Towards Participatory Sensing of Regions of Interest with Adaptive Sampling Rate

Carlos Henrique de O. M. André[a], Dianne S. V. Medeiros[b,*], Miguel Elias M. Campista[a]

[a] *Universidade Federal do Rio de Janeiro - PEE-COPPE/Del-Poli/GTA*
[b] *Universidade Federal Fluminense - PPGEET/MídiaCom*

Abstract

Participatory Sensing (PS) is a known paradigm of collaborative networks which provides incentives for users to participate in sensing tasks of Regions of Interest (RoIs). A challenge in wireless networking, however, is to balance the amount of data collected by users without imposing excessive load to the network. In this direction, this paper proposes a centralized system to adapt the sampling rate assigned to each crowdsourcing participant sensor. The sampling rate is computed based on the standard deviation of samples collected from a given RoI. The results obtained via simulations show a tradeoff between the sampling rate and the number of crowdsourcing participants. The more crowdsourcing participants, the lower must be the individual sampling rate and the amount of data transferred. This strategy can increase the data delivery rate taking into account the available short contact times, even though it requires a larger number of sensors.

Keywords: Participatory sensing, VANET, Adaptive sampling

1. Introduction

Participatory Sensing (PS) [1] introduces large-scale distributed sensing systems capable of collecting huge amounts of data from Regions of Interest (RoI).

*Corresponding author
Email addresses: choma@gta.ufrj.br (Carlos Henrique de O. M. André), diannescherly@id.uff.br (Dianne S. V. Medeiros), miguel@gta.ufrj.br (Miguel Elias M. Campista)

These systems count on users' resources, such as smartphones, to provide low-cost data sensing at high sampling rates. Users' resources, however, are not free to use. In this case, even though applications can improve performance using data from multiple sources, participatory sensing systems must deal with user incentives to become feasible [2]. Tasks and incentive mechanisms can be combined in a central entity that identifies RoIs and users eligible to handle the required sensing tasks. For instance, Waze offers information about traffic conditions using the data provided by users, who receive points to rank up in the application [3].

A large number of data sources and their dynamics can lead to networking issues concerning bandwidth and connectivity. Such issues are, respectively, a consequence of the significant amount of generated data and the intermittent Internet connectivity of sensing devices. These challenges are similar to those from Vehicular Ad Hoc Networks (VANETs) [4]. In the era of the Internet of Things (IoT), the VANET has evolved to a new paradigm, known as the Internet of Vehicles (IoV) [5]. In IoV, besides users, smart sensors and actuators can be installed on vehicles and vehicular infrastructure [6]. Consequently, the service provided depends on the ability of geo-distributed nodes, mobile or not, to receive data from or to send data to a central entity on the Internet. The volume of data generated by the devices, the constraints of the devices, and the vehicular mobility, make PS even more challenging in IoV scenarios. Therefore, we can reduce the number of samples collected by each participant in a given RoI, adjusting individual sampling rates according to the number of participants. The more participants, the lower the individual sampling rates without accuracy loss.

Several PS systems have been proposed in the literature for IoV scenarios. SensingBus, for instance, improves weather monitoring in a city using sensors embedded in urban buses [7]. The idea is to gratuitously complement the data collected by meteorological stations with finer-grained data from buses, as they move around the city. Similarly to SensingBus, Mohan et al. [8] propose Nericell, which uses sensors embedded in smartphones, such as accelerometer,

microphone, and GPS, to determine road and traffic conditions. Yet, Zhou et al. [9] propose a collaborative system to predict the position of buses along their trajectory. To this end, they use the signal power between cellular towers and mobile devices. Even though these works focus on collecting data using mobile crowdsourcing participants, none of them uses adaptive sampling rates. This means that, whenever participants perform a sensing task, they collect data at a fixed sampling rate. The As-Air application [10] and the work from Weinschrott et al. [11] are, however, two counterexamples. As-Air measures the air quality, which varies over time, using adaptive sampling rates. Depending on the air environment, As-Air adapts each node sampling rate using reinforcement learning to allow optimized air quality monitoring. Following a similar idea, Weinschrott et al. propose a system to discover mobile objects that can increase the sampling rate for the sake of robustness. Higher sampling rates are used to increase the probability of detecting an object of interest. Adaptive sampling is, indeed, a technique that has been used so far. As far as we know, however, there is still a gap concerning combining the adaptive sampling within IoV systems based on Participatory Sensing.

This paper proposes a participatory sensing system that differs from the aforementioned works [10, 11]. We investigate the feasibility of embedding in the participatory system a functionality to adapt the individual sampling rates of participants, according to both temporal and spatial aspects of IoV systems. Moreover, our proposed participatory system addresses both the control of inconsistent data and the sampling rate used to sense the RoI. To this end, we assume the existence of a centralizing node, in the cloud, with elastic capacity. This central node receives the data collected by all participants and sets the sampling rate of each one in each RoI accordingly. The key idea is to reduce the sampling rate of each participant by distributing the sensing task to all participants in the same RoI. This reduces the amount of energy spent per participant; at the same time, it maintains the measurement accuracy. In this paper, we assume that the system is used to evaluate the traffic conditions of a city and, thus, that crowdsourcing participants are vehicles providing samples

of instantaneous speed. If the collected data vary significantly on a road segment in the RoI, we consider that erroneous measurements are likely to occur and, therefore, the sampling rate per participant is increased. Similarly, if the variability on a road segment is small, the system considers the possibility of reducing the sampling rate. We use as sample values the registers from a one-day dataset of the city bus fleet of Seattle, USA [12], assuming that the sensed data is the bus speed. We show that higher data loads can be delivered only if a small number of nodes is competing to access the wireless medium. Besides, we conclude that it is possible to compensate for lower individual sampling rates with a higher number of nodes, depending on the sensed data variability on a given segment. This approach reduces the load on each crowdsourcing participant, saving energy, improving the delivery rate, and avoiding the transfer of large data volume. Moreover, results indicate that the adaptive sampling rate can reduce sampling errors without impacting the data load.

This paper is organized as follows. Section 2 introduces the notions of participatory sensing. Section 3 describes the network architecture considered and proposes the sensing system with an adaptive sampling rate. In Section 4 we characterize the datasets used in this paper, whereas in Section 5, we analyze our participant system proposal. Finally, Section 6 concludes this paper and discusses future research directions.

2. Participatory Sensing Overview

Currently, the sensing power embedded in vehicles or even in personal mobile devices grows at a fast pace. This is a consequence of the evolution of mobile technology and the miniaturization of electronic components. If these devices are used together for sensing tasks, it is possible to build more robust, scalable, and low-cost sensing systems. Consequently, more sophisticated applications can be developed, such as the ones for Smart Cities [13]. Combining the partial data collected by each user, we compose a complete view of the entire RoI. Hence, relying on the participation of users to sense the environment, using their

personal devices, is the focus of the research paradigm known as Participatory Sensing (PS). Several applications have been developed for PS, such as natural resources management [14], urban planning and monitoring [15, 8], and public healthcare [16].

In a PS system, tasks are assigned to users' devices to allow data collection in the RoI [17], analytics, and sharing. PS systems enable to collect data in more extensive areas due to the number of crowdsourcing participants and their mobility, thus increasing the knowledge about the RoI. The data collected by all participants needs processing, which can be performed on an external central entity. Hence, a resource-rich server in the cloud can play the role of processing entity of a PS system, also providing storage capacity. The use of cloud resources has been often considered to support PS systems [18].

User participation in data collection and transfer is stimulated through incentive mechanisms. In PS systems, depending on the reward, a user can be motivated to collect a large volume of data during a short interval. For instance, users can inform their speed to the system in exchange for discounts on services or targeted advertising, according to their profiles. The system can also yield, as reward, useful information obtained from the processed data collected by the crowdsourcing participants, such as traffic information, as in Google Maps or Waze. Exhaustion of users' resources can lead to the collection of faulty data. Hence, there must exist a tradeoff between the quantity and the quality of the collected data.

PS can be applied in a multitude of scenarios. In this paper, we focus on Vehicular Ad Hoc Networks (VANETs), exploring the areas of urban planning and monitoring. In this scenario, PS can be used to plan a new traffic detour, to create a new bus stop, or to design the expansion of roads, to cite a few examples. Data is inserted into the system by crowdsourcing participants, whose participation is fundamental. Nevertheless, this participation can incur on networking issues. If a large volume of data is collected within the RoI, the transfer between participants' devices (On-Board Units - OBUs) and Road Side Units (RSUs) can be an issue due to short contacts. If external servers

do not entirely receive the data, the system performance may become at least temporarily impaired. In this paper, we use a dataset from the city of Seattle (Section 4) as the basis of our PS system scenario. In this dataset, we assume that the city buses are the crowdsourcing participants and each bus contributes to the system with information about its own speed. It is worth mentioning that other datasets could be used without loss of generality.

3. Adaptive Sampling Approach for Participatory Sensing Systems

One of the main challenges for PS systems is to efficiently cover an RoI without overloading both participants and the system itself. A possible solution could be to increase the bandwidth available for collected data transfers or to increase the number of available RSUs. As a consequence, participants would need to upload fewer data at each RSU, if we assume that the number of contacts increases proportionally. The shortcoming of these solutions, however, is an increased cost of implementation and maintenance. Hence, instead of increasing the number of RSUs, or the available bandwidth to transmit the collected data, this paper investigates the feasibility of a participatory sensing system based on adaptive sampling rate, which varies according to each segment condition in an RoI. The sampling rate depends on the sensed data standard deviation on a given road segment within an RoI. The idea is to increase the sampling rate overtime on each segment if the standard deviation is high or to reduce it otherwise. Such adaptability is vital to minimize the overall sampling error, considering the entire RoI, which is obtained from the consolidated data. The sampling rate dynamics must be taken into account so that: (i) the RoI is entirely sensed providing accurate measurements compared with real values and (ii) neither the participating users nor the wireless or the access network can become overloaded. This section presents the network scenario considered in this paper and describes the PS system proposed with an adaptive sampling rate.

3.1. Vehicular network scenario

We consider a vehicular network composed by a central server, access points (Roadside Units - RSUs), and participants equipped with sensing-enabled devices (On-board Units - OBUs), as shown in Figure 1. The central server is responsible for processing the data collected by the crowdsourcing participants. Based on the information obtained from the processed data, the central server offers services to external users, such as information about traffic conditions. This server must have elastic capacity to support the system and, thus, it is located in the cloud, so as not to act as a bottleneck. The RSUs are connected to the Internet through an access network, which is typically wired. Finally, the OBUs cannot communicate using cellular infrastructure and, thus, they can reach the central server only through the RSUs. This communication uses the IEEE 802.11p standard.

Authentication and security issues related to the proposed system are out of the scope of this paper in its current version. Therefore, we assume for the sake of simplicity that participants do not collude. Hence, the central server is only capable of validating received samples taking into account that most

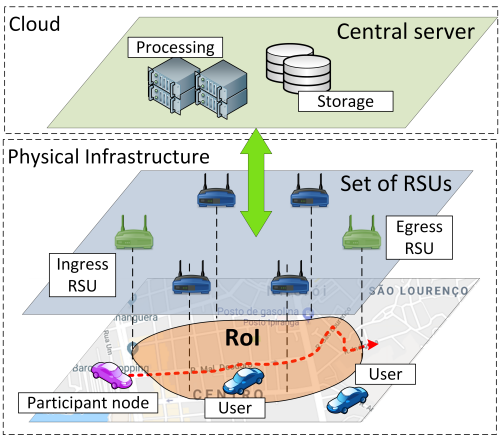


Figure 1: The vehicular network scenario considered in this paper. Crowdsourcing participants start collecting data as they receive a sensing task at the RoI entrance from an ingress RSU. Afterward, the participants deliver the collected data to an egress RSU when leaving the RoI.

participants are legitimate. To avoid collusion attacks, we could envision extensions to our system with current directions found in the literature. In general, to avoid the participation of malicious users, the participatory sensing system must use authentication, authorization, and access control mechanisms. The central server is a trusted authority, and it communicates directly to RSUs. These RSUs could be used to authenticate participants, based on efficient VANET authentication protocols, such as RAISE [19]. This protocol can be adapted to our system to verify message authenticity. Whenever a participant enters an area covered by an RSU, the RSU assigns a unique shared symmetric secret key and a pseudo ID with her. The symmetric key is used by the participant to generate a symmetric message authentication code, which will be further attached to all its messages. As all the RSUs know the key for generating this code, any RSU could be used to verify the message. Upon authenticating participants, the system can avoid inside outbreaks by tracking the actions of each participant by using a reputation scheme. The system must maintain a reputation score associated with each participant, based on the participant performance in sensing tasks [20]. The trustworthiness of each participant is then evaluated, and the system can consider only the contribution from trusted participants. It is possible to combine our proposal with the reputation system proposed by Huang et al. [21]. The central server in our scenario operates as the application server in the proposal of Huang et al., which associates a reputation score to each participant. This score reflects the level of trust perceived by the central application server about the data uploaded by the participant over time. Trust can also be provided in a hardware-like solution. For instance, Dua et al. [22] propose to use a trusted sensing peripheral to attest the sensor data integrity at the source, making the system resilient to collusion, as local integrity check is inherently resilient to it. It is worth mentioning that detecting collusion attacks is a complex research area in wireless networking as a whole, given the network dynamics [23].

Figure 1 shows a participant node that, upon entering the Region of Interest (RoI), has the opportunity to receive requests to contribute with the participa-

tory sensing system. The central server sends this request, and it is delivered to the user through an RSU at the RoI entrance (ingress RSU). To simplify, an RoI is composed of road segments delimited by RSUs. In our scenario, there are no RSUs within the RoI and, because OBUs do not communicate with the cellular network, the update of the sampling rate does not happen continuously. This is, in fact, an assumption we make for the sake of offloading data traffic from mobile networks [24, 25, 26]. Nevertheless, this is not a requirement as the server may contact the participant node at the RoI entrance and collect the data at the RoI exit, assuming it tracks the participant with data from a GPS embedded in the vehicle, for instance.

In the RoI, a participant periodically collects sample measurements, respecting the sampling rate defined by the central server for a given road segment. When the crowdsourcing participant leaves the RoI, she delivers the collected data to the central server through an RSU at the exit (egress RSU). The RSU sends to the central server the collected data from different participants in the same RoI. The central server, in turn, consolidates the received data and possibly offers services to external users. In this paper, incentive mechanisms are not addressed, even though we assume that users are rewarded somehow if they participate in the sensing system.

3.2. Operation of the adaptive PS system

In this paper, the crowdsourcing participants know the adapted rate at which they collect samples on each segment in their trajectory. The trajectory considered is the one falling within the Region of Interest (RoI), as observed in Figure 1. The rate adaption depends on the sampling error of each segment, which is periodically measured by a central server. According to this error, the central server calculates the value of the sampling rate on each segment that will reduce the overall error in the RoI. Then, the central server sends requests to the RSUs in the system, containing the recommended sampling rate for each segment. When a participant enters the RoI, the RSU at the entry point forwards a request to the participant, so that she can adapt her sampling rate according

to the needs of each segment within the RoI. We represent a RoI, $\mathcal{R}$, by a set of segments ς_j, i.e., $\mathcal{R} = \{\varsigma_1, \ldots, \varsigma_m\}$, where m is the number of segments within $\mathcal{R}$. The set of crowdsourcing participants p_i is denoted as $\mathcal{P} = \{p_1, \ldots, p_n\}$, where n is the total number of users in the system. Each user $p_i \in \mathcal{P}$ collects samples that contain the position, the timestamp, and the value of the sampled measurement. Then, each user p_i registers the sample tuple in a data trace T_i, which is further transferred to the egress RSU. In our scenario, each sample stores the value of the vehicle speed. We highlight that the proposed sensing system does not lose generality if other sensors or other scenarios are used.

When entering the RoI at the ingress RSU, a participant p_i has the opportunity to receive a vector $\boldsymbol{\pi}^{(\boldsymbol{\Delta t})} = \langle \pi_1^{(\Delta t)}, \ldots, \pi_m^{(\Delta t)} \rangle$ containing the description of the sensing task to be performed in the RoI during the interval Δt, as shown in Figure 1. Figure 2 illustrates the vector $\pi^{(\Delta t)}$ and provides further details about the content of its corresponding elements.

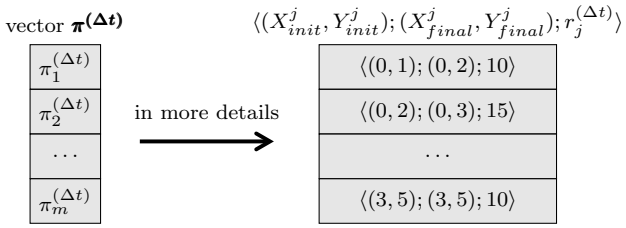
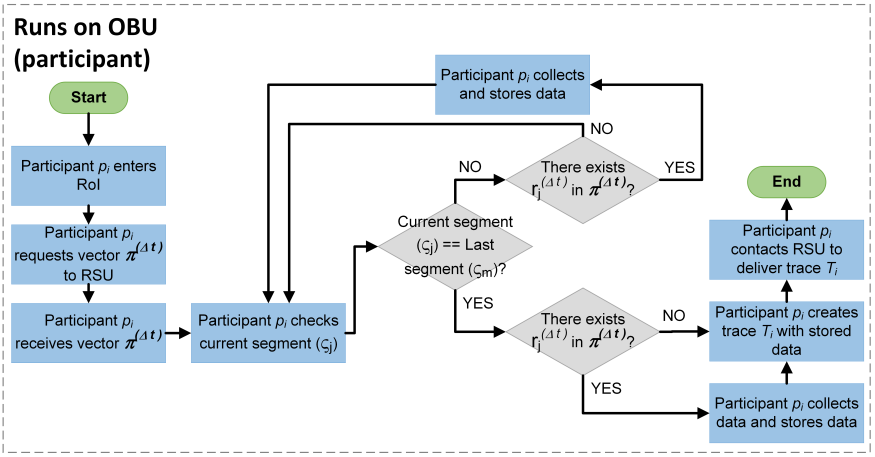
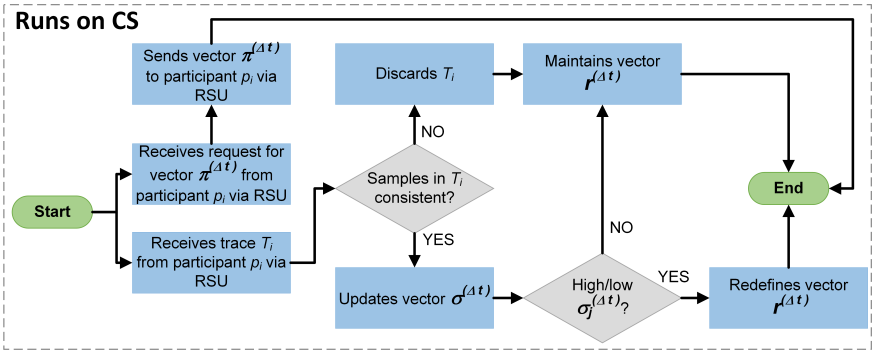


Figure 2: The vector $\boldsymbol{\pi}^{(\boldsymbol{\Delta t})} = \langle \pi_1^{(\Delta t)}, \ldots, \pi_m^{(\Delta t)} \rangle$ and, in more details, the content of each element in $\boldsymbol{\pi}^{(\boldsymbol{\Delta t})}$, composed of the initial and the final coordinates of a segment ς_j in the RoI and its corresponding recommended sampling rate ($r_j^{(\Delta t)}$). The values for the coordinates as well as for the recommended sampling rates are integers for the sake of simplicity.

Figure 3(a) shows the interaction between the participant and the RSU. The participant p_i requests vector $\boldsymbol{\pi}^{(\boldsymbol{\Delta t})}$ to the RSU upon entering the RoI. Participant p_i receives vector $\boldsymbol{\pi}^{(\boldsymbol{\Delta t})}$ from the RSU as a data structure, composed by a list of tuples that describes the segments of the RoI and their respective recommended sampling rates. Each segment ς_j in $\boldsymbol{\pi}^{(\boldsymbol{\Delta t})}$ is represented by its initial and final coordinates, denoted by (X^j_{init}, Y^j_{init}) and $(X^j_{final}, Y^j_{final})$, respectively. The participant knows her own trajectory and uses only the sampling rates that



(a) Process executed on the OBU of a participant.



(b) Process executed on the Central Server (CS).

Figure 3: Flowchart of the processes executed on the central server and on the participant's OBU. Participants request vector $\boldsymbol{\pi^{\Delta t}}$ to an RSU when entering an RoI and collects data according to the sample rates and segments information provided in $\boldsymbol{\pi^{\Delta t}}$. The central server communicates with the participant's OBU via the RSUs, sending the sensing tasks and receiving data. Upon receiving a new trace $\boldsymbol{T_i}$ from a crowdsourcing participant $\boldsymbol{p_i}$, the central server executes the actions shown in the flowchart.

correspond to the segments through which she travels. Hence, before each collection task, the participant continuously checks the current segment, ς_j, and verifies if it is the last segment, ς_m, received in vector $\boldsymbol{\pi^{(\Delta t)}}$. If it is not, p_i checks if vector $\boldsymbol{\pi^{(\Delta t)}}$ contains a sampling rate for the current segment, $r_j^{(\Delta t)}$, so that p_i can collect and store the sampling data. When the participant reaches

the last segment described by vector $\boldsymbol{\pi}^{(\boldsymbol{\Delta t})}$, *i.e.*, $\pi_m^{(\Delta t)}$, and there is a sampling rate for this segment within the vector, the participant collects the data and store it to further create trace T_i. If there is no sampling rate for the segment, the participant creates trace T_i with the previously stored data. Upon creation of trace T_i, the participant contacts the RSU to deliver the trace.

It is noteworthy that, if the system knows the route of each participant p_i, vector $\boldsymbol{\pi}^{(\boldsymbol{\Delta t})}$ can be summarized to contain only the segments through which p_i will travel. Even in the more general case, not necessarily all the segments of p_i's trajectory need to be described in vector $\boldsymbol{\pi}^{(\boldsymbol{\Delta t})}$. If the sampling rates of two consecutive segments are the same, and if they have the same direction and sense, the system can aggregate both segments in a single tuple. The collected data is offloaded through the egress RSU. It should be noted that, although vector $\boldsymbol{\pi}^{(\boldsymbol{\Delta t})}$ is individual for each vehicle, consecutive vehicles arriving with a $\Delta t + \varepsilon$ difference in time, where $\varepsilon > 0$, can receive vectors with the same description and recommended sampling rates if no change is observed in the segments of the RoI. Nevertheless, if a modification happens, consecutive vehicles receive different vectors.

The central server verifies whether the samples have inconsistencies, which can happen due to failures on the sensing system or malicious actions[1], upon receiving a trace T_i from participant p_i. After validating the received samples, i.e., checking if the values are feasible for the RoI, the central server computes the standard deviation of the samples received for each segment ς_i and updates the standard deviation vector considering the time interval Δt. The updated standard deviation vector is denoted as $\boldsymbol{\sigma}^{(\boldsymbol{\Delta t})} = \langle \sigma_1^{(\Delta t)}, \ldots, \sigma_m^{(\Delta t)} \rangle$. It is noteworthy that $\sigma_j^{(\Delta t)}$, corresponding to the standard deviation of segment ς_j, is calculated considering the samples received in the previous traces in the same interval Δt plus the samples received in the last trace. The standard deviation is then used to infer the measurement error for each segment, as this error cannot be obtained directly. Based on this error, if at least one segment with

[1]The problem of data consistency is addressed in our technical report available at [27]

high or low standard deviation exists, the sampling rate for this segment must be better adapted. Thus, we define the vector $\boldsymbol{r^{(\Delta t)}} = \langle r_1^{(\Delta t)}, \ldots, r_m^{(\Delta t)} \rangle$, which contains the recommended sampling rate for all segments in $\mathcal{R}$ during the time interval Δt. Note that a sampling rate $r_j^{(\Delta t)}$ is recommended for the segment ς_j. Figure 3(b) shows the process executed by the central server to update the standard deviation $\boldsymbol{\sigma^{(\Delta t)}}$ and the recommended sampling rate $\boldsymbol{r^{(\Delta t)}}$ vectors. To investigate the feasibility of using adaptive sampling rate in the participatory system, in our evaluations, we use at most three different sampling rates, i.e., $c_{min}R$, R, $c_{max}R$, where $c_{min} < 1$ and $c_{max} > 1$ are constants selected according to the measured standard deviation. The mechanism to dynamically choose the values of c_{min} and c_{max} is not in the scope of the current work.

Each application running on the system is responsible for setting the thresholds that define the lower and upper limits of the standard deviation. The thresholds are set according to the tolerated sampling error on each road segment, which depends on upstream application requirements. As discussed by Lee and Hoh [28], different applications may have different minimum acceptable accuracy levels. Hence, if the application requires higher accuracy, the central server must be able to adjust the sampling rate of each participant accordingly. Upon processing the collected data, the central server can infer if the sampling error leads to the accuracy required by the current application. Then it can adjust, again, the data collection with higher or lower sampling rates. This closes the data collection loop without the participation of external users, who play only the role of data consumers. In our paper, we consider that external users are only aware of the quality of the application provided by the central server and that the data collection process is totally transparent to them. In addition, we assume that the central server is able to maintain the required application accuracy by adjusting the sampling rate of each participant. If the maximum sampling rate achievable by all participants does not satisfy the application requirements, the system will need more participants. This problem, however, is not exclusive of our proposal as it can affect any PS system.

4. Dataset

We analyze the impact of the sampling rate adaption in a city according to requests from the central server. To this end, we use the *Ad Hoc City* [12] dataset, which contains the daily movement of the bus fleet of Seattle, in Washington, USA. The dataset was collected for one month, from October 31^{st}, 2001, to December 2^{nd}, 2001, totaling 125 MB of data. The dataset represents the real movement of city buses. The mobility events of each bus id_{bus} traveling on route id_{route} are registered as a 6-tuple as follows: $\langle d; t; id_{bus}; id_{route}; (x, y)\rangle$, where ($x$,$y$) are the Cartesian coordinates of the bus position on date d at time t. The bus route is composed of consecutive segments in the RoI.

In this paper, we select a typical case within those available on the trace to validate our proposal. The goal is to show that increasing, or even reducing, the number of samples collected in the RoI is feasible and can improve the PS system without overloading the participants. Hence, we analyze the trace of a typical Wednesday, October 31^st, 2001, which includes the mobility of n buses (which in our case represent the set $\mathcal{P}$ of participants) on 236 segments. The bus speed is not provided in the original dataset. We use the Euclidian distance to compute bus speeds as the segments in the analyzed scenario are mainly straight. Hence, we consider the bus position and the time interval between two consecutive tuples of the same bus. Then, if we have the tuples $\langle d; t_1; id_{bus}; id_{route}; (x_1, y_1)\rangle$ and $\langle d; t_2; id_{bus}; id_{route}; (x_2, y_2)\rangle$ from bus id_{bus}, the speed magnitude can be approximated to $\frac{\sqrt{(x_2-x_1)^2+(y_2-y_1)^2}}{t_2-t_1}$. This speed is appended to the second mobility tuple, obtained at instant t_2. The same procedure is repeated considering instants t_2 and t_3, and so on until the last tuple of bus id_{bus}. In the end, we compute all speeds to all buses, enriching the dataset with this additional information.

Considering buses as the participants of the PS system, we can think of each tuple as the contribution of a participant to the PS system. As such, the dataset has 376,491 samples, with an average of avg samples per participant bus and an average of 1,595 samples per route. Figure 4 shows the Cumulative

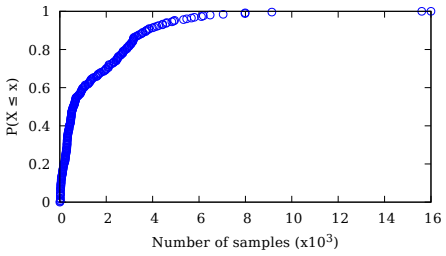


Figure 4: CDF of the number of samples per route on October 31^{st}, 2001, provided by the Ad Hoc City dataset used in this paper.

Distribution Function (CDF) of the number of samples per route in the dataset. We observe that if we gather together 90% of the routes, we obtain only 4,000 samples, approximately. This means that approximately 93 routes concentrate the majority of samples. The route with the highest number of samples is Route 007 ($id_{route} = 007$), which provides 15,601 samples. Hence, in this paper, we analyze only this route. The methodology and the results can be reproduced for any other route.

The speeds of buses are calculated based on the displacement registered in the original data, as mentioned before. The obtained results reveal some unreal values, which are not taken into account (Figure 3). The unreal values appear because of communication issues involving the GPS receiver and the GPS satellite constellation. Communications may fail because of a temporary lack of connectivity, caused by non-line-of-sight links, just to cite one typical problem. This would incur unreal positions and consequently unreal speed values. These speeds are not considered in our results, as they would also affect the sampling rate variation, possibly adding complexity to the analysis. We removed such unreal speeds from the dataset before running simulations and, therefore, our results remain impartial. To disregard the unreal values, the set of samples of each route is submitted to a validation process. We first verify whether the bus position given by the sample is within the coordinates of the city of Seattle. Then, we verify if the estimated speed, v_i, for the sample is within the interval

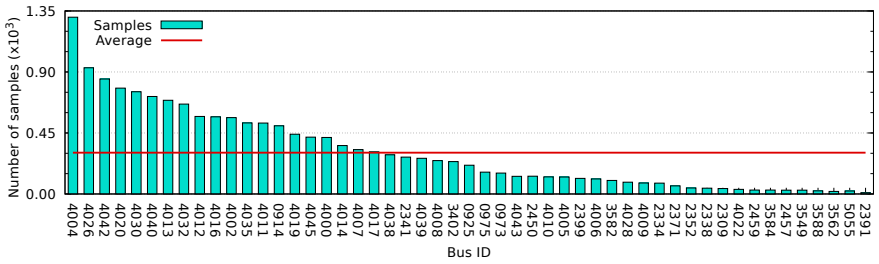


Figure 5: Total number of contributions per bus in Route 007 on October 31^{st}, 2001.

[0,v_{max} km/h], which defines the reasonable range of speeds for the scenario. The maximum bus speed allowed in the metropolitan area of Seattle[2] is approximately 56 km/h. In this paper, however, we consider $v_{max} = 110$ km/h, which is the maximum speed allowed in the entire city of Seattle[2]. The validation process applied to Route 007 results in the exclusion of 55 samples, which are equivalent to 0.35% of the total number of samples, being thus, negligible.

4.1. Characterization of Route 007

We characterize Route 007, considering its entire extent, which is approximately 15.45 km. The average duration of each trip in this route during weekdays is 44 minutes. Figure 5 shows the number of samples for each bus in Route 007, organized in decreasing order of number of samples. The x-axis represents the bus identification (id_{bus}), while the y-axis is the number of samples for the x bus. The average number of samples is shown in this figure by the horizontal red line, and it is equal to 305 samples. The Route 007 had 51 buses moving around on October 31^{st}, 2001, and the buses that collected the largest number of samples are $id_{bus} = \{4004, 4026, 4042\}$, participating with 1,304, 931, and 848 samples each, respectively. The buses with the poorest participation are $id_{bus} = \{2391, 3562, 5055\}$, with 9, 19, and 22 samples each, respectively.

The average speed along the day in Route 007 is 19.91 km/h. Aiming to

[2]Information available on the Seattle department of transportation website, `http://www.seattle.gov/transportation/sdotfaqs.htm`, accessed in July 2017

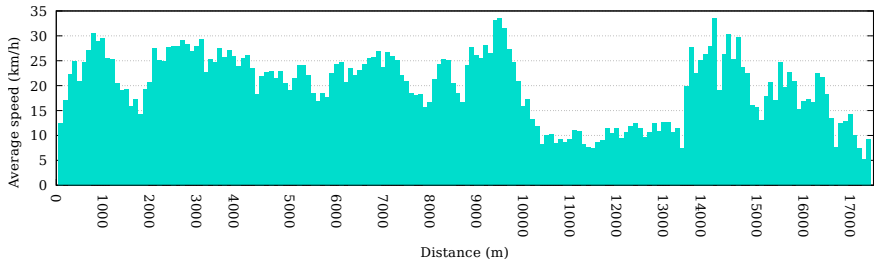
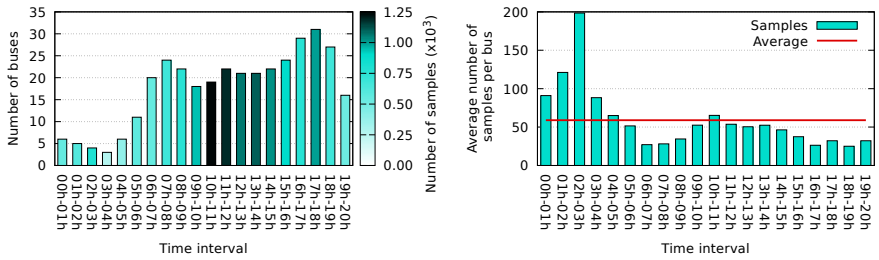


Figure 6: Average bus speed on Route 007 obtained from the subdivision of this route into segments of 100 m in length. The ingress RSU is at $x = 0$ and the egress RSU is at $x =$ 17,400 m.

evaluate this route in more detail, we divide the route into smaller segments of 100 m in length, and we compute the average speed during the day for each segment. Figure 6 shows the average speed obtained for each segment. We observe that the maximum average speed on Route 007 is 34 km/h. Moreover, buses do not move at their maximum allowed speed in the city of Seattle[2] on any segment of Route 007 during the entire day. This indicates the presence of traffic jams, bus stops, or several traffic lights in the area. We also observe in Figure 6 that some segments present a significantly lower average speed, indicating that there must be a higher concentration of traffic lights, bus stops, or a more intense traffic jam.

Figure 7(a) shows the number of buses and samples per hour in Route 007 on October 31^{st}, 2001. The x-axis represents the time interval of the day, while the y-axis gives the total number of buses at the x interval. The color grid shows the total number of samples at each interval. The darker the blue, the higher the number of samples. This figure shows that the number of buses that contributes to the sampling system varies between 3 and 31 buses, depending on the time interval analyzed. The number of samples obtained at each time interval is not proportional to the number of buses participating in the sensing system at that time interval. For instance, we observe that 31 buses participate in the system between 17 and 18 hours and that they collect 998 samples. The highest number of samples, however, is obtained between 10 and 11 hours,



(a) Number of buses overtime.

(b) Average number of samples overtime.

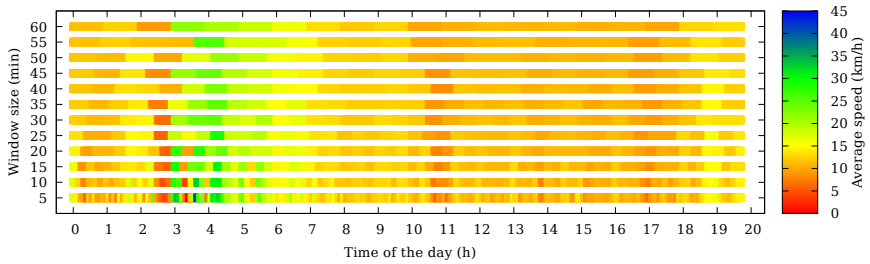
Figure 7: (a) Number of buses and samples per hour in Route 007, on October 31^{st}, 2001. (b) Average number of samples per bus at each hour of the day.

totaling 1,240 samples collected by 19 buses. Between 3 and 4 hours, we have the poorest participation, totaling 265 samples collected by 3 buses. The average number of buses per hour on the analyzed day is 19, and the average number of samples per hour is 777 samples.

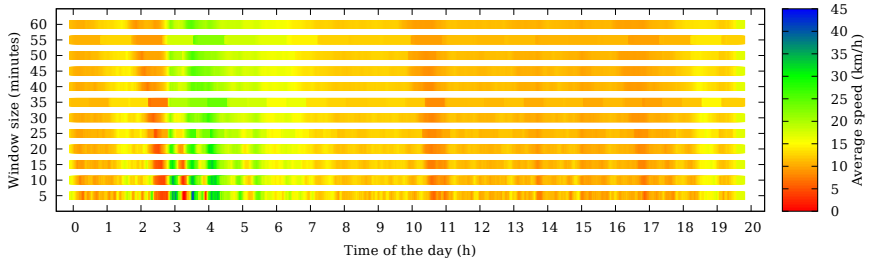
Figure 7(b) shows the average number of samples per bus at each hour of the day. The total average is 58 samples per bus, and the interval between 2 and 3 hours has the highest average per bus, totaling 199 samples per bus on average. This happens because at this time of the day there are very few buses that are responsible for a significant number of samples, as shown in Figure 7(a). Between 10 and 11 hours, the interval with the highest number of samples, the average number of samples per bus is 59. Smaller averages are interesting because this means that the volume of data sent by a single bus to an RSU is not high. Nevertheless, the average value cannot be very small because the samples together must remain a representative for the entire RoI.

5. Evaluation of the Proposed Sensing System

The proposed system considers both temporal and spatial aspects of the RoI. The spatial aspect is taken into account as we divide the RoI into segments, ς_i, while the temporal aspect is considered when we divide time into intervals, Δt. To evaluate the proposed sensing system, first, we need to define the most



(a) Fixed-time window.



(b) Sliding window.

Figure 8: Analysis of the size of the time window $\boldsymbol{\Delta t}$ comparing the fixed-time and sliding window approaches.

suitable value for the time window, Δt, to be considered for the evaluation. Then, using the chosen time window, we can evaluate the performance of our system in terms of delivery rate, data load, network load, and sampling error.

5.1. Time window evaluation

The time window Δt is important to obtain the average speed of the buses and the number of collected samples per interval. As these features depend on Δt and they are essential to the operation of the system, we investigate which Δt is the most suitable for the size of the time window. To this end, we consider sliding and fixed-time windows. In both approaches, we assume that the first window begins at time 0 h and the last window finishes at time 20 h, which is the total duration of the trace for route $id_{route} = 007$ on October 31^{st}, 2001.

We investigate the size of the time window, Δt, varying the range of possible values from 5 to 60 minutes and considering time steps of 5 minutes. Figure 8(a)

shows the result for the fixed-time window approach and Figure 8(b), for the sliding window approach. In these figures, the color of each pair (x,y) represents the average speed calculated at time x using a time interval of y minutes. In both approaches, we observe that shorter windows capture greater and finer variations in the average speed. The behavior obtained for both approaches is very similar, but there is some difference at the moment of transition between two consecutive fixed-time windows compared to the corresponding interval using the sliding window approach. It is noticeable that when we increase the window size in both approaches, we have a reduction in information granularity. Besides, when using the fixed-time window approach, there is also a loss of information at each transition. Nevertheless, the loss of information does not change, significantly, the interpretation of the current state of the route. Even when considering a fixed-time window of 60 minutes, we can still observe the same overall behavior of the sliding window with $\Delta t > 35$ minutes. Without loss of generality and because fixed-time windows are easier to implement, in the remaining of this paper, we use a fixed-time window of 60 minutes to evaluate our proposed sensing system.

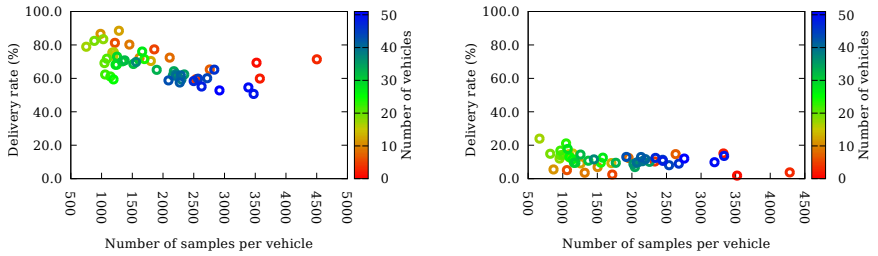
5.2. Performance evaluation

We evaluate the performance of the proposed sensing system in terms of data delivery rate, sampling error, and data and network loads. The collected measurement is the speed of buses on a predefined segment of the route. The network administrator defines these segments. In our scenario, we consider that Route 007 is divided into segments of 100 m. The mobility and communication in the scenario are simulated using the network simulator NS-3, version 3.26[3].

5.3. Data delivery rate

The evaluation of the data delivery rate verifies the tradeoff between the number of nodes and the volume of data transferred per node. The goal is

[3]Available on `https://www.nsnam.org/overview/780what-is-ns-3/`, accessed in July 2017



(a) Egress RSU (RSU at distance x = 17,300 m, seen in Figure 6).

(b) Ingress RSU (RSU at distance $x = 0$ m, seen in Figure 6).

Figure 9: Delivery rate at each RSU located at the egress and ingress points on Route 007, varying the number of crowdsourcing participant buses and the data load produced by each one.

to deliver the data collected by buses within the Region of Interest (RoI) to the RSUs. To accomplish that, we analyze the network capacity and calculate the data delivery rate through simulations. We vary the number of buses and the number of samples per bus to be transferred to an RSU placed at the left extremity (ingress RSU at distance $x = 0$ m) and another RSU placed at the right extremity (egress RSU at distance $x = 17{,}300$ m) of Route 007, as seen in Figure 6.

Figure 9 presents the relation between the delivery rate, the average load transferred from buses to RSUs, and the number of buses competing to access the medium during the data transfer. The delivery rate at the egress RSU is higher when the average load per bus is lower, e.g., 1,000 samples per bus. The delivery rate decreases as the average load increases, reaching up to $\approx 60\%$ for a data load of $\approx 2{,}500$ samples per bus. This can be due to the short duration of contacts between the buses and the RSU, which may not be sufficient to transfer the entire data completely. The number of buses contending for the medium also reduces the delivery rate. For instance, considering the same load, 1,500 samples per bus, the delivery rate is $\approx 65\%$ if we have ≈ 25 buses, and $\approx 80\%$ for ≈ 10 buses. This reduction can be due to the increase in the number of collisions when more than one bus tries to transfer data simultaneously to

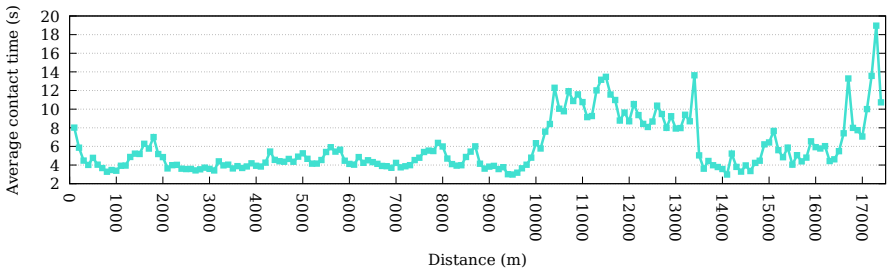


Figure 10: Average contact time between buses and RSUs placed at each 100 m. It is clear that buses contacting the egress RSU (x = 17,300) have much longer lasting contact times than the ones contacting the ingress RSU ($x = 0$).

the same RSU. It is noteworthy that, for higher loads, above 3,500 samples per bus, data delivery has higher rate when very few buses contend for the medium.

At the ingress RSU, both the number of buses and the load per bus similarly influence the delivery rate. Even for lower loads and fewer buses, the delivery rate is low. The difference between the behaviors observed for the egress and ingress RSUs is a consequence of the shorter contact time between buses and the ingress RSU. Figure 10 shows the average contact time along Route 007 to corroborate the expectation. The average contact time with the ingress RSU is much shorter than with the egress RSU. This is justified by the higher speed observed for buses traveling on the road segments closer to the ingress RSU (Figure 6). At both ingress and egress RSUs, data delivery of higher loads happens only when the number of buses contending for the medium is small. For instance, to have a chance to succeed in the data delivery, the number of buses must be lower than 5 when more than 3,500 samples per vehicle exist. At the ingress RSU, however, this delivery is highly inefficient due to the short contact time between the few buses and the RSU.

Considering the egress RSU, where the average contact time is sufficient to have high delivery rates, we can conclude that higher delivery rates can be obtained when several buses participate in the system and the individual data load of each bus is low. Hence, it is not interesting to overload buses, requesting them to sample a high volume of data individually. This is important because

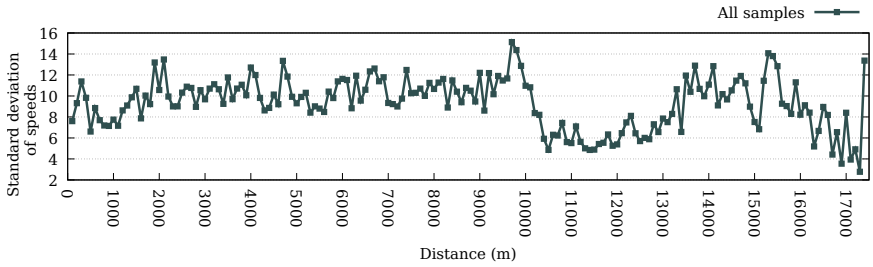


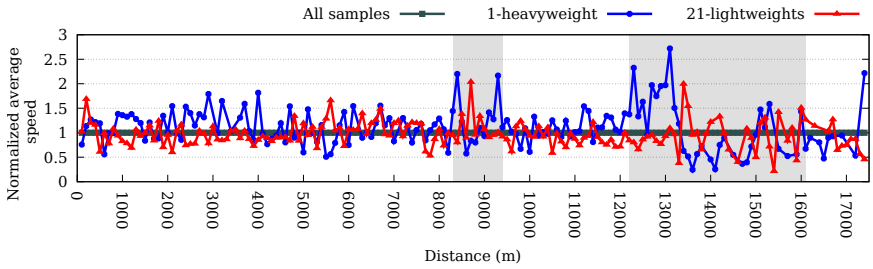
Figure 11: Standard deviation of the speeds along Route 007 taking into account all possible samples in the raw scenario.

it confirms that the PS system operation is improved when the data load is distributed among the participants. Moreover, even considering the participant viewpoint, data overloading is not attractive because it can quickly deplete participants' resources.

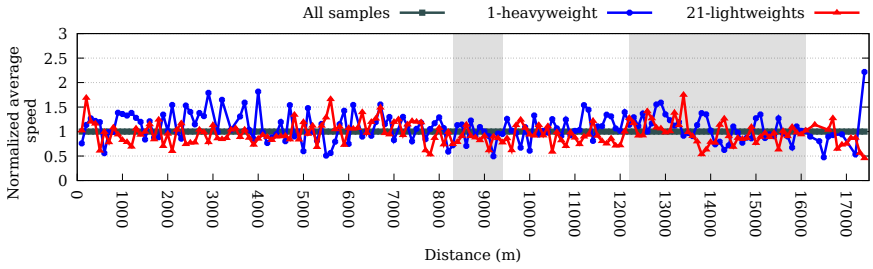
5.4. Sampling error

In an ideal scenario, the participatory sensing system knows precisely the volume of data needed to reproduce the sensed information with small error. This is advantageous, as this knowledge would avoid under or oversampling. Consequently, we could avoid estimation errors and waste of resources, respectively. As such ideal condition does not exist, we consider that, even though the proposed system cannot determine the exact volume of data required for exact results, it can observe regions needing more or less samples. This is based on the analysis of the data variability, i.e., the standard deviation of the samples considered throughout the different segments. Figure 11, for instance, shows the standard deviation of the speed along Route 007 taking into account all possible samples. It is quite easy to observe that in some segments the standard deviation is higher whereas in other it is lower. Hence, the system must be able to define upper and lower tolerated levels to determine regions requiring more or less samples. In this paper, as discussed next, we define this regions by visual inspection.

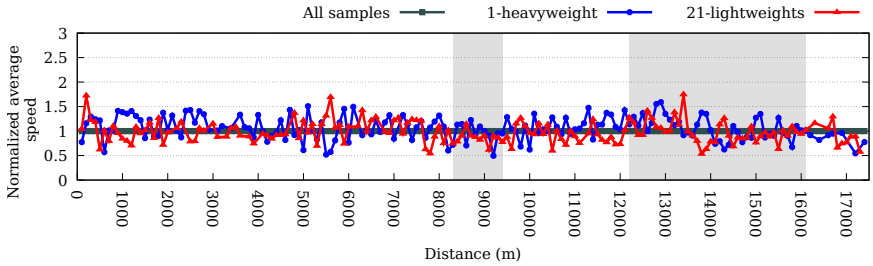
Unlike Figure 11, Figure 12 shows the variability of the average speed mea-



(a) Raw sensing system, all segments have a constant average sampling rate equal to R.



(b) Proposed sensing system working on the segments with higher variability (filled in gray), the sampling rates in the gray regions are equal to $2R$.



(c) Proposed sensing system working on the segments with lower (filled in white) and higher variability (filled in gray), the sampling rates in the white regions are equal to $0.5R$ and in the gray regions, $2R$.

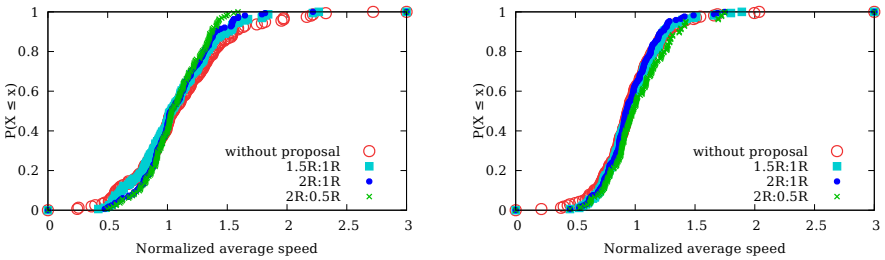
Figure 12: Normalized average speed of one bus ("1-heavyweight") and 21 buses that collect the same volume of data as the 1-heavyweight bus ("21-lightweights") on the 100-meter-long road segments. The gray region represents the segments with greater differences in the average speed of buses.

Table 1: Buses within the "1-heavyweight" and "21-lightweight" bus groups.

Group	**Bus identification**
1-heavyweight	4004
21-lightweights	2309, 2334, 2338, 2352, 2391, 2399, 2457, 2459, 3549, 3562, 3582, 3584, 3588, 4005, 4006, 4009, 4010, 4022, 4028, 4043, 5055

surements with and without the proposed sensing system, considering that not all buses collect all possible samples. Instead, we consider only the bus that collects the largest number of samples, called hereinafter "1-heavyweight", and the set of buses that sum up the same number of collected samples compared to the "1-heavyweight" bus, hereinafter called the "21-lightweights", which includes 21 buses. The identification of the buses in each set is shown in Table 1. This separation is important to find out if it is better to assign the task to only one participant with many resources or to divide the task into smaller ones distributed among several participants with limited resources. In Figure 12, the averages are normalized by the average speed considering all the samples in the dataset for each 100 m segment. The adaption of the sampling rate occurs when the system determines that more samples must be collected in a given region where higher variability is observed, or when the system detects a region with lower variability and reduces the sampling rate. The regions with higher variability are filled in gray, meaning that the variability around the horizontal line, representing the normalized average speed (equals to 1), is greater; whereas the regions with lower variability are filled in white. Even though in this paper we determine these regions by visual inspection for the sake of simplicity, other techniques could fit well. We first show the result of the raw sensing system, i.e., without our proposal, in Figure 12(a). Then we apply our proposal only to the higher variability regions to obtain the result shown in Figure 12(b). For last, we apply our proposal to both variability regions, obtaining the result in Figure 12(c).

Figure 12(a) shows the results considering an average sampling rate, R, where $R = \frac{\#\text{ of samples}}{\#\text{ of segments}}$, approximately equal to 7.5 in all segments. This



(a) 1-heavyweight.

(b) 21-lightweights.

Figure 13: CDF of the normalized average speeds considering the entire Route 007.

means that 1,304 samples are collected throughout the 174 segments of Route 007. Note that the averages have higher variability on the road segments filled in gray. If the sampling rate is twice as fast, i.e., $2R = 15$, only for the segments with higher variability (filled in gray), we already observe an improvement, as shown in Figure 12(b). The adaption of the sampling rate can also be performed in segments with lower variability (white region). In this case, it is possible to reduce the volume of data collected in lower variability regions, simultaneously avoid oversampling and save participants' resources. Figure 12(c) shows the result considering a sampling rate of $0.5R$ in segments with lower variability (white region) and sampling rates of $2R$ in regions with higher variability (gray region). We can verify that reducing the volume of data collected in the lower variability region results in data variability similar to the one shown in Figure 12(b), and we still have a significant reduction of the measurement error compared with the original configuration with adaptive sampling. Hence, we verify the possibility to compensate for the higher sampling rate in higher variability regions, using fewer data in lower variability regions, but without losing information.

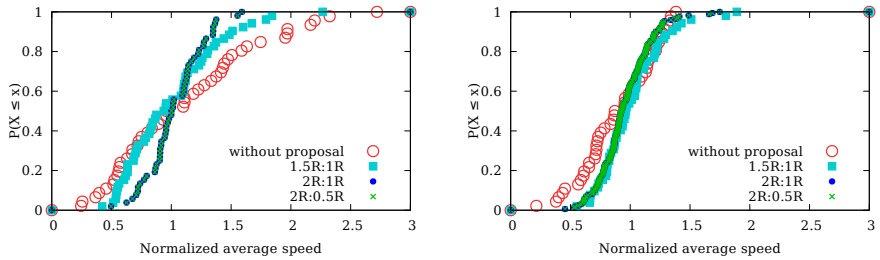
Figure 13 shows the CDF of the normalized average speeds using different combinations of the following sampling rates: $1R$, $1.5R$ and $2R$ for the higher variability regions (gray regions), and $1R$ and $0.5R$ for the lower variability regions (white regions). This is equivalent to plot all the normalized average speeds shown in the y-axis of Figure 12 as a CDF, separating the results obtained

for the 1-heavyweight and the 21-lightweights. As such, the closest to a step function in $x = 1$ is the CDF curve, the smaller is the difference between the real and the measured values of the average speed. Table 2 shows how the sampling rates are combined to obtain the results in Figure 13. We observe that, when the sampling rate in the gray regions is increased from $1R$ to $1.5R$ or $2R$, maintaining the sampling rate in the white region constant at $1R$, the differences of average speeds are reduced (closer to a step function at $x = 1$) for both the 1-heavyweight bus, Figure 13(a), and the 21-lightweight buses, Figure 13(b). This is represented by the configurations $1.5R : 1R$ and $2R : 1R$. Such a reduction is small because the sampling rate is increased only for the gray region, which represents only a few segments. The same trend is observed for configuration $2R : 0.5R$, even though the result for 21-lightweight buses is quite similar to the original configuration, as we will explain shortly.

If we isolate only the segments with higher variability, as shown in Figure 14, we observe more clearly that the differences between average speeds reduce when the sampling rate increases, even for the configuration $2R : 0.5R$. Note that the measured average speeds become closer to the ones existing in the dataset as the CDF curve approaches a step curve at $x = 1$. Hence, measured values lower than $x = 1$ represent average speeds lower than the ones found in the dataset, whereas measured values higher than $x = 1$ represent average speeds higher than the ones found in the dataset. The difference is more significant to the 1-heavyweight group, seen in Figure 14(a). Further analyzing the 21-lightweight

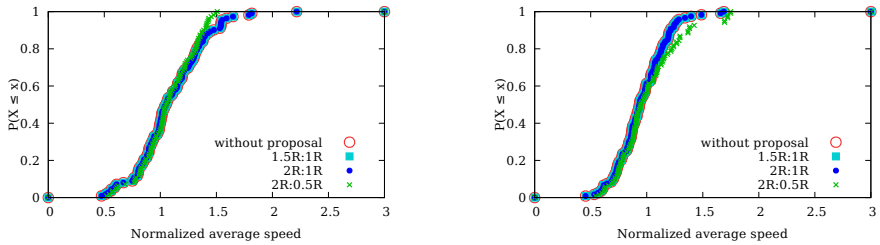
Table 2: Combination of the sampling rates used in the simulation of the proposed system.

Name	**Segments with higher variability (gray regions)**	**Segments with lower variability (white regions)**
Without the proposal	$1R$	$1R$
$1.5R : 1R$	$1.5R$	$1R$
$2R : 1R$	$2R$	$1R$
$2R : 0.5R$	$2R$	$0.5R$



(a) 1-heavyweight. (b) 21-lightweights.

Figure 14: CDF of the normalized average speeds in the segments with higher variability (gray regions).



(a) 1-heavyweight. (b) 21-lightweights.

Figure 15: CDF of the differences of average speed in the segments with lower variability (white region).

group, we observe that the impact is smaller because the samples are randomly picked within the region with higher variability. Hence, the number of picked samples may not cover all the segments in the gray region equally. This explains the performance of the configuration $2R : 0.5R$ for the 21-lightweight buses seen in Figure 13.

Isolating the CDF only for the segments with lower variability, as seen in

Table 3: RMSE of average speeds considering the entire Route 007.

Group	**without proposal**	$\mathbf{1.5R : 1R}$	$\mathbf{2R : 1R}$	$\mathbf{2R : 0.5R}$
1-heavyweight	0.422	0.342	0.290	0.264
21-lightweights	0.441	0.387	0.366	0.350

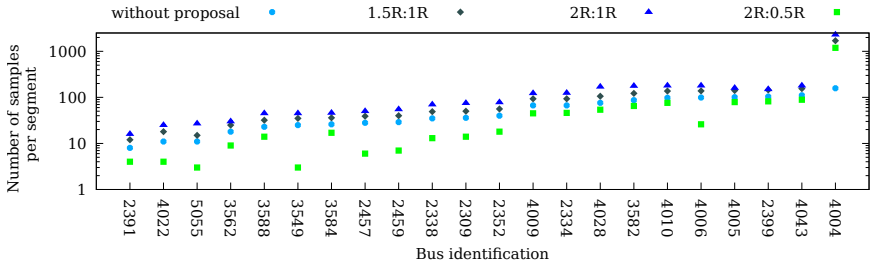
Figure 15, the behavior of the CDF curve with the rate adaption is similar to the curve without the adaption, i.e., without the proposed system. Note that the configurations $1.5R : 1R$ and $2R : 1R$ have the same result because they only modify the sampling rate in regions with higher variability (gray regions). The configuration $2R : 0.5R$, which applies the $0.5R$ sampling rate to the white region, can maintain the measurement error low for both 1-heavyweight and 21-lightweight groups. It is important to highlight that, as the variability in the white region is smaller, inserting a little bit more variability is tolerable as a tradeoff to save participants' resources.

Table 3 confirms the results showing the reduction of the Root Mean Square Error (RMSE) of the average speeds compared with those found at the original dataset. It is noticeable that the error tends to reduce in the entire route when the sampling rate is increased on the segments with higher variability (gray regions). The RMSE is even smaller when we simultaneously decrease the sampling rate of the segments with lower variability (white regions) and increase the sampling rate to $2R$ in the segments with higher variability (gray regions). This result is very interesting if we compare it with the results of the $2R : 1R$ configuration. We claim that this is a consequence of the samples used.

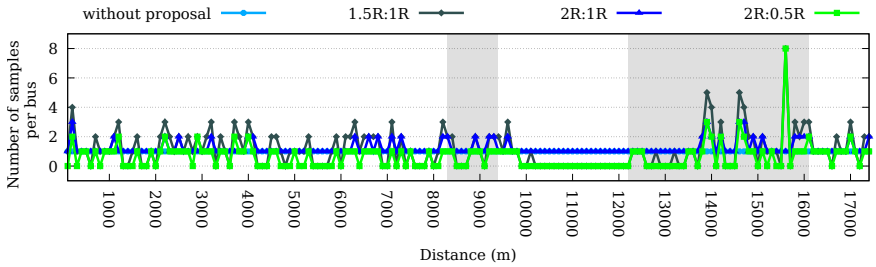
We would like to emphasize that, even using only two gray regions, which were visually chosen, we can prove the feasibility of the proposed PS system. The same conclusion could be possible even for different types of collected data, e.g., we could be interested on road luminosity and not on vehicular speeds. More elaborated strategies, not based only on data variability, could also be used to improve the system. Nevertheless, we chose to maintain the simplicity of the system to prove that even relying on simple choices, an adaptive PS system for a highly dynamic environment such as IoV is feasible.

5.5. Data load

We investigate the volume of data carried by each participant, as this volume can increase in road segments with higher variability (gray region) and can reduce in road segments with lower variability (white region). In Route 007,



(a) Number of samples per segment for each bus.



(b) Number of samples per bus on each segment.

Figure 16: Performance comparison of the data load over buses and in road segments in the raw and in the proposed systems.

there is a total of 51 buses traveling through 174 road segments. Not all of them adapt the sampling rate and, moreover, not all segments need higher sampling rates. Figure 16 shows a comparison between the data load with and without the adaptive system for both 1-heavyweight and 21-lightweight buses, according to the scenarios described in Table 2.

Figure 16(a) shows the comparative results for buses. The x-axis represents the bus identification, and the y-axis represents the number of samples per segment for the x bus. We observe a reduction of the number of samples on each segment when the proposed sensing system is used with the configuration $2R : 0.5R$, meaning that the data load for each bus decreases. This happens because the rate assigned in the white region ($0.5R$) compensates the higher rate applied in the gray region ($2R$). This is true only for the 21-lightweight group because they present a data load distributed among the participants,

compared to the 1-heavyweight bus ($id_{bus} = 4004$). Again, we confirm that by distributing the sampling task, we can save individual resources. Using configurations $1.5R : 1R$ and $2R : 1R$, the number of samples per segment grows, consequently increasing the load for each bus, as we only change the sampling rate in the higher variability region (gray region).

Figure 16(b) compares the load per bus on each road segment. In the x-axis, we have the distance the segment is from the ingress RSU of Route 007, and the y-axis shows the number of samples per bus. The measurements are normalized according to the results obtained without the proposal. In the configuration $2R : 0.5R$, we obtain, for most of the segments, a more significant reduction of the number of samples per bus. The only segment that presents a higher load is the one at 15,400 m, where there exists a greater concentration of samples owned by the 1-heavyweight bus. This segment is also in the region of higher variability, meaning that the 1-heavyweight bus was in charge of collecting a great number of samples, incurring a high data load. This problem could be mitigated if more buses collected samples in this segment as proposed in the 21-lightweight alternative.

We show in Table 4 the summary of important observations related to the system configuration with the best results found for the configuration $2R : 0.5R$. Concerning the crowdsourcing participants (buses), we observe that 43.14% of them changed their sampling rate. Among these, $\approx$ 95.46% reduced their sampling rate. This adjustment represents a $\approx$ 67.49% reduction in the data load of participants. Besides, in $\approx$ 57.46% of the segments, there is a sampling rate change. In most of them, the system configuration reduced the sampling rate in $\approx$ 86%, showing that the adaption of the sampling rate is important for routes similar to Route 007. This result confirms that in most cases, considering either buses or segments, the sampling rate is over-dimensioned and can be reduced to save participants' resources.

It is also important to discuss the performance of the proposed system when the RSU fails to update the sampling rates assigned to a participant. Participants execute only the sampling rates received in the vector $\boldsymbol{\pi}^{(\boldsymbol{\Delta t})}$ at the time

Table 4: Comparison of the volume of data with and without the proposed sensing system, considering the point of view of buses and segments and using the configuration $2R : 0.5R$.

Point of view	**Sampling rate change**	**Decreased sampling rate**	**Reduced data load**
Bus	43.14%	95.46%	67.49%
Segment	57.46%	86.00%	67.49%

interval Δt. After the time interval Δt and also out of the segments listed in the vector $\boldsymbol{\pi}^{(\boldsymbol{\Delta t})}$, participants do not contribute to the participatory system. This could lead to participants that could not contribute because of lack of reliability, which is an effect mitigated at a global picture by the total number of participants per segment. The sampling error can be maintained at an acceptable value, as a consequence of crowdsensing.

5.6. Network load

It is important to highlight that the sampling rate could be increased for the entire route if needed, as the system is still far from the maximum number of samples in the dataset and the maximum capacity of the network, as illustrated in Figure 9. The Route 007 has a total of 15,546 samples, but in the experiment with the smallest RMSE ($2R : 0.5R$), as few as 1,696 samples are collected, representing a reduction of 67.49% in the number of samples, as presented in Table 4. This value does not represent a problem, as it is smaller than the total number of samples in the dataset and still has a high probability of being successfully delivered to the RSU during the contact.

6. Conclusions and Future Work

The main goal of this paper was to investigate and to present a mechanism to adjust the data sampling rate in a Region of Interest (RoI) and to show that it is possible to perform this adjustment in a Participatory Sensing (PS) system developed for high mobility RoIs, such as Vehicular Ad Hoc Networks

(VANETs). Hence, this paper proposed an adaptive participatory sensing system that concerns about data consistency and considers both spatial and temporal characteristics of an RoI. At the same time, the proposed sensing system guarantees a reduction of the sampling error. This reduction is achieved for segments within regions of higher data variability. In regions of lower variability, decreasing the sampling rate can increase the sampling error. Nevertheless, such growth is not significant. To evaluate the proposed sensing system, we divide the RoI into segments, and we assume that the measurements vary within a time interval Δt. In our Participatory Sensing system, participants collect samples obeying a sampling rate defined by a central server, and the accuracy of the system depends on this sampling rate. The higher the sampling rate, the better the results. Increasing the sampling rate indefinitely, however, is not a smart approach to participatory sensing systems, which depend on the participants' devices to collect data. There must be a tradeoff between the system accuracy, the volume of data transferred to the communication infrastructure, and the data load imposed to the crowdsourcing participants. The consistency of the collected data must also be verified before the processing server can use it to generate useful information. Hence, the PS system proposed in this paper tackles these problems, considering the adjustment of the sampling rate on each segment as a function of the variability of the measurements. Segments with higher variability shall have higher sampling rates, whereas segments with lower variability can have lower sampling rates.

Our system is evaluated using a real dataset that registers the location of city buses. We consider that buses are responsible for collecting samples about their own speeds, in order to determine the average speed experimented along the analyzed route. The results showed that higher data volume can be successfully delivered only if there is a small number of buses competing for the medium. This shows that distributing sensing tasks is indeed the most appropriate solution for data sensing. Besides, the adaptive sampling rate reduces sampling errors and further decreases the volume of data that needs to be transferred to the processing server. Changing the sampling rate of as much as $\approx 43\%$ of

buses, the proposed system reduces 67.49% of the data load on the system.

As future work, we intend to evaluate more routes and other scenarios consisting of different types of vehicles, such as personal vehicles and taxis. The idea is to automate the identification of the best sampling rate per segment for each participant. We also intend to develop a framework that can consolidate and manage the data of a participatory sensing system. Finally, we will also consider reliability as a requirement for communications between crowdsourcing participants and RSUs.

Acknowledgments

The authors thank CAPES, CNPq, FAPERJ, and FAPESP for their financial support. This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001. This work was partially funded by grants n^o 15/24494-8 and n^o 15/24490-2 from FAPESP.

References

[1] Y. Xiao, P. Simoens, P. Pillai, K. Ha, M. Satyanarayanan, Lowering the barriers to large-scale mobile crowdsensing, in: 14th Workshop on Mobile Computing Systems and Applications, 2013, pp. 9:1–9:6.

[2] F. Restuccia, S. K. Das, J. Payton, Incentive mechanisms for participatory sensing: Survey and research challenges, ACM Transactions on Sensor Networks 12 (2) (2016) 13:1–13:40.

[3] V. Ribeiro Neto, D. S. V. Medeiros, M. E. M. Campista, Analysis of mobile user behavior in vehicular social networks, in: International Conference Network of the Future (NOF 2016), 2016, pp. 1–5.

[4] Y. Liu, J. Niu, J. Ma, L. Shu, T. Hara, W. Wang, The insights of message delivery delay in vanets with a bidirectional traffic model, Journal of Network and Computer Applications 36 (5) (2013) 1287–1294.

[5] M. Miche, T. M. Bohnert, The internet of vehicles or the second generation of telematic services, ERCIM News 77 (2009) 43–45.

[6] R. Yu, Y. Zhang, H. Wu, P. Chatzimisios, S. Xie, Virtual machine live migration for pervasive services in cloud-assisted vehicular networks, in: 8th International ICST Conference on Communications and Networking in China (CHINACOM), 2013, pp. 540–545.

[7] P. H. C. Caminha, F. F. da Silva, R. G. Pacheco, R. de Souza Couto, P. B. Velloso, M. E. M. Campista, L. H. M. K. Costa, SensingBus: Using bus lines and fog computing for smart sensing the city, IEEE Cloud Computing 5 (5) (2018) 58–69.

[8] P. Mohan, V. N. Padmanabhan, R. Ramjee, Nericell: rich monitoring of road and traffic conditions using mobile smartphones, in: ACM conference on Embedded network sensor systems, 2008, pp. 323–336.

[9] P. Zhou, Y. Zheng, M. Li, How long to wait?: predicting bus arrival time with mobile phone based participatory sensing, in: 10th international conference on Mobile systems, applications, and services, 2012, pp. 379–392.

[10] Y. Zeng, K. Xiang, Adaptive sampling for urban air quality through participatory sensing, Sensors (Basel) 17 (11) (2017) 1–16.

[11] H. Weinschrott, J. Weisser, F. Durr, K. Rothermel, Participatory sensing algorithms for mobile object discovery in urban areas, in: IEEE International Conference on Pervasive Computing and Communications (PERCOM), 2011, pp. 128–135.

[12] J. G. Jetcheva, Y.-C. Hu, S. PalChaudhuri, A. K. Saha, D. B. Johnson, CRAWDAD dataset rice/ad_hoc_city (v. 2003-09-11), `http://crawdad.org/rice/ad_hoc_city/20030911/bus_mobility`, traceset: bus_mobility (Sep. 2003). `doi:10.15783/C73K5B`.

[13] J. Wang, C. Jiang, K. Zhang, T. Q. S. Quek, Y. Ren, L. Hanzo, Vehicular sensing networks in a smart city: Principles, technologies and applications, IEEE Wireless Communications 25 (1) (2018) 122–132.

[14] M. K. McCall, P. A. Minang, Assessing participatory GIS for community-based natural resource management: claiming community forests in Cameroon, Geographical Journal 171 (4) (2005) 340–356.

[15] E. D'Hondt, M. Stevens, A. Jacobs, Participatory noise mapping works! an evaluation of participatory sensing as an alternative to standard techniques for environmental monitoring, Pervasive and Mobile Computing 9 (5) (2013) 681–694.

[16] M. N. K. Boulos, S. Wheeler, C. Tavares, R. Jones, How smartphones are changing the face of mobile and participatory healthcare: an overview, with example from eCAALYX, Biomedical engineering online 10 (1) (2011) 24:1–24:14.

[17] J. A. Burke, D. Estrin, M. Hansen, A. Parker, N. Ramanathan, S. Reddy, M. B. Srivastava, Participatory sensing, in: World Sensor Web Workshop, ACM Sensys, 2006, pp. 1–5.

[18] D. Estrin, K. M. Chandy, R. M. Young, L. Smarr, A. Odlyzko, D. Clark, V. Reding, T. Ishida, S. Sharma, V. G. Cerf, et al., Participatory sensing: applications and architecture, IEEE Internet Computing 14 (1) (2010) 12–42.

[19] C. Zhang, X. Lin, R. Lu, P. Ho, X. Shen, An efficient message authentication scheme for vehicular communications, IEEE Transactions on Vehicular Technology 57 (6) (2008) 3357–3368.

[20] S. Reddy, K. Shilton, J. Burke, D. Estrin, M. Hansen, M. Srivastava, Evaluating participation and performance in participatory sensing, in: Proc. of the International Workshop on Community and Social Applications of Networked Sensing Systems, 2008, pp. 1–5.

[21] K. L. Huang, S. S. Kanhere, W. Hu, Are you contributing trustworthy data?: The case for a reputation system in participatory sensing, in: Proceedings of the 13th ACM International Conference on Modeling, Analysis, and Simulation of Wireless and Mobile Systems, 2010, pp. 14–22.

[22] A. Dua, N. Bulusu, W.-C. Feng, W. Hu, Towards trustworthy participatory sensing, in: Proc. of the 4th USENIX conference on Hot topics in security, 2009, pp. 1–6.

[23] C. Miao, Q. Li, L. Su, M. Huai, W. Jiang, J. Gao, Attack under disguise: An intelligent data poisoning attack mechanism in crowdsourcing, in: Proceedings of the 2018 World Wide Web Conference, 2018, pp. 13–22.

[24] J. Whitbeck, Y. Lopez, J. Leguay, V. Conan, M. D. de Amorim, Push-and-track: Saving infrastructure bandwidth through opportunistic forwarding, Journal of Pervasive and Mobile Computing 8 (5) (2012) 682–697.

[25] N. Cheng, N. Lu, N. Zhang, X. S. Shen, J. W. Mark, Vehicular WiFi offloading: Challenges and solutions, Vehicular Communications 1 (1) (2014) 13–21.

[26] B. Baron, P. Spathis, H. Rivano, M. D. de Amorim, Offloading massive data onto passenger vehicles: Topology simplification and traffic assignment, IEEE/ACM Transactions on Networking 24 (6) (2016) 3248–3261.

[27] C. H. O. M. André, D. S. V. Medeiros, M. E. M. Campista, A methodology to assess data consistency in vehicular networks using participatory sensing, Tech. Rep. GTA-17-31, GTA/PEE/UFRJ (2017).

[28] J.-S. Lee, B. Hoh, Dynamic pricing incentive for participatory sensing, Pervasive and Mobile Computing 6 (6) (2010) 693 – 708.