

一种面向信息质量的移动群智感知协作方法

张继德¹, 袁 印¹, 刘亚志¹, 张新英², 杨宗帅³

(1. 华北理工大学 信息工程学院, 河北 唐山 063009; 2. 承德广播电视大学 信息技术教研室, 河北 承德 067000; 3. 同济大学 信息工程学院, 上海 201800)

摘 要: 现有的感知信息质量研究主要集中在节点的招募、选择和感知任务的分配阶段, 缺少对感知任务执行过程的优化。为此, 借鉴移动设备云中的感知任务迁移理念, 设计基于效用的感知任务迁移算法, 提出一种移动群体内节点间直接协作的感知方法。对感知任务的执行阶段进行优化, 以解决移动设备的异构性与感知信息质量需求之间的矛盾。实验结果表明, 与随机选择算法和基于多任务的参与者选择算法相比, 该算法可有效提高感知数据覆盖率和感知任务完成率。

关键词: 移动群智感知; 效用; 感知任务迁移; 协作感知; 信息质量

中文引用格式: 张继德, 袁 印, 刘亚志, 等. 一种面向信息质量的移动群智感知协作方法[J]. 计算机工程, 2017, 43(7): 15-21.

英文引用格式: Zhang Jide, Yuan Yin, Liu Yazhi, et al. A Quality of Information Oriented Collaboration Method for Mobile Crowd Sensing[J]. Computer Engineering, 2017, 43(7): 15-21.

A Quality Information Oriented Collaboration Method for Mobile Crowd Sensing

ZHANG Jide¹, YUAN Yin¹, LIU Yazhi¹, ZHANG Xinying², YANG Zongshuai³

(1. College of Information Engineering, North China University of Science and Technology, Tangshan, Hebei 063009, China;

2. Information Technology Teaching and Research Office, Chengde Radio and TV University, Chengde, Hebei 067000, China;

3. College of Information Engineering, Tongji University, Shanghai 201800, China)

【Abstract】 The existing research on sensing information quality mainly focuses on the mobile nodes recruiting, selecting and sensing task allocating, lacking optimizing the execution of sensing tasks. This paper proposes Sensing Task Migrating (STM) method based on utility among the distributed heterogeneous mobile sensing devices which can collect sensing data collaboratively. The process of task execution is optimized to solve the contradiction between the heterogeneity of mobile devices and the quality requirement of sensing information. Experimental results show that, compared with random selecting algorithm and participant selecting algorithm based on multi-tasking, the algorithm improves the ratio of sensing data coverage and the ratio of sensing task finished.

【Key words】 Mobile Crowd Sensing (MCS); utility; Sensing Task Migrating (STM); collaborative sensing; information quality

DOI: 10.3969/j.issn.1000-3428.2017.07.003

0 概述

移动群智感知(Mobile Crowd Sensing, MCS)技术^[1-2]和云计算技术是当前移动计算与移动互联网领域的研究热点。移动群智感知的基本思想是通过随人移动的移动终端设备收集共享各类感知数据, 然后利用收集到的感知数据为用户提供各项新颖服

务。例如 Common Sense 项目通过手持设备收集感知数据监测环境^[3]; 文献[4]通过鼓励公交车上的乘客利用智能手机收集、共享周围小区的无线信号强度、运动状态、噪音等信息来估计公交车的行驶路径和到站时间; 文献[5]利用智能手机的相机拍水的照片, 通过分析这些照片的比色成分来检测水的质量; 文献[6]利用麦克风收集所在位置的噪音程度, 以较

基金项目: 国家自然科学基金(61370197)。

作者简介: 张继德(1963—), 男, 副教授, 主研方向为数据库技术; 袁 印, 硕士研究生; 刘亚志, 讲师、博士; 张新英, 讲师; 杨宗帅, 学士。

收稿日期: 2016-07-15 **修回日期:** 2016-08-16 **E-mail:** breadtreeyy@163.com

小的人力和时间开销建立实时细粒度的噪音地图。

从移动群智感知的基本特性和应用中可以发现,采集到的感知信息质量是决定移动群智感知技术能否走向大规模应用的关键因素。现有的研究大多集中在移动节点的选择和招募上,通过选择出最优的移动节点来完成感知任务从而达到提高感知信息质量的目标。文献[7]研究参与者选择算法,探索了基于参与者时间、空间位置及任务区域要求匹配程度进行参与者选择的方案。文献[8]通过对任务的空间需求及参与者的历史移动轨迹的匹配程度来选择参与者,该方法假设移动节点的历史移动轨迹是已知的,不利于参与者的隐私保护。在多任务环境下的参与者选择与单任务环境下的参与者选择存在很大的不同,文献[9]提出了多任务环境下的参与者选择问题,选出采集到的数据的价值和消耗的成本之间的差值最高的参与者。文献[10]提出了一种面向多感知任务的参与者选择算法(简称 DPS),该算法根据参与者的转移概率来选择出最合适的参与者,最大化满足不同任务对不同数据的需求。

文献[11]将移动群智感知过程分为 4 个步骤:感知任务的建立,分配,执行和数据整合。当前的感知信息质量保障方法大多集中在感知任务的分配阶段,缺少对感知任务执行过程的优化。文献[12]针对节点的移动性、社会性及复杂性,实现了基于分级激励的感知任务众包分配。事实上,在感知任务执行过程中通过引入节点间的协作^[13]能进一步提高感知信息质量。在感知任务的执行过程中机会地将感知任务迁移到固定的感知节点能够提高感知信息的质量,同时还能够降低移动感知设备的能源消耗^[14]。

在移动云计算领域,任务迁移 computing task offloading^[15]能够将复杂的计算任务迁移到云端进行,降低智能移动终端的计算负担,提高计算任务的执行效率^[16]。在移动群智感知领域,文献[17]尝试利用节点间的机会性相遇传输感知数据,但没有涉及到感知设备间的感知任务迁移协作。

综上所述,在感知信息质量方面,当前关于移动群智感知的研究主要集中在感知任务的建立和分配阶段。然而,在一个动态演化的异构群体中,将感知任务由固定的移动节点来完成显然不利于感知数据质量的提高。本文引入移动设备云中感知任务的迁移^[18]理念,提出基于效用的感知任务迁移算法,对感知任务的执行阶段进行优化来提高移动群智感知信息质量。

1 感知系统结构

本节主要介绍移动群智感知系统模型,如图 1 所示。该系统由感知数据服务中心、网络 and 一组自由移动的备选移动节点组成。其中,感知数据服务中心负责统筹、管理整个感知系统;感知数据服务中心负责不定时地发放感知任务并选择若干移动节点来完成感知任务。感知数据服务中心还负责收集和存储由移动节点上传的感知数据,并根据这些数据向用户提供新颖服务。

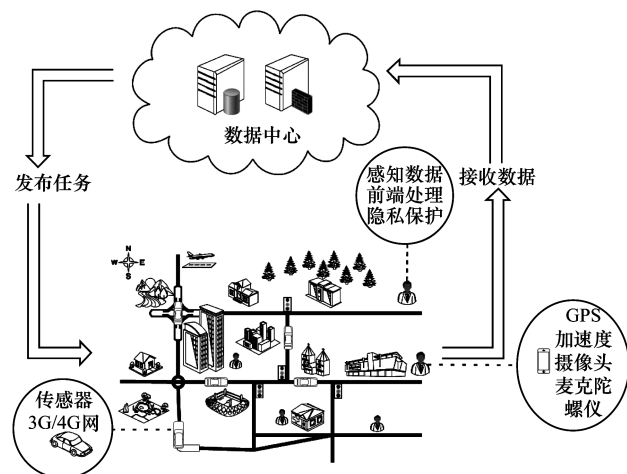


图 1 群智感知系统结构

移动节点是指集成了丰富感知接口的移动设备,如智能手机、平板电脑、车辆中的移动设备等。移动节点能够收集多种类型的感知数据,并自愿完成感知任务,令 α 表示一个移动节点。移动节点可以通过网络和其他移动节点交换移动轨迹、感知效用等信息,并通过网络将感知数据上传给感知数据服务中心。

系统中的移动节点都装备 2 种无线网络接口:一种是基于 802.11 协议的短距离通信接口,通过这个接口,临近的移动节点能组成自组网,协作收集感知数据;另一种是 3G 或 4G 的无线网络接口,移动节点通过这种网络接口与感知数据服务中心相连,接收来自感知数据服务中心的感知任务,或上传感知数据。

1.1 感知数据模型

本文将要采集的感知数据的地理区域称之为目标感知区域,用 I 表示,并将其划分为若干子区域。令 $I = \{1, 2, \dots, N\}$ 表示感知区域 I 内子区域的集合,其中 I 为有限集, $i (1 \leq i \leq N, i \in I)$ 。在子区域任意一点采集到的感知数据的值都能代表此子区域的值。令 $M = \{1, 2, \dots, m\}$ 表示感知系统中的 m 种感

知数据类型,令 $W = \{w_1, w_2, \dots, w_l\}$ 表示系统中感知数据保质期的集合,其中 w_l 表示对第 l 种感知数据的保质期,令 $Y_\alpha = \{Y_1^\alpha, Y_2^\alpha, \dots, Y_l^\alpha\}$ 表示节点 α 的感知能力,定义:

$$Y_l^\alpha = \begin{cases} 0 \\ 1 \end{cases} \quad (1)$$

其中, $y_l^\alpha = 0$ 表示节点 α 不能收集 l 型感知数据; $y_l^\alpha = 1$ 表示节点 α 能收集 l 型感知数据。移动节点一般在移动过程中采集所在子区域的感知数据。根据感知目标区域的划分方法可知,移动节点在其子区域内的移动会持续一段时间,令 ξ_i^α 表示移动节点 α 进入子区域 i 的时刻, θ_i^α 表示移动节点在子区域 i 的停留时间。节点 α 在 t 时刻在区域 i 收集 l 型感知数据的函数可以表示为:

$$h_{il}^\alpha(t) = \begin{cases} y_l^\alpha, & \xi_i^\alpha \leq t \leq \xi_i^\alpha + \theta_i^\alpha + w_l \\ 0, & \text{其他} \end{cases} \quad (2)$$

1.2 移动模型

移动节点的分布和运动具有随机性,为了克服移动节点这种特性带来的影响,需要研究移动节点的运动规律。移动模型是用来描述移动节点的移动模式(包括移动节点的位置、速度和加速度的变化等),决定节点如何移动,是对节点移动方式的一种抽象。移动模型参数的选择对仿真结果的影响很大。本文令 $\{x_n, n \geq 0\}$ 表示节点移动的随机过程,如果 $x_n = i$, 那么称该移动节点在时刻 t 处于状态 i (例如节点在 t 时刻所处的子区域)。在一般情况下,节点下一时刻的状态只与节点前一时刻所处的状态有关,因此:

$$p\{x_{n+1} = j | x_1 = j_1, x_2 = j_2, \dots, x_n = j_n\} \quad (3)$$

即节点的移动过程为离散的马尔科夫链。对于任意的 $n, t \geq 0, i, j \in I$ 有:

$$p_{ij}^{(n)}(t) = p\{x_{t+n} = j | x_t = i\} \quad (4)$$

其中, $p_{ij}^{(n)}(t)$ 表示移动节点在 t 时刻处于子区间 i 的条件下,经 n 步转移到子区间 j 的 n 步转移概率。马尔科夫链 $\{x_n, n \geq 0\}$ 的一步转移概率 $p_{ij}(t)$ 可以确定所有的 n 步转移概率 $p_{ij}^{(n)}$ 。

证明:由 $C - K$ 方程,记 $\mathbf{p}^{(n)} = (p_{ij}^{(n)})_{i,j \in I}$, $\mathbf{p} = \mathbf{p}^{(1)} = (p_{ij})_{i,j \in I}$ 。称 $\mathbf{p}^{(n)}$ 为马尔科夫链的 n 步转移概率矩阵。此时, $C - K$ 方程可表示为:

$$\mathbf{p}^{(m+n)} = \mathbf{p}^{(m)} \mathbf{p}^{(n)} \text{ 且 } \mathbf{p}^{(n)} = \mathbf{p}^{(n)}$$

综上所述,移动节点的转移概率矩阵可表示为:

$$\mathbf{p}_{ij}^{(n)} = (p_{ij}^{(n)})_{i,j \in I} \quad (5)$$

2 移动节点的效用模型

移动节点的效用是用来定义移动节点在未来一段时间内的感知能力。令 U_α 表示移动节点在未来一段时间 t_u 内的效用值。移动节点的效用是用来衡量移动节点完成感知任务的一个标准,其直接影响到移动节点在进行任务迁移时与之协作的移动节点的选择,进而影响到收集的感知数据质量的优劣程度。

在感知系统中运行效用计算算法会消耗一定的计算资源。因此,只需要预测移动节点在未来一段有限的时间内的移动轨迹,并非移动节点运行的全过程,所以,本文为了限制效用计算带给感知系统额外的负担,设置了一个时间上限 t_u , 其中移动节点效用估计过程中的轨迹预测以 t_u 为时间上限。同时,感知系统以 t_u 作为系统的更新周期,更新参与移动节点的感知效用值,并根据移动节点的效用值选择出合适的移动节点来收集感知数据,从而用有限的开销收集到较高质量的感知数据。

2.1 移动节点的效用函数

本文用 u_i^α 来描述移动节点 α 在子区域 i 内的节点效用值:

$$u_i^\alpha = \sum_{l=1}^M \sum_{t=0}^{t_u} y_l^\alpha h_{il}^\alpha(t) \quad (6)$$

其中, u_i^α 表示移动节点 α 在未来一段时间 t_u 内,对子区域 i 内所有类型的感知数据所具备的感知能力; $h_{il}^\alpha(t)$ 表示节点 α 在访问子区域 i 的过程中采集 l 型感知数据的有效时间区间。当节点在 t 时刻到达子区域 i 并停留了 τ 时间长度,即 $\xi_i^\alpha = t, \theta_i^\alpha = \tau$, 则:

$$\sum_{t=0}^{t_u} h_{il}^\alpha(t) = \tau + w_l \quad (7)$$

在任意条件下:

$$\sum_{t=0}^{t_u} h_{il}^\alpha(t) = \sum_{t=0}^{t_u} \sum_{\tau=0}^{t_u-1} p(\xi_i^\alpha = t) p(\theta_i^\alpha = \tau) (\tau + w_l) \quad (8)$$

其中, $p\{\xi_i^\alpha = t\}$ 表示节点 α 到达子区域 i 的时间的概率; $p\{\theta_i^\alpha = \tau\}$ 表示节点 α 在子区域 i 停留 τ 时间的概率。

本文将目标感知区域划分为 N 个子区域,那么节点 α 的感知效用可以表示为:

$$U_\alpha = \sum_{i=1}^N u_i^\alpha \quad (9)$$

2.2 移动节点在区间停留时刻的分布

对于任意一个给定的状态,比如 0,其单步转移概率为 p (转出概率为 $1 - p$)。 θ_i^α 为节点 α 在该区域的停留时间,则利用时齐特点可知:

$$p\{\theta_i^\alpha = n\} \\ = p\{x(1) = \dots = x(n-1) = 0, x(n) \neq 0 \mid x(0) = 0\} \quad (10)$$

利用马氏性:

$$\begin{aligned} p\{\theta_i^\alpha = n\} \\ &= p\{x_1 = 0 \mid x_0 = 0\} * p\{x_2 = 0 \mid x_1 = 0\} * \dots \\ &* p\{x_{n-1} = 0 \mid x_{n-2} = 0\} * p\{x_n \neq 0 \mid x(n-1) = 0\} \\ &= \underbrace{p * p * \dots * p}_{n-1} * (1-p) \\ &= (1-p) * p^{n-1}, n \geq 0 \end{aligned} \quad (11)$$

因此,移动节点的区间停留时间服从几何分布。

2.3 移动节点到达时刻分布

定义 1 对于任意给定的 $i, j \in I$, 称随机变量:

$$T_{ij} = \min\{n: x_0 = i, x_n = j, n \geq 1\} \quad (12)$$

为从状态 i 出发首次进入状态 j 的时刻, 简称首次到达时间。

定义 2 对于任意的 $i, j \in I$, 称:

$$f_{ij}^{(n)} = p\{T_{ij} = n \mid x_0 = i\} \quad (13)$$

为节点在 0 时从状态 i 出发, 经 n 步首次到达状态 j 的概率。

定义 3 对于任意给定的 $i, j \in I$, 称:

$$\begin{aligned} f_{ij} &= \sum_{1 \leq n < \infty} f_{ij}^{(n)} \\ &= \sum_{1 \leq n < \infty} p\{T_{ij} = n \mid x_0 = i\} \\ &= p\{T_{ij} < \infty\} \end{aligned} \quad (14)$$

为节点从 0 时从状态 i 出发经过有限步转移后迟早到达状态 j 的概率。

$$p\{T_{ij} = \infty\} = f_{ij}^{(\infty)} = 1 - f_{ij} \quad (15)$$

表示节点从 0 时从状态 i 出发, 经过有限步转移后不能到达状态 j 的概率。

一旦确定了首达概率, 就可以计算节点到达时刻的概率。节点从状态 i 开始最终达到状态 j 的概率可由下式给出:

$$q_{ij}^{(\infty)} = \sum_{n=1}^{\infty} f_{ij}^{(n)} = f_{ij} \quad (16)$$

其中, q_{ij} 表示节点从状态 i 出发最终到达状态 j 的概率。

3 基于效用的感知任务迁移算法

基于效用的感知任务迁移算法是指携带有移动设备的节点, 在随机的移动过程中机会地与其他移动节点相遇后, 将感知任务迁移给效用更高的移动节点。如果没有遇到比自己效用更高的节点, 则该节点将继续携带感知任务。例如, 当移动节点 a, b 随着自身移动并且相遇后, a, b 可以在相遇的时候比较对感知任务 x 其各自的效用, 如果节点 b 对任

务 x 的效用更高的话, 则将感知任务 x 由节点 a 迁移到节点 b 。

下面详细介绍基于效用的感知任务迁移算法, 其伪代码如下:

算法 1 感知任务迁移算法

输入 Sensing tasks Γ ; the set of mobile users Ω ;

data requirement of each task $R^q, q \in \Gamma$;

sensing capability of each mobile user $m \in S_m^q$;

interval between samples Δt ;

transition matrix obtained from historic traces $p(\Delta t)$;

the length of two mobile users $\alpha, \beta, L(\alpha, \beta), \alpha, \beta \in \Omega$;

initial number of selected mobile users N

输出 Selected mobile users as set S

```

1. Set of unselected mobile users  $\varphi \leftarrow \Omega$ , set of selected
mobile users  $S \leftarrow \text{NULL}$ ;
2. The sensing task  $q$ ;
3.  $\text{flag} \leftarrow 0$ ;
4. While  $\text{flag} < N$  do
5.  $m \leftarrow$  random select a mobile user from  $\varphi$ 
6.   If  $R^q \in S_m^q$  then
7.      $S \leftarrow S + \{m\}$ 
8.    $\text{flag} \leftarrow \text{flag} + 1$ 
9. End if
10. End while
11. For each mobile user  $\alpha$  in  $S$  do
12.   Calculate  $U_\alpha$ ;  $M \leftarrow \text{NULL}$ ;  $S' \leftarrow \text{NULL}$ ;
13.   For each mobile user  $\beta$  in  $S$  do
14.     If  $L(\alpha, \beta) \leq 4$  then
15.        $M \leftarrow M + \{\beta\}$ 
16.     End if
17.   End for
18.   For each mobile user  $\beta$  in  $M$  do
19.     If  $R^q \in S_m^q$  then
20.       Calculate  $U_\beta$ 
21.     End if
22.     If  $U_\alpha < U_\beta$  then
23.        $S' \leftarrow S' + \{\beta\}$ 
24.     End if
25.   End for
26.    $S \leftarrow S + S'$ 
27.    $S \leftarrow S - \{\alpha\}$ 
28. End for

```

首先, 初始化所有数据, 感知任务集 Γ , 并且 $\forall q \in \Gamma$. 感知任务需要收集的数据类型为 R^q . 异构群体内所有的移动节点的集合为 Ω , S_m^q 表示节点 m 对任务 q 的感知能力, 根据历史移动轨迹得到转移概率矩阵, 数据采样时间间隔为 Δt , $L(\alpha, \beta)$ 表示节点 α 与 β 之间的距离, 当 $L(\alpha, \beta) \leq 4$ 时认为这 2 个节点相遇。

N 表示感知数据服务中心选择的要完成感知任务的移动节点数量。然后,对所有携带感知任务的节点,当它们在移动的过程中与其他节点相遇后,进行对所携带任务的效用进行比较,如果遇到比自己效用更高的节点,就将该任务迁移给该节点,否则就继续携带感知任务。最后,在系统进行更新时,对所有携带感知任务的节点进行感知数据的采集与传输。

4 实验与结果分析

采用通过 GPS 追踪获得的 T-Drive^[19-20] 文件模拟用户的移动轨迹,该文件包含了 900 个移动节点大约一周时间的移动轨迹,约 1 500 个 GPS 点,其采样间隔为 5 s。通过以下步骤建立仿真平台:

1) 仿真中所有轨迹均分布在北京四环路周围,故将北京四环路周围的矩形感知区域作为目标感知区域。该感知区域的纬度范围是(39.840 018°。N, 39.993 971°。N),经度范围是(116.276 206°。E, 116.494 243°。E)。移动节点的轨迹并没有限制在目标感知区域范围内,假设不在目标感知区域内的子区域的区域号为-1,目标感知区域内的子区域的区域号为0,1,2等。

2) 在模拟多任务场景下的移动群智感知,感知数据服务中心并没有明确生成具体数目的感知任务,不定时向移动节点发送感知任务。为了简化仿真场景而又不失其一般性,假设每个感知任务需要采集感知数据类型小于4个。

3) 移动群智感知系统的感知时间历时大约 533 315 s(大约 6 d)。系统的更新周期为 1 200 s,也就是感知数据每 1 200 s 采集一次。因为感知数据是有有效期的,所以在仿真中将感知数据的生命周期设置为系统周期的 10 倍。

4) T-Drive 文件中记录的 900 个移动节点被认为是备选参与者,即 $|\Omega| = 900$ 。设定感知系统中共有 10 种传感器。每个移动节点的感知能力随机生成,对于每种类型的传感器,移动节点都有 50% 的概率配备有该类型的传感器,即 $y_i^a = \text{rand}(0, 1)$ 。

为验证所提算法 STO 的性能,实验中还另外实现了 2 种参与者选择算法:随机选择算法(RS)和基于多任务的参与者选择算法(DPS)。随机选择算法随机选择参与者直到符合数量需求为止。而 DPS 算法则是根据参与者的转移概率的高低选择最合适的参与者。本文涉及到的 3 种算法均由 Python 脚本编写。

下面主要对多任务场景下的感知任务迁移算法的性能进行评估。本文中的感知数据组成形式为[移动节点,时间,区域,感知数据类型(A,B,C等)]。当前,对感知信息质量的度量标准并不统一,文献[21]提出运用时空覆盖质量和感知信息质量来衡量采集

到的感知数据是否充足和可信。本文用感知数据覆盖率 and 感知任务完成率 2 个标准来衡量算法的性能。

1) 选择的节点数对感知数据的影响

本节主要介绍当选择的节点数不同时对感知数据覆盖率和感知任务完成率的影响。备选参与者总数为 900,分别选出 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60 个移动节点参与完成感知任务。每组数据的最终结果为 5 次数据采样结果取平均值。图 2 给出了当选择的节点数不同时,其感知数据覆盖率的结果。当选择的节点个数比较少时,如 5, 10, 15, 20 个时 DPS 算法和 STO 算法性能相差不大,但是随着选择节点数的增加,STO 算法的感知数据覆盖率迅速上升,当选择的节点数为 60 时,感知数据覆盖率达到最大值 81.8%,高于 DPS 算法和 RS 算法。图 3 描述了节点数不同时感知任务完成率的结果。DPS 算法和 RS 算法的曲线变化比较缓慢,随着选择的节点数目的迅速增多,STO 算法的曲线变化比较大。当选择的节点数为 60 时,感知数据的完成率达到最大值 68.2%,远高于 DPS 算法和 RS 算法的结果。

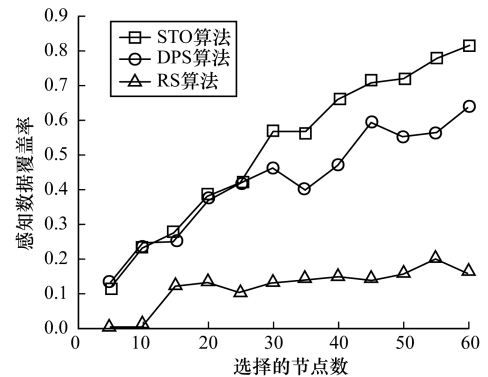


图2 选择节点数对感知数据覆盖率的影响

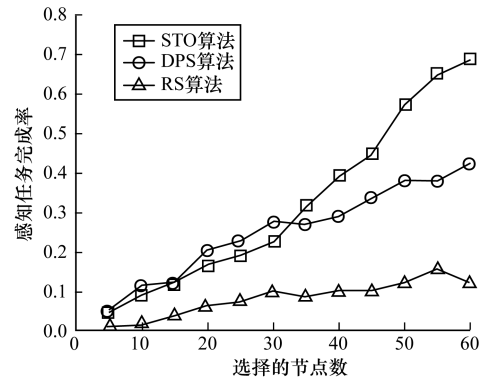


图3 选择节点数对感知任务完成率的影响

产生上述结果的原因一方面是因为 RS 算法的节点选择策略为随机选取,没有考虑到节点的感知能力和移动轨迹对节点采集感知数据的影响。另一方面是 STO 算法采用基于马尔科夫链原理建立的移动模型,这比 DPS 算法中建立的移动模型更准确,

而且在 STO 算法中,移动节点在携带感知任务移动过程中还能通过节点间的本地化协作将感知任务迁移给效用更高的移动节点,有利于感知数据的采集。

2) 备选参与者总数对感知数据的影响

本节主要验证备选参与者总数对感知数据的影响。移动模型中备选参与者的数目分别为 100, 200, 300, 400, 500, 600, 700, 800, 900 条轨迹。对于不同数目的备选参与者,其最后的结果由 5 次实验结果取平均值。图 4 给出了当备选参与者总数不同时感知数据覆盖率的结果。随着备选参与者数目的增多,感知数据覆盖率快速提升,并在 900 时达到最大值 81.8%。当备选参与者数目比较少时,STO 算法和 DPS 算法相差不大,但随着备选参与者总数的增多,STO 算法的优势逐渐体现出来,其性能表现远远高于 DPS 算法和 RS 算法。

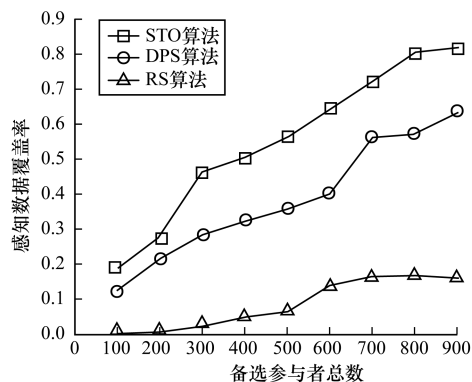


图4 备选参与者总数对感知数据覆盖率的影响

图5描述了备选参与者的数目对感知任务完成率的影响。当备选参与者的数目为100时,3种算法的感知任务完成率都较低,STO算法和DPS算法相差不大,都略优于RS算法。随着备选参与者数目的增多,STO算法和DPS算法的感知任务完成率迅速攀升,RS算法的感知任务完成率曲线变化缓慢。当备选参与者数目为900时,STO算法的感知任务完成率的最大值为68.2%,而DPS算法的感知任务完成率的最大值为52.6%。实验结果表明STO算法的性能优于DPS算法和RS算法。产生这样结果的原因是RS算法选择参与者时采取随机选择的方法,随着备选参与者数目的不断增多,感知任务完成率的变化幅度也不太大。STO算法根据节点的效用选择节点,比DPS算法中根据节点转移概率选择节点更优,更有利于感知任务的完成。此外,STO算法利用了节点间的相互协作,通过比较效用的大小选择出更优的节点来完成感知任务,这意味着将有更多的节点参与到感知任务完成中,备选参与者的数

目越多越有利于找到更多并且效用更高的节点来完成感知任务。

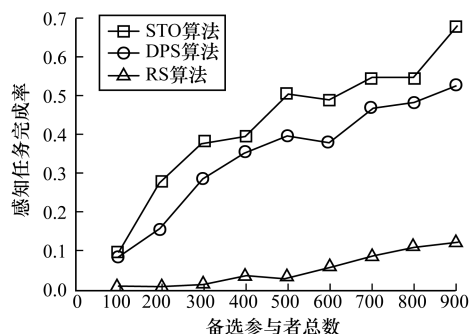


图5 备选参与者总数对感知任务完成率的影响

3) 区域划分对感知数据的影响

本节主要介绍感知区域的划分对感知数据的影响。首先将目标感知区域分别划分为 2×2 , 3×3 , 4×4 , 5×5 , 6×6 , 7×7 , 8×8 , 9×9 , 10×10 个子区域,选择完成感知任务的节点数为60个。图6给出了感知区域划分不同感知数据覆盖率的结果。随着子区域数目的不断增多,感知数据覆盖率曲线呈先升后降的趋势,在 5×5 时达到最大值88.2%,也就是说将目标感知区域划分为 5×5 个子区域,最有利于采集感知数据。图7描述了当目标感知区域划分为不同数目的子区域时,感知任务完成率的结果。将目标感知区域划分为 5×5 时,STO算法的感知任务完成率最高(最大值为85.5%)。当感知区域划分为100个子区域时,感知任务完成率最低(最小值为68.6%)。产生这样的结果的原因一方面是因为当目标感知区域划分的子区域个数比较少时,感知系统对感知数据的要求比较低,感知任务的完成率就较高,反之,感知任务的完成率就比较低。另一方面原因是当目标感知区域的子区域个数不断增多时,对节点的移动模型和节点效用模型都产生很大的影响。

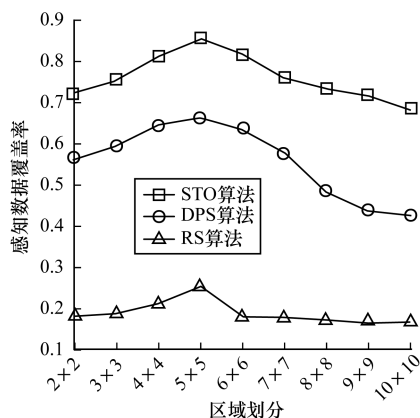


图6 区域划分对感知数据覆盖率的影响

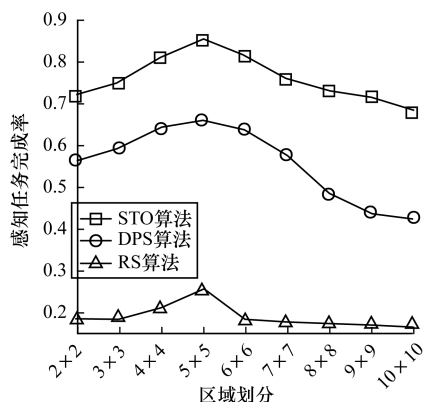


图7 区域划分对感知任务完成率的影响

5 结束语

本文提出基于效用的感知任务迁移方法,对感知任务的执行过程进行优化。从仿真结果中可以看出,利用 STO 算法采集到的感知数据,无论是在感知数据的覆盖率方面,还是在感知任务的完成率方面,都优于移动群智感知系统中的其他方法。但由于效用计算会消耗移动设备的资源,如何能用最小的能源消耗完成感知任务将是下一步要研究的问题。

参考文献

- [1] Ganti R K, Ye F, Lei H. Mobile Crowdsensing: Current State and Future Challenges[J]. IEEE Communications Magazine, 2011, 49(11): 32-39.
- [2] Guo B, Yu Z, Zhou X, et al. From Participatory Sensing to Mobile Crowd Sensing[J]. Computer Science, 2014, 6(4): 593-598.
- [3] Dutta P, Aoki P M, Kumar N, et al. Common Sense: Participatory Urban Sensing Using a Network of Handheld air Quality Monitors[C]//Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems. New York, USA: ACM Press, 2009: 349-350.
- [4] Zhou P, Zheng Y, Li M. How Long to Wait? Predicting Bus Arrival Time With Mobile Phone Based Participatory Sensing[J]. IEEE Transactions on Mobile Computing, 2014, 13(6): 1228-1241.
- [5] Minkman E, Overloop P J V, Sanden M C A V D. Citizen Science in Water Quality Monitoring: Mobile Crowd Sensing for Water Management in the Netherlands[C]//Proceedings of Conference on World Environmental and Water Resources Congress. New York, USA: ACM Press, 2015: 216-225.
- [6] Liu Lubin, Zhu Yanmin. Noise Collection and Presentation System Based on Crowd Sensing [J]. Computer Engineering, 2015, 41(10): 160-164.
- [7] Reddy S, Estrin D, Srivastava M. Recruitment Framework for Participatory Sensing Data Collections [C]//Proceedings of International Conference on Pervasive

- Computing. Helsinki, Finland: [s. n.], 2010: 138-155.
- [8] Tuncay G S, Benincasa G, Helmy A. Autonomous and Distributed Recruitment and Data Collection Framework for Opportunistic Sensing[J]. ACM Sigmobile Mobile Computing & Communications Review, 2012, 16(4): 50-53.
- [9] Duan L, Kubo T, Sugiyama K, et al. Incentive Mechanisms for Smartphone Collaboration in Data Acquisition and Distributed Computing[C]//Proceedings of IEEE INFOCOM'12. Washington D. C., USA: IEEE Press, 2012: 1701-1709.
- [10] Song Z, Liu C H, Wu J, et al. QoI-aware Multitask-oriented Dynamic Participant Selection With Budget Constraints [J]. IEEE Transactions on Vehicular Technology, 2014, 63(9): 4618-4632.
- [11] Zhang D, Wang L, Xiong H, et al. 4W1H in Mobile Crowd Sensing [J]. IEEE Communications Magazine, 2014, 52(8): 42-48.
- [12] Rachuri K K, Efstratiou C, Leontiadis I, et al. Smartphone Sensing Offloading for Efficiently Supporting Social Sensing Applications[J]. Pervasive & Mobile Computing, 2014, 10(2): 3-21.
- [13] 安 键, 桂小林, 何其昌, 等. 群智感知中基于层次分析法的众包机制[J]. 北京邮电大学学报, 2015, 38(5): 37-41.
- [14] Jaimes L G, Chakeri A, Lopez J, et al. A Cooperative Incentive Mechanism for Recurrent Crowd Sensing[C]//Proceedings of IEEE SoutheastCon'15. Washington D. C., USA: IEEE Press, 2015: 125-132.
- [15] Fernando N, Seng W L, Rahayu W. Mobile Cloud Computing: A Survey [J]. Future Generation Computer Systems, 2013, 29(1): 84-106.
- [16] Shi C, Lakafosis V, Ammar M H, et al. Serendipity: Enabling Remote Computing Among Intermittently Connected Mobile Devices [C]//Proceedings of the 13th ACM International Symposium on Mobile Ad Hoc NETWORKING and Computing. New York, USA: ACM Press, 2012: 145-154.
- [17] Ma H, Zhao D, Yuan P. Opportunities in Mobile Crowd Sensing [J]. IEEE Communications Magazine, 2014, 52(8): 29-35.
- [18] Mtibaa A, Fahim A, Harras K A, et al. Towards Resource Sharing in Mobile Device Clouds: Power Balancing Across Mobile Devices [C]//Proceedings of ACM SIGCOMM Workshop on Mobile Cloud Computing. New York, USA: ACM Press, 2013: 51-56.
- [19] Yuan J, Zheng Y, Xie X, et al. Driving with Knowledge from the Physical World [C]//Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. San Diego, USA: ACM Press, 2011: 316-324.
- [20] Yuan J, Zheng Y, Zhang C, et al. T-drive: Driving Directions Based on Taxi Trajectories [C]//Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems. New York, USA: ACM Press, 2015: 99-108.
- [21] 赵 东, 马华东, 刘 亮. 移动群智感知质量度量与保障[J]. 中兴通讯技术, 2015, 21(6): 3-5.