

# 移动无线传感器网络中分布式重聚类算法研究

徐超杰, 俞 晖, 罗汉文

(上海交通大学 电子信息与电气工程学院, 上海 200240)

**摘 要:** 移动无线传感器网络中,节点的移动性影响着层次化聚类之后的网络结构,从而影响聚类内部节点间通信时的数据送达率与能耗.为了降低节点移动性的影响,本文提出了一种分布式重聚类算法.该算法基于已聚类网络,利用粒子滤波算法对节点当前位置进行估计,并结合移动模型预测下一时刻位置;处于聚类边界的非簇头节点周期性地评估自身是否需要重聚类,并在需要时通过与所属聚类及目标聚类的簇头节点通信,将自身重聚类到目标聚类中.仿真结果表明,在重聚类周期较小时,该算法能够使节点在移动过程中保持合理的通信距离,并在数据送达率与能耗方面优于现有的算法.

**关键词:** 移动无线传感器网络; 聚类; 分布式; 重聚类; 数据送达率; 能耗

**中图分类号:** TN 929.5   **文献标志码:** A   **文章编号:** 1000-5137(2016)02-0202-07

## 0 概 述

移动无线传感器网络由大量移动节点组成,这些节点被布置于目标区域中以完成诸如目标跟踪与环境条件监测等任务.节点将采集到的数据发送至汇聚节点或者服务器进行处理,由于节点能量有限,为了延长网络的生命周期,必须对节点进行高效利用<sup>[1-2]</sup>.

在无线传感器网络中,层次化聚类方法有利于降低网络能耗与提高数据发送效率等.文献[3]提出的 LEACH 是一种自适应的聚类算法,一些节点被随机地选为簇头,其他节点加入到距其最近的簇头节点形成聚类,该算法周期性地对簇头节点进行轮转以平衡节点之间能量消耗差异.文献[4]提出了一种分布式聚类算法,通过迭代过程实现聚类与簇头节点的选取,该算法根据节点的剩余能量对簇头节点进行轮转.考虑到节点的移动性,文献[5]提出了 LEACH-mobile 算法,该算法要求移动节点在移动时对所属聚类进行声明.

在已聚类的移动无线传感器网络中,节点的移动性随机、动态地影响着网络的拓扑结构,使得节点未处于合适的聚类中,非簇头节点与簇头节点之间的通信距离增加,导致通信能耗增加、数据送达率降低<sup>[6]</sup>,降低了网络的服务质量.因此需要重聚类过程,使节点加入到合适的聚类中.在已有的聚类协议中,重聚类算法主要分为集中式重聚类算法与分布式重聚类算法.在集中式重聚类算法中,重聚类过程由簇头节点或汇聚节点控制,非簇头节点处于从属地位,并且重聚类主要目的是轮转簇头节点,非簇头节点仅在簇头节点轮转时才能被重聚类.在分布式重聚类算法中,非簇头节点可以评估自身是否需要重聚类,并在合适的时机进行重聚类.文献[7]提出了一种集中式重聚类算法,将簇头节点与汇聚节点控制的重聚类过程进行结合.文献[8]提出的 self-incentive and semi-reclustering(SISR)是一种分布式重聚类算法,该算法主要处理簇头节点轮转时导致的非簇头节点重聚类问题,允许聚类边界区域的非簇头节点在必要时重聚类到更合适的聚类中.但是,SISR 中重聚类过程仅在目标聚类下一次簇头轮转开始时

收稿日期: 2016-03-04

通信作者: 俞 晖,中国上海市闵行区东川路 800 号,上海交通大学电子信息与电气工程学院,邮编:200240, E-mail: yuhui@sjtu.edu.cn.

才能进行, 导致较长的重聚类周期. 在大多数聚类协议中, 节点的位置主要通过 GPS 或者其他定位方式获得, 这将引入较大的能耗, 同时也使得这些协议在 GPS 等定位方式无法使用的场景中失效.

本文作者提出了一种分布式重聚类算法, 该算法基于已聚类的移动无线传感器网络, 利用粒子滤波算法结合惯性传感器数据实现对节点当前时刻位置的估计, 并根据节点移动模型预测节点下一时刻的位置; 该算法允许处于聚类边界区域的非簇头节点对自身是否需要重聚类进行评估, 并在必要时通过与所属聚类及目标聚类的簇头节点进行通信, 自主地完成重聚类过程.

## 1 系统模型

移动无线传感器网络由移动节点组成, 初始时这些节点被随机布置在目标区域中且初始位置已知, 之后节点不能通过 GPS 或者其他类似定位方式获取位置. 在运动过程中, 每个节点都可以获得自身的运动速度与方向. 本节主要介绍节点的移动模型以及节点之间通信时的能耗模型.

### 1.1 节点移动模型

假设节点的运动遵从特定的移动模型. Gauss-Markov 移动模型中<sup>[9]</sup>, 假定每个时间间隔中运动情况不变, 时间间隔  $k$  (时刻  $k$  与时刻  $k+1$  之间) 的运动情况与时间间隔  $k-1$  中运动情况相关, 并能够通过改变随机度参数获得不同随机程度的移动模型, 能够很好地模拟节点的运动情况. 时间间隔  $k$  中运动速度与方向的更新规则为:

$$\begin{aligned} v_k &= \alpha v_{k-1} + (1 - \alpha) \bar{v} + \sqrt{(1 - \alpha^2)} v_{x_{k-1}}, \\ d_k &= \alpha d_{k-1} + (1 - \alpha) \bar{d} + \sqrt{(1 - \alpha^2)} d_{x_{k-1}}, \end{aligned} \quad (1)$$

其中,  $v_{k-1}$  与  $d_{k-1}$  为时间间隔  $k-1$  中节点的运动速度与方向;  $\bar{v}$  与  $\bar{d}$  为常量, 即当  $k \rightarrow \infty$  时平均运动速度与方向;  $v_{x_{k-1}}$  与  $d_{x_{k-1}}$  为高斯随机变量;  $\alpha$  为随机度参数.

初始时, 每个节点都具有初始速度  $v_0$  与方向  $d_0$ ; 在时刻  $k$ , 节点位置更新规则为:

$$\begin{aligned} x_k &= x_{k-1} + v_{k-1} t \cos d_{k-1}, \\ y_k &= y_{k-1} + v_{k-1} t \sin d_{k-1}, \end{aligned} \quad (2)$$

其中,  $(x_{k-1}, y_{k-1})$  为节点在时刻  $k-1$  时的位置,  $t$  为每个时间间隔的固定长度.

为了获得更加真实的移动模型, 对 Gauss-Markov 移动模型进行了限定, 即限定节点运动速度在一定范围内变化, 相邻时间间隔内节点运动方向变化也限定在一定范围内.

### 1.2 能耗模型

主要考虑节点之间通信时的能耗, 根据发送端与接收端之间的距离, 分别采用自由空间与多径衰落信道模型. 发送端将  $l$  比特数据发送到距离其  $d$  处接收端的能耗计算为:

$$E_{Tx}(l, d) = \begin{cases} l \varepsilon_{elec} + l \varepsilon_{fs} d^2 & d < d_0 \\ l \varepsilon_{elec} + l \varepsilon_{mp} d^4 & d \geq d_0 \end{cases}. \quad (3)$$

其中,  $\varepsilon_{elec}$  为发送端发送每比特数据所消耗能量;  $\varepsilon_{fs}$  为自由空间信道模型下放大每比特数据所消耗能量,  $\varepsilon_{mp}$  为多径衰落信道模型下对应消耗能量;  $d_0$  为临界距离值, 其计算为:

$$d_0 = \sqrt{\varepsilon_{fs} / \varepsilon_{mp}}. \quad (4)$$

接收端接收  $l$  比特数据所消耗能量计算为:

$$E_{Rx}(l) = l \varepsilon_{elec}. \quad (5)$$

## 2 分布式重聚类算法

本文作者提出的分布式重聚类算法基于已层次化聚类的移动无线传感器网络, 主要包括节点利用粒子滤波算法<sup>[10]</sup>对当前自身位置进行估计, 节点根据移动模型预测下一时刻自身位置, 处于聚类边界区域的非簇头节点通过与所属聚类的簇头节点通信, 对自身是否需要重聚类进行估计, 并在需要重聚类

时,通过与所属聚类及目标聚类的簇头节点进行通信,将自身重聚类到目标聚类中.

### 2.1 节点位置估计

由于节点不能通过 GPS 或者其他定位方式来获取自身的位置. 采用粒子滤波算法对节点在每个时刻的位置进行估计. 定义节点在时刻  $k$  的状态为:

$$\mathbf{s}_k = \begin{bmatrix} \mathbf{p}_k \\ \vartheta_k \end{bmatrix} = \begin{bmatrix} x_k \\ y_k \\ \vartheta_k \end{bmatrix}, \quad (6)$$

其中,  $\mathbf{p}_k = (x_k, y_k)$  与  $\vartheta_k$  分别为节点在时刻  $k$  的位置与运动方向. 根据航迹推算法,  $\mathbf{s}_k$  的更新模型为:

$$\mathbf{s}_k = \mathbf{s}_{k-1} + v_{k-1} t \begin{bmatrix} \cos \vartheta_{k-1} \\ \sin \vartheta_{k-1} \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \Delta \vartheta_k \end{bmatrix}, \quad (7)$$

其中,  $v_{k-1}$  与  $\vartheta_{k-1}$  分别为节点在时间间隔  $k-1$  的运动速度与方向;  $\Delta \vartheta_k$  为从时间间隔  $k-1$  到时间间隔  $k$  过程中节点运动方向的变化;  $t$  为每个时间间隔的固定值.

在粒子滤波算法中,节点在时刻  $k$  的状态  $\mathbf{s}_k$  的先验概率分布  $p(\mathbf{s}_k)$  可由许多随机分布在节点位置周围的粒子  $s_k^i$  及其概率分布来估计,即:

$$p(\mathbf{s}_k) \approx \sum_{i=1}^{N_k} \omega_k^i \delta(\mathbf{s}_k - s_k^i), \quad (8)$$

其中,  $N_k$  为粒子数目;  $\omega_k^i$  为粒子  $s_k^i$  的归一化权重值;  $\delta(\cdot)$  为狄拉克函数.

初始时,根据每个节点的初始位置生成  $N_0$  个粒子,其归一化权重均为  $\omega_0^i = 1/N_0$ . 在时间间隔  $k-1$  中,节点获得自身的运动速度与方向后,节点根据时刻  $k-1$  自身的状态  $\mathbf{s}_{k-1}$ ,利用公式(7)计算得到时刻  $k$  的状态观察值  $\tilde{\mathbf{s}}_k$ ,每个粒子的状态也将根据公式(7)进行更新. 时间间隔  $k-1$  中粒子滤波算法的迭代过程主要包括如下过程:

(1) 更新阶段:时刻  $k-1$  到时刻  $k$  粒子的更新过程中,每个粒子  $s_{k-1}^i$  发展出  $N_{\text{evolve}}$  个新粒子来代替其自身. 在时刻  $k$ ,新粒子  $s_k^i$  的非归一化权重值  $\eta_k^i$  与归一化权重值  $\omega_k^i$  分别为:

$$\eta_k^i = \omega_{k-1}^i \cdot N(s_k^i | \mu_k, \delta_k) \cdot \frac{1}{N_{\text{evolve}}}, \quad (9)$$

$$\omega_k^i = \frac{\eta_k^i}{\sum_{i=1}^{N_k} \eta_k^i}, \quad (10)$$

其中,  $N(\cdot)$  为新粒子  $s_k^i$  产生的概率,  $s_k^i$  服从高斯分布,  $\mu_k$  与  $\delta_k$  用于近似该高斯分布的均值与方差,其计算分别为:

$$\mu_k = \frac{\sum_{i=1}^{N_k} s_k^i}{N_k}, \quad (11)$$

$$\delta_k = \sum_{j=1}^{N_k} \omega_{k-1}^j (s_k^j - \mu_k) (s_k^j - \mu_k)^T. \quad (12)$$

其中,  $N_k$  为时刻  $k$  的粒子数目;  $\omega_{k-1}^i$  为粒子  $s_{k-1}^i$  的归一化权重值.

(2) 粒子有效数目阈值计算:为了保持粒子状态具有收敛性,每个时刻都对粒子进行筛选. 定义粒子有效数目阈值为  $N_k^{\text{eff-thr}}$ ,其计算过程包括:(1) 由节点在时刻  $k$  的状态观察值  $\tilde{\mathbf{s}}_k$ ,计算该时刻每个粒子与  $\tilde{\mathbf{s}}_k$  之间的欧氏距离,并将这些距离值排序得到升序集合  $\mathbf{S}_d$ ;(2) 计算  $\mathbf{S}_d$  中元素平均值  $\bar{s}$ ,并将  $\mathbf{S}_d$  中大于  $\bar{s}$  的元素值放入集合  $\mathbf{S}_d^{\text{up}}$ ,其余元素组成集合  $\mathbf{S}_d^{\text{low}}$ ,二者中元素顺序与  $\mathbf{S}_d$  中一致;(3) 计算  $\mathbf{S}_d^{\text{low}}$  中全部元素与  $\mathbf{S}_d^{\text{up}}$  中前  $i$  个元素的平均值  $S_d^{\text{low}+i}$ ,以及  $\mathbf{S}_d^{\text{up}}$  中剩余元素的平均值  $S_d^{\text{up}-i}$ ;(4) 对  $i$  与  $N_k^{\text{eff-thr}}$  的

计算为:

$$i = \arg \max_i \left( \frac{S_d^{\text{up}-i} - S_d^{\text{low}+i}}{s} \right), \quad (13)$$

$$N_k^{\text{eff-thr}} = N_d^{\text{low}} + i. \quad (14)$$

其中,  $N_d^{\text{low}}$  为  $S_d^{\text{low}}$  中元素个数.

(3) 重采样阶段: 为了避免所有的权重值集中于某些粒子而导致目标空间不能很好覆盖, 对具有较低权重值的粒子进行重采样. 在时刻  $k$ , 定义粒子的有效数目为:

$$N_k^{\text{eff}} = \frac{1}{\sum_{i=1}^{N_k} (\omega_k^i)^2}. \quad (15)$$

若  $N_k^{\text{eff}}$  大于  $N_k^{\text{eff-thr}}$ , 则将  $N_k^{\text{eff}} - N_k^{\text{eff-thr}}$  个权重值最低的粒子滤除, 并用具有最高权重值的同等数量粒子进行替代, 粒子的权重值通过公式(10)进行归一化. 在该过程中, 引入了服从学生氏  $t$  分布的噪声以防止迭代过程收敛于少量粒子.

(4) 预测阶段: 在时刻  $k$ , 节点的状态为

$$s_k = \sum_{i=1}^{N_k} \omega_k^i \cdot s_k^i. \quad (16)$$

## 2.2 节点位置预测

结合节点在时间间隔  $k-1$  中的运动情况, 节点移动模型中对节点运动速度及相邻时间间隔中运动方向变化的限定, 节点在时间间隔  $k$  的运动速度范围与运动方向变化范围可由公式(1)计算得到, 分别表示为  $V_{\text{range}} = [v_{\text{min}}, v_{\text{max}}]$  与  $D_{\text{range}} = [-\theta_1, \theta_2]$ . 因此在时刻  $k+1$  节点的可能位置被限定为图1中扇形阴影区域.

$V_{\text{range}}$  中每个可能的速度值  $v_i$  在公式(1)中均有对应的高斯随机变量  $v_{x_{k-1}}$ ,  $D_{\text{range}}$  中每个可能的运动方向变化值  $\theta_j$  在公式(1)中也存在对应的高斯随机变量  $d_{x_{k-1}}$ , 因此  $v_i$  与  $\theta_j$  的概率值分别等于对应  $v_{x_{k-1}}$  与  $d_{x_{k-1}}$  出现的概率值. 由于  $v_{x_{k-1}}$  与  $d_{x_{k-1}}$  相互独立,  $v_i$  与  $\theta_j$  对应的节点可能位置  $s_{ij}$  出现的概率为  $p(s_{ij}) = p(v_i) \cdot p(\theta_j)$ . 将  $V_{\text{range}}$  与  $D_{\text{range}}$  中速度值与方向变化值进行离散化, 计算每一对  $v_i$  与  $\theta_j$  对应可能位置  $s_{ij}$

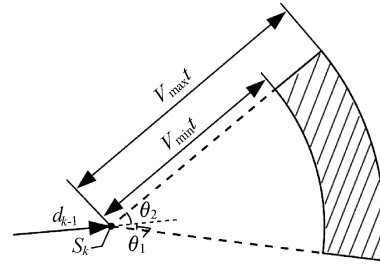


图1 节点位置预测中可能位置示意图

的集合  $S$  以及对应的概率值  $p(s_{ij})$  的集合  $P$ . 从  $S$  中随机选取  $N_{\text{rand}}$  个可能位置, 这些可能位置对应的概率值集合为  $P_{\text{rand}}$ , 将  $P_{\text{rand}}$  中概率值排序, 并选择概率值最高的  $N_{\text{sel}}$  个概率值为集合  $P_{\text{sel}}$ , 对应的可能位置组成集合  $S_{\text{sel}}$ . 归一化集合  $P_{\text{sel}}$  中的概率值得到集合  $P_{\text{norm}}$ , 则根据  $S_{\text{sel}}$  中节点可能位置及其对应的归一化概率值集合  $P_{\text{norm}}$  可以计算节点在时刻  $k+1$  的预测位置:

$$\hat{s}_{k+1} = \sum_{i=1}^{N_{\text{sel}}} p_i s_i, \quad p_i \in P_{\text{norm}}, s_i \in S_{\text{sel}}. \quad (17)$$

## 2.3 分布式重聚类过程

每个重聚类周期开始时, 非簇头节点将当前时刻自身的估计位置与下一时刻的预测位置发送给所属聚类的簇头节点, 簇头节点整合聚类内所有节点的位置数据后与邻近聚类的簇头节点交换位置数据, 并将邻近聚类及本聚类中节点的估计位置信息发送给聚类中处于边界区域的非簇头节点(简称边界节点), 边界节点评估自身是否需要进行重聚类即计算对所属聚类及各个邻近聚类之间的从属度. 节点  $i$  对聚类  $C$  的从属度计算定义为:

$$\psi(i, C) = \frac{1}{\sum_{j \in C} [d(i, j)]^2}, \quad (18)$$

其中,  $d(i, j)$  为节点  $i$  与  $j$  之间的欧式距离. 对于节点  $i$ , 具有较大从属度的聚类更适合其加入. 边界节点

将具有最高从属度的聚类作为当前时刻最适合其加入的聚类  $C_{\text{now-opt}}$ , 若  $C_{\text{now-opt}}$  与当前所属聚类  $C_{\text{now}}$  不同, 则说明当前时刻该边界节点需要进行重聚类.

需要进行重聚类的边界节点首先向  $C_{\text{now}}$  的簇头节点发送重聚类请求信息,  $C_{\text{now}}$  的簇头节点将  $C_{\text{now}}$  中重聚类请求情况与邻近聚类的簇头节点进行交换, 并将  $C_{\text{now}}$  与邻近聚类中节点的预测位置及重聚类请求情况发送给请求重聚类的边界节点. 该边界节点再次进行重聚类评估, 得到下一时刻最适合其加入的聚类  $C_{\text{next-opt}}$ , 若  $C_{\text{next-opt}}$  与  $C_{\text{now}}$  不同, 即重聚类过程中不存在乒乓效应, 该边界节点能够重聚类到聚类  $C_{\text{now-opt}}$  中, 否则当前重聚类过程不进行. 若该边界节点能够进行重聚类, 则向  $C_{\text{now}}$  的簇头节点发送重聚类确认信息, 并向  $C_{\text{now-opt}}$  的簇头节点发送请求加入信息, 在下一时刻脱离聚类  $C_{\text{now}}$  并加入聚类  $C_{\text{now-opt}}$ . 另外, 本文作者提出的分布式重聚类算法中实行簇头节点轮转机制, 轮转准则为节点的剩余能量以及节点在聚类中所处的位置.

### 3 仿真结果与分析

本文作者提出了一种移动无线传感器网络中的分布式重聚类算法, 为了评价该算法的效果, 本文作者利用 Matlab 平台进行了仿真实验. 在本节中, 该分布式重聚类算法简称为 DRC 算法. 仿真实验中主要仿真参数设置如表 1 所示.

表 1 主要仿真参数设置

参数	值	参数	值
目标区域	300 m × 300 m	时间间隔长度	1 s
节点数目	300	随机度 $\alpha$	0.75
节点运动速度范围	4 ~ 6 km/h	相邻时间间隔运动方向变化范围	$[-\pi/6, \pi/6]$
节点初始能量	2J	$\epsilon_{\text{mp}}$	0.0013 $\mu\text{J}/\text{bit}/\text{m}^4$
$\epsilon_{\text{elec}}$	50 nJ/bit	$\epsilon_{\text{elec}}$	10 $\mu\text{J}/\text{bit}/\text{m}^2$

本文作者对 DRC 算法在不同重聚类周期  $T$  (单位: 时间间隔) 条件下的性能进行了评估, 并与文献 [8] 中提出的 SISR 算法进行了对比, 每次仿真实验时长为 1000 个时间间隔, 非簇头节点在每个时刻向所属聚类的簇头节点发送长度为 1000 比特的采集数据.

在已聚类的移动无线传感器网络中分别应用 DRC 算法 ( $T=1, 2, 4, 8, 16$ ) 与 SISR 算法, 统计每个时刻非簇头节点与对应的簇头节点之间的平均通信距离, 图 2 为节点之间平均通信距离的累积分布函数曲线对比. 从图 2 中可以看出,  $T=1$  时 DRC 算法对应的节点之间平均通信距离值最小, 随着  $T$  增加, 节点之间平均通信距离增大; SISR 算法相较于 DRC 算法 ( $T=1, 2, 4, 8, 16$ ), 由于对节点的重聚类处理不及时, 导致节点之间平均通信距离较大.

在仿真实验中, 利用公式 (3) 与公式 (5) 计算节点通信造成的能耗, 并根据节点之间的通信距离对数据送达率<sup>[6]</sup>进行计算. DRC 算法 ( $T=1, 2, 4, 8, 16$ ) 与 SISR 算法关于平均数据送达率的累积分布函数曲线对比如图 3 所示. 结合图 2 与图 3 可以看出, 随着  $T$  增加, 节点之间平均数据送达率降低, 节点之间通信距离的增加导致了节点之间数据送达率的降低. 相比 SISR 算法, DRC 算法 ( $T=1, 2, 4, 8, 16$ ) 能够使节点之间通信时保持更高的数据送达率.

图 4 为 DRC 算法 ( $T=1, 2, 4, 8, 16$ ) 与 SISR 算法对应的节点平均剩余能量变化曲线对比. 从图 4 中

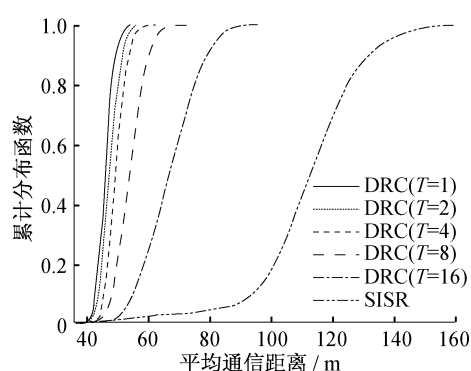


图 2 节点之间平均通信距离累积分布函数

可以看出,随着  $T$  从 1 增大到 8, DRC 算法对应的节点剩余能量变化速度逐渐减缓,说明随着  $T$  增大,重聚类通信次数减少,其引起的能耗降低;然而在  $T = 16$  时,节点剩余能量变化速度与  $T = 4$  时相当,说明节点之间通信距离较大导致了节点之间单次通信耗能增加;而 SISR 算法由于节点之间通信距离过大,导致其对应的节点通信能耗较大. 因此相比 SISR 算法, DRC 算法 ( $T = 1, 2, 4, 8, 16$ ) 能耗较低.

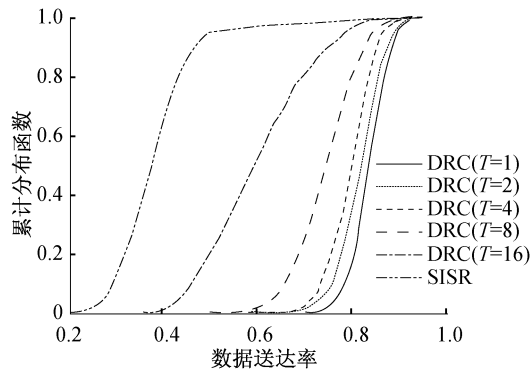


图3 数据送达率累积分布函数曲线对比

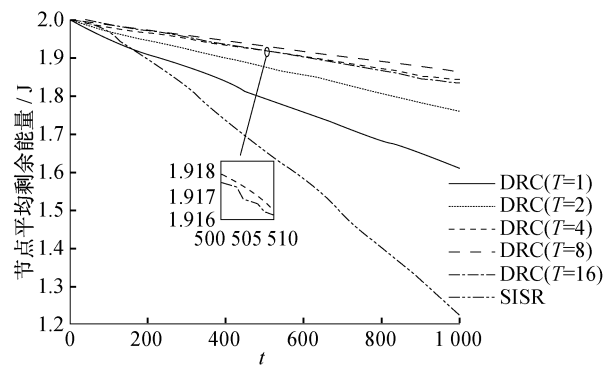


图4 节点平均剩余能量变化曲线对比图

通过仿真实验可以看出,在  $T$  取值较小(如  $T = 1, 2, 4, 8, 16$ )时,即对节点重聚类请求的处理较为及时的情况下, DRC 算法在数据送达率及能耗方面的性能优于 SISR 算法.

## 4 结束语

为了解决移动无线传感器网络中节点移动性引起的重聚类问题,本文作者提出了一种分布式重聚类算法. 该算法基于已聚类的移动无线传感器网络,利用粒子滤波算法结合惯性传感器数据实现对节点当前时刻位置的估计,并根据节点移动模型预测下一时刻节点的位置;该算法允许处于聚类边界区域的非簇头节点对自身是否需要重聚类进行评估,并在必要时自主地完成重聚类过程;该算法还考虑了节点重聚类过程中的乒乓效应. 仿真结果表明,在重聚类周期较小时,该算法在数据送达率与能耗方面的性能优于现有算法.

## 参考文献:

- [ 1 ] Sayyed A, Becker L B. A survey on data collection in mobile wireless sensor networks (MWSNs) [ M ] // Koubaa A, Dios J R M. Cooperative Robots and Sensor Networks. Berlin: Springer, 2015.
- [ 2 ] Zhao M, Yanf Y, Wang C. Mobile data gathering with load balanced clustering and dual data uploading in wireless sensor networks [ J ]. IEEE Transaction on Mobile Computing, 2015, 14(4): 770 - 785.
- [ 3 ] Heinzelman W R, Chandrakasan A, Balakrishnan H. Energy-efficient communication protocol for wireless microsensor networks [ C ] // IEEE. IEEE the 33rd annual Hawaii international conference on System sciences. Hawaii: IEEE, 2000.
- [ 4 ] Younis O, Fahmy S. Heed: A hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks [ J ]. IEEE Transaction on Mobile Computing, 2004, 3(4): 366 - 379.
- [ 5 ] Kim D S, Chung G Y J. Self-organization routing protocol supporting mobile nodes for wire-less sensor network [ C ] // IEEE. IEEE First International Multi-Symposiums on Computer and Computational Sciences. Hangzhou: IEEE, 2006.
- [ 6 ] Zubiga M, Krishnamachari B. Analyzing the transitional region in low power wireless links [ C ] // IEEE. IEEE First Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks. Santa Clara: IEEE, 2004: 517 - 526.
- [ 7 ] Guanathillake A, Samarasinghe K. Energy efficient clustering algorithm with global & local re-clustering for wireless sensor

- networks [J]. World Academy of Science, Engineering and Technology, 2013, 7(7): 45 – 52.
- [ 8 ] Baek J, An S K, Fisher P. Dynamic cluster header selection and conditional re-clustering for wireless sensor networks [J]. IEEE Transaction On Consumer Electronics, 2010, 56(4): 2249 – 2257.
- [ 9 ] Mwila M K, Djouani K, Kurien A. An efficient approach to node localisation and tracking in wireless sensor networks [C]//IEEE. IEEE Global Communications Conference ( GLOBECOM). Austin; IEEE, 2014.
- [ 10 ] Jing W, Zhao H, Lin X, et al. Selectively iterative particle filtering and its applications for target tracking in WSNs [C]//IEEE. IEEE Global Communications Conference ( GLOBECOM). Atlanta; IEEE, 2013.

## Study on distributed re-clustering algorithm for mobile wireless sensor networks

XU Chaojie, YU Hui, LUO Hanwen

(School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China)

**Abstract:** In mobile wireless sensor networks, node mobility influences the topology of the hierarchically clustered network, thus affects packet delivery ratio and energy consumption of communications in clusters. To reduce the influence of node mobility, a distributed re-clustering algorithm is proposed in this paper. In this algorithm, basing on the clustered network, nodes estimate their current locations with particle algorithm and predict the most possible locations of next time basing on the mobility model. Each boundary node of a cluster periodically estimates the need for re-clustering and re-cluster itself to the optimal cluster through communicating with the cluster headers when needed. The simulation results indicate that, with small re-clustering periods, the proposed algorithm can be effective to keep appropriate communication distance and outperforms existing schemes on packet delivery ratio and energy consumption.

**Key words:** mobile wireless sensor networks; cluster; distributed; re-clustering; packet delivery ratio; energy consumption

(责任编辑:顾浩然)