A Data Collection Method Based on the Region Division in Opportunistic Networks

Yaqing Ma [1], Shukui Zhang [1,2], Chengkuan Lin [1], and Lingzhi Li [1]

[1] School of Computer Science and Technology
Soochow University, Suzhou, Jiangsu 215006, China
yqma@stu.suda.edu.cn, zhangsk2000@163.com

[2] Jiangsu High Technology Research Key Laboratory for Wireless Sensor Networks
Nanjing, Jiangsu 210003, China

Abstract ─ The popularity of wearable devices and smart phones provide a great convenience for large-scale data collection. Owing to the non-uniform distribution of mobile sensors, the data quantity collected from different regions has a wide variation. So we design the region division algorithm that divides area into different density grades and sets appropriate sampling frequency on different regions. Furthermore, we propose Circle of Time Slice (CoTS) and Cardinal Number Timing Method (CNTM) to solve the sampling error when nodes move from one area to another. On this basis, we propose the Data Collection Algorithm Based on the Sampling Frequency (DC-BSF) to reduce the data redundancy. Simulations demonstrate that the method proposed in this paper can reduce data redundancy under the condition of achieving high coverage.

Index Terms ─ Data collection, region division, sampling frequency, time slice cycle.

I. INTRODUCTION

With the rapid development of mobile communication and sensing technology, a number of innovative applications and services have emerged. In particular, the popularity of portable devices (e.g., smart phones, wearable devices, etc.) and vehicle sensors (e.g., GPS), provide efficient ways to sense physical objects and environmental conditions on a large scale. It greatly expanded the dimensions of human perception and changed the way people perceive the world [1].

Mobile Crowd Sensing (MCS), where individuals with sensing and computing devices collectively share data and extract information to measure phenomena of common interest [2]. Sensor nodes collect data adaptively in the mobile process. It forms a Mobile Opportunistic Networks (MONs) when data is transferred between sensor nodes [3]. The data transmission depends on the cooperation of nodes to fulfill a "store – carry – process – forward" mode.

Sensors are embedded in a taxi and collect data periodically. Since different taxis always have heterogeneous mobility regions with some randomness, they could make different contributions to the coverage. The mobile trajectory is more intensive in hot regions, like shopping malls, stations and so on. So there are a large number of sensors in these regions. If they collect data at the same sampling frequency as sensors in sparse region, it may lead to serious data redundancy. On the contrary, the mobile trajectory is relatively sparse in remote areas. The data coverage is low in the sparse area. Sensors in these regions need to be set a higher sampling frequency. So we divide the region according to the density of taxi trajectory. Then we set the sampling frequency of sensors in regions of different density. At last, we propose an effective sensing mechanism that can reduce data redundancy in the premise of high coverage. The main contributions of this paper are shown as follows.

1) We design the Region Division algorithm (D-RG) to divide the entire area into regions of different grades, according to density of nodes' trajectory.
2) We propose the Circle of Time Slice and Cardinal Number Timing Method to solve the sampling error resulted from nodes moving from one region to another.
3) We propose the Data Collection Algorithm Based on the Sampling Frequency that can reduce data redundancy in the premise of high coverage.

The remainder of this paper is organized as follows. Section 2 reviews the related work. Section 3 describes the system model and two algorithms. In Section 4, we evaluate the performance of our model by real mobile traces. Finally, we conclude the paper in Section 5.

II. RELATED WORK

Recently the concept of mobile crowd sensing has

Submitted On: September 27, 2016
Accepted On: December 4, 2016

attracted the attention of many researchers. Challenges in this research include localized analytics, resource limitation, privacy, security, data integrity, architecture and so on [1]. The 4W1H characterize the major research issues in the MCS life cycle [4]. The noise-mapping system can intuitively present the urban noise level by encoding levels with colors [5]. COUPON is a cooperative framework for building sensing maps in mobile opportunistic network [6]. The CarTel is a distributed mobile sensor computing system [7].

As more and more sensors are integrated on mobile devices, the application field of mobile crowd sensing is constantly expanding [8]. In the paper [9], smart parking was used as a case study to investigate features of crowdsourcing that may apply to other mobile applications. Participatory sensing can be applied in environmental monitoring [10]. The Intelligent Traffic System (ITS) is based on mobile crowd sensing [11]. Smart P2P model can optimize the search process [12].

Due to the different scenarios and purposes, the type of data needed is different. So many scholars have studied the methods of data collection. ITAMP is a new sparsely adaptive algorithm with high recovery rate and fast reconstructing speed [13]. In paper [14], the data collection method of wireless sensor networks in gateway was proposed. Another data acquisition method was proposed in paper [15], heterogeneous multi-source multi-mode sensory based on data quality.

The data sensing mechanism is the key to obtaining valid data in the process of data collection. Researchers have proposed some sensing mechanisms. Researchers designed two cooperative schemes to optimize the system performances in terms of sensing quality, delivery delay and energy consumption [16]. Researchers proposed a city hot spot event sensing method based on mobile crowd sensing. This method can discover and classify hot spots [2]. Previous studies have considered the characteristics of nodes mobility and the spatial-temporal correlation among sensory data. But they ignored the moving speed of sensors, residence time in each area and the sampling error resulted from nodes moving from one region to another. In view of the above problems, we propose a new data collection algorithm. It can reduce data redundancy under the condition of achieving high coverage.

III. PERCEPTION PROCESS

In this part, we divide the entire sensing area into different grades according to the data of pre-sampling firstly. Furthermore, we propose the Circle of Time Slice and Cardinal Number Timing Method to solve the sampling error resulting from nodes moving from one region to another. Finally, we propose the Data Collection Algorithm Based on the Sampling Frequency (DC-BSF).

A. Division of sensing area

Although the movement trajectory possesses some randomness, it has characteristics of dense or sparse. The data of taxi movement trajectory is provided by Korea Advanced Institute of Science and Technology (KAIST) [17]. The graph of movement trajectory is shown in Fig. 1.

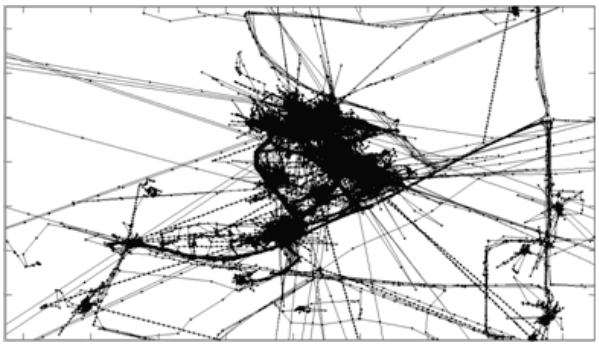

Fig. 1. The graph of taxi movement trajectory.

In order to further analyze the characteristics of the trajectory of sensors, we divide the entire sensing area into the same size grid cells $G=\{g_1,g_2\cdots,g_m\}$, and sensors $(s_1,s_2\cdots,s_n)$ are embedded in mobile vehicles [7], as illustrated in Fig. 2.

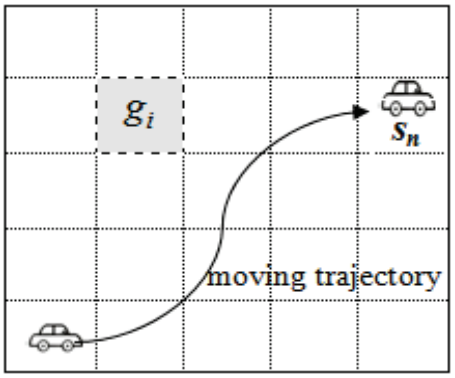

Fig. 2. The sensing area with same size grid cell.

The cell-grid (g_i) shown in Fig. 2 is the smallest unit of region division. In order to divide the sensing area according to the density of the vehicle trajectory, we need to calculate the density of trajectory firstly. The statistical method is shown in Algorithm 1. It counts the number of trajectories in each grid cell according to the data of pre-sampling.

Statistical results are stored in a table like Table 1. The a_i in the *i*-th row represents the number of trajectory in a grid cell. The u_i in *i*-th row represents the total number of grids that include a_i trajectories.

Algorithm 1: Statistical method of sampling times

```
Input:
    vecData;  // Pre sampled data
    tabsize; // Grid size
    Xnum; // the number of grid on the X axis
    Ynum; // the number of grid on the Y axis
Output:
    vecTabData; // Sampling times
    mapSumData; //the number of grid sampled n times
Initialize vecTabData is NULL;
for i ← 0 to vecData.size do
    index_x ← floor(vecData[i].x / tabsize) + 1;
    index_y ← floor(vecData[i].y / tabsize) + 1;
    // floor() is a Integral Formula.
    vecTabData[index_x][index_y]
    ←vecTabData[index_x][index_y] + 1;
repeat
for i ← 0 to vecTabData.size do
  for j ← 0 to vecTabData[i].size do
    if (mapSumData.find( vecTabData[i][j] ))
      then  mapSumData(vecTabData[i][j])
            ←mapSumData(vecTabData[i][j]) + 1;
    else
      mapSumData(vecTabData[i][j]) ← 1;
    endif
  repeat
repeat
```

Table 1: Statistical table

Number of tracks (a_i)	Number of grids (u_i)
$a_0 = 0$	u_0
$a_1 = 1$	u_1
⋮	⋮
$a_n = n$	u_n

We set a information table stored at each node and the table is dynamic. The information table includes the current time, data (collected and transmitted), sampling frequency ($f(l_k)$) and current location, as illustrated in Table 2. Current location is stored in a two-dimensional array. tab[i][j] represents that the location of a cell-grid is the i-th row and j-th column of the area. *level_k* represents that the grade of the grid is *k*. The formula *level_k* = f (l_k) in Table 1 means that the sampling frequency of sensors located in *level_k* is f (l_k).

Table 2: Storage table in sensors

Current Location	Frequency	Data	
$tab[i][j] = level\_k$	$level\_k = f(l_k)$	*Collect*	*Transmit*

The traditional K-means algorithm [18] is a typical distance-based clustering algorithm. Distance between the surrounding sensors and the center sensor was used as the evaluation index of similarity. Therefore, we propose a method based on the number of trajectory in each cell-grid to divide sensing area. We divide the entire area into three levels according to the data of Table 1. The steps are shown bellow, and the algorithm is shown in Algorithm 2;

$$sum = \frac{1}{3} \times \sum_{i=0}^{n} (a_i \times u_i), \tag{1}$$

where *sum* represents one-third of the total number of trajectory in all grids. There is a constant p that satisfies the Equation (2):

$$\begin{cases} a_0 + a_1 + \cdots + a_p \le sum \\ a_0 + a_1 + \cdots + a_p + a_{p+1} > sum \end{cases}, \ p \in [0, n). \tag{2}$$

We divide the grids that satisfy the above conditions into the region of low sampling frequency (*level_1*). As shown in Fig. 3, the different textures represent different region grades. Similarly, there is a constant q that satisfies the Equation (3):

$$\begin{cases} a_{p+1} + a_{p+2} + \cdots + a_q \le sum \\ a_{p+1} + a_{p+2} + \cdots + a_{q+1} > sum \end{cases}, \ q \in [p+1, n). \tag{3}$$

We divide the grids satisfying the above conditions into the region of intermediate sampling frequency (*level_2*), as shown in Fig. 3. The rest of the grids are divided into the region of high sampling frequency (*level_3*).

Algorithm 2: D-RG algorithm

```
Input: a_i , u_i
Output: a_p , a_q
Initialize s ← 0;
for i ← 0 to n do
    s ← s + a_i × u_i ;
    //Determine the boundary value of the level region.
    if( s >= sum ) then
        Get a boundary value: a_i ;
        s ← 0; // get the next boundary value.
    end if
repeat
```

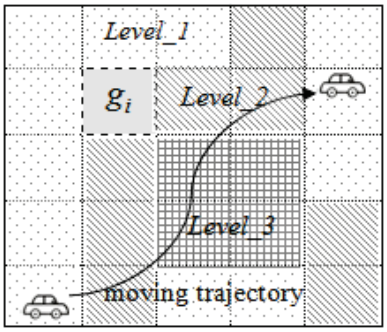

Fig. 3. The graph of region grades represented by different textures.

B. The algorithm of data collection

The sensor performs the sampling task, when the sampling time is reached and the sensor does not contain the data of the current grid. Due to the limited transmission range of sensors, the transmission delay is relatively serious. So we set different sampling frequency for sensors in different regions. Sensors can collect data according to their own sampling frequency, rather than entirely relying on collaboration with other sensors. Therefore, it can reduce the interdependence between sensors.

In the pre-sampling process, sampling period is T_1 (30 seconds). Considering the number of grids that are over-sampled and the region grade, we set the different sampling frequency for sensors in different grade regions. The time interval of sampling in *level_k* is shown bellow:

$$T_{l_k} = 2^k \times T_1 \times \frac{\sum(a_i \times u_i)}{\sum u_i},$$
$$i \in [0, p) \ or \ [p+1, q) \ or \ i \in [q+1, n], \ k = 1, 2, 3. \quad (4)$$

So the sampling frequency in *level_k* is shown below:

$$f(l_k) = \lambda \times \frac{1}{T_{l_k}}, \ (k = 1, 2, 3; \lambda \ is \ a \ constant). \quad (5)$$

In the sensing process, sensors may move from one region to another. However, the sampling frequency of sensors in these regions is different. If sensors stay in one region for a short time, it may miss some data. From the trajectory of the sensor in Fig. 4, we can see that the sensor moves to the *level_2* region and then leaves here. Under the condition of the same speed, the sensor stays in the *level_2* region for a short time. If the dwell time is much shorter than the sampling interval of the sensor, the data in that region may be missed.

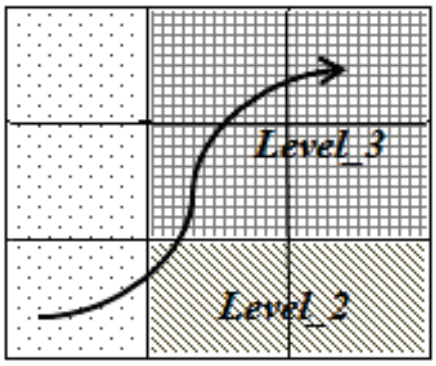

Fig. 4. Moving trajectory in different grade regions.

So we propose the Circle of Time Slice (CoTS) and Cardinal Number Timing Method (CNTM) to solve the sampling error resulted from sensors moving from one area to another. The Time Slice aims to divide the time axis into many short time periods. The sampling interval is much longer than the time slice. CoTS means that sensors check information at the beginning of each time slice. The sensor requires checking its current location, whether it contains the data of current grid and whether it has reached the sampling time. CNTM is a method of calculating time length, as shown in Equation (6):

$$cur\_time = cur\_time - \frac{const\_init\_time}{time\_k}, \quad (6)$$

where *cur_time* indicates how long it takes to reach the sampling instants. The *time_k* represents the sampling interval in *level_k* region. The *const_init_time* is the lowest common multiple of *time_k* (k = 1, 2, 3). After a time slice, the *cur_time* is shortened by *const_init_time*/ *time_k*. This also ensures that the longer the sampling interval, the more times that sensors check information during the sampling interval. The data collection algorithm based on the sampling frequency is shown as follows.

Algorithm 3: DC-BSF

Input: *time_k*, *const_init_time*, *transimit_length*
Output: the data required;
Initialize *cur_time* ← *const_init_time*;
while(during the sampling time **) do**
  if(current location is included in level_k **) then**
    cur_time = *cur_time*−*const_init_time* / *time_k*;
    // the circle of time slice
  end if
  for i ← 0 **to** n **do**
    Traverse all nodes;
    if(*dis* <= *transimit_length* **) then**
      // *dis* is the distance between two nodes.
      Process and forward the collected data;
    end if
  repeat
  if(*cur_time* <= 0 &&
    Data of current grid is not included in the node.**)**
  then Perform sampling task;
  end if
repeat

C. Analysis of algorithm

In algorithm DC-BSF, T, $x * y$, n and m respectively represent the sampling period, the number of grids in the entire area, sensors and grids that have been sampled. The algorithm is mainly completed in a while circulation and the total number of circulation is t. The following three steps are performed in each cycle.

1) Sensors check the grade of the region that the sensor is located and the sampling interval in this grade. So the time complexity in this part is

$\Theta\left((\log_2 x)\times\log_2 y\right)$.

2) Sensor nodes transmit data between each other. The time complexity in this part is $\Theta\left(n\times x\times y\times(\log_2 x)\times\log_2 y\right)$.

3) Sensor nodes check whether it contains the data of current grid. The time complexity in this part is $\Theta\left((\log_2 x)\times\log_2 y\right)$.

So the time complexity of algorithm DC-BSF is $\Theta\left(t\times n\times x\times y\times(\log_2 x)\times\log_2 y\right)$.

IV. PERFORMANCE EVALUATION

In order to verify the feasibility and superiority of DC-BSF, we carry out the following evaluation based on the real mobility traces provided by Korea Advanced Institute of Science and Technology [17]. The simulation platform we used is Opportunistic Network Environment (ONE) [19]. ONE is a free simulation platform developed in Java language. It can be used to simulate Opportunistic Network and Delay Tolerant Network. The simulation parameter is shown in Table 3.

Table 3: Simulation parameter

Parameter	Value
Time	35ks (about 10 hours)
Area	30km×30km
The number of nodes	92
Communication range	100m
Size of level_k	500m
λ	0.5; 0.8; 1; 1.5; 2; 5
Size of grid	100; 200… ; 900; 1000m

We evaluate the feasibility of DC-BSF algorithm in terms of coverage and data redundancy as follows.

In Fig. 5, we compare the coverage with various values of λ by ten sizes of grid from 100 to 1000 m. The legend in this figure represents different grid size. We can obviously find that the coverage rate increases with the increase of λ regardless of the size of grid. When the coefficient of sampling frequency (λ) increases from 0.2 to 0.8 the coverage rate increase greatly, but the coverage level is low. When λ > 0.8, the coverage rate reaches a high level and keeps stable. So the optimum range of λ is from 0.8 to infinity considering the coverage rate only.

Data redundancy is an inevitable problem under the condition of high coverage rate. In order to reach the best compromise between coverage rate and data redundancy, we compare the data redundancy with various values of λ by ten sizes of grid from 100 to 1000 m. And the simulation parameter is shown in Table 3. From Fig. 6, we can see that the DC-BSF can achieve a lower data redundancy and the data redundancy increases slowly when λ is less than 2. On the contrary, the data redundancy is large and increases rapidly when λ > 2. So the optimum range of λ is from negative infinity to 2 considering the data redundancy only.

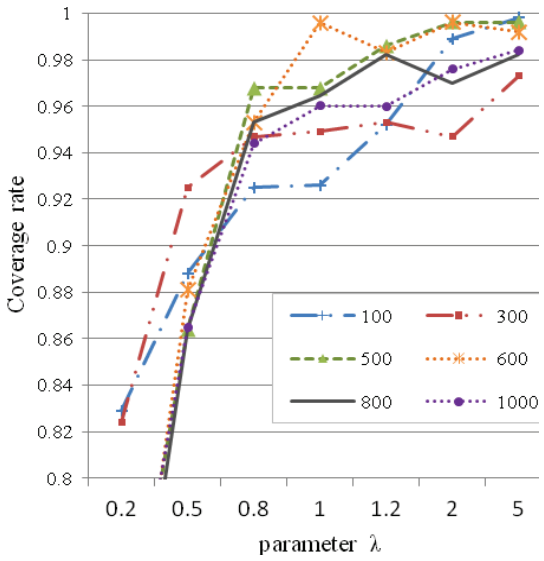

Fig. 5. Coverage rate with various values of λ.

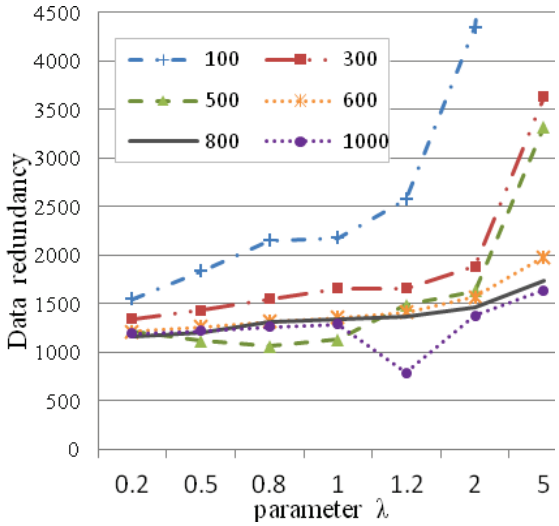

Fig. 6. Data redundancy with various values of λ.

Therefore, considering the best compromise between coverage rate and data redundancy, we take the intersection of the two parts. From the two sets of simulation above, we can get a conclusion that the optimum range of λ is from 0.8 to 2. It means that the data collection method based on the sampling frequency can reduce data redundancy in the condition of achieving

high coverage when the range of λ is from 0.8 to 2.

The simulation results above verified the feasibility of the method we proposed. In order to verify the efficiency of DC-BSF proposed in this paper, we compare it with Cooperative Sensing (CS) [6] in terms of coverage and redundancy. The Cooperative Sensing method is based on spatial-temporal correlation among sensory data.

We have $\lambda = 1$ in our DC-BSF and set the parameter k=1 in CS. Other parameters in the simulation are the same as in Table 2. As shown in Fig. 7, we compare the DC-BSF with CS about the coverage rate by using ten kinds of grid size. These two methods both get a high coverage rate and the gap between them is small. And then from Fig. 8, we can see that compared to the data redundancy of CS, it is significantly less of DC-BSF. So the method we proposed in this paper outperforms the CS relatively, although the DC-BSF does not achieve its best performance.

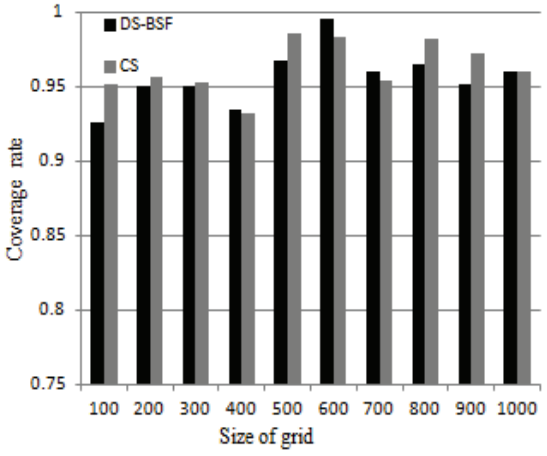

Fig. 7. Coverage rate of two kinds of sampling mechanisms in different grids.

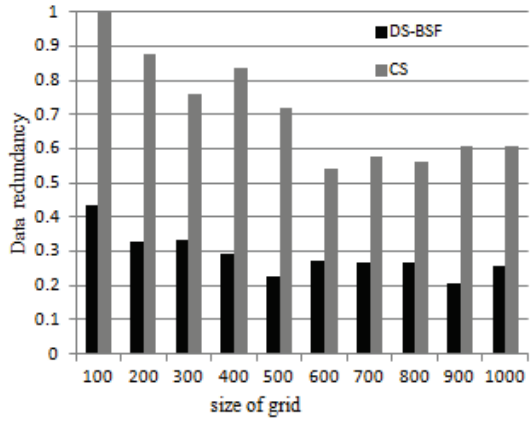

Fig. 8. Data redundancy of two kinds of sampling mechanisms in different grids.

V. CONCLUSION AND FUTURE WORK

In this paper, we propose the Division of Region Grade algorithm to divide the entire area into three regions of different grades. Furthermore, we put forward two concepts: CoTS and CNTM. These two methods can solve the sampling error resulting from nodes moving from one area to another. Last but not least, we proposed the Data Collection Algorithm Based on the Sampling Frequency that can reduce data redundancy in the premise of high coverage.

The model proposed in this paper has strict constraints. In the future, we plan to further study and improve it.

ACKNOWLEDGMENT

This work is supported by National Natural Science Foundation of China under Grants No. 61070169, 61201212, 61572340 and Natural Science Foundation of Jiangsu Province under Grant No. BK2011376 and Production-Teaching-Research Prospective of Jiangsu Province No. BY2012114 and Suzhou Key Laboratory of Converged Communication No SKLCC2013XX, SZS0805 and Application Foundation Research of Suzhou of China No. SYG201239 and "Six Talent Peak" high-level personnel selection and training foundation of Jiangsu Province under Grant No. 2014-WLW-010.

REFERENCES

[1] R. K. Ganti, F. Ye, and H. Lei, "Mobile crowd sensing: Current state and future challenges," *IEEE Communications Magazine*, vol. 49, no. 2, pp. 32-39, 2011.

[2] B. Guo, Z. Yu, X. Zhou, et al., "From participatory sensing to mobile crowd sensing," *IEEE International Conference*, pp. 593-598, 2014.

[3] P. Luciana, P. Andrea, and C. Marco, "Opportunistic networking: Data forwarding in disconnected mobile ad hoc networks," *IEEE Communications Magazine*, vol. 44, no. 11, pp. 134-141, 2006.

[4] D. Zhang, L. Wang, H. Xiong, et al., "4W1H in mobile crowd sensing," *IEEE Communications Magazine*, vol. 52, no. 8, pp. 42-48, 2014.

[5] W. Wu, B. Guo, and Z. Yu, "Crowd sensing based urban noise map and temporal-spatial future analysis," *Journal of Computer-Aided Design & Computer Graphics*, vol. 26, no. 4, pp. 638-643, 2014.

[6] D. Zhao, H. Ma, S. Tang, et al., "COUPON: A cooperative framework for building sensing maps in mobile opportunistic networks," *IEEE Transactions on*, vol. 26, no. 2, pp. 392-402, 2014.

[7] B. Hull, V. Bychkovsky, Y. Zhang, et al., "CarTel: A distributed mobile sensor computing system," *ACM*, pp. 125-138, 2006.

[8] H. Huang, Q. Ding, and L. Li, "Research on mobile terminal crowdsourcing," *Computer and*

Development, vol. 24, no. 6, pp. 6-9, 2014.
[9] X. Chen, N. E. Santos, and M. Ripeanu, "Crowdsourcing for on street smart parking," *New York, NY, USA: ACM*, pp. 1-8, 2012.
[10] V. Kotovirta, T. Toivanen, R. Tergujeff, et al., "Participatory sensing in environmental monitoring experiences," *Proc of 2012 Sixth International Conferences on IMIS*, Palermo: [s. n.], pp. 155-162, 2012.
[11] K. Ali, D. Al Yaseen, A. Ejaz, et al., "CrowdITS: Crowdsourcing in intelligent transportation systems," *IEEE Conferences on WCNC*, pp. 3307-3311, 2012.
[12] G. Chatzimilioudis, A. Konstantinidis, C. Laoudias, et al., "Crowd sourcing with smart phones," *IEEE Internet Computing*, vol. 16, no. 5, pp. 36-44, 2012.
[13] L. Lv, "Research on data acquisition and reconstruction algorithm of Internet of things sensor based on compressed sensing theory," *Nankai University*, pp. 1-95, 2011.
[14] L. Yao, Z. Zhao, N. An, and W. Wen, "Data acquisition and processing of wireless sensor networks in gateway," *Chinese Journal of Scientific Instrument*, pp. 1577-1578, 2008.
[15] Q. Ma, Y. Gu, T. Zhang, and G. Yu, "A heterogeneous multi-source multi-mode sensory data acquisition method based on data quality," *Chinese Journal of Computers*, vol. 36, no. 10, pp. 2120-2131, 2013.
[16] D. Zhao, "Research on data collection and incentive mechanisms in mobile crowd sensing network," *Beijing University of Posts and Tele-communications*, 2014.
[17] H. Xia, J. Chen, M. Marathe, et al., "Synthesis and refinement of detailed subnetworks in a social contact network for epidemic simulations," *4th International Conference on Social Computing, Behavioral-cultural Modeling and Prediction*, pp. 366-373, 2011.
[18] J. Liu, *Introduction to Social Network Analysis [M]*. Social Science Academic Press, 2004.
[19] J. Sun, *Opportunistic Network Routing Algorithm*. Post & Telecom Press, 2013.

Yaqing Ma is a student in the school of Computer Science and Technology, Soochow University, Suzhou, China. Her research interests include Wireless Sensor Networks, mobile computing and Mobile crowd sensing.

Shukui Zhang received his Ph.D. degree in Computer Science from the University of Science and Technology of China. Currently, he is a Professor and Doctoral Supervisor in the School of Computer Science and Technology at the Soochow University. His main research interest is ad hoc and wireless sensor networks, mobile computing, distributing computing, intelligent information processing, parallel and distributed systems etc. School of Computer Science and Technology, Soochow University, Suzhou, 215006, P.R. China.

Cheng-Kuan Lin received his B.S. degrees in Science Applied Mathematics from the Chinese Culture University in 2000; and received his M.S. degrees in Mathematic from the National Central University in 2002. He obtained his Ph.D. in Computer Science from the National Chiao Tung University in 2011.

His research interests include graph theory, design and analysis of algorithms, discrete mathematics, wireless sensor networks, mobile computing, wireless communication, wireless applications, and parallel and distributed computing.

Lingzhi Li received the Ph.D. degree in Computer Application from Nanjing University of Aeronautics and Astronautics in 2006. He is working as an Associate Professor in School of Computer Science & Technology, Soochow University, Suzhou, China. His research interests include network coding, vehicular network and wireless networks.