method. Energy-efficient GPS-based location sensing methods were presented in [33], [46]. In [33], RAPS, a rate-adaptive localization system for smartphone applications, was presented. RAPS turns on GPS when and only when the location uncertainty exceeds a threshold. In RAPS, energy-efficient sensors, such as accelerometer and digital compass, are used to estimate user's movements and locations. In [46], authors proposed to increase sleep time of GPS sensors by using a piggybacking method to share location information among location-dependent applications.

However, these related works were all focused on a single mobile phone. We, however, are interested in minimizing total energy consumption of mobile phones in a cloud while still satisfying certain sensing coverage constraints. We believe that collaborative sensing is a viable solution, in which the cloud can be used for coordinating sensing activities.

Only few recent works addressed collaborative sensing with mobile phones. In [26], the authors presented analytical results on the rate of information reporting by uncontrolled mobile sensors needed to cover a given geographical area, and demonstrated the feasibility of using existing software and standard protocols for information reporting and retrieval to support a large system of uncontrolled mobile sensors using a testbed. In [42], the authors introduced mechanisms for automated mapping of urban areas that provide a virtual sensor abstraction to applications. They also proposed spatial and temporal coverage metrics for measuring the quality of acquired data. In [40], the authors proposed a protocol, Aquiba, that exploits opportunistic collaboration of pedestrians. Its performance was studied via simulations.

Collaborative sensing has been well studied for mobile sensor networks. Centralized and distributed collaborative sensing algorithms have been proposed in [4], [37], [39], [44], [45] to address different coverage and connectivity problems in mobile sensor networks (where sensor mobility can be controlled to achieve certain sensing coverage). Specifically, In [4], Chin *et al.* formally defined and evaluated exposure in mobile sensor networks with the presence of obstacles and noise. They also developed algorithms to find the upper and lower bounds on exposure using the time expansion graph. In [39], Tan *et al.* proposed to let mobile sensors collaborate with static sensors and move reactively to achieve the required detection performance. The accuracy has been shown to be improved as the measurements of mobile sensors have higher signal-to-noise ratios after movement. The authors also developed a sensor movement scheduling algorithm that can achieve near-optimal detection performance within a given delay bound. In [37], Saipulla *et al.* explored the fundamental limits of sensor mobility on barrier coverage, studied how to efficiently improve barrier coverage using mobile sensors with limited mobility, and presented a sensor mobility scheme that constructs the maximum number of barriers with minimum sensor moving distance. Several distributed algorithms were presented for a sensing coverage problem in [44], which do not need any location or distance information. In [45], Zhou *et al.* considered how to deploy mobile sensors into an existing sensor network to enhance its connectivity and coverage, and presented a dynamic programming algorithm and a virtual force based heuristic algorithm under the assumption that each sensor is equipped with GPS. A distributed mobility management scheme for mobile sensor networks was presented in [47]. The proposed scheme considers node movement decisions as part of a distributed optimization problem which integrates mobility-enhanced improvement in the quality of target tracking data with the associated negative consequences of increased energy consumption due to locomotion, potential loss of network connectivity, and loss of sensing coverage.

The algorithms proposed for a mobile sensor network cannot be applied to an S^2aaS system because of uncontrollable mobility of mobile users. However, we believe that collaborative sensing can still be used in mobile phone sensing to reduce sensing redundancy and improve energy-efficiency. We need to consider a fundamental problem for mobile phone sensing (which has not been well addressed yet): given a set of target points or a target region, a set of mobile phones and a deadline, find a sensing schedule (which specifies when to sense for each mobile phone) such that the total energy consumption is minimized subject to a coverage constraint. Scheduling algorithms that solve this problem can be used on sensing servers to schedule sensing activities of mobile phones (recruited using an incentive mechanism). Note that sensing scheduling algorithms will only be used for opportunistic sensing applications since mobile users manually control sensing activities in participatory sensing applications.

In our preliminary work [38], we studied the feasibility and effectiveness of applying the collaborative sensing approach to mobile phone sensing applications. We tried to answer this question: how much energy saving can potentially be achieved by using a sensing platform (such as a cloud system) to coordinate sensing activities of mobile phones? By solving the optimization problem described above under a strong assumption that the moving trajectory of each mobile user is known in advance, we came up with a graph model, Virtual Sensor Graph (VSG) to assist problem solving [38]. The importance of the VSG lies in the fact that the sensing scheduling problem described above can be transformed to a minimum-cost-flow-like problem. We then proposed an Linear Programming (LP) based algorithm to obtain minimum energy sensing schedules (that can ensure full coverage of given roadways) in polynomial time. We also addressed individual energy consumption and fairness by presenting an algorithm to find fair energy-efficient sensing schedules which balancing energy consumption among participating mobile users. It has been shown by simulation results based on real energy consumption and location data that compared to traditional sensing without collaborations, collaborative sensing achieves over 80% power savings. This finding well justifies the effectiveness of collaborative sensing in mobile phone sensing applications. Even though these algorithms can produce optimal solutions, they cannot be applied in the reality due to the difficulty for predicting mobile users' moving trajectories. Practical algorithms need to be developed for these mobile phone scheduling problems without making such an assumption, which will be discussed in Section V.

IV. Incentive Mechanism Design

In this section, we discuss incentive mechanisms, which will be used to attract participation of mobile users.

Even though many researchers have developed different mobile phone sensing applications, there are few studies on the incentive mechanism design. In [36], Reddy *et al.* developed recruitment frameworks to enable the system to identify well-suited participants for sensing services based on geographic and temporal availability as well as participation habits. However, they focused only on user selection, not incentive mechanism design. In [7], Danezis *et al.* developed a sealed-bid second-price auction to motivate user to allow precise information about their locations to be collected. However, the utility of the platform was neglected in the design of auction. In [19], Lee and Hoh studied the economic models of user participation incentive in participatory sensing applications. They designed and evaluated a reverse auction based dynamic price incentive mechanism, where users can sell their sensed data to a service provider with users' claimed bid prices. However the proposed incentive mechanism is focused only on minimizing and stabilizing incentive cost while maintaining adequate number of participants by preventing users from dropping out of participatory sensing applications, without considering the truthfulness, which is an important property. In [11], Duan *et al.* analyzed and compared different incentive mechanisms for motivating the collaboration of smartphone users on both data acquisition and distributed computing applications. They designed a reward-based collaboration mechanism, where the client announces a total reward to be shared among collaborators, and the collaboration is successful if there are enough users who are willing to collaborate. However, they formulated their problems based on a simple model without carefully addressing users' contributions.

The void has been filled by a recent publication [43], in which Yang *et al.* designed several incentive mechanisms to motivate users to participate in mobile sensing applications. Two types of incentive mechanisms are proposed: *platform-centric incentive mechanisms* and *user-centric incentive mechanisms*. In a platform-centric incentive mechanism, the platform has the absolute control over the total payment made to users, and users can only tailor their actions to cater for the platform. Whereas in a user-centric incentive mechanism, the roles of the platform and users are reversed. To assure himself of the bottom-line benefit, each user announces a reserve price, the smallest price at which he/she is willing to sell a service. The platform then selects a subset of users and pay each of them an amount that is no smaller than the user's reserve price. In the platform-centric model, a user participating in mobile phone sensing will earn a payment that is no lower than its cost. However, it needs to compete with other users for a fixed total payment. In the user-centric model, each user asks for a price for its service. If selected, the user will receive a payment that is no lower than its asked price. Unlike the platform-centric model, the total payment is not fixed for the user-centric model. Hence, the users have more control over the payment.

The platform-centric incentive mechanism is modeled as a Stackelberg game. There are two stages in such a mechanism: In the first stage, the platform announces its reward; in the second stage, each user strategizes its sensing time to maximize its own utility. Therefore, the platform is the leader and the users are the followers in this Stackelberg game. Meanwhile, both the platform and the users are players. The strategy of the platform is its reward. The strategy of a user is its working time. Based on some utility functions, the authors of [43] presented a Stackelberg Game based approach for the platform-centric model to compute a Stackelberg equilibrium, which maximizes the utility of the platform, and ensures that no user has the incentive to change his/her strategy.

In addition, the auction theory [16] can be leveraged to design incentive mechanisms for the user-centric model. A reverse auction based incentive mechanism was presented for the user-centric model in [43]. An auction takes as input the bids submitted by the users, selects a subset of users as winners, and determines the payment to each winning user. The proposed incentive mechanism has been formally proved to be computationally efficient, individually-rational, profitable and truthful.

V. Future Research Directions

In this section, we point out future research directions:

GPS-less Mobile Phone Scheduling: First of all, practical algorithms need to be designed to schedule sensing activities (i.e., determine when/where to sense) of mobile phones without full knowledge of mobile users' moving trajectories. In addition, GPS is energy-hungry and keeping GPS on during the whole sensing procedure is not feasible since it may drain the battery very quickly. Other approaches, such as WiFi or cellular signals, can also be used to obtain location information, which consume much less energy but provide less accuracy. Hence, GPS-less algorithms are needed for sensing scheduling. The scheduling problem becomes very challenging without accurate location information, which is illustrated in Fig. 2, where disks are used to show the sensing coverage region. First, a probabilistic coverage model needs to be developed to calculate the probability that a target point (or area) is covered if a mobile phone is scheduled to sense at a location which it believes to be (x, y) (The actual location may not be (x, y)), and the coverage probability given by a sensing schedule. Second, a simple and practical method is needed to predict the mobility of mobile users based on historical data in the near future (next few seconds). Moreover, efficient algorithms (based on the coverage model and the mobility prediction algorithm) need to be designed to solve the scheduling problem. Again, mobile phone scheduling discussed here is only related to opportunistic sensing.

Sensing Task Scheduling on a Mobile Phone: A sensing task will be assigned to multiple mobile phones. Correspondingly, a mobile phone may be used to process multiple sensing

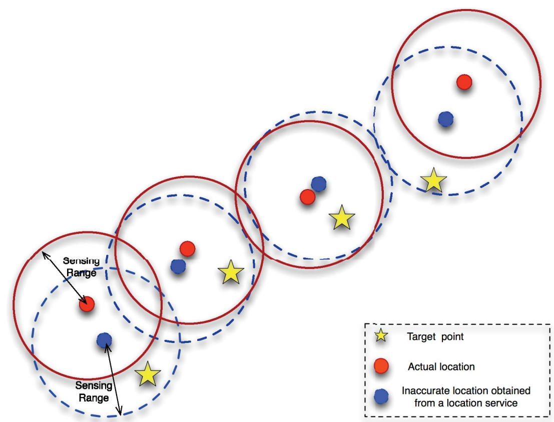


Fig. 2. Sensing without accurate location infromation.

tasks. Hence, sensing task scheduling algorithms are also needed to schedule multiple sensing tasks on a mobile phone. The following optimization problem needs to be addressed: given a set of sensing tasks (on a mobile phone), each with certain temporal requirement (i.e., must be completed at a particular time or during a certain period), spatial requirement (i.e., must be performed at a particular location or in a certain area), or both, find a schedule with minimum energy consumption for performing these tasks such that the given requirements are met. To the best of our knowledge, this problem has not been studied yet. One trivial solution is to treat each sensing task as an independent task and handle them one by one. However, this may not be energy-efficient because multiple sensing tasks may share one or multiple sensing actions (e.g., request location information from GPS). The best way may be to group multiple correlated tasks together by exploiting the temporal-spatial correlations between them, schedule sensing actions associated with them and determine when to conduct common sensing actions based on user mobility status with the objective of minimizing energy consumption and satisfying the temporal and spatial requirements.

Privacy-preserving Incentive Mechanisms: Incentive and privacy have been addressed separately in the context of mobile sensing but has not been considered simultaneously. For example, in the incentive mechanisms designed in [43], privacy issues were not considered at all. It is important to design incentive mechanisms that can enhance user privacy. Having privacy protection will encourage more mobile users to participate in sensing activities. Initial work was done in a very recent paper [21]. In this paper, the authors proposed two privacy-aware incentive schemes, which allow each mobile user to earn credits by contributing data without leaking which data it has contributed, and in the meanwhile ensure that dishonest users cannot abuse the system to earn unlimited amount of credits. The first scheme considers scenarios where a Trusted Third Party (TTP) is available. It relies on the TTP to protect user privacy, and thus has very low computation and storage cost at each mobile user. The second scheme removes the assumption of TTP and applies blind signature and commitment techniques to protect user privacy. We believe

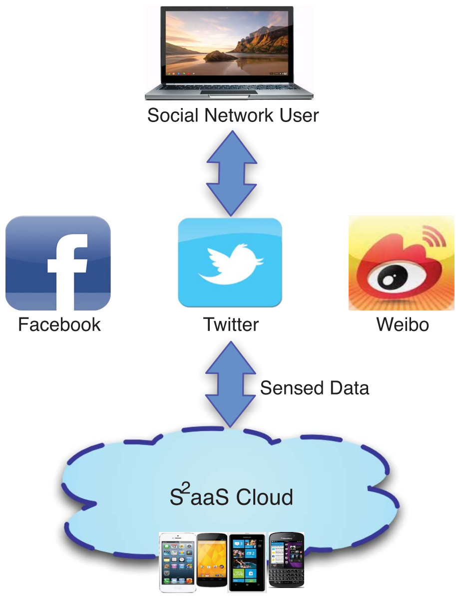


Fig. 3. Mobile phone sensing based social networking.

this line of research can make a significant impact on mobile phone sensing.

A Reputation System of Mobile Users: It is very important to establish a common reputation system of mobile users for various mobile phone sensing applications. In the current incentive mechanisms [43], mobile users are selected purely based on their bids. It would be interesting to study user selection schemes based on the reputation of individual users. Moreover, users' reputations can be used to enhance the reliability of sensed data provided by mobile users. For example, mobile users' reputations can be used as weights for generating the final sensing results. A mobile user in an S^{2}aaS cloud may be involved in various applications. A unified and fair approach needs to be developed to adjust the reputation score(s) of a mobile user based on his/her performance in different applications in terms of various metrics (efficiency, quality of sensed data, etc), which is very challenging but has not yet been studied.

Mobile Phone Sensing based Social Networking: Social networks have been making a significant impact on people's life. Marrying mobile phone sensing with social networking can benefit both systems. As illustrated in Fig. 3, on one hand, a popular social networking system, such as Facebook, Twitter and Weibo, serves as a perfect platform for sharing data collected via mobile phone sensing; on the other hand, mobile phone sensing can substantially enrich social networking activities by providing various context information of mobile users, such as location, moving states, etc. The CenceMe [27] represents the first system that combines the inference of the presence of individuals (e.g., walking, in conversation, at the gym) via mobile phone sensing with sharing of this information through social networking systems such as Facebook and MySpace. In [10], the authors designed and implemented a crowd-sourced sensing and collaboration system over Twitter, and demonstrated their system using two applications: a crowd-sourced weather radar, and a participatory noise-mapping application. In [34], a mobile phone sensing based platform, SociableSense, was developed to capture user behavior in office environments, while providing users with a quantitative measure of their sociability and that of colleagues. It will be very interesting to develop new social networking applications based on mobile phone sensing.

VI. Conclusion

In this paper, we introduced a new concept, Sensing as a Service (S^{2}aaS), and identified unique challenges of developing an S^{2}aaS cloud, which include: 1) support for various sensing applications; 2) energy-efficiency; 3) incentive mechanism design. We then reviewed existing systems and methods, presented viable solutions, and pointed out future research directions. Specifically, we described the basic functionalities that an S^{2}aaS cloud needs to have and proposed to use scripts to describe various sensing tasks and enable secure and flexible loading of them to different smartphone platforms. Moreover, we introduced energy-efficient sensing scheduling problems and pointed out the right directions for developing effective scheduling algorithms. In addition, we discussed two models for incentive mechanism design, platform-centric model and user-centric model, and described the desirable properties for incentive mechanisms under both models as well as the future research directions.

References

[1] (2012). *Apache HTTP Server* [Online]. Available: http://httpd.apache.org/

[2] (2006). *BigTable* [Online]. Available: http://en.wikipedia.org/wiki/BigTable

[3] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Comput. Surv.*, vol. 41, no. 3, pp. 1–72. Jul. 2009.

[4] T. L. Chin, P. Ramanathan, K. K. Saluja, and K. C. Wang, "Exposure for collaborative detection using mobile sensor networks," in *Proc. IEEE Int. Conf. Mobile Adhoc Sensor Syst. Conf.*, Nov. 2005, pp. 750–758.

[5] C. Cornelius, "Anonysense: Privacy-aware people-centric sensing," in *Proc. 6th Int. Conf. Mobile Syst., Appl. Services*, Jun. 2008, pp. 211–224.

[6] S. Consolvo, "Activity sensing in the wild: A field trial of ubifit garden," in *Proc. 6th Int. Conf. Mobile Syst., Appl. Services*, Apr. 2008, pp. 1797–1806.

[7] G. Danezis, S. Lewis, and R. Anderson, "How much is location privacy worth," in *Proc. 4th Workshop Econ. Inf. Security*, May 2005, pp. 1–13.

[8] T. Das, "PRISM: Platform for remote sensing using smartphones," in *Proc. 8th Int. Conf. Mobile Syst., Appl., Services*, Jun. 2010, pp. 63–76.

[9] J. Dean and S. Ghemawat, "MapReduce: Simplified data processing on large clusters," *ACM Commun.*, vol. 51, no. 1, pp. 107–113, Jan. 2008.

[10] M. Demirbas, "Crowd-sourced sensing and collaboration using Twitter," in *Proc. IEEE Int. Symp. World Wireless Mobile Multimedia Netw.*, Jun. 2010, pp. 1–9.

[11] L. Duan, "Incentive mechanisms for smartphone collaboration in data acquisition and distributed computing," in *Proc. IEEE Inf. Conf.*, Jan. 2012, pp. 1701–1709.

[12] Q. Feng, L. Liu, and Y. Dai, "Vulnerabilities and countermeasures in context-aware social rating services," *ACM Tran. Internet Technol.*, vol. 11, no. 3, pp. 1–11, Jan. 2012.

[13] S. Gaonkar, "Micro-blog: Sharing and querying content through mobile phones and social participation," in *Proc. 6th Int. Conf. Mobile Syst., Appl., Services*, Jun. 2008, pp. 174–186.

[14] E. Koukoumidis, L. S. Peh, and M. R. Martonosi, "SignalGuru: Leveraging mobile phones for collaborative traffic signal schedule advisory," in *Proc. 9th Int. Conf. Mobile Syst., Appl., Services*, Jul. 2011, pp. 127–140.

[15] R. J. Kauffman, H. Lai, and C. T. Ho, "Incentive mechanisms, fairness and participation in online group-buying auctions," *Electron. Commerce Res. Appl.*, vol. 9, no. 3, pp. 249–262, May–Jun. 2010.

[16] V. Krishna, *Auction Theory*. New York, NY, USA: Academic, 2009.

[17] N. D. Lane, "A survey of mobile phone sensing," *IEEE Commun. Mag.*, vol. 48, no. 9, pp. 140–150, Sep. 2010.

[18] H. J. Lee, "Ubiquitous healthcare service using Zigbee and mobile phone for elderly patients," *Int. J. Med. Informat.*, vol. 78, no. 3, pp. 193–198, Mar. 2009.

[19] J. Lee and B. Hoh, "Sell your experiences: A market mechanism based incentive for participatory sensing," in *Proc. IEEE Int. Conf. Pervas. Comput. Commun.*, Apr. 2010, pp. 60–68.

[20] P. Leijdekkers and V. Gay, "A self-test to detect a heart attack using a mobile phone and wearable sensors," in *Proc. 21st IEEE Int. Symp. Comput.-Based Med. Syst.*, Jun. 2008, pp. 93–98.

[21] Q. Li and G. Cao, "Providing privacy-aware incentives for mobile sensing," in *Proc. IEEE Int. Conf. Pervas. Comput.*, Mar. 2013, pp. 1–9.

[22] H. Lu, "SoundSense: Scalable sound sensing for people-centric sensing applications on mobile phones," in *Proc. 7th Int. Conf. Mobile Syst., Appl., Services*, Jun. 2009, pp. 165–178.

[23] H. Lu, "Bubble-sensing: Binding sensing tasks to the physical world," *J. Pervas. Mobile Comput.*, vol. 6, no. 1, pp. 58–71, Feb. 2010.

[24] H. Lu, "The Jigsaw continuous sensing engine for mobile phone applications," in *Proc. 8th ACM Conf. Embedded Netw. Sensor Syst.*, 2010, pp. 71–84.

[25] S. M. Lui, "Participation incentive mechanisms in peer-to-peer subscription systems," in *Proc. 35th Ann. Hawaii Int. Conf. Syst. Sci.*, Jan. 2002, pp. 3925–3931.

[26] S. Madhani, M. Tauil, and T. Zhang, "Collaborative sensing using uncontrolled mobile devices," in *Proc. Int. Conf. Collaborative Comput., Netw., Appl. Worksharing*, 2005, pp. 1–8.

[27] E. Miluzzo, "Sensing meets mobile social networks: The design, implementation and evaluation of the CenceMe application," in *Proc. 6th ACM Conf. Embedded Netw. Sensor Syst.*, 2008, pp. 337–350.

[28] E. Miluzzo, "Darwin phones: The evolution of sensing and inference on mobile phones," in *Proc. 8th Int. Conf. Mobile Syst., Appl., Services*, 2010, pp. 5–20.

[29] P. Mohan, V. N. Padmanabhan, and R. Ramjee, "Nericell: Rich monitoring of road and traffic conditions using mobile smartphones, in *Proc. 6th ACM Conf. Embedded Netw. Sensor Syst.*, 2008, pp. 323–336.

[30] M. Mun, "PEIR, the personal environmental impact report, as a platform for participatory sensing systems research," in *Proc. 7th Int. Conf. Mobile Syst., Appl., Services*, Jun. 2009, pp. 55–68.

[31] M. Musolesi, "Supporting energy-efficient uploading strategies for continuous sensing applications on mobile phones," *Pervas. Comput., Lect. Notes Comput. Sci.*, vol. 6030, pp. 355–372, May 2010.

[32] E. Ngai and J. Xiong, "Adaptive collaborative sensing using mobile phones and stationary sensors," in *Proc. IEEE/IFIP 41st Int. Conf. Dependable Syst. Netw. Workshops*, Jun. 2011, pp. 280–285.

[33] J. Paek, J. Kim, and R. Govindan, "Energy-efficient rate-adaptive GPS-based positioning for smartphones," in *Proc. 8th Int. Conf. Mobile Syst., Appl., Services*, 2010, pp. 299–314.

[34] K. K. Rachuri, C. Mascolo, M. Musolesi, and P. J. Rentfrow, "SociableSense: Exploring the trade-offs of adaptive sampling and computation offloading for social sensing," in *Proc. 17th Ann. Int. Conf. Mobile Comput. Netw.*, Sep. 2011, pp. 73–84.

[35] R. K. Rana, "Ear-phone: An end-to-end participatory urban noise mapping system," in *Proc. 9th ACM/IEEE Int. Conf. Inf. Process. Sensor Netw.*, Aug. 2010, pp. 105–116.

[36] S. Reddy, D. Estrin, and M. Srivastava, "Recruitment framework for participatory sensing data collections," in *Proc. 8th Int. Conf., Pervas.*, May 2010, pp. 138–155.

[37] A. Saipulla, "Barrier coverage with sensors of limited mobility," in *Proc. 11th ACM Int. Symp. Mobile Hoc Netw. Comput.*, Sep. 2010, pp. 201–210.

[38] X. Sheng, J. Tang, and W. Zhang, "Energy-efficient collaborative sensing with mobile phones," in *Proc. IEEE INFOCOM*, Mar. 2012, pp. 1916–1924.

[39] R. Tan, G. Xing, J. Wang, and H. C. So, "Exploiting reactive mobility for collaborative target detection in wireless sensor networks," *IEEE Trans. Mobile Comput.*, vol. 9, no. 3, pp. 317–332, Mar. 2010.

[40] N. Thepvilojanapong, S. I. Konomi, Y. Tobe, Y. Ohta, M. Iwai, and K. Sezaki, "Opportunistic collaboration in participatory sensing environments," in *Proc. 5th ACM Int. Workshop Mobility Evolving Internet Archit.*, Sep. 2010, pp. 39–44.

[41] A. Thiagarajan, "VTrack: Accurate, energy-aware road traffic delay estimation using mobile phones," in *Proc. 7th ACM Conf. Embedded Netw. Sensor Syst.*, May 2009, pp. 85–98.

[42] H. Weinschrott, F. Durr, and K. Rothermel, "StreamShaper: Coordination algorithms for participatory mobile urban sensing," in *Proc. IEEE 7th Int. Conf. Mobile Adhoc Sensor Syst.*, Nov. 2010, pp. 195–204.

[43] D. Yang, G. Xue, X. Fang, and J. Tang, "Crowdsourcing to smartphones: Incentive mechanism design for mobile phone sensing," in *Proc. 18th Ann. Int. Conf. Mobile Comput. Netw.*, Dec. 2012, pp. 173–184.

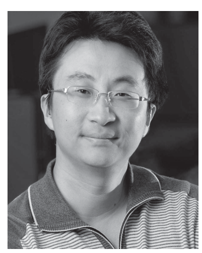
[44] C. W. Yu, "Coverage algorithms in GPS-less wireless mobile sensor networks," in *Proc. Int. Conf. Mobile Technol., Appl., Syst.*, Apr. 2008, pp. 1–79.

[45] S. Zhou, M. Y. Wu, and W. Shu, "Finding optimal placements for mobile sensors: Wireless sensor network topology adjustment," in *Proc. IEEE Circuits Syst. Symp. Emerging Technol., Frontiers Mobile Wireless Commun.*, Jun. 2004, pp. 529–532.

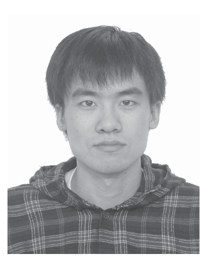
[46] Z. Zhuang, K.-H. Kim, and J. P. Singh, "Improving energy efficiency of location sensing on smartphones," in *Proc. 8th Int. Conf. Mobile Syst., Appl., Services*, Jun. 2010, pp. 315–330.

[47] Y. Zou and K. Chakrabarty, "Distributed mobility management for target tracking in mobile sensor networks," *IEEE Trans. Mobile Comput.*, vol. 6, no. 8, pp. 872–887, Aug. 2007.

Xiang Sheng (M'11) is currently pursuing the Ph.D. degree at the Department of Electrical Engineering and Computer Science, Syracuse University, Syracuse, NY, USA. He received the B.E. degree in electrical engineering from the Nanjing University of Science and Technology, Nanjing, China, in 2008, and the M.S. degree in electrical engineering from Syracuse University in 2010. His current research interests include wireless networking, mobile computing, and green communications.



Jian Tang (M'13) is an Assistant Professor with the Department of Electrical Engineering and Computer Science, Syracuse University, Syracuse, NY, USA. He received the Ph.D. degree in computer science from Arizona State University, Tempe, AZ, USA, in 2006. His current research interests include wireless networking, cloud computing, and green networking. He received the NSF CAREER Award in 2009. He served as a symposium Co-Chair for the IEEE Globecom 2011 Wireless Networking Symposium, the ICNC 2012–2013 Wireless Networks Symposium, the WOCC 2013 Wireless Symposium and IWCMC 2013 General Symposium. He has served on the technical program committees of many international conferences, such as the IEEE Infocom, ICC, and Globecom.



Xuejie Xiao is currently pursuing the Ph.D. degree at the Department of Electrical Engineering and Computer Science, Syracuse University, Syracuse, NY, USA. He received the B.S. degree in software engineering from Tongji University, Shanghai, China, in 2011. He was a Software Engineer Intern with Google, Shanghai, from 2010 to 2011. His current research interests include cloud computing, high performance computing systems, large-scale distributed systems, and big data analysis.

Guoliang Xue (F'11) is a Professor of computer science and engineering with Arizona State University, Tempe, AZ, USA. He received the Ph.D. degree in computer science from the University of Minnesota, Minneapolis, MN, USA, in 1991. His current research interests include survivability and security issues in networking, quality of service provisioning, and human centric computing with collective intelligence. He has published over 200 refereed technical papers, many of which appeared in top journals, such as the IEEE/ACM TRANSACTIONS ON NETWORKING, the *IEEE Journal on Selected Areas in Communications*, the IEEE TRANSACTIONS ON MOBILE COMPUTING, and top conferences, such as ICNP, INFOCOM, MOBICOM, MOBIHOC, and NDSS. He received the Best Paper Awards at ICC'2012, GLOBECOM'2011, ICC'2011, and MASS'2011, as well as a Best Paper Runner-up Award at ICNP'2010. He was a Keynote Speaker at the LCN'2011-IEEE Conference on Local Computer Networks. He serves on the editorial boards of IEEE/ACM TRANSACTIONS ON NETWORKING and the *IEEE Network Magazine*. He served as an Associate Editor and Advisory Board Member of the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS. He was a TPC Co-Chair of the IEEE INFOCOM'2010 and is a member of the INFOCOM Standing Committee. He was the IEEE Communications Society Distinguished Lecturer from January 2010 to December 2011.