Weighted Adaptive Opportunistic Scheduling Framework for Smartphone Sensor Data Collection in IoT

Thejaswini M[1], Bong Jun Choi[2*]
[1] Department of Computer Science and Engineering,
Indian Institute of Information Technology, Bhagalpur, India
[e-mail: mthejaswini.cse@iiitbh.ac.in]
[2] School of Computer Science and Engineering & School of Electronic Engineering,
Soongsil University, Seoul, Korea
[e-mail: davidchoi@soongsil.ac.kr]
*Corresponding author: Bong Jun Choi

Abstract

Smartphones are important platforms because of their sophisticated computation, communication, and sensing capabilities, which enable a variety of applications in the Internet of Things (IoT) systems. Moreover, advancements in hardware have enabled sensors on smartphones such as environmental and chemical sensors that make sensor data collection readily accessible for a wide range of applications. However, dynamic, opportunistic, and heterogeneous mobility patterns of smartphone users that vary throughout the day, which greatly affects the efficacy of sensor data collection. Therefore, it is necessary to consider phone users mobility patterns to design data collection schedules that can reduce the loss of sensor data. In this paper, we propose a mobility-based weighted adaptive opportunistic scheduling framework that can adaptively adjust to the dynamic, opportunistic, and heterogeneous mobility patterns of smartphone users and provide prioritized scheduling based on various application scenarios, such as velocity, region of interest, and sensor type. The performance of the proposed framework is compared with other scheduling frameworks in various heterogeneous smartphone user mobility scenarios. Simulation results show that the proposed scheduling improves the transmission rate by 8 percent and can also improve the collection of higher-priority sensor data compared with other scheduling approaches.

1. Introduction

Internet of Things (IoT) technology is emerging rapidly and will become part of our day-to-day lives in the future by connecting anything, anywhere, at any time [1], [2], [3], [4], [5]. IoT comprises nearly all industries and applications such as home automation, building automation, health care, education, environment maintenance, intelligent transportation, smart appliances, smart campuses, smart grids, and smart cities [6], [7], [8], [9], [10]. Some believe that in the near future, hundreds of billions of devices will be connected to each other through the Internet using various communication technologies.

Smartphones are a rich platform for IoT as they not only are equipped with multiple communication technologies and powerful processing power, but can also be equipped with many different types of sensors such as ambient light and proximity detector, magnetometer, accelerometer, gyroscope, camera, microphone, GPS, barometer, temperature sensor, humidity sensor, fingerprint scanner, and heart rate monitor [11], [12], [13]. Moreover, with continued advancements in hardware technology, we can expect to have smartphones equipped with even more advanced and sophisticated sensors [13], [14], [15], [16]. Smartphones can also be used as relays or data gathering devices for wireless sensors networks and vehicular sensor networks.

Using smartphone sensing to facilitate IoT applications raises many research issues; one of the main issues is designing efficient scheduling for smartphone sensor data acquisition. Human-carried smartphones enable opportunistic networking due to the uncertainty of user mobility patterns. Opportunistic data collection refers to collecting smartphone sensor data whenever a smartphone opportunistically comes in contact with a designated destination such as the base station, sink node, central/cloud server.

The mobility patterns of smartphone users vary daily based on their activities (walking, running, etc.) and modes of transportation, and the success of smartphone sensor data transmissions is largely affected by these mobility patterns. Users dynamic, unpredictable, heterogeneous, and opportunistic mobility leads to unstable communication, increasing data delay and loss. Furthermore, smartphone sensor data collection is data-oriented, which means that to facilitate pervasive IoT applications, sensor data needed for the targeted application are given high priority rather than simply collecting raw sensor data. Therefore, a novel approach is needed for scheduling data collection from smartphones. The scheduling approach should also be adaptive to the unpredictable and opportunistic nature of mobility patterns of smartphone users to reduce unsuccessful data transmissions.

This paper provides a new weighted adaptive opportunistic scheduling framework for collecting smartphones sensor data. The main objectives of the proposed work are to reduce the loss of important smartphone sensors data, which is application-oriented and also to reduce the rate of unsuccessful transmissions, which is mobility dependent. This is achieved by providing prioritized scheduling that involves the calculation of adaptive scheduling duration and assignment of appropriate weights to application parameters, mobility parameters, and geographic region of interest. In particular, we make the following contributions:

- We propose a generalized weighted scheduling framework that can collect application-specific sensor data from smartphones with dynamic and heterogeneous mobility patterns by considering the velocity, the geographic region of interest, and the priority of sensors.

- The proposed scheduling framework is adaptive to the dynamic mobility patterns of smartphone users to achieve high transmission of valued sensor data.
- We demonstrate improved sensor data collection and transmission compared with randomized and distance-based scheduling frameworks under a scenario with both walking and vehicle mobility patterns.

The remainder of this paper is organized as follows. Section 2 presents related work. Section 3 describes the system model for the proposed work. Section 4 gives a description of the proposed weighted adaptive opportunistic scheduling framework. Section 5 discusses the evaluation of simulation results, and Section 6 concludes the paper.

2. Related Work

There are many scheduling algorithms in the literature for improving throughput and quality of service in mobile communications; the different algorithms use various physical, network, and application layer parameters in their scheduling approach. For example, a physical layer parameter, signal-to-interference ratio, is used for scheduling vehicle node data rates [17]. For this work, we gave higher priority to the nodes that were moving away from the base station. Some researchers used network parameters such as priority and deadline of packets to schedule time-critical IoT smart city applications to reduce packet loss rate, average waiting time, and end-to-end delay [18]. Others used application parameters such as trust, reputation, and usefulness of information to schedule a pothole detection application for smart cities to reducing data traffic and bandwidth usage [19].

There are also some smartphone-based opportunistic data collection frameworks in the literature, specifically to facilitate IoT and smart city applications. Some researchers proposed an opportunistic mobile crowdsensing framework for collecting sensor data from pedestrian mobile devices in smart cities with the aim of minimizing the cost of sensing and reporting sensed data [20]. Other authors proposed a cost-efficient smart device application for opportunistic sensor data collection to monitor indoor air quality [21]. In opportunistic mobile networks, there are more chances to miss required data due to heterogeneous mobility patterns of smart device users, and efficient data collection scheduling methodologies are required to reduce the loss of mobile data. There are also many mobile data collection scheduling methods proposed based on mobility parameters. The velocity of users (high or low), determined from the channel quality indicator, has been used to schedule LTE resources blocks [22].

Some researchers proposed a shortest-job-first random walk mobility scheduling algorithm to collect mobile node data, with scheduling time based on mobility and operation length of mobile nodes; expanding the transmission power of data collection nodes is also considered for decreasing incomplete tasks [23]. Researchers have also used flight length and pause time of a truncated Levy walk mobility model for scheduling mobile nodes [24]; these researchers gave higher priority to the mobile nodes that were more likely to visit larger areas and have smaller pause times. Other investigators proposed a human mobility-based weighted algorithm for selecting an efficient cluster head that could collect cell phone sensor data [25, 26]. The authors of those studies considered such mobility parameters as flight length and distance and mobile node velocity to be important parameters in addition to the residual energy limit of mobile nodes. A scheduling algorithm based on the duration of stay was also proposed for mobile crowd sensed data collection [27]; those authors gave higher priority to mobile nodes that stayed within range of the base station for a short period to reduce data loss.

However, when collecting sensor data from smartphones to support IoT applications, it is necessary to combine adaptive and application-oriented scheduling approaches to cope with the unpredictable, heterogeneous, and opportunistic nature of user mobility and to minimize the loss of required sensor data. To the best of our knowledge, none of the authors of the above-mentioned existing works provided combined approaches. Thus, we aim to provide a new weighted adaptive opportunistic scheduling framework for smartphone sensor data collection. Specifically, the mobility parameters (velocity, geographical region) and application parameters (specific sensor data) are combined using a weight assignment model; the weight assignments are formulated so that they can be applied readily to meet the various requirements of different IoT applications. Consideration of other parameters such as duration of stay inside the base station, arriving rate, and departure rate is not feasible as an accurate prediction of these parameters requires high computational complexity.

Moreover, we consider a heterogeneous smartphone mobility scenario that involves both vehicular mobility (where smartphone users are traveling in vehicles) and pedestrian (users are on foot) mobility. Researchers have proposed and considered heterogeneous mobility scenarios, such as a vehicle-to-pedestrian broadcasting algorithm based on the Manhattan mobility model (vehicular mobility model) and the SLAW model (human mobility model) [28]. However, our proposed scheduling framework collects data from smartphones under heterogeneous mobility scenario and can be easily applied to support various data collection application scenarios.

3. System Model

3.1 Communication Scenario

As shown in **Fig. 1**, a fixed base station BS with a maximum communication range of D_{max} is placed at the center of a simulation area of m^2. Let H and W represent the total number of vehicle nodes and pedestrian nodes in the simulation area, respectively. Here, we use the term *pedestrian node* to indicate smartphone users on foot and *vehicle node* to indicate users in vehicles. We use the term *mobile node* to denote both pedestrian and vehicle nodes; we consider these two nodes because they are the most common mobility patterns of smartphone users. However, other mobile nodes can be easily added to our framework because we use generic mobility characteristics to configure the framework. We assume that mobile nodes are embedded with sensors and GPS (to obtain velocity and current location values) and they have adequate storage space. We also assume that smartphones always have some sensor data (application related, either self-generated or collected from other devices) to transmit to the BS within the communication range.
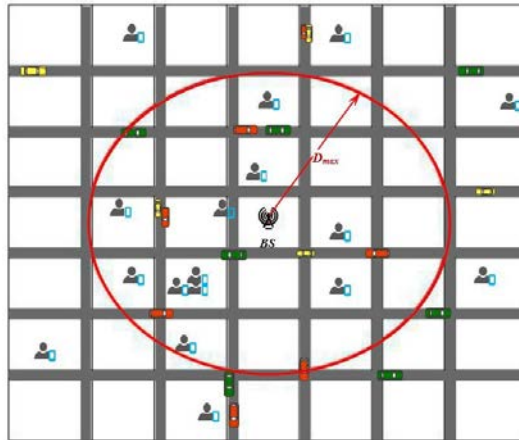
Fig. 1. Basic communication scenario.

Fig. 2. Regions based on the communication range.

Fig. 3. Regions based on road-lane

3.2 Mobility Model

To verify the effects of different mobility patterns on our proposed framework under various scenarios, we use a mixture of some well-defined mobility models. Human walking patterns closely follow Levy walk (LW) patterns [29]. Authors of [30] used outdoor GPS traces

collected from many volunteers to study human mobility patterns; proposed the LW mobility model, which depicts the statistical properties of human mobility. Therefore, we use the LW mobility model [30] to generate mobility traces of the pedestrian nodes; for the vehicle nodes, we use the well-known Manhattan mobility model using MobiSim [31], [32]. Other authors have provided more details on the statistical features of the LW and Manhattan mobility models [30, 31, 32, 33]. Roads are considered to be two-way, and the movement of the vehicle nodes is restricted to roads; the movement of pedestrian nodes is not restricted to roads.

3.2 Mobility Model

The received signal power from a transmitting source mobile node located D meters away, $P_r(D)$, is calculated using the Friis free space model [34], [35], given as

$$P_r(D) = P_t G_t G_r \frac{\lambda^2}{(4\pi D)^2},\qquad(1)$$

where P_t, G_t, G_r, and λ represent transmitted power, transmitter gain, receiver gain, and the wavelength of the transmitted wave, respectively. The received signal strength (RSS) [18, 31, 34] is calculated as

$$RSS = P_r(d_0) + 10\eta\log\frac{d_0}{D} + X_\sigma,\qquad(2)$$

where $P_r(d_0)$ and η represent received power at the reference distance d_0 and path loss exponent, respectively. X_σ is a Gaussian random variable with zero mean and variance in dB.

4. Proposed Scheduling Framework

This section provides a description of the proposed weighted adaptive opportunistic scheduling framework. The pseudocode of the framework is given in Algorithm 1.

_____

Algorithm 1: Weighted Adaptive Opportunistic Scheduling Framework
_____

1: Let M be the total number of mobile nodes in an area A where $M = H + W$ and I is the iteration number.
2: $I = 0$
3: $simTime = simStart$
4: **while** $simTime \leq simEnd$ **do**
5: $I = I + 1$
6: BS broadcasts $beaconInit$ packet
7: BS waits T_{wait} to receive replies from mobile nodes
8: $N(I)$ ($N(I) \subseteq M$) mobile nodes send $beaconReply$ packet to BS
9: BS computes the schedule duration $T_{schedDur}(I)$ of the iteration (Equation (7))
10: **for** every $N(I)$ mobile node **do**
11: BS computes the weights of mobile nodes (Equation (4))
12: **end for**
13: BS sort mobile nodes in descending order of weights
14: BS broadcasts the time slot schedule to mobile nodes specified in $sched$ packet
15: Each mobile node transmits data according to its schedule
16: **if** ($T_{schedDur}(I) \leq T_{schedDur}(I)^{lim}$) **then**
17: $T_{iteration}(I) = T_{wait} + T_{sched} + T_{schedDur}(I)$
18: $simTime = simTime + T_{iteration}(I)$

19: **else**
20: $T_{iteration}(I) = T_{wait} + T_{sched} + T_{schedDur}(I)^{lim}$
21: $simTime = simTime + T_{iteration}(I)$
22: **end if**
23: **end while**

Algorithm 2: Assignment of weights to the velocity and region of interest parameters

---

1: Assigning weight to velocity parameter
2: **if** $v_j(t_p) \in category_v$, where $v = 1,2,\ldots,V$ **then**
3: $W_{Velocity_j} = w_v^{velocity} * \gamma$ (Equation 6))
4: **end if**
5: Assigning a weight to the region of interest parameter
6: **if** $(x_j(t_p), y_j(t_p))$ lies in $region_r$, where $r = 1,2,\ldots,R$ **then**
7: $W_{RoI_j} = w_r^{region}$
8: **end if**

4.1 Initiation from Base Station

At the beginning of every scheduling iteration, the BS broadcasts a $beaconInit$ packet that consists of the ID of the BS (ID_{BS}), location coordinate of the BS (x_{BS}, y_{BS}), and a list of parameters to be obtained from mobile nodes. The requested parameters are a location coordinate of the mobile node, the velocity of the mobile node, and available sensors on the mobile node. Then, the BS waits until it receives replies from mobile nodes for T_{wait} computed as

$$T_{wait} > T_{beaconInit} + T_{beaconProc} + T_{beaconReply}, \quad (3)$$

where $T_{beaconInit}$ is the maximum time taken for the $beaconInit$ packet to reach all mobile nodes in the communication range of the BS, $T_{beaconProc}$ is the time mobile nodes take to process the required information, and $T_{beaconReply}$ is the maximum time taken for the BS to receive $beaconReply$ packets from the mobile nodes in the communication range of the BS.

4.2 Replies from Mobile Nodes

Once mobile node j, where $j = 1,2,\ldots,N$, receives a $beaconInit$ packet, it sends back the requested parameters to the BS using a $beaconReply$ packet. In detail, $beaconReply$ packets consist of the ID of the mobile node (ID_j), location coordinate of the mobile node at the time t_p ($x_j(t_p), y_j(t_p)$a), the velocity of the mobile node ($v_j(t_p)$), and the IDs and corresponding types of sensors ($sensorID_{ji}, sensorType_{ji}$) where $i = 1,2,\ldots,C$. Because each mobile node can have different number and types of sensors, C may differ for each node.

4.3 Calculating Weights

Based on the information obtained from the $beaconReply$ packets received from the mobile nodes, the BS assigns the weights of each mobile node. The weight of node j is computed as:

$$W_j = W_{Sensors_j} + W_{RoI_j} + W_{Velocity_j}, \quad (4)$$

where $W_{Sensors_j}$, W_{RoI_j}, and $W_{Velocity_j}$ are the weights of sensors, the region of interest (RoI), and mobile node velocity, respectively. Higher transmission priorities are given to those nodes with higher assigned weights.

The total weight of sensors of mobile node j ($W_{Sensors_j}$) is assigned based on the availability of sensors on the mobile node and their weights decided by the BS based on their relative importance to the assumed application. We consider that there are K important sensors to support a specific application. First, the availability of sensor type k on mobile node j is indicated using a binary variable s_{jk} where $k = 1, 2, ..., K$; here, s_{jk} equals 1 if k is available on j and 0 otherwise. Each sensor type is also given a weight w_k^{sensor} that is assigned based on the specific requirements of the assumed application. For example, for an air pollution monitoring application, a higher weight is given to the CO_2 sensors among the different sensor types. Finally, the total weight of sensors of mobile node j is calculated as

$$W_{Sensors_j} = \sum_{k=1}^{K} w_k^{sensor} \times s_{jk}. \qquad (5)$$

Consideration of multiple applications and collecting multiple sensors data at any Ith schedule iteration will lead to another set of complex tasks. Schedule duration, packets size, slot time parameters should be reconsidered when there are multiple applications. Performance in terms of collecting particular application related sensors data will decrease as multiple applications occupy data transmission slots. Therefore, it is better to consider the specific application at a time and sensor data which fit packet size $beaconReply$ (**Table 1**. Simulation Parameters). If there is a greater number of sensor types and data exceeds $beaconReply$ packet size, then it can be scheduled to collect in consecutive schedule iterations.

The pseudocode for calculating the weight of a mobile node's RoI is given in Algorithm 2. The weight of an RoI of mobile node j (W_{RoI_j}) is assigned based on the geographical region of interest around the BS, and the regions can be defined based on the application scenario, for example, based on the BS communication range or on-road lanes as shown in **Fig. 2** and **Fig. 3**, respectively. Here, each region r is given a positive integer weight between 1 and R, and its weight is assigned based on the specific requirements of the application; for example, a higher weight is given to the region closer to the edge of the BS to reduce the number of unsuccessful transmissions. Mobile nodes closer to the edge are more likely to move out of the communication range of the BS in the near future than those near the BS.

The pseudocode for calculating the weight of velocity of a mobile node is given in Algorithm 2. The weight of velocity of mobile node j ($W_{Velocity_j}$) is assigned based on its velocity at time t_p ($v_j(t_p)$). Velocities are divided into V number of categories, and each category is assigned with a positive integer between γ and $V \times \gamma$ and γ is calculated as

$$\gamma = R + K. \qquad (6)$$

The categories can be defined based on the mobility model and the specific requirements of the application; for example, if data from pedestrians is considered more important, then the category with $\leq 1.0 m/s$ is given a higher weight than other categories. Also, a higher weight can be given to the fast-moving nodes because they are more likely to move out of the BS communication range in the near future than the slow-moving nodes. Note that relatively higher weight is given to the weight of mobile node velocity than other weights (weights of sensors and geographical region of interest) as velocity more significantly affects the transmission rate of mobile nodes. Multiplying γ (Equation (6)) with V increases the weight of velocity parameter compared to the weight of sensors and geographical region of interest.

Finally, the weights of mobile nodes are normalized, and the scheduling duration (in seconds) of the given iteration (I) is computed adaptively based on the total number of mobile nodes available ($N(I)$) in the range of the BS as

$$T_{schedDur}(I) = \lfloor \frac{N(I)}{nodes\_per\_second} \rfloor, \qquad (7)$$

where $nodes\_per\_second$ is the number of nodes allowed to transmit their respective sensor data in one second. Here, each node can transmit its sensor data for $= \lfloor \frac{1000}{nodes\_per\_second} \rfloor ms$. It is important to note that the proposed scheduling approach is adaptive: $T_{schedDur}(I)$ varies at each iteration. We set a limit for $T_{schedDur}(I)$ because long schedule durations increase delays and may cause loss of important sensor data from incoming mobile nodes that are to be scheduled in the next iteration. The maximum schedule duration of an iteration is calculated as

$$T_{schedDur}(I)^{lim} = \lceil \frac{Avg\_TotalNodes}{nodes\_per\_second} + 1 \rceil \qquad (8)$$

where $Avg\_TotalNodes$ is the average number of mobile nodes in the range of the BS at any given time instance. After computing the weights of mobile nodes, the BS broadcasts a $sched$ packet that contains the schedule of transmission slots of mobile nodes for the given iteration. Data transmission of a mobile node j is considered successful if the RSS of j is greater than the receiver sensitivity threshold Rx_{sen}; otherwise, the data transmission is considered unsuccessful. The availability or absence of any sensor type k on mobile node j is assigned with a random binary value; at each successful transmission, data of respective sensor type from any mobile node, if available, is counted as one unit of data.

5. Simulation Results and Discussion

In this section, we present the performance evaluation of the proposed weighted adaptive opportunistic scheduling framework; we implement the proposed framework using MATLAB, and the simulation parameters are shown in **Table 1**. For a proof of concept, the simulation is conducted for a simple case with 100 vehicle and 100 pedestrian nodes in an area of $500\ m \times 500\ m$. As mentioned in our system model, pedestrian nodes have no restricted path; their minimum, average, and maximum velocities are 0.52 m/s, 1.35 m/s, and 2.82 m/s, respectively, with an average velocity close to the average human walking velocity [36]. The minimum and maximum velocities of vehicle nodes are 2.0 m/s and 10.0 m/s, respectively.

We use the term WAP (**w**eighted **a**daptive o**pp**ortunistic) to denote the proposed framework. Because there is no comparable framework for collecting data from smartphone sensors under heterogeneous human mobility scenarios, we compare our scheduling approach with two other general scheduling strategies, termed $Distance$ and $Random$. In $Distance$ scheduling, mobile nodes are sorted and assigned time slots according to their relative distances from the BS at time t_p, calculated as

$$\delta_j(t_p) = \sqrt{(x_{BS} - x_j(t_p))^2 + (y_{BS} - y_j(t_p))^2}. \qquad (9)$$

To reduce the number of unsuccessful transmissions, the mobile nodes that are closer to the edge or far away from the BS are given a higher priority. In the $Random$ scheduling, mobile nodes are chosen randomly, and their time slots are also assigned randomly. The number of sensor types (K) is set to 6, and named $sensor_1, sensor_2, ..., sensor_6$; here, $sensor_6$ is given the highest priority and $sensor_1$ the lowest. There are three regions of interest ($R = 3$) and five categories of velocity ($V = 5$). Usually, chances of unsuccessful data transmissions are more from mobile nodes which are at the end of communication range of BS

and moving with high velocity. To reduce unsuccessful data transmission rate, a higher weight is given to the nodes with higher velocity and to the region closer to the edge of the BS communication range.

5.1 Results under the Base Scenario

The numbers of pedestrian and vehicle nodes in the communication range of the BS in each iteration are shown in **Fig. 4**; their averages are shown using straight horizontal lines. The result shows that there is more vehicle than the pedestrian, which is because the number of incoming and outgoing vehicles increases as the velocity increases. The varying scheduling duration ($T_{schedDur}(I)$) is shown in **Fig. 5**. Our proposed framework can adaptively change the scheduling duration for each iteration according to the number of available mobile nodes ($N(I)$) in the communication range; the total number of iterations is 60 and $T_{schedDur}(I)^{lim}$ is 26 s. **Fig. 6** and **Fig. 7** show the number of successful and unsuccessful transmissions, respectively, averaged over all iterations. The standard errors for both figures are 0.49, 0.58, and 0.52 for WAP, $Distance$, and $Random$ methods, respectively, which are negligible. The proposed scheduling framework is shown to reduce the rate of unsuccessful transmissions by 8% and 10% compared with the others. Under $Distance$ scheduling, mobile nodes closer to the edge of the BS communication range are given high priority. Fast-moving vehicle nodes (or mobile nodes with high velocity) which are scheduled, but given low priority (as priority is based on distance) cause more unsuccessful transmissions. This verifies our idea that considering only a distance parameter is not a good choice to reduce unsuccessful transmissions. Still, the proposed scheduling framework achieves a higher rate of successful transmissions by considering velocity as an important parameter. Because the availability or absence of any sensor type k of mobile node j is assigned a random binary value, the number of different sensors types available averaged from the iterations varies as shown in **Fig. 10(a)**. In **Fig. 8** and **Fig. 9**, because the priority of the different type of sensor is not considered in either $Distance$ or $Random$ scheduling, those methods lose more sensor data than our proposed scheduling framework. Even though the availability of high-priority sensors ($sensor_6$ and $sensor_5$) is low (**Fig. 10(a)**), they receive a higher percentage of data than other considered sensors (**Fig. 10(b)**). This is achieved by assigning higher weights to $sensor_6$ and $sensor_5$ than to other sensor types.
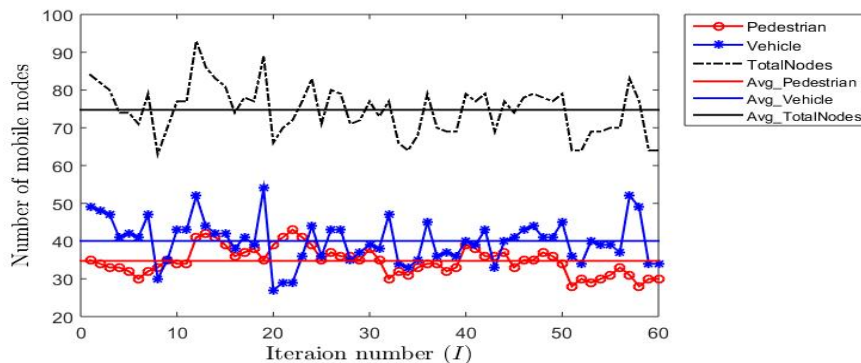

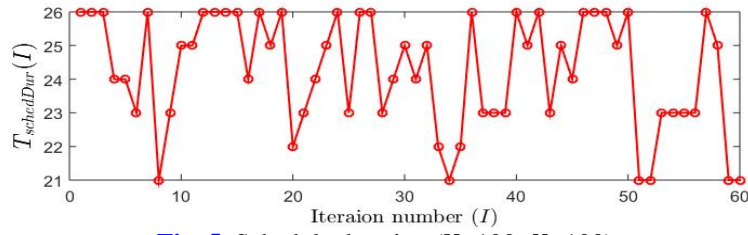**Fig. 4.** Number of mobile nodes within communication range (V=100, H =100).

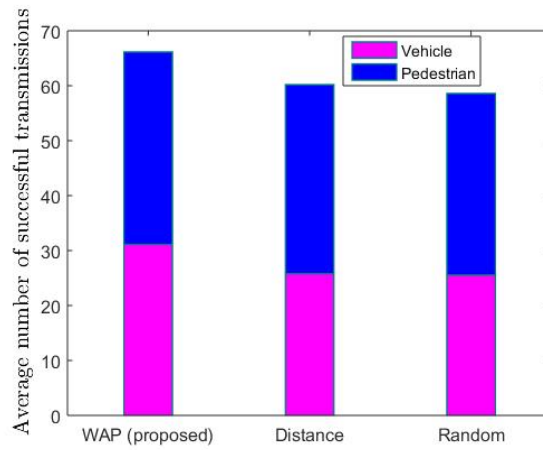**Fig. 5.** Schedule duration (V=100, H=100).

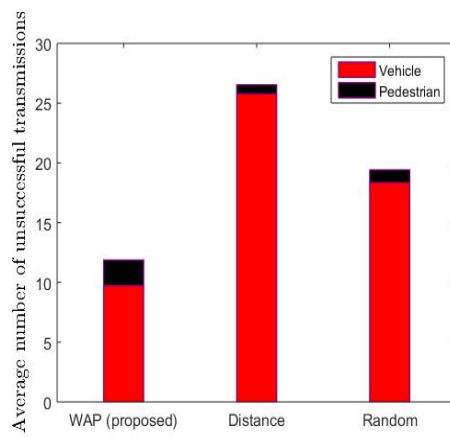**Fig. 6.** Average number of successful transmissions (V=100, H=100).

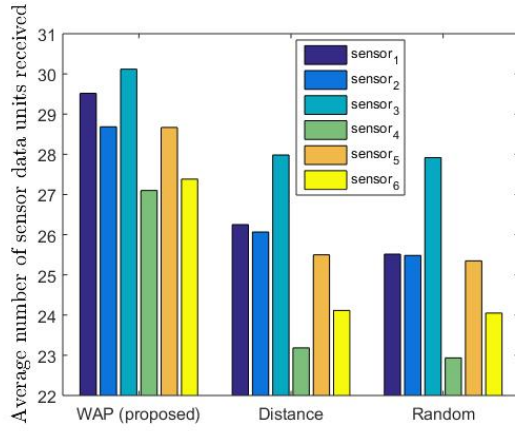**Fig. 7.** Average number of unsuccessful transmissions (V=100, H=100).

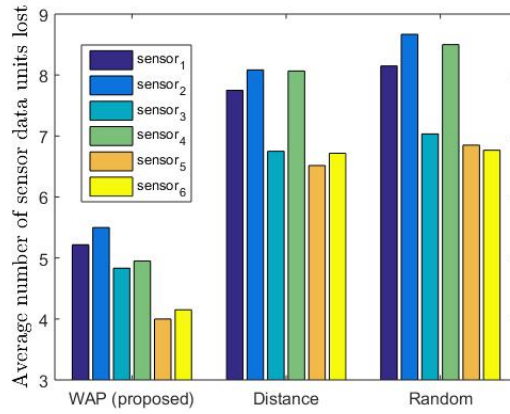**Fig. 8.** Average number of sensor data units received (V=100, H=100).

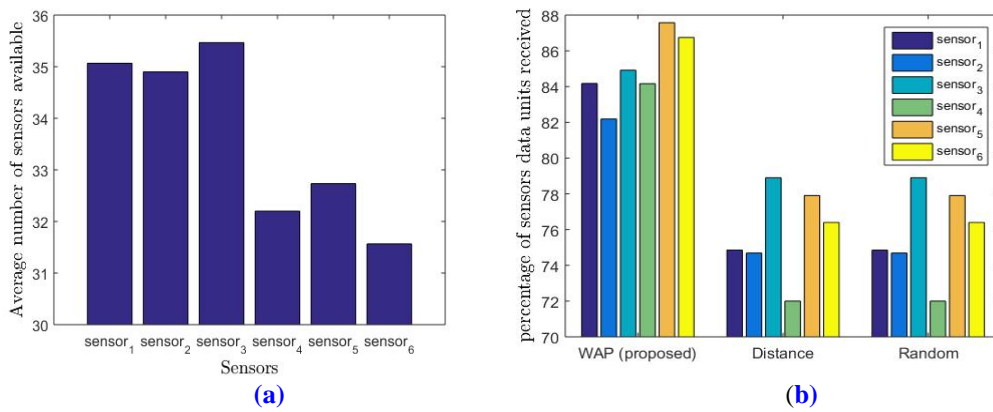**Fig. 9.** Average number of sensor data units lost (V=100, H=100).

(a)   **(b)**

Fig. 10. (a) Average number of sensors available (V=100, H=100). **(b)** Percentage of sensor data units received (V=100, H=100).

5.2 Impact of High Node Density

To analyze the impact of the density of mobile nodes, we increase H and W from 100 to 400, keeping the other parameters value same. **Fig. 11** shows a similar trend from Figure **Fig. 4** for the numbers of pedestrian and vehicle nodes in the communication range of the BS. there are more vehicle nodes than the pedestrian nodes. The number of iterations has decreased to 18 and the value of $T_{schedDur}(I)^{lim}$ has increased to 98 s as shown in **Fig. 12**. this is because there are now a larger number of mobile nodes available under BS range. The proposed approach still achieves a higher rate of successful transmissions and a lower rate of unsuccessful transmissions in comparison with other approaches under a higher density of mobile nodes as shown in **Fig. 13** and **Fig. 14**. The standard errors for the average number of successful/unsuccessful mobile nodes are negligible at 3.03, 3.28, and 3.87 for WAP, $Distance$, and $Random$ methods, respectively. **Fig. 17(a)** shows the number of availability of different types of sensors averaged from iterations and Figure **Fig. 17(b)** shows the percentage of respective sensors data received. From **Fig. 15** and **Fig. 16,** even under the high density of mobile nodes, a collection of sensor data units (with respect to any k-th type sensor) is not affected for proposed scheduling framework compared to others.
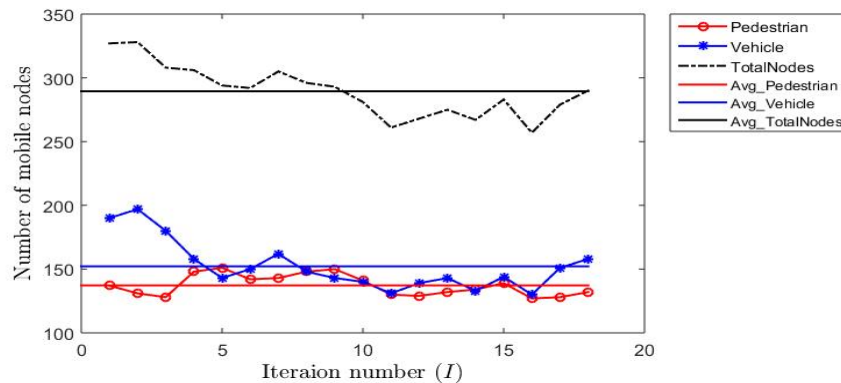

Fig. 11. Number of mobile nodes within communication range (High density: $V = 400$, $H = 400$)
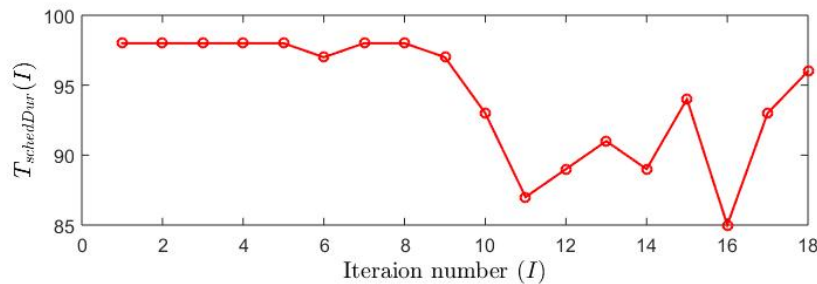

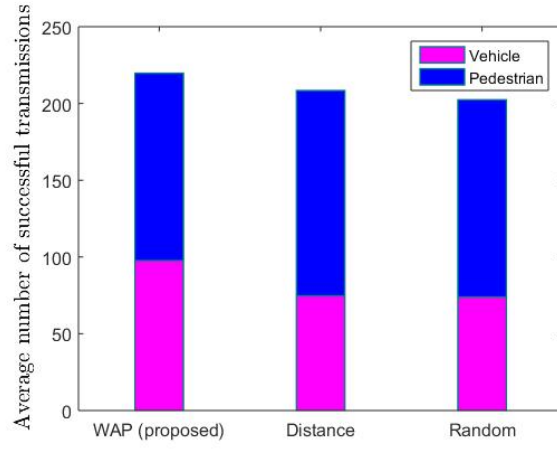**Fig. 12.** Schedule duration (High density: $V = 400$, $H = 400$)

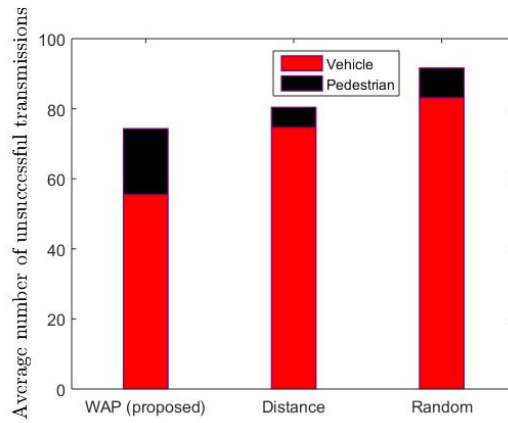**Fig. 13.** Average number of successful transmissions (High density: $V = 400$, $H = 400$)

Fig. 14. Average number of unsuccessful transmissions (High density: $V = 400$, $H = 400$)
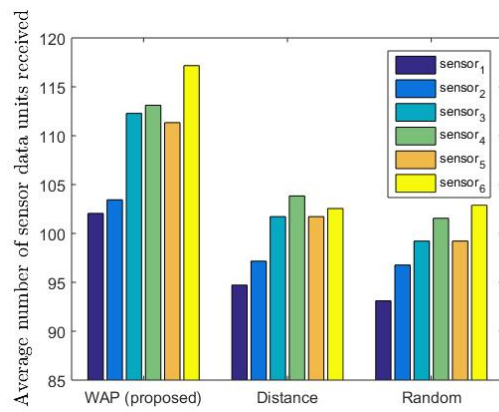

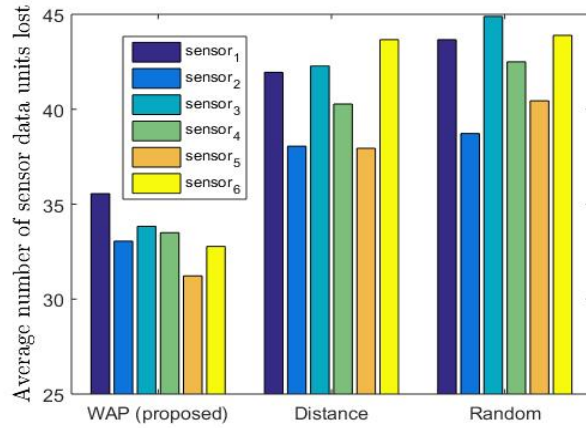**Fig. 15.** Average number of sensors data units received (High density: $V = 400$, $H = 400$)

Fig. 16. Average number of sensors data units lost (High density: $V = 400, H = 400$)
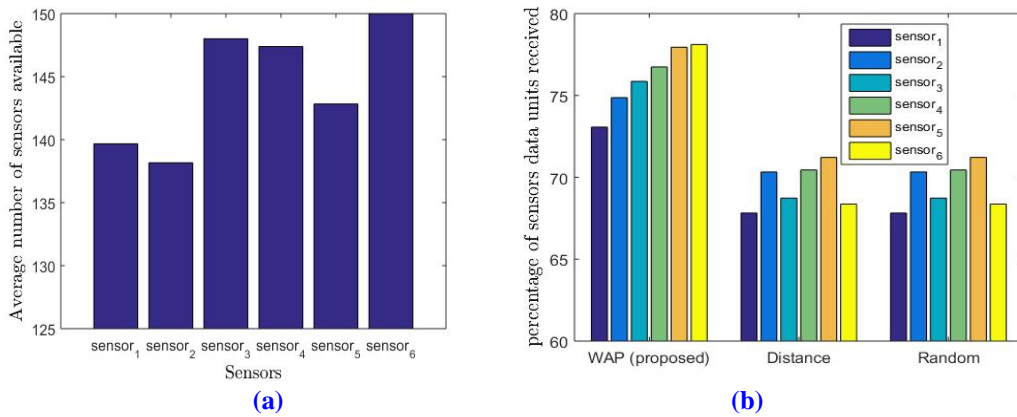

(a)   **(b)**

Fig. 17. (a) Average number of sensors available (High density: $V = 400, H = 400$), **(b)** Percentage of sensor data units received (High density: $V = 400, H = 400$)

5.3 Impact of Scheduling Duration Limit

We propose that the value of the scheduling duration $T_{schedDur}(I)$ has an upper limit ($T_{schedDur}(I)^{lim}$). Without the upper limit, the scheduling duration can increase to a very large value which can create large delays for the high priority data in the succeeding iteration. To verify our argument, we conduct simulations with and without an upper limit. This means that the lines 16 and 19-22 of Algorithm 1 are not executed. The results in **Fig. 18** shows a similar trend as before. However, as shown in **Fig. 19**, there are many $T_{schedDur}(I)$ values exceeding 98 seconds. Here, the total number of iterations is 18. **Fig. 20** and **Fig. 21** show the total number of successful and unsuccessful transmissions, respectively. In **Fig. 20,** the total number of successful transmissions is greater when $T_{schedDur}(I)^{lim}$ is not considered but unsuccessful transmissions are also greater as shown in **Fig. 21**. The impact of the schedule duration limit is more evident in **Fig. 21(a)** and **Fig. 21(b)**. Without $T_{schedDur}(I)^{lim}$, the loss rate of sensor data units increases, specifically for the sensor types with high priorities (i.e., $k = 6, 5, 4$), as shown in Figure **Fig. 21(b)**. This is because the proposed approach considering $T_{schedDur}(I)^{lim}$ reduces the number of long scheduling iterations by avoiding long delays that

cause loss of important sensor data. Therefore, $T_{schedDur}(I)^{lim}$ is essential for both avoiding unsuccessful transmissions and reducing the loss of important sensor data.
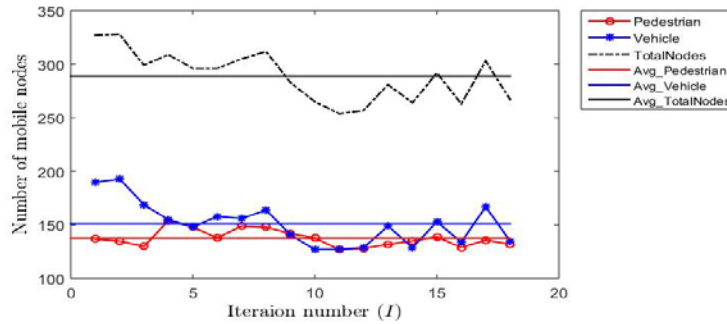

Fig. 18. Number of mobile nodes within communication range ($V = 400$, $H = 400$, no scheduling duration limit).
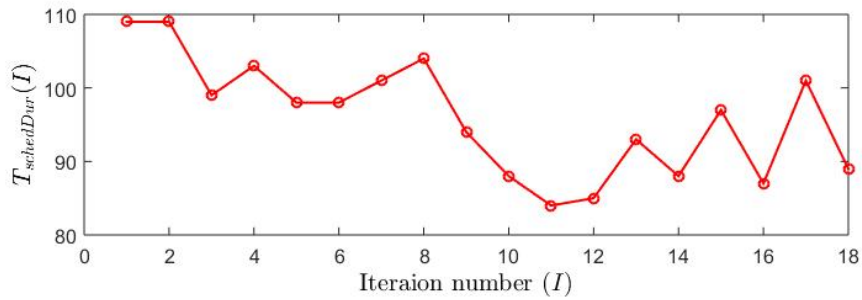

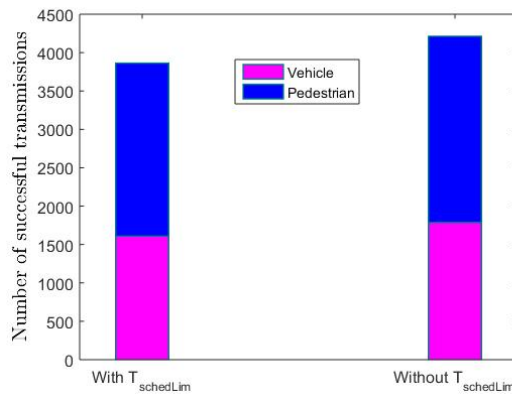**Fig. 19.** Schedule duration ($V = 400$, $H = 400$, no scheduling duration limit).

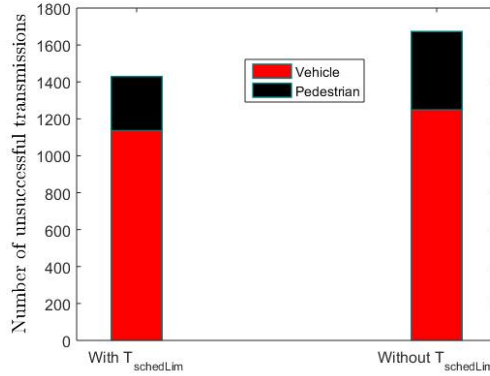**Fig. 20.** Number of successful transmissions ($V = 400$, $H = 400$, no scheduling duration limit).

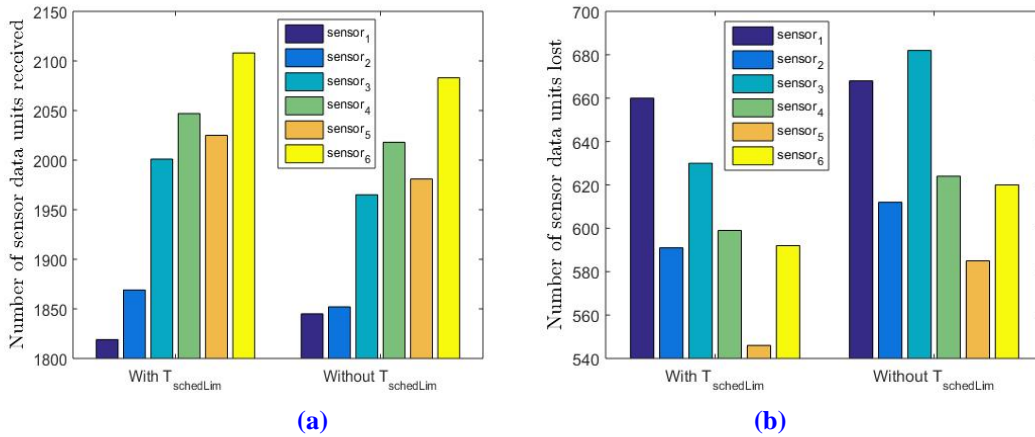**Fig. 21.** Number of unsuccessful transmissions ($V = 400$, $H = 400$, no scheduling duration limit).

(a)   **(b)**

Fig. 22. (a) Number of sensor data units received ($V = 400$, $H = 400$, no scheduling duration limit). **(b)** Number of sensor data units lost ($V = 400$, $H = 400$, no scheduling duration limit).

Table 1. Simulation Parameters

Parameter	Value
Simulation area (A)	$500\ m \times 500\ m$
Simulation duration ($simTime$)	$1900\ s$
Simulation start ($simStart$)	$5\ s$
Simulation end ($simEnd$)	$1800\ s$
Mobility Model Parameter	
Total number of pedestrian nodes (W)	100, 400
LW flight parameter α	1.0
LW pause parameter β	1
LW average flight length	5-500 m
LW pause time	10-60 s
LW location sample interval	$1\ s$
Min, Avg, Max pedestrian node velocity	0.50, 1.40, 3.78 m/s
Total number of vehicle nodes (H)	100, 400
Vehicle pause time	0-60 s
Vehicle speed minimum-maximum	2-10 m/s
Vehicle location sample interval	$1\ s$

Communication Network Parameter	
Number of base stations (BS)	1
Frequency (IEEE 802.11)	2.4 GHz
Transmission power (IEEE 802.11)	15 dBm
Reference distance (d_0)	1 m
Receiver sensitivity (Rx_{sen})	-68 dBm
Path loss exponent (η)	2
Data rate	100 $kbps$
Number of $nodes\_per\_second$	3
Size of $beaconInit$ packet	300 bytes
Size of $beaconReply$ packet	400 bytes
Length of $T_{beaconProc}$	5 s
Weight Parameter	
Number of sensor types (K)	6
Weights of sensor types (w_k^{sensor})	k (i.e, 1, 2, 3, 4, 5, 6)
Number of regions (R)	3
Weights of regions (w_r^{region})	r (i.e., 1, 2, R)
Number of velocity categories (V)	5
Weights of velocity ($w_v^{velocity}$)	v (i.e., 1, 2, V)
$w_1^{velocity} * \gamma \ (v_j(t_p) \le 1.0 \ m/s)$	9
$w_2^{velocity} * \gamma \ (1.0 \le v_j(t_p) \le 1.4 m/s)$	18
$w_3^{velocity} * \gamma \ (1.4 \le v_j(t_p) \le 3.0 \ m/s)$	27
$w_4^{velocity} * \gamma \ (3.0 \le v_j(t_p) \le 5.0 \ m/s)$	36
$w_5^{velocity} * \gamma \ (v_j(t_p) > 5.0 \ m/s)$	45

5.4 Network Complexity

In this paper, the proposed weighted adaptive opportunistic scheduling approach is explained in terms of network framework which involves transmission/broadcasting and receiving of packets between *BS* and *mobile node(s)*. Therefore, overall computation complexity (or network computation complexity) of the proposed scheduling approach is the addition of computation complexity at a *BS* and *mobile node(s)*. As given in Algorithm 1, each scheduling iteration runs for maximum $T_{schedDur}(I)^{lim}$ or less then $T_{schedDur}(I)^{lim}$. At *BS*, the algorithm cost of weight calculation for a mobile node is very low as it includes basic multiplications and addition tasks. (Equation (4), (5) and (6)). The time cost to compute the weight of mobile *node(s)* at any I^{th} scheduling iteration is *O(N)*, where *N* is the number of mobile nodes at any I^{th} iteration that sends $beaconReply$ packet to *BS*. At *mobile node(s)*, the main computation includes sensing and processing of $beaconReply$ and $sched$ packets.

6. Conclusion

It is very challenging to collect smartphone sensor data when user mobility is dynamic and heterogeneous. In this work, a weighted adaptive opportunistic scheduling framework is proposed for collecting sensor data from smartphones required by applications. The scheduling duration is adaptive and varies according to the number of mobile nodes available in the range of the *BS*. Also, the schedule duration has an upper limit to avoid long scheduling

delays that can lead to missing high-priority sensor data. The simulation results under a heterogeneous mobility scenario (the two main mobility modes of smartphone users are combined) show that the proposed scheduling framework, which comprehensively considers both mobility and application parameters, can achieve higher data collection performance than other frameworks. Since the simulated results demonstrate the effectiveness of the proposed framework, it will be extended in the future to implement in real-time. The real-time applicability of the proposed scheduling approach involves the implementation of two co-operative network protocols running on the base station and on smartphones, respectively. Also, the proposed scheduling framework will be extended to consider more realistic scenarios that include real mobility traces, geographical obstacles, interference effects, and consideration of energy consumption.

Acknowledgment

References

[1] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A survey on enabling technologies, protocols, and applications," *IEEE Communications Surveys Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015. Article (CrossRef Link)

[2] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Computer Networks*, vol. 54, no. 15, pp. 2787 – 2805, 2010. Article (CrossRef Link)

[3] G. Choudhary and A. K. Jain, "Internet of Things: A survey on architecture, technologies, protocols and challenges," in *Proc. of 2016 International Conference on Recent Advances and Innovations in Engineering (ICRAIE)*, pp. 1–8, Dec 2016. Article (CrossRef Link)

[4] S. Nalbandian, "A survey on Internet of Things: Applications and challenges," in *Proc. of 2015 International Congress on Technology, Communication and Knowledge (ICTCK)*, pp. 165–169, Nov 2015. Article (CrossRef Link)

[5] S. H. Shah and I. Yaqoob, "A survey: Internet of Things (IoT) technologies, applications and challenges," in *Proc. of 2016 IEEE Smart Energy Grid Engineering (SEGE)*, pp. 381–385, Aug 2016. Article (CrossRef Link)

[6] Q. Ou, Y. Zhen, X. Li, Y. Zhang, and L. Zeng, "Application of Internet of Things in smart grid power transmission," in *Proc. of 2012 Third FTRA International Conference on Mobile, Ubiquitous, and Intelligent Computing*, pp. 96–100, June 2012. Article (CrossRef Link)

[7] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of Things for smart cities," *IEEE Internet of Things Journal*, vol. 1, no. 1, pp. 22–32, Feb 2014. Article (CrossRef Link)

[8] S. M. R. Islam, D. Kwak, M. H. Kabir, M. Hossain, and K. S. Kwak, "The Internet of Things for health care: A comprehensive survey," *IEEE Access*, vol. 3, pp. 678–708, 2015. Article (CrossRef Link)

[9] T. M. Bojan, U. R. Kumar, and V. M. Bojan, "An Internet of Things based intelligent transportation system," in *Proc. of 2014 IEEE International Conference on Vehicular Electronics and Safety*, pp. 174–179, Dec 2014. Article (CrossRef Link)

[10] A. Corradi, G. Curatola, L. Foschini, R. Ianniello, and C. R. D. Rolt, "Smartphones as smart cities sensors: Mcs scheduling in the participant project," in *Proc. of 2015 IEEE Symposium on Computers and Communication (ISCC)*, pp. 222–228, July 2015. Article (CrossRef Link)

[11] W. Z. Khan, Y. Xiang, M. Y. Aalsalem, and Q. Arshad, "Mobile phone sensing systems: A survey," *IEEE Communications Surveys Tutorials*, vol. 15, no. 1, pp. 402–427, First 2013. Article (CrossRef Link)

[12] Z. Ma, Y. Qiao, B. Lee, and E. Fallon, "Experimental evaluation of mobile phone sensors," in *Proc. of 24th IET Irish Signals and Systems Conference (ISSC 2013)*, pp. 1–8, June 2013. Article (CrossRef Link)

[13] L. Fenghua, B. Yu, W. Dandan, W. Wei, and N. Li, "Smartphones for sensing," *Science Bulletin*, vol. 61, no. 3, pp. 190–201, 2016. Article (CrossRef Link)

[14] S. Aram, A. Troiano, and E. Pasero, "Environment sensing using smartphone," in *Proc. of 2012 IEEE Sensors Applications Symposium Proceedings*, pp. 1–4, Feb 2012. Article (CrossRef Link)

[15] J. Li, G. Yu, Y. Lu, C. Hsiung, A. Hannon, D. Kim, and S. Dennis, "Nanotechnology-based cell-all phone-sensors for extended network chemical sensing," in *Proc. of 2012 IEEE Sensors*, pp. 1–4, Oct 2012. Article (CrossRef Link)

[16] A. Anton and M. Statheropoulos, "Chemical sensors integrated with mobile phones for remote medical diagnostics: State-of-the-art and beyond," in *Proc. of 2014 4th International Conference on Wireless Mobile Communication and Healthcare - Transforming Healthcare Through Innovations in Mobile and Wireless Technologies (MOBIHEALTH)*, pp. 234–237, Nov 2014. Article (CrossRef Link)

[17] S. H. Ali and V. C. M. Leung, "Mobility assisted opportunistic scheduling for downlink transmissions in cellular data networks," in *Proc. of IEEE Wireless Communications and Networking Conference 2005*, vol. 2, pp. 1213–1218, March 2005. Article (CrossRef Link)

[18] T. Qiu, K. Zheng, M. Han, C. L. P. Chen, and M. Xu, "A data emergency-aware scheduling scheme for Internet of Things in smart cities," *IEEE Transactions on Industrial Informatics*, pp. 1–1, 2017. Article (CrossRef Link)

[19] A. Alkhelaiwi and D. Grigoras, "Scheduling crowdsensing data to smart city applications in the cloud," in *Proc. of 2016 IEEE 12th International Conference on Intelligent Computer Communication and Processing (ICCP)*, pp. 395–401, September 2016. Article (CrossRef Link)

[20] A. Capponi, C. Fiandrino, D. Kliazovich, and P. Bouvry, "Energy efficient data collection in opportunistic mobile crowdsensing architectures for smart cities," in *Proc. of 2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 307–312, May 2017. Article (CrossRef Link)

[21] A. Zhalgasbekova, A. Zaslavsky, S. Saguna, K. Mitra, and P. P. Jayaraman, "Opportunistic data collection for IoT-based indoor air quality monitoring," in *Proc. of Internet of Things, Smart Spaces, and Next Generation Networks and Systems, Cham: Springer International Publishing*, pp. 53–65, 2017. Article (CrossRef Link)

[22] S. V. George, L. Mathews, and S. S. Pillai, "Heterogeneous user mobility based scheduling scheme and transmission mode selection in downlink LTE/LTE-a systems," in *Proc. of 2015 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT)*, pp. 78–83, Dec 2015. Article (CrossRef Link)

[23] S. B. Eisenman, H. Lu, and A. T. Campbell, "Halo: Managing node rendezvous in opportunistic sensor networks," in *Proc. of Distributed Computing in Sensor Systems, Springer Berlin Heidelberg*, pp. 273–287, 2010. Article (CrossRef Link)

[24] M. B. Shah and J. L. Rathod, "Efficient scheduling algorithm for query processing in opportunistic sensor network under human mobility model," in *Proc. of 2014 5th International Conference - The Next Generation Information Technology Summit (Confluence)*, pp. 401–405, Sept 2014. Article (CrossRef Link)

[25] M. B. Shah, P. P. Verma, S. N. Merchant, and U. B. Desai, "Human mobility based stable clustering for data aggregation in single-hop cell phone based wireless sensor network," in *Proc. of 2011 IEEE International Conference on Advanced Information Networking and Applications*, pp. 427–434, March 2011. Article (CrossRef Link)

[26] M. B. Shah, P. S. Tamhankar, S. N. Merchant, and U. B. Desai, "A realistic weighted clustering algorithm for data gathering in single hop cell phone-based sensor network," in *Proc. of 2011 IEEE GLOBECOM Workshops*, pp. 1253–1257, Dec 2011. Article (CrossRef Link)

[27] Thejaswini M, P. Rajalakshmi, and U. B. Desai, "Duration of stay based weighted scheduling framework for mobile phone sensor data collection in opportunistic crowd sensing," *Peer-to-Peer Networking and Applications*, vol. 9, no. 4, pp. 721–730, Jul 2016. Article (CrossRef Link)

[28] S. E. Odongo, J. Jhihoon, and H. Dong, "A broadcast scheme for vehicle-to-pedestrian safety message dissemination," *International Journal of Distributed Sensor Networks*, vol. 13, no. 11, pp. 1-19, 2017. Article (CrossRef Link)

[29] G. M. Viswanathan, S. V. Buldyrev, S. Havlin, M. G. E. da Luzk, E. P. Raposok, and H. E. Stanley, "Optimizing the success of random searches," *NATURE*, vol. 401, pp. 911–914, October 1999. Article (CrossRef Link)

[30] I. Rhee, M. Shin, S. Hong, K. Lee, S. J. Kim, and S. Chong, "On the levy-walk nature of human mobility," *IEEE/ACM Transactions on Networking*, vol. 19, no. 3, pp. 630–643, June 2011. Article (CrossRef Link)

[31] S. M. Mousavi, H. R. Rabiee, M. Moshref, and A. Dabirmoghaddam, "Mobisim: A framework for simulation of mobility models in mobile ad-hoc networks," in *Proc. of Third IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob 2007)*, pp. 82–82, Oct 2007. Article (CrossRef Link)

[32] Mobisim. [Online]. Available: http://masoudmoshref.com/old/myworks/documentpages/mobility_simulator.htm

[33] B. Birand, M. Zafer, G. Zussman, and K. W. Lee, "Dynamic graph properties of mobile networks under levy walk mobility," in *Proc. of 2011 IEEE Eighth International Conference on Mobile Ad-Hoc and Sensor Systems*, pp. 292–301, October 2011. Article (CrossRef Link)

[34] W. H. Tranter, K. S. Shanmugan, T. S. Rappaport, and K. L. Kosbar, *Principles of Communication Systems Simulation with Wireless Applications*, Prentice-Hall Professional Technical Reference, 2004. Article (CrossRef Link)

[35] W. C. Y. Lee, *Wireless and Cellular Telecommunication*, McGraw-Hill, 2006. Article (CrossRef Link)

[36] Wikipedia, Preferred walking speed. [Online]. Available: https://en.wikipedia.org/wiki/Preferred_walking_speed

Bong Jun Choi is an associate professor at the School of Computer Science & Engineering and jointly at the School of Electronic Engineering, Soongsil University, Seoul, Korea. Previously, he was an assistant professor at the Department of Computer Science, State University of New York Korea, Korea, and concurrently a research assistant professor at the Department of Computer Science, Stony Brook University, USA. He received his B.Sc. and M.Sc. degrees from Yonsei University, Korea, both in Electrical and Electronics Engineering, and the Ph.D. degree from University of Waterloo, Canada, in Electrical and Computer Engineering. His current research focuses on energy-efficient networks, distributed mobile wireless networks, smart grid communications, and network security. He is a member of the IEEE and the ACM.

Thejaswini M received the Ph.D. degree in computer science and engineering at the Indian Institute of Technology Hyderabad, Hyderabad, India in 2016. She is currently an assistant professor at the Indian Institute of Information Technology, Bhagalpur, India. Her research interests include mobile communications, mobile sensor networks, cloud computing, internet of things and mobile applications for smart cities.