# Dynamic Resource Discovery based on Preference and Movement Pattern Similarity for Large-Scale Social Internet-of-Things

Zhiyuan Li, Rulong Chen, Lu Liu, *Member, IEEE* and Geyong Min*, Member, IEEE*

***Abstract*—Given the wide range deployment of disconnected delay-tolerant social Internet-of-Things (SIoT), efficient resource discovery remains a fundamental challenge for large-scale SIoT. The existing search mechanisms over the SIoT do not consider preference similarity and are designed in Cartesian coordinates without sufficient consideration of real-world network deployment environments. In this paper, we propose a novel resource discovery mechanism in a 3-dimensional Cartesian coordinate system with the aim of enhancing the search efficiency over the SIoT. Our scheme is based on both of preference and movement pattern similarity to achieve higher search efficiency and to reduce the system overheads of SIoT. Simulation experiments have been conducted to evaluate this new scheme in a large-scale SIoT environment. The simulation results show that our proposed scheme outperforms the state-of-the-art resource discovery schemes in terms of search efficiency and average delay.**

***Index Terms*—social Internet of things, resource discovery, cosine similarity, preference, movement pattern.**

## I. Introduction

THE integration of social networking concepts into the Internet-of-Things (IoT) has led to a burgeoning topic of research, so called Social Internet of Things (SIoT) paradigm, according to which the smart objects are capable of establishing social relationships in an autonomous way with respect to their owners. The benefits are those of improving scalability in resource/service discovery when the SIoT is made of a large number of heterogeneous nodes, similarly to what happens with social networks among humans [1]. SIoT is turning out to be a successful paradigm for peer-to-peer communications, being driven by the following properties:

- Smart objects belonging to the same community due to their owners often have common interests.
- Smart objects carried by their owners with common interests usually exhibit similarity in their movements and behavior patterns.
- Smart objects carried by their owners with common interests tend to meet each other frequently.

However, leveraging the above mentioned social properties for designing an efficient resource discovery mechanism over SIoT is an acute issue in the context of resource sharing in SIoT.

The existing resource discovery mechanisms in SIoT can be classified into three categories: social connectivity-based methods, movement pattern-based methods and preference similarity-based methods. The social connectivity-based methods [2-9] utilize the long-term social ties existing between the nodes for building communities, in such a way that those having frequent encounters in the past will belong to the same community. Once the communities are built, resource sharing is achieved among the nodes within the communities by social-connectivity. The movement pattern-based methods [10-13] utilize the trajectories of the mobile users to build the communities, in such a way that nodes exhibiting similar movement and behaviour patterns will belong to the same community. Both the above two strategies are benefitted with low transmission delay but also have similar disadvantages, such as management overhead and low search efficiency since both the two approaches do not consider preference similarity. The preference similarity-based methods [14-15] can show better search efficiency, which utilizes the interests and shared resources of the mobile users to form the communities with nodes having similar preferences. However, the disadvantages of those approaches can be attributed to its topology mismatch and higher traffic cost, since preference similarity-based methods do not consider the information regarding the geographical location of the mobile nodes.

In order to overcome the drawbacks of the resource discovery mechanisms mentioned above, this paper proposes a novel Resource Discovery mechanism based on the Preference and Movement pattern similarity (RDPM). Firstly, we extract the preferences of the nodes from their profile table and resources, and also movement pattern from their trajectories using the AGNES clustering method [16]. Then the cosine similarity of the preferences and movement pattern of the nodes

Manuscript received April 15, 2015. This work was supported in part by the National Natural Science Foundation of China under Grant No.61202474, the Project Funded by China Postdoctoral Science Foundation No. 2015M570469, the Natural Science Foundation of Jiangsu Province under Grant No.BK20130528 and the Senior Professional Scientific Research Foundation of Jiangsu University under Grant No.12JDG049.

Z.Y. Li and R.L. Chen are with the Department of Internet of Things, School of Computer Science and Telecommunications Engineering, Jiangsu University, Zhenjiang, China (e-mail: lizhiyuan@ujs.edu.cn).

L. Liu is with the Department of Computing and Mathematics, University of Derby, Derby, UK and the Department of Internet of Things, School of Computer Science and Telecommunications Engineering, Jiangsu University, Zhenjiang, China (e-mail: l.liu@derby.ac.uk).

G. Min is with the Department of Mathematics and Computer Science, University of Exeter, Exeter, UK (e-mail: g.min@exeter.ac.uk).

is generated for the purpose of building sub-community in a 3-dimensional Cartesian coordinate system in order to improve the efficiency of resource discovery. Next, virtual global communities are formed by utilizing the similarity found among sub-communities, ultimately to improve the searching performance for the resources. Finally, we design a resource discovery algorithm that can dynamically adjust the search radius in order to balance the performance and communication overhead.

The remainder of the paper is structured as follows: Section II introduces the related works. Section III presents the network model. Our proposed resource discovery algorithm based on preference and movement pattern similarity is elaborated in Section IV. The experimental evaluation and performance results are presented in Section V. Finally, Section VI concludes this paper.

## II. Related Work

In this section, the existing resource discovery mechanisms are analyzed by classifying them into three categories – social connectivity-based methods, movement pattern-based methods and preference similarity-based methods.

### *A. Social connectivity-based methods*

Li and Wu [4] proposed a Mobile cOmmunity-based Pub/Sub scheme (MOPS) to promote the content-based service by utilizing the long-term neighboring relationship between the nodes. In this approach, the node community is built in a distributive fashion based on the encounter frequencies of the nodes. A contact duration aware framework was proposed [5] to model the content dissemination process for MSNs in order to reduce the content dissemination delay. ICast [6] adopts the behavior-aware mechanism which regularly extracts weak ties, i.e. nodes having rare encounters, to ensure remarkable performance results in content delivery. By analyzing the tight-coupled relationship between human and opportunistic connection of smart things (e.g., mobile phones, vehicles), [7] proposed opportunistic Internet of Things. It enables information sharing and dissemination within/among opportunistic communities that are formed with the movement and opportunistic contact nature of human. [8] proposed a community detection scheme based on friend relationship to solve the problem in integrated IoT and Social Network (SN) architecture. Zheng and Wang [9] proposed a graph-based social aware algorithm in which cellular links and D2D links are established according to social ties and social contributions of users for efficient multi-file dissemination. However, such methods are subjected to lower efficiencies in their resource discovery mechanisms since they do not consider preference similarity.

### *B. Movement pattern-based methods*

Human trajectories often show a higher degree of temporal and spatial regularity, with each individual being characterized by a time independent characteristic length scale and also a significant probability of returning to a few highly frequented locations [10], which tends mobile users to exhibit similar movement and behavior patterns. Such movement patterns of a group of nodes can be extracted by recording the node movements. A movement pattern-aware optimal routing for social delay tolerant networks was proposed in the works of [11], where both a local search scheme and a tabu-search scheme are utilized in finding the optimal set. The tabu-search based routing shows the ability to guide the evaluation of relay node sets into optical node sets. [12] analyzed firstly the social relation between mobile-aware nodes, and then mined the activity rules of nodes and the community property and guided to select the mobile-aware nodes in target regions, so as to improve the discovery efficiency of nodes in objective regions, and increase the success ratio of service discovery. An efficient proactive service discovery protocol was proposed [13], which can leverage both the social behavior and the people mobility. Simulation results show that the protocol achieves increased efficiency in discovering services. Such movement pattern-based methods can enjoy lower resource discovery delay and transmission delay than the social connectivity-based methods. However, they do not fully utilize the preference of users, and hence the success rate of the resource discovery is still low.

### *C. Preference similarity-based methods*

An adaptive content sharing protocol to facilitate the implementation of MSNs over peer-to-peer networks was proposed [14] for preference driven communication. The proposed protocol takes into account the information about user's interests, preference based content storing and forwarding, and host mobility in a disconnected delay tolerant MANET, which can improve the performance of content dissemination. A P2P content-based file sharing system, namely SPOON, was proposed [15] for disconnected MANETs. The system uses an interest extraction algorithm to derive a node's interests from its files for content-based file searching. For efficient file searching, SPOON groups common-interest nodes that frequently meet each other, as communities. It exploits the node mobility for the purpose of categorizing the nodes into two types, in such a way that the nodes with more frequent encounters with the community members are categorized as stable nodes and the nodes frequently visiting other communities are categorized as highly mobile nodes. Stable nodes are generally assigned as community coordinates for intra-community searching and the highly mobile nodes are assigned as community ambassadors for inter-community searching. In general, the success rate of resource discovery of the preference similarity-based methods is better than two other methods. However, it incurs topology mismatch, which leads to higher communication overhead since they lack multidimensional context-aware information in their resource sharing mechanisms.

## III. Network Model

The architecture of the proposed RDPM model can be classified into three layers as shown in Fig. 1. Each node represents a counterpart of a physical smart device at the physical layer. Firstly, the physical area is divided into square units of equal area. In each square unit, the nodes belonging to the same sub-community have similar preferences and show consistent movement patterns. Then, similar sub-communities are merged into a global virtual community. For instance, hundreds of football fans are watching a World Cup soccer tournament in an open-air stadium. In the stadium, they spontaneously form a football sub-community. Meanwhile, fans outside the stadium form another football sub-community by watching TV in pubs or at home. But all of them together form the football global virtual community. The sub-communities together form the global virtual community and also have their corresponding local characteristics. For example, in the sub-community network layer, node 2 has two neighbors, as node 1 and node 3, respectively. Furthermore, considering the global community network layer, node 1 and node 3 are the neighbors of node 2. This strategy greatly reduces the traffic overheads incurred by the topology mismatch and improves the efficiency of resource discovery.

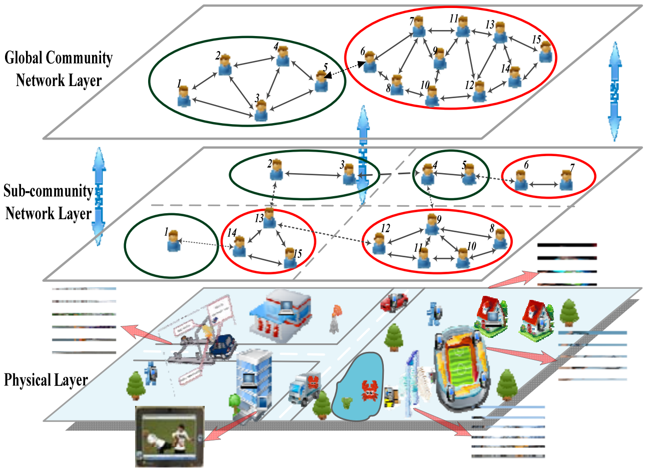


Fig. 1. Three-layer network structure of RDPM.

## IV. The Design of RdPm

### A. Preference Extraction

Users with common preferences tend to join the same community, and their shared resources or services are often similar. Hence, we extract the interests and preferences of the nodes using their profile lists and shared resources or services.

#### 1) Resource or service Representation

The properties of the resources or services are multi-dimensional, and the association properties of the nodes are generally weak. The differences in the association properties can only be attributed to the occurrence frequency of the keywords. For resource $R_i$, its preference vector is defined as

$$R_i = \{(k_1, w_1);(k_2, w_2);...;(k_n, w_n)\} . \quad (1)$$

where $k_i$ refers to the $i$ *th* keyword and $w_i$ is the weight of the $i$ *th* keyword. They are sorted by weights in descending order.

#### 2) Preference Representation

The shared resources are required to be clustered in order to be representative rather than a single entity. For example, the common resource vectors are extracted from $R_u$ and $R_v$, utilizing the common keywords and corresponding weights. Then, Formula (2) is used to calculate the similarity among the resource $u$ and the resource $v$.

$$sim(u,v) = \frac{\sum_{i=1}^{m} w_{ui} * w_{vi}}{\sqrt{\sum_{i=1}^{m} w_{ui}^2} * \sqrt{\sum_{i=1}^{m} w_{vi}^2}} . \quad (2)$$

where $m$ is the total number of common keywords, $w_{ui}$ and $w_{vi}$ represent the weights of the $i$ *th* common keyword in the two resources, respectively.

In a similar way, a global similarity matrix could be obtained by comparing the similarities of all the resources. The element of the similarity matrix is $sim^i_{(u,v)}$, and then they are sorted in a descending order to generate a resource similarity list. Formula (3) is used to choose the top $r$ higher similarity resources.

$$r = \arg\min \left| \frac{\sum_{i=1}^{r} sim^i_{(u,v)}}{\sum_{i=1}^{k} sim^i_{(u,v)}} - \rho \right|, 0 < \rho \le 1, 0 < r \le k \quad (3)$$

where $k$ is the length of the resource similarity list and $\rho$ is a user-specific parameter to determine top $r$ resources with higher similarity to represent the interests of a mobile user. The parameter could be set according to the specific user's requirement. Generally speaking, the higher $\rho$, the higher $r$. When $\rho = 1$, all the resources will be selected ($r$=$k$). While $\rho = 0$, none of the resources will be selected ($r$=0).

Next, the preference vector should be re-calculated for the top $r$ higher similarity resources to represent the interests of a mobile user, as shown below

$$w_{k_j} = \sum_{i=1}^{r} w_{R_i}^{k_j} / r . \quad (4)$$

$$R_n^p = (k_j, w_{k_j}), j > 0 . \quad (5)$$

where $w_{R_i}^{k_j}$ denotes the weight of the keyword $k_j$ in the resource $R_i$, and $R_n^p$ represents the preference vector of the node $n$.

### B. Movement Pattern Analysis and Extraction

#### 1) Current location

The current location information of nodes can be obtained via the Global Position System (GPS). The current location of a node $i$ is represented by the vector $l(x_i, y_i, z_i, t_i)$, where $(x_i, y_i, z_i)$ is three-dimension coordinates of the node $i$ and $t_i$ represents the timestamp that the node $i$ stays at the position $(x_i, y_i, z_i)$.

#### 2) Stay region

The stay region $A$ is a physical space where a user stayed over a time threshold $t_0$. Suppose that $l_i(x_i, y_i, z_i, t_i)$ is the node's position at time $t_i$, and $l_j(x_j, y_j, z_j, t_j)$ is the node's position at time $t_j$, then the sojourn time of a user is $T_{ij}$, and the value $T_{ij}$ is equal to $(t_j - t_i)$. When $T_{ij}$ is larger than the threshold value $t_p$ and the distance between the position $l_i$ and the position $l_j$ is smaller than a threshold value $r_p$, we conclude that the mobile users are in the same stay region. The shape of the region $A$ is a sphere whose radius is $r$ and the center is $(x_i, y_i, z_i)$.

*3) Movement pattern representation*

In order to analyze and extract the movement patterns of a node, the motion trajectories of the nodes are required including the location, the start time and the end time. Formula (6) is used to represent the motion trajectory of a node $i$.

$$T_i = (A_1(x_1, y_1, z_1,\ list(t_{start}, t_{end}), \delta_1), \dots \\ ,\ A_n(x_n, y_n, z_n, list(t_{start}, t_{end}), \delta_n)) . \tag{6}$$

where $t_{start}$ represents the start time when a node moves into the region $A_i$, and $t_{end}$ represents the end time when the corresponding node moves out of the region $A_i$. The list is an array storing multiple time records. The weight value $\delta_i$ in sphere $A_i$ is calculated using (7).

$$\delta_i = \frac{\sum_{i=0}^{m-1}(list(i).\,t_{end} - list(i).\,t_{start}))}{t_{total}} . \tag{7}$$

where $m$ is the length of a list, and $t_{total}$ represents the total time.

*C. Community Construction*

*1) Sub-community construction*

The preferences are combined with the movement pattern of the mobile users in order to build a sub-community.

① *Preference similarity*

Formula (8) is used to calculate the preference similarity $S_{ij}^{p}$ between node $i$ and node $j$.

$$S_{ij}^{p} = \frac{\sum_{k=1}^{m} w_{ik} * w_{jk}}{\sqrt{\sum_{k=1}^{m} w_{ik}^2} * \sqrt{\sum_{k=1}^{m} w_{jk}^2}} . \tag{8}$$

where $m$ is the total number of common keywords of the preference vector $S_{ij}^{p}$, and $w_{ik}$ and $w_{jk}$ represent the weights of the $k$ *th* common keyword of the preference vector $S_i^p$ and $S_j^p$.

② *Movement pattern similarity*

As shown in Fig. 2, the motion trajectory of node *i* and node *j* is the same from time $t_a$ to time $t_b$. This shows the similarity of movement pattern.

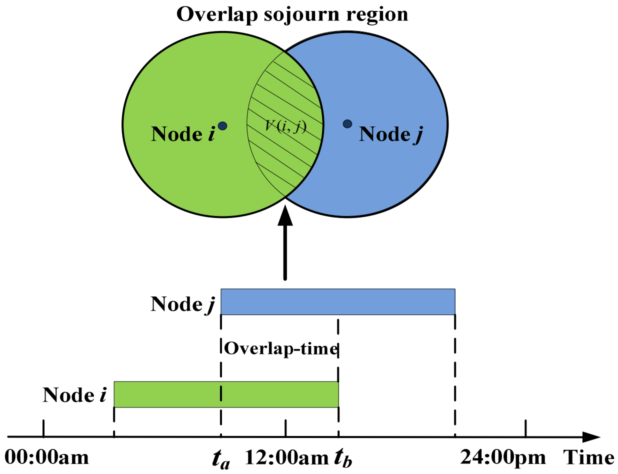


Fig. 2. Overlap sojourn region at the same period.

Equation (9) is used to calculate the movement pattern similarity $S_{ij}^{mp}$ between node $i$ and node $j$.

$$S_{ij}^{mp} = \frac{3\sum_{i=1}^{n}\sum_{j=1}^{n} V(i,j)\delta_i * \delta_j}{4n\pi r^3 \sqrt{\sum_{i=1}^{n}\delta_i^2} * \sqrt{\sum_{j=1}^{n}\delta_j^2}} . \tag{9}$$

where $V(i, j)$ is the overlap space between the volume of the sojourn region $A_i$ of the node *i* and the volume of the sojourn region $A_j$ of node *j* at the same time period, both $\delta_i$ and $\delta_j$ are the weight values of the sojourn regions $A_i$ and $A_j$, respectively.

③ *Construction procedure*

Algorithm 1. Construction of the sub-community

```
Input: preference and movement pattern vectors of a node $N_i$
and its neighbors
Output: sub-community member list
Begin
for each $N_{nb}$ of $N_i$ do
 {
    $N_i$. brocastsTo $N_{nb}$
    $N_{nb}$.computers $S_{i.nb}^{t}$
    if $S_{i.nb}^{t} > T_{sim}$
    {
        $N_{nb}$.add $N_i$ To CommList
        $N_{nb}$.sendjoinmessageTo $N_i$
```

```
      }
  }
    if acceptedmessage is joinmessage
        $N_i$.addN$_{nb}$To CommList
    Return subCommlist
End
```

Formula (10) is used to calculate the total similarity between node $i$ and node $j$.

$$S_{ij}^{t} = \alpha S_{ij}^{p} + \beta S_{ij}^{mp}, \alpha + \beta = 1. \quad (10)$$

If $S_{ij}^{t}$ is greater than a predefined threshold $T_{sim}$, node $i$ joins the sub-community of node $j$. In the initial process of sub-communities construction, node $i$ will broadcast the "Hello" message to its neighbors, and the neighbor nodes use Formula (10) to compute their total similarity with node $i$. The neighboring node is added to the sub community $m$ when the total similarity value $S_{ij}^{t}$ is larger than the predefined threshold value. To reduce the overhead, each node saves the top $k$ neighbors. The pseudo code for the construction of the sub-community is shown in Algorithm 1. Regarding $T_{sim}$ in Algorithm 1, $T_{sim}$ is directly related to the size of each sub-community. $T_{sim}$ should be set according to the similarity between community members and the size of each sub-community. Generally speaking, the higher $T_{sim}$, the smaller the sub-community. When $T_{sim}$ =0, there should be only one sub-community where its size is equal to the global community. When $T_{sim}$ =1, each sub-community will be the node itself.

After the successful construction of the sub-community, the management node calculates the central location vector of the community which is shown below

$$(x_i = \frac{1}{n}\sum_{k=1}^{n} x_k, y_i = \frac{1}{n}\sum_{k=1}^{n} y_k, z_i = \frac{1}{n}\sum_{k=1}^{n} z_k). \quad (11)$$

where $n$ is the number of members in the sub-community.

*2) Node role assignment and sub-community merger*

In each sub-community, there are three types of nodes, including administrators, ambassadors and ordinary nodes.

An Administrator plays the managing role in a sub-community, and maintains the indexes of all resources belonging to the sub-community. Hence a stable and reliable node is chosen as the administrator in a sub-community. The administrator has two important tasks: One is to help ordinary nodes to search resources within or out of the sub-community, and the other is to periodically collect the information of the ordinary users in the sub-community and to extract the interest vector using the AGNES clustering method.

The primary responsibility of an ambassador is to bridge the sub-communities in order to form the global virtual community. To this end, the ambassador regularly moves in and out of the sub-communities.

Then, we give a procedure of sub-community merger.

Step 1: Community preference vectors extraction

Administrator periodically collects the preference vectors information of the nodes in the sub-community, and calculates the $sim(i, j)$ value using Formula (2), where $i$ and $j$ represent any two nodes in the same sub-community, respectively. Then the same clustering method with the preference extraction is leveraged to get the preference vector of sub-community $R_c^p$. Analogously, we can get the preference vector of global community by calculating the similarity between the sub-community preference vectors.

Step 2: Sub-community merger

The mobility of the ambassadors is utilized to carry the preference vectors of the sub-communities in order to merge similar sub-communities. When an ambassador moves into another sub-community, it firstly calculates the similarity of the interest vector between the two sub-communities. If this similarity is larger than the threshold value, the two sub-communities are then merged to form a bigger virtual community. Ideally, all the similar sub-communities with common interests will be associated together to construct a global community. It is noted that the preference vector for the global community is also need to be calculated.

### *D. Message Routing*

Message routing and data forwarding on the two dimensional plane will reduce the efficiency of resource discovery and transmission. As a result, a greedy algorithm is used to improve the efficiency of resource discovery in the three dimensional Cartesian coordinate system. Fig. 3 is a diagram of message routing. The source node $S$ plans to construct a path $\overline{SD}$ to the target node $D$. After a node $B$ suddenly moves out of the communication range of node $S$, this node needs to calculate its angle with nodes $E, F$ and $G$. As shown in Fig.3, $\beta_1$ is the angle between $\overline{SD}$ and $\overline{SE}$, $\beta_2$ is the angle between $\overline{SD}$ and $\overline{SF}$, $\beta_3$ is the angle between $\overline{SD}$ and $\overline{SG}$. Now, node $S$ chooses node $E$ as the next hop since $\beta_1$ being the minimum forwarding angle. After that, node $C$ is also chosen as the next hop. To ensure quick message forwarding, the sum value $\sum_{k=0}^{m-1}\beta_k$ should be minimum, where, $m$ is the $m$ $th$ forwarding node.

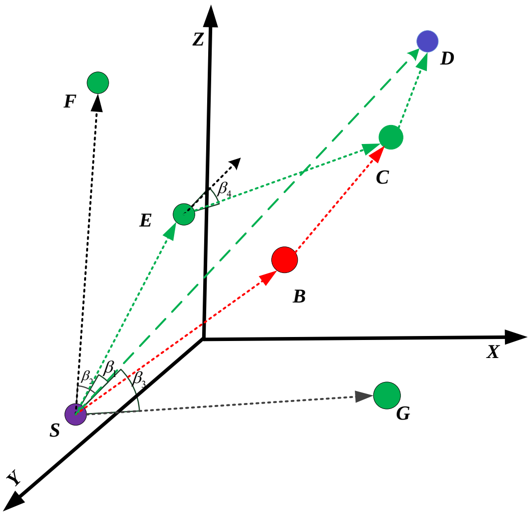


Fig. 3. Diagram of message routing.

### E. Resource Discovery Process

```
Algorithm 2. Search within the sub-communities

Input: the request vector of a node $N_i$
Output: resource holder list
Begin
if $S^p_{Q,C} < T_Q$ then
    $N_i$ .sendQueryTo $N_{Ad}$
    else $N_i$ .rank $N_{nb}$ By $S^p_{Q,Nb}$
    {
      for each $N_{nb}$ of $N_i$ do
      {
        $N_i$ .sendQueryTo $N_{nb}$ with message routing
         Count++
        if($N_{nb}$ ==resource holder) then
        Return $N_{nb}$
        else if (Query.hops<MaxHop) then
            $N_i$.sendQueryTo $N_{next}$
            Query.hops++
        if Count>k then
          break
      }
     if(SearchResults == null) then
       $N_i$ .sendQueryTo $N_{Ad}$
    }
End
```

#### 1) Searching within the sub-community

The requester calculates the similarity between the query vector and its sub-community interest vector. If this similarity is lower than the threshold value, the query message is sent out of the local sub-community. Otherwise, the query message is sent to the top $k$ neighbors with higher similarity. The neighbor continues to forward this query message to its neighbors if it lacks the requested resource. TTL (time-to-live) is used to control the search depth and avoid excessive overhead. The TTL value decreases one when the message is forwarded every time. The resource discovery will eventually stop when the TTL value becomes zero. Algorithm 2 shows the pseudo code of the resource discovery process within a sub-community.

#### 2) Searching within the global communities

The Fig.4 illustrates an example of searching within the global communities. If the requested resource is not found within a sub-community 1-1, the query message will be forwarded to its similar neighboring sub-communities 1-2, 1-3, and 1-4 by the ambassador node 1, the ambassador node 2 and the ambassador node 3 respectively. And the search within the sub-community is recalled again. The search within the global community stops when either the TTL value of the query message becomes zero or the requested resources is discovered.

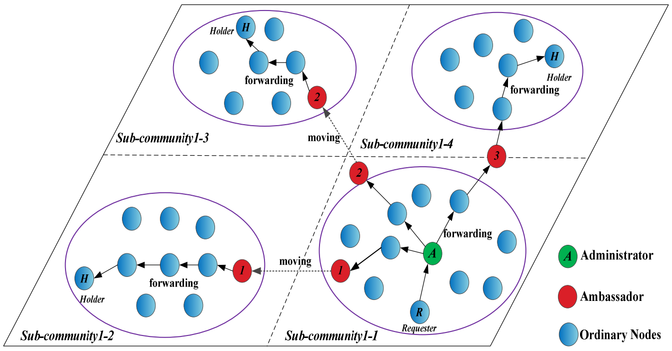


Fig. 4. Searching within the global communities

The searching process within the global communities is described in Algorithm 3.

```
Algorithm 3. Searching within the global communities

Input: the request vector of $N_i$
Output: resource holder list
Begin
   $N_{Ad}$. rank$C_{sub}$ByCentralLocation
    for each $C_{sub}$ of $N_{Ad}$ do
    {
      $N_{Ad}$. sendQueryTo $C_{sub}$ with message routing
      Count++
      Searching within sub-communities()
      if Count>k then
        break
        }
      if(SearchResults == null) then
        $N_i$ .sendQueryTo $N_{Ad}$
      }
  End
```

### 3) Global searching over social Internet of Things

Fig. 5 illustrates an example of global searching over the social IoT.

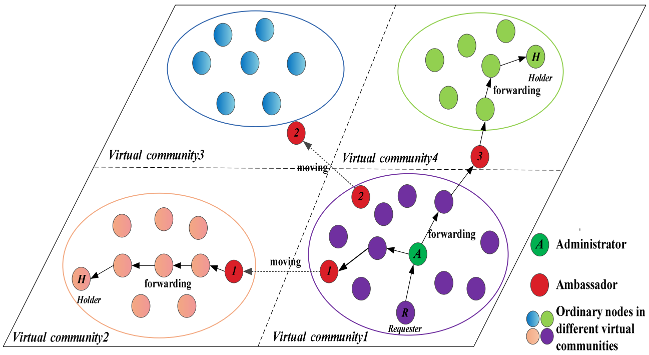


Fig. 5. Global searching over the social Internet of Things.

If the search within a global community is not successful, the global search over the social IoT is triggered. The similarity between the query vector and other virtual communities including virtual community 2, virtual community 3 and virtual community 4 are calculated, respectively. And then, the query message is quickly forwarded to the virtual community with higher similarity through the message routing. Now, searching within the sub-communities of the corresponding global community is triggered again.

The global searching process over social Internet of Things is described in Algorithm 4.

Algorithm 4. Global searching over social Internet of Things

```
Input: the request vector of $N_i$
Output: resource holder list
Begin
    $N_{Ad}$. rank$C_{viru}$By $S^P_{Q,C}$
    for each $C_{viru}$ of $N_{Ad}$ do
    {
      $N_{Ad}$. sendQueryTo $C_{viru}$ with message routing
      Count++
     Searching within the global communities()
      if Count>k then
        break;
      }
End
```

## V. Performance Evaluation

### A. Simulation Environment and Parameters Setting

The opportunistic network environment (ONE) [17] has been used as the simulation platform in order to evaluate the performances of our scheme. As shown in Fig. 6, the mobile social activities of the users over the SIoT are based on the campus environment of Jiangsu University in China.

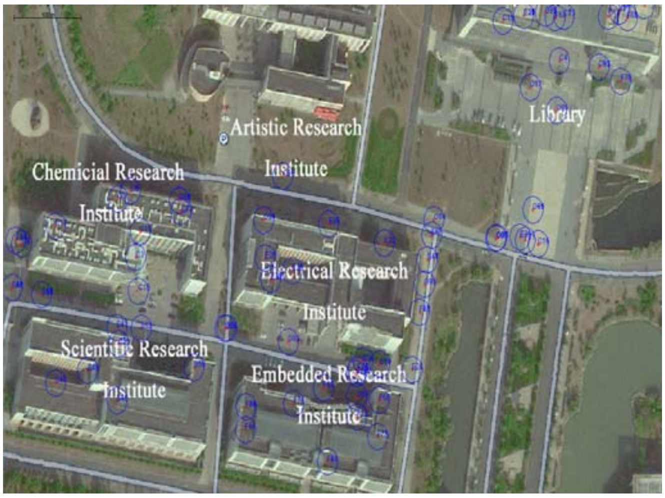

Artistic Research Institute
Library
Chemicial Research Institute
Electrical Research Institute
Scientific Research Institute
Embedded Research Institute

Fig. 6. The geographic map.

100 mobile users have been classified into four interest groups and one movement group, such that each group comprises 20 users. The four interest groups include the chemical interest group, the electrical interest group, the scientific research institute's interest group and the embedded system interest group, accordingly. The movement group consists of random walk users along the roads, as shown in Fig. 6. The responsibility of the random walk users is to help the member nodes in the interest groups to carry and forward the message and data. The corresponding parameter configuration information is shown in Table 1.

TABLE I. Simulation Parameters

| Parameters | Description | Default values |
|---|---|---|
| MovementModel.world Size($m^2$) | Simulation scenario area | 1000 × 800 |
| Scenario.endTime(hour) | Simulation time | 12 |
| Scenario.nrofHosts | Number of nodes | 100 |
| Scenario.nrofHostGroups | Number of groups | 5 |
| BtInterface.type | Communication interface type | SimpleBroadcast Interface |
| BtInterface.transmit Speed(Kbps/s) | Bandwidth | 250 |
| BtInterface.transmit Range(m) | Transmission range | 10~20 |
| Group.movementModel | Movement model of nodes within a group | ShortestPathMap BasedMovement |
| Mobility Speed(m/s) | Walking speed of nodes | 0.5~1.5 |
| Message.Size(k) | Size of message | 500~1024 |

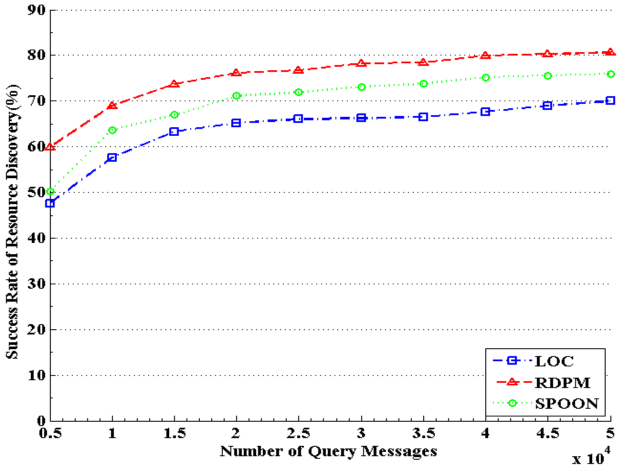
| | | |
|---|---|---|
| Message. Interval(s) | Message interval | 1~2 |
| Message.TTL(min) | The time to live value for the forwarded message | 300 |
| $\rho$ | The threshold value for choosing the top ***r*** higher similarity resources | *0.5* |
| $T_{sim}$ | The threshold value for construction of the sub-community | *0.3* |
| $T_Q$ | The threshold value for resource query within the sub-communities | *0.5* |
| $\alpha$ | The weight with preference similarity parameter. | *0.6* |
| $\beta$ | The weight with movement pattern similarity parameter. | 0.4 |

Our proposed scheme RDPM is compared with the two related methods - geographic location-based scheme (LOC) [12] and the preference similarity-based scheme (SPOON) [15] in terms of the success rate of resource discovery and the average delay.

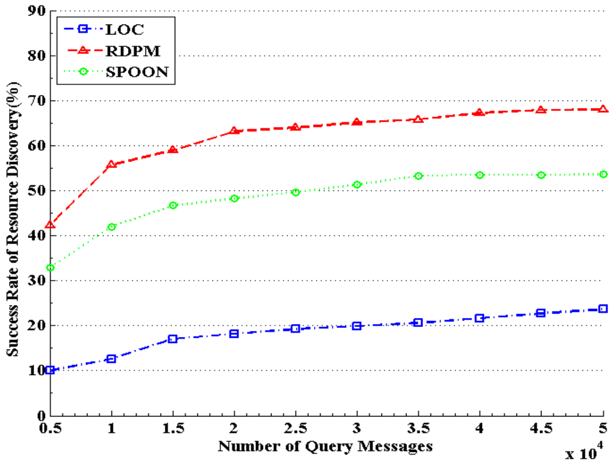
The success rate of resource discovery is computed as the ratio of the number of successful queries to the total number of the queries.

The average delay is the average time used for successful queries.

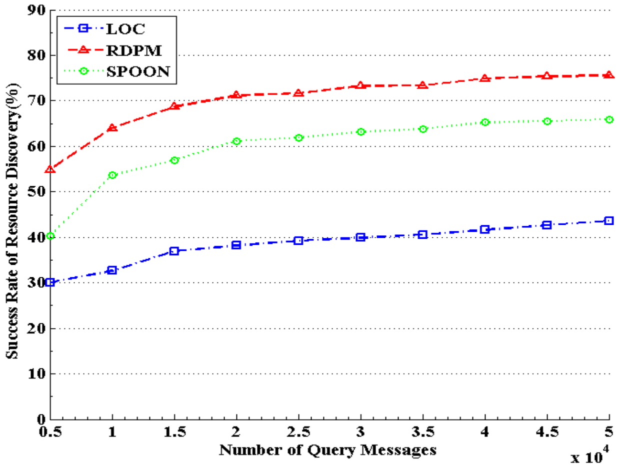
### *B. Success Rate of Resource Discovery*



(a) Intracommunity search efficiency.



(b) Intercommunity search efficiency.



(c) Global search efficiency.

Fig. 7. Search efficiency.

We evaluate the search efficiency of the three schemes based on the intra-community search, the inter-community search and the global search. Fig. 7(a) depicts the intra-community search efficiency comparisons, Fig. 7(b) shows the inter-community search efficiency comparisons, and Fig. 7(c) shows the global search efficiency comparisons. The LOC scheme leverages the similarities of the movement patterns among mobile users for the purpose of constructing communities to search the desired resources. However, this scheme does not exploit the content similarity of the resources, which makes it less efficient than both the RDPM scheme and the SPOON scheme. The SPOON scheme selects the mobile users with higher preference similarity over the overlay social networks in order to construct a community but it does not consider the physical communication distance between the users. Thus, the success rate of resource discovery of the SPOON scheme is significantly lower than that of the RDPM scheme. The RDPM scheme combines the advantages of the SPOON with LOC scheme, and hence, it can apparently improve the efficiency of resource discovery.

### *C. Average Delay*

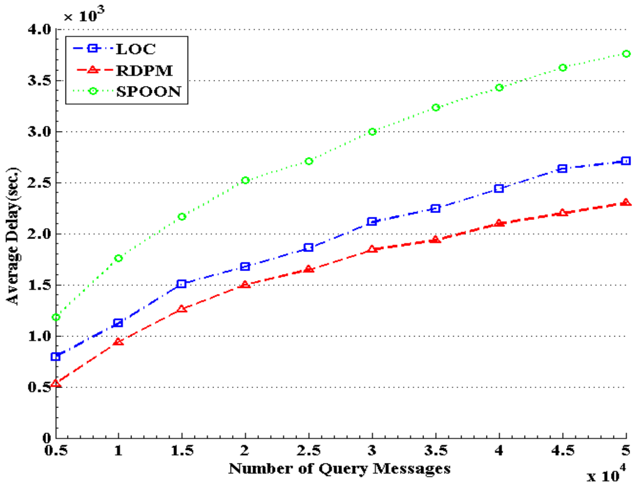


Fig. 8. Average delay.

Fig. 8 shows the comparison of the three schemes in terms of the average delay. Simulation results show that the average delay of the SPOON is larger among the three. This is because the query message is forwarded to the node with the highest preference similarity in the SPOON scheme. However, if the optimal forwarding node is out of the communication range of the requesting node, the transmission of the query message will incur additional waiting time in opportunistic wireless ad-hoc networks, which results in higher average delay. The resource discovery of the LOC scheme can only support forwarding in proximate SIoT. Hence, the average delay of the LOC scheme is lower than that of the SPOON scheme. Nevertheless, the LOC scheme does not utilize the preference similarity of the mobile users. And thus, the resource discovery of the LOC scheme incurs longer delay time than our proposed scheme. Our scheme leverages the preference similarity of the mobile users to construct interest communities in proximate SIoT. As shown in Fig. 8, the average delay of the RDPM scheme is lower than the other two schemes.

## VI. Conclusion And Future Work

This paper proposes a resource discovery algorithm based on preference and movement pattern similarity in disconnected and delay-tolerant social Internet of Things. Our proposed scheme is based on preference and movement pattern similarity as well as incorporates the 3-dimension geographical location awareness to achieve the higher search efficiency and reduce the system overheads for social Internet of Things. Our algorithm has been simulated and evaluated in an opportunistic social Internet of Things environment. Simulation results show that our proposed scheme outperforms the state-of-the-art resource discovery schemes in terms of the search efficiency and the average delay. To further improve the efficiency of our resource discovery algorithm and to reduce the wait time of the nodes in mobile social Internet of Things, we plan to utilize the characteristic of day-to-day human social behavior to design an effective behavior prediction model to solve the technological problem as our future work.

## Acknowledgement

This work was partially supported by the National Natural Science Foundation of China under Grant No.61202474, the Natural Science Foundation of Jiangsu Province under Grant No.BK20130528 and the Senior Professional Scientific Research Foundation of Jiangsu University under Grant No.12JDG049.

## References

[1] L. Atzoria, A. Ierab, G. Morabitoc, M. Nitti, "The social Internet of things (SIoT) – when social networks meet the internet of things: concept, architecture and network," Computer Networks, vol. 56, no.16, pp. 3594-3608, Nov. 2012.

[2] S.M. Allen, M.J. Chorley, G.B. Colombo, E. Jaho, M. Karaliopoulos, I. Stavrakakis, et al, "Exploiting user similarity and social links for micro-blog forwarding in mobile opportunistic networks," Pervasive and Mobile Computing, vol. 11, April 2014, pp. 106-131, April 2014.

[3] S.H. Chen, G.J. Wang, and W.J. Jia. 2014 June. "Cluster-group based trusted computing for mobile social networks using implicit social behavioral graph." Future Generation Computer Systems. [Online]. Available:http://www.sciencedirect.com/science/article/pii/S0167739X14001265.

[4] F. Li and J. Wu, "MOPS: Providing content-based service in disruption-tolerant networks," Proc. the 29th IEEE Int. Conf. on Distributed Computing System(ICDC' 09), IEEE Press, Jun. 2009,pp.526-533.

[5] Y. Li, Q. Li, D.P. Jin, S. Li, H. Pan and L.G. Zeng, "Contact duration aware evaluation for content dissemination delay in mobile social network," Wireless Communications & Mobile Computing, vol. 15, no.3, pp. 527-537, Feb. 2015.

[6] E. Pagani, L. Valerio and GP. Rossi "Weak social ties improve content delivery in behavior-aware opportunistic networks," Ad Hoc Networks, vol. 25, no.PB, pp.314-329, Feb. 2015.

[7] B. Guo, Z. Yu, X. Zhou, and D.Zhang, "Opportunistic IoT: Exploring the social side of the internet of things," Proc. the 16th IEEE Int. Conf. on Computer Supported Cooperative Work in Design (CSCWD' 12), IEEE Press, May 2012,pp.925-929.

[8] S. Misra, R. Barthwal, M. S. Obaidat, "Community detection in an integrated Internet of things and social network architecture," Proc. the 54th IEEE Global Communications Conf. (GLOBECOM'12), IEEE Press, Dec 2012, pp. 1647-1652.

[9] Z.J. Zheng, T.Y. Wang, L.Y. Song, Z. Han and J.J. Wu, "Social-aware multi-file dissmination in device-to-device overlay networks," Proc. of the 33rd IEEE Int. Conf. on INFOCOM, Computer Communications Workshops, IEEE Press, Apr. 2014, pp. 219-220.

[10] J. An, X. Gui, W. Zhang, J. Jiang, "Nodes Social Relations Cognition for Mobility-Aware in the Internet of Things," Proc. of the 4th IEEE Int. Conf. Internet Things Cyber, Phys. Soc. Comput., iThings/CPSCom, IEEE Press, Oct. 2011, pp. 687-691.

[11] M. C. Gonzales, C. A. Hidalog and A. L. Barabasi, "Understanding individual human mobility patterns," Nature, vol. 453, pp. 779-782 , Jun 2008.

[12] L. You, J.B. Li, C.J. Wei and L.J. Hu, "MPAR: A movement pattern-aware optimal routing for social delay tolerant networks," Ad Hoc Networks, vol.24, no. PA, Jan. 2015, pp.228-249.

[13] M. Girolami, P. Barsocchi and S. Chessa, "A social-based service discovery protocol for mobile Ad Hoc networks" Proc. 12th Annual Int. Conf. on Ad Hoc Networking Workshop (MED-HOC-NET, 2013), pp. 103-110.

[14] B. Qureshi, G.Y. Min and D. Kouvatsos, "An adaptive content sharing protocol for P2P mobile social networks," Proc. of the 24th IEEE Int. Conf. on Advanced Information Networking and Applications Workshops, Apri 2010, pp. 413-418.

[15] K. Chen, H.Y. Shen and H.B. Zhang, "Leveraging social networks for P2P content-based file sharing in disconnected MANETs," IEEE Transactions on Mobile Computing, vol. 13, pp. 235-249, Feb 2014.

[16] L. Kaufman and P.Rousseeuw, "Finding groups in data: an introduction to cluster analysis,"John Wiley and Sons, 1990.

[17] K. Ari, O. Jrg and K. Teemu, "The one simulator for DTN protocol evaluation," Proc. of the 2nd Int. Conf. on Simulation Tools and Techniques(ICST'09), Mar 2009.

**Zhiyuan Li** is an associate professor in the School of Computer Science and Telecommunication Engineering at Jiangsu University. He received his Ph.D. degree in information and communication engineering from Nanjing University of Posts and Telecommunications, Nanjing, in December 2011. He is a membership of China Computer Federation and ACM. His research interests include social Internet of Things, mobile peer-to-peer computing, vehicle networks, and mobile cloud computing.

**Rulong Chen** is a graduate research student in the School of Computer Science and Telecommunication Engineering at Jiangsu University. He is a student membership of China Computer Federation and ACM. His research interests include mobile social networks and resource sharing.

**Lu Liu** is the Professor of Distributed Computing at the School of Computing and Mathematics in the University of Derby and adjunct professor in the School of Computer Science and Communication Engineering at Jiangsu University. Prof. Liu received his Ph.D. degree from University of Surrey (funded by DIF DTC) and M.S. in Data Communication Systems from Brunel University. He is the Fellow of British Computer Society and Member of IEEE. Dr. Liu's research interests are in areas of cloud computing, service-oriented computing, peer-to-peer computing, virtual computing and system of systems engineering.

**Geyong Min** is the Professor and Director of High Performance Computing and Networking (HPCN) Research Group at the University of Exeter, UK. He received the PhD degree in Computing Science from the University of Glasgow, UK, in 2003, and the B.Sc. degree in Computer Science from Huazhong University of Science and Technology, China, in 1995. He joined the University of Bradford as a Lecturer in 2002, became a Senior Lecturer in 2005 and a Reader in 2007, and was promoted to a Professor in Computer Science in 2012. His main research interests include Next-Generation Internet, Analytical Modelling, Cloud Computing, and Big Data.