

## 基于分组的分布式节点调度覆盖算法

李小龙<sup>1</sup> 林亚平<sup>1,2</sup> 胡玉鹏<sup>1</sup> 刘永和<sup>3</sup>

<sup>1</sup>(湖南大学计算机与通信学院 长沙 410082)

<sup>2</sup>(湖南大学软件学院 长沙 410082)

<sup>3</sup>(德州大学阿灵顿分校计算机科学与工程系 阿灵顿 76019)

(xiaolong-lee@163.com)

### A Subset-Based Coverage-Preserving Distributed Scheduling Algorithm

Li Xiaolong<sup>1</sup>, Lin Yaping<sup>1,2</sup>, Hu Yupeng<sup>1</sup>, and Liu Yonghe<sup>3</sup>

<sup>1</sup>(College of Computer and Communication, Hunan University, Changsha 410082)

<sup>2</sup>(College of Software, Hunan University, Changsha 410082)

<sup>3</sup>(Department of Computer Science and Engineering, University of Texas at Arlington, Arlington, USA, 76019)

**Abstract** Recent research has found that management of node duty circles, termed sensor scheduling, is able to save significant energy and prolong the lifetime of sensor networks. Sensor scheduling schemes could be categorized into two types in terms of scheduling features: round-based scheduling scheme and subset-based scheduling scheme respectively. Compared with round-based scheduling scheme, subset-based scheduling scheme is an effective solution because it is only carried out once after deployment. When sensor nodes are randomly deployed to a target region, existing subset-based scheduling algorithms can not guarantee that any subset of sensor nodes is uniformly distributed over the target region. Motivated by these reasons, firstly a maximum similarity distribution model is set up and an approximation algorithm, the subset-based coverage-preserving distributed scheduling algorithm is proposed. The algorithm is robust to clock asynchrony of nodes in different clusters, hence it is applicable to large-scale sensor networks, for which precise time synchronization is very hard. In addition, the analytical results for the theoretical upper bound of average coverage rate are presented, while nodes are randomly distributed over the target region. The experimental simulations demonstrate that this algorithm has the ability that sensor nodes in each subset are rather uniformly distributed over the target area, and available coverage rate approaches the upper bound.

**Key words** sensor network; coverage; node scheduling; subset; maximum similarity distribution

**摘要** 当部署的传感器节点服从随机分布时, 现有的基于分组的节点调度算法不能保证各个组内的节点均匀分布在目标区域。基于以上原因, 建立了极大相似分布模型, 并提出了极大相似分布的一种近似求解算法: 基于分组的分布式节点调度覆盖算法。算法仅需要簇内的节点维持时钟同步, 簇之间节点的时钟异步对于覆盖效果的影响可以忽略, 因此适用于难以维持整个网络保持时间同步的大型传感器网络。此外, 给出了在节点随机分布的条件下, 采用分组调度时平均覆盖率的理论上界值。仿真实验表明, 提出的这种算法能使各个组内的传感器节点较为均匀地分布在目标区域, 获得的平均覆盖率接近于上界值。

**关键词** 传感器网络; 覆盖; 节点调度; 分组; 极大相似分布

中图法分类号 TP393

无线传感器网络是由大量计算、通信及存储能力有限的传感器节点组成的特殊 Ad-Hoc 网络,在军事和民用领域具有广泛的应用,如战场监视、环境和交通监测、灾难救助等,是目前一个非常活跃的研究领域<sup>[1]</sup>. 传感器节点体积较小,节点提供的能量有限,且能量难以补充. 如何节约节点能量、最大化网络生命周期是传感器网络研究中的中心问题<sup>[2]</sup>. 由于传感器节点密度高,如果每个节点都处于工作状态会造成信息冗余,增加网络能量消耗,降低网络通信效率. 因此,在满足任意时刻处于活动状态下的节点能够对目标区域进行有效监控的条件下,可把所有节点划分为合适的节点子集,这些节点子集采用轮换活跃/休眠的节点调度方式,达到减少能量消耗和延长网络生存时间的目的. 节点调度算法根据调度的特点可分为两种:基于轮次的调度算法<sup>[3-6]</sup>和基于分组的调度算法<sup>[7]</sup>. 基于轮次的调度算法要求传感器节点在每一轮的开始执行一次算法,按照某种竞争机制从所有节点中选择若干个节点作为活动节点,这种算法在传感器网络的生存时间内执行了多次. 而基于分组的调度算法在传感器节点部署后仅执行一次,通过分组将所有传感器节点划分到若干个组内,在算法完成之后,依次调度每一组的传感器节点作为活动节点.

文献[8]提出了一个基于节点状态调度的分布式覆盖算法. 每个节点根据邻近区域内的节点信息,判断自身传感任务是否可由邻近节点完成. 如果可替代,节点广播状态通告消息之后进入休眠状态,需要继续工作的节点执行传感任务. 文献[3]提出了一个基于监测的分布式密度控制算法 PEAS. 在任意时刻,一部分传感器节点处于活动状态,而其他的传感器节点处于休眠状态. 处于休眠状态的传感器节点定时苏醒,检查在它的邻近区域是否有传感器节点处于活动状态. 如果有则继续休眠. 否则它从休眠状态转换到活动状态. 由于 PEAS 算法中的某些节点可能持续工作,导致过早死亡,造成网络中的节点能耗不均匀,影响覆盖质量. 文献[4]对文献[3]进行了改进,节点根据能量大小竞选活动节点,实现了节点能耗均衡. 随机调度覆盖算法<sup>[7]</sup>是一种基于分组的节点调度覆盖算法,它的基本思想是每个传感器节点随机赋予  $1 \sim K$  的某个值  $i$  ( $K, i \in N^*$ ),并将自身分配到第  $i$  组. 在执行完算法之后,依次调度每一组的传感器节点对目标区域进行监测. 基于分组的随机调度算法与基于轮次的调度算法相比,具有能耗小(算法在传感器网络的生存时

间内仅执行了一次)和各节点间能量消耗均匀等优点. 但是由于位置非常临近的传感器节点可能分配到同一组或者是少数几组,导致目标区域内节点分布不均匀,目标区域中某些小区域节点分布过于密集,增加了该区域的通信冲突和数据冗余. 而有些小区域节点分布过于稀疏,导致传感器节点不能对目标区域进行有效监控,并且降低了网络的连通性. 此外,目前大多数的调度覆盖算法<sup>[3-6]</sup>均要求整个网络保持严格的时间同步,而在实际应用中,对于大型的传感器网络,维持整个网络的时间同步是极其困难的,且时间同步需要额外的通信开销和能量消耗.

基于以上原因,本文从使位置非常临近的传感器节点平均分配到不同组内这一目的考虑,建立了极大相似分布模型,证明了该问题属于 NP-难问题,并提出了极大相似分布的一种近似求解算法:基于分组的集中式节点调度覆盖算法. 由于集中式算法不具有好的扩展性,难以适用于大型的传感器网络,为此,本文进一步提出了改进的近似求解算法:基于分组的分布式节点调度覆盖算法. 这种算法对节点之间的时间同步要求较低:当节点密度较大时,算法仅需要簇内的节点维持时间同步,对于不同的簇,节点之间的时间异步对于算法的性能没有影响,因此适用于大型传感器网络. 此外,本文给出了节点服从随机分布的条件下,采用分组调度时平均覆盖率的理论上界值. 仿真实验表明,本文提出的这种算法能使各个组内的传感器节点较为均匀地分布在目标区域,获得的平均覆盖率接近于上界值.

## 1 问题描述及分析

### 1.1 网络模型

传感器节点往往随机分布在目标区域,即节点在部署之前,预先不知道自己在目标区域内的位置. 现假设传感器网络具有如下性质:

- 1) 传感器节点部署密集;
- 2) 传感器节点的感知模型为圆形区域感知模型,且节点的传感半径均为  $r$ ;
- 3) 所有节点部署后不再移动;
- 4) 传感器网络并不是层次结构,也就是说,传感器网络是平面的;
- 5) 部署后节点能够获取自身的准确的位置信息.

### 1.2 问题分析

由于传感器节点随机分布在目标区域,因此目标区域中的小区域之间的节点密度分布稠密不均.

假设某个极小的区域  $\epsilon$  内分布了  $K \times L$  ( $K \geq 1, L \geq 1$ ) 个传感器节点, 如果将所有节点分成  $K$  个不相交的组, 为了使组之间在目标区域内节点分布均匀, 这  $K \times L$  个节点应该平均分配到  $K$  个组内. 采用随机调度覆盖算法<sup>[7]</sup>, 每组恰好包含  $L$  个分布在区域  $\epsilon$  内的传感器节点的概率为式(1):

$$P = \begin{cases} 1, & K = 1, \\ \prod_{i=K}^1 \binom{L \times i}{L} \times \frac{1}{i^L} \times \left(1 - \frac{1}{i}\right)^{L \times i - L}, & K > 1, \end{cases} \quad (1)$$

其中,  $S_i$  为第  $i$  组中分布在区域  $\epsilon$  内的节点个数,  $i = 1, 2, \dots, K$ . 由式(1)可知, 当  $L = 1$  组数  $K$  逐渐增大时, 每组恰好包含了 1 个分布在区域  $\epsilon$  内的传感器节点的概率迅速减小. 当  $K = 3$  时, 概率小于 25%, 当  $K = 7$  时概率几乎为 0. 也就是说, 存在着一些分组在区域  $\epsilon$  内没有分配到传感器节点, 导致在某些时刻处于活动状态下的节点不能有效覆盖区域  $\epsilon$ . 由于在区域  $\epsilon$  内分布的节点个数为固定值  $K$ , 显然可以推出, 存在着一些分组包含 2 个或 2 个以上分布在区域  $\epsilon$  内的节点, 导致在某些时刻区域  $\epsilon$  内处于活动状态下的节点分布过于密集, 此时在区域  $\epsilon$  内的节点通信冲突较大, 获得的传感数据冗余度高. 对于任意大小的  $K$  和  $L$  ( $K \geq 1, L \geq 1$ ), 使得这  $K$  个组内的传感器节点较为均匀地分布在目标区域是本文需要解决的关键问题.

### 1.3 极大相似分布模型

将  $N$  个传感器节点分为  $K$  组,  $K$  组之间分布均匀程度如何进行量化是本文考虑的问题之一. 在给出具体数学模型之前, 首先给出一些定义:

**定义 1.** 假设  $N$  个传感器节点分布在目标区域内, 已知节点  $i$  的位置坐标  $(x_i, y_i), i = 1, 2, \dots, N$ . 对于节点  $j$  ( $j \in P = \{1, 2, \dots, N\}$ ), 到节点  $j$  距离最短的节点称为节点  $j$  在节点集  $P$  中的最匹配节点.

**定义 2.** 假设均包含  $L$  个节点的节点集  $A = \{a_1, a_2, \dots, a_L\}$  和节点集  $B = \{b_1, b_2, \dots, b_L\}$  分布在同一目标区域,  $A \cap B = \emptyset$ . 对于所有的  $A$  和  $B$  之间的匹配, 当  $\sum_{i=1}^m |a_i - b'_i|$  达到最小值时, 称该值为节点集  $B$  到节点集  $A$  的距离. 用  $|A - B|$  来表示, 如式(2)所示:

$$\begin{cases} |A - B| = \min \sum_{i=1}^m |a_i - b'_i|, \\ |a_i - b'_i| = \sqrt{(x_i - x'_i)^2 + (y_i - y'_i)^2}, \\ i = 1, 2, \dots, L, \end{cases} \quad (2)$$

其中  $b'_1, b'_2, \dots, b'_L$  各不相同, 且  $b'_i \in B, i = 1, 2,$

$\dots, L$ .

**定义 3.** 假设一个由  $K \times L$  个传感器节点组成的传感器网络  $S$  随机分布在目标区域, 节点  $i$  的位置坐标为  $(x_i, y_i), i = 1, 2, \dots, K \times L$ . 将所有的节点平均分配到  $K$  组, 传感器网络的分布相似度等于第  $i$  组到第  $j$  组的距离之和,  $i$  和  $j \in \{1, 2, \dots, K\}$ , 且  $i \neq j$ . 即分布相似度等于式(3):

$$\sum_{i=1}^K \sum_{j=1, j \neq i}^K \|X_i - X_j\|, \quad (3)$$

其中,  $\|X_i\| = L, i = 1, 2, \dots, K, K \geq 2, S = \{X_1, X_2, \dots, X_K\}$ . 当任意一组包含的节点不同时, 可能导致传感器网络的分布相似度增大或者减小. 当传感器网络的分布相似度最小时, 称这种分配达到了极大相似分布.

**定理 1.** 当  $K$  为大于等于 2 的整数值时, 传感器网络的极大相似分布问题属于 NP 难问题.

证明. 略.

### 1.4 平均覆盖率的理论值上限

**定理 2.** 假设在面积为  $S$  的区域内随机部署  $K \times L$  个节点, 节点的传感范围均为  $r$ . 若将  $K \times L$  个节点分配到  $K$  个不相交的组内, 无论采用什么分配方案, 在忽略边界因素影响的条件下,  $K$  组节点依次轮流等时间工作时, 得到的平均覆盖率的期望值不会大于:

$$1 - \frac{K \times C_{K \times L}^0 p^0 q^{K \times L}}{K} - \dots - \frac{1 * C_{K \times L}^{K-1} p^{K-1} q^{K \times L - K + 1}}{K}, \quad (4)$$

其中,  $p = \frac{\pi r^2}{S}, q = 1 - \frac{\pi r^2}{S}$ .

证明. 考虑目标区域内的任意一点  $A$ . 由于节点是随机分布的, 因此对于任意一个节点, 没有覆盖到点  $A$  的概率是  $q \Rightarrow K \times L$  个节点均没有覆盖到点  $A$  的概率为  $C_{K \times L}^0 p^0 q^{K \times L}$ . 在  $K \times L$  个节点中, 恰好有  $K$  个节点覆盖到点  $A$  的概率为  $C_{K \times L}^K p^K q^{K \times L - K}$ . 当点  $A$  处于没有被任何节点覆盖的情况下, 无论这  $K \times L$  个节点如何分组, 各个组依次轮流等时间工作时, 该点没有被覆盖的时间与网络运行时间的比率为  $K/K, \dots$ , 在点  $A$  恰好被  $K-1$  个节点覆盖的情况下, 理想的分配方案是将这  $K-1$  个节点分配到  $K-1$  个不同的组, 点  $A$  没有被覆盖的时间与网络运行时间的比率为  $1/K$ . 在点  $A$  被  $K$  个节点覆盖的情况下, 理想的分配方案是将这  $K$  个节点分配到  $K$  个不同的组, 点  $A$  没有被覆盖的时间

与网络运行时间的比率为  $0/K$ 。由于点  $A$  具有任意性,综合上述,命题成立。

注意,上述这种理想的分配方案是相互冲突的,即不可能实现的。例如,假设在目标区域内点  $1, 2, 3$  与节点  $A, B, C$  的覆盖关系如图 1 所示,连线表示左边的点被右边的节点覆盖。从图中可以看出,每个点均被其中的 2 个节点所覆盖,如果将节点  $A, B, C$  划分为 2 组,无论这 3 个节点如何分组,也不能满足点  $1, 2, 3$  在 2 组节点依次工作时均被覆盖。

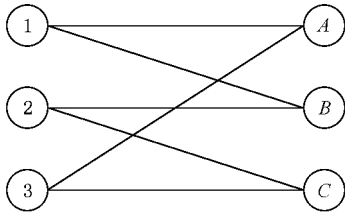


Fig. 1 A scenario.  
图 1 一个示例

为了量化在一种分配方案下冲突对于目标区域覆盖率的影响,我们在这里给出了一些定义。

定义 4. 假设在目标区域内所有传感器节点被划分成  $K$  个不相交的组。当  $K$  个组依次轮流等时间工作时,在一个工作周期  $T$  内,点  $A$  被覆盖的时间与  $T$  的比率称为点  $A$  的覆盖率  $CR$ (coverage rate),用  $C_A$  表示。

定义 5. 假设在目标区域内点  $A$  被  $n$  个节点覆盖,若某种分配方案将所有节点划分成  $K$  个不相交的组时,这  $n$  个节点在各个组内的个数分别为  $m_1, m_2, \dots, m_K$ ,称点  $A$  在该分配方案下的冲突度为

$$n - \sum_{j=1}^K 1 | m_j \geq 1. \quad (5)$$

我们用  $U_A$  来表示,  $U_A$  的实际意义是虽然  $n$  个节点分组后,每个节点都参与执行了监测点  $A$  的任务,然而,其中有  $U_A$  个节点,它们的存在并没有增加节点的覆盖率。以图 1 为例,如果某种分配方案将节点  $A, B$  划分到第 1 组,节点  $C$  划分到第 2 组,对于点 1 来说,点 1 的冲突度为 1。由定义 4 和定义 5 可以推出  $C_A$  和  $U_A$  之间存在如下关系:

$$C_A = \frac{n - U_A}{K}. \quad (6)$$

定义 6. 假设监测目标是一个包含  $t$  个点的点集  $P$ ,若某种分配方案将所有节点分成  $K$  个不相交的组时,集合内每个点的冲突度之和称为点集  $P$  在该分配方案下的冲突度,我们用  $U_P$  来表示。点集  $P$  内所有点的平均覆盖率称为  $P$  的覆盖率,用  $C_P$  表

示。如果  $P$  中每一点均被  $n$  个节点覆盖,可以推出:

$$C_P = \frac{n}{K} - \frac{U_P}{tK}, \quad (7)$$

其中  $t = \|P\|$ ,  $K$  为分组个数。当  $U_P$  取最小值时  $C_P$  为最大值。

当目标区域内分布了若干个传感器节点时,目标区域可以被划分成若干个子区域,其中子区域为被相同的传感器节点集覆盖的点的集合。因此,目标区域可以用有限个点来表示。由于节点是随机分布,子区域的个数不是固定值,在这里我们仅计算目标区域内平均的子区域个数。假设在面积为  $S$  的区域内随机放置了  $N$  个节点,且节点的位置坐标都不相同。现在考虑对于某个节点  $A$ ,周围有  $m$  个节点的传感区域与节点  $A$  的传感区域相交的情况。根据这  $m$  个节点与  $A$  的传感区域相交,且节点位置均不相同,可以推出,在  $A$  的传感区域内最少有  $m + 1$  个子区域。由于在  $A$  和这  $m$  个节点的传感区域内,最多只能形成  $m(m + 1) + 1$  个子区域,因此在  $A$  的传感区域内子区域的个数不会大于  $m(m + 1) + 1 - m$ 。我们取两者的均值作为对实际情况的估计。在  $A$  的传感区域内有  $m(m + 1)/2 + 1$  个子区域。对于任意节点来说,在它的传感区域内子区域个数的期望值为

$$E\left(\frac{m(m + 1) + 2}{2}\right) = \frac{E(m)}{2} + \frac{E(m^2)}{2} + E(1) = \frac{2\pi r^2}{S} \times N + \frac{8\pi^2 r^4}{S^2} \times N^2 + N \times \frac{2\pi r^2}{S} \times \left(1 - \frac{4\pi r^2}{S}\right) + 1. \quad (8)$$

目标区域内平均的子区域个数为式(9):

$$\left(\frac{2\pi r^2}{S} \times N + \frac{8\pi^2 r^4}{S^2} \times N^2 + N \times \frac{2\pi r^2}{S} \times \left(1 - \frac{4\pi r^2}{S}\right) + 1\right) \frac{S}{\pi r^2}. \quad (9)$$

由于节点是随机分布,子区域的面积大小并不一致,为了简化其计算,我们假设子区域的面积均相同。因此,被  $m$  个节点覆盖的子区域的个数与这些子区域的面积成正比。被  $m$  个节点覆盖的子区域的个数为

$$\left(\frac{2\pi r^2}{S} \times N + \frac{8\pi^2 r^4}{S^2} \times N^2 + N \times \frac{2\pi r^2}{S} \times \left(1 - \frac{4\pi r^2}{S}\right) + 1\right) \frac{S}{\pi r^2} \times C_N^m \left(\frac{4\pi r^2}{S}\right)^m \left(1 - \frac{4\pi r^2}{S}\right)^{N-m}. \quad (10)$$

由以上可知,计算均被  $m$  个节点覆盖的子区域

集合的最大覆盖率可以转换成如下问题:点集  $P$  中的每一个点均随机被  $N$  个节点中的  $m$  个节点覆盖, 点集的个数为式(10), 当将  $N$  个节点划分成  $K$  个不相交的分组时, 求点集  $P$  的最大覆盖率.

由于该问题属于 NP 难问题, 我们设计了一个启发式近似求解算法:

初始化. 随机将这  $N$  个节点分配到  $K$  个组内,  $i = 1$ .

1) 计算节点  $i$  属于哪一组时, 点集对应的冲突度最小, 不妨假设为  $j$ . 初始化  $j = 1$ .

① 节点  $i$  属于第  $j$  组时, 点集对应的冲突度值.

②  $j++$ , 重复①, 直到  $j = K$ .

2) 将节点  $i$  划分到第  $j$  组.

3)  $i++$ , 跳转到第 1) 步和第 2) 步, 一直到所有节点所属分组不再发生变化.

当  $L \times L = 300m \times 300m$ ,  $N = 2000$ ,  $r = 10m$ , 图 2 为  $m, K$  与对应的点集  $P_m$  的最大覆盖率之间的关系. 通过图 2 及式(10), 我们可以得到  $K$  与目标区域的平均覆盖率的理论值上限  $CR_{up}$  之间的关系, 如表 1 所示.

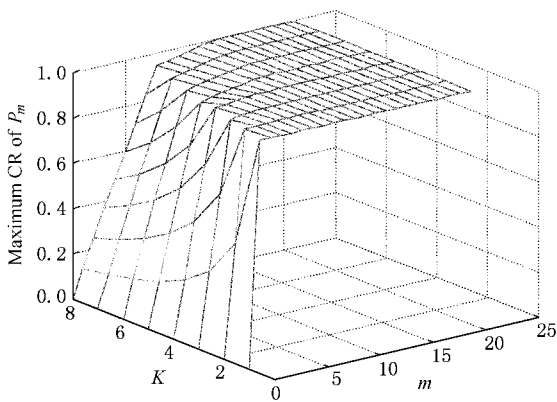


Fig. 2 Maximum CR of  $P_m$  VS.  $K$  and  $m$ .

图 2  $m, K$  与  $P_m$  的最大覆盖率之间的关系示意图

Table 1  $K$  vs.  $CR_{up}$

表 1  $K$  与  $CR_{up}$  之间的关系

$K$	$CR_{up}$
1	0.9991
2	0.9954
3	0.9836
4	0.9615
5	0.9158
6	0.8699
7	0.8165
8	0.7594

## 2 极大相似分布的近似求解算法

### 2.1 基于分组的集中式节点调度覆盖算法

通过定理 1, 我们得到极大相似分布问题属于 NP 难问题, 本文首先提出了极大相似分布的集中式近似求解算法. 下面给出该算法的一般步骤:

步骤 1. 任意选择一个未分配的节点  $i$  分配到第 1 组.

步骤 2. 从未分配的节点中选择节点  $i$  的最匹配节点分配到第 2 组.

步骤 3. 从未分配的节点中选择节点  $i$  的最匹配节点分配到第 3 组.

.....

步骤  $K$ . 从未分配的节点中选择节点  $i$  的最匹配节点分配到第  $K$  组.

重复步骤 1 至步骤  $K$ , 直到所有的节点都已被分配.

### 2.2 基于分组的分布式节点调度覆盖算法

在传感器网络中, 节点规模巨大, 将所有传感器节点的坐标信息发送给某一个节点进行集中处理会大量增加整个网络的通信开销, 不具有好的扩展性. 本文在基于分组的集中式节点调度覆盖算法的基础上, 提出了分布式节点调度覆盖算法.

首先由汇聚节点随机选择某一个节点, 设置它处于工作状态, 其他节点的初始状态均为准备状态.

#### 1) 节点为工作状态时的算法描述

当节点  $X$  为工作状态时, 节点首先广播查询消息 IM (inquiry message), 还没有分配组号的邻居节点收到查询消息后将发送自己的位置信息 PI (position information) 给  $X$ , 包括位置坐标和 ID 号, 此时  $X$  在  $T_w$  时间内完成如下工作: 在收到若干节点的 PI 消息后, 从这些节点中选择  $K-1$  个与自己最匹配的节点, 将自身的组号赋值为 1, 根据距离的大小依次把这  $K-1$  个节点的组号赋值为 2, 3, ...,  $K$ , 然后把这包括组号的消息 AMGN (announcement message of group number) 发送给对应的节点. 对于剩余节点  $S_i$ ,  $X$  根据这些节点到  $X$  之间的距离分别计算这些节点的延迟时间  $T_d = |S_i - X| \times T_0$ , 并把包含延迟时间的消息 AMDT (announcement message of delay time) 发送给对应的节点. 在完成上述工作后, 节点  $X$  从工作状态转变成休眠状态. 算法描述如图 3 所示.

```

Let  $X$  denote a node in working state
 $X$ . Broadcast(  $IM$  )
 $X$ . Wait_Receive(  $PIs$  )
 $n = \text{size of } PIs$ 
 $ID_i = PIs_i$ ,  $ID$  segment
Sort_Ascending(  $|ID_i - X|$ , ...,  $|ID_n - X|$  )
 $X$ . subset = 1
For(  $i = 2$ ;  $i < n + 1$ ;  $i++$  )
    If  $\text{index\_list of } |ID_i - X| < K + 1$ 
         $AMGN$ . subset =  $\text{index\_list of } |ID_i - X|$ 
         $X$ . Send(  $AMGN$  ) to  $ID_i$ 
    Else
         $AMDT$ . delaytime =  $t_0 \times \text{Distance}$ 
         $X$ . Send(  $AMDT$  ) to  $ID_i$ 
End for
 $X$ . State = Sleep
    
```

Fig. 3 Procedure block.

图3 程序块

2) 节点为接收状态时的算法描述

当节点  $X$  为接收状态时,将不再响应工作节点发送的查询消息.在  $T_w$  时间之内,节点一直等待接收  $AMGN$  或  $AMDT$ .若节点  $X$  接收到组号为  $x$  的  $AMGN$ ,节点将自己的组号赋值为  $x$ ,之后节点  $X$  从接收状态转变成休眠状态.若节点  $X$  接收到延迟时间为  $T_d$  的  $AMDT$ ,节点首先判断  $T_d$  与自身的等待时间  $T_c$  之间的大小关系,若  $T_c > T_d$ ,节点将  $T_c$  赋值为  $T_d$ ,若  $T_c \leq T_d$ ,节点不改变  $T_c$  的值,之后节点  $X$  转变成候选工作状态.

3) 节点为候选工作状态时的算法描述

当节点  $X$  为候选工作状态时,若在  $T_c$  完成之前没有收到查询消息,节点  $X$  从候选工作状态转变成工作状态.若在  $T_c$  之前收到查询消息,该节点将自身等待时间再延长  $T_w$ ,以等待周围的工作节点完成上述工作过程,并将自身状态转变成接收状态.

4) 节点为准备状态时的算法描述

当节点  $X$  为准备状态时,如果在某一个时间  $T_m$  收到了工作节点的查询消息,节点  $X$  需要将自身的位置信息  $PI$  发送给工作节点,并将等待时间设置为极大值,之后节点  $X$  转变成接收状态.

3 时钟异步对于覆盖的影响分析

在具体分析时钟异步对于覆盖的影响之前,我们首先给出簇的定义:

定义7. 对于一个给定的传感器网络,当分组数为固定值  $K$  时,在执行基于分组的分布式节点调度覆盖算法的过程中,对于任意一个处于过工作状态

的节点,令该节点为  $A$ , $A$  曾从周围未分配的节点中选择了  $K - 1$  个最匹配节点,我们将  $A$  与这  $K - 1$  个节点统称为一个簇.

从基于分组的分布式节点调度覆盖算法的描述可知,任意一个处于工作状态的节点  $A$  都从周围未分配的节点中选择  $K - 1$  个最匹配节点.根据最匹配节点的定义,在未分配的节点中,这  $K - 1$  个节点是距离  $A$  最近的节点集.然后, $A$  将自身和这  $K - 1$  个节点平均分配到  $K$  个组.当传感器节点密度较大且  $K$  较小时,这  $K$  个节点距离极其临近,因此节点  $A$  的传感任务可由簇内的任意一个节点代替完成.如图4所示,在该区域内分布着密集传感器节点,若将所有节点划分成3组,由于  $A, B, C$  这3个节点极其临近,因此  $A, B, C$  可相互代替完成它们的传感任务.在监测目标区域的过程中,若簇内的节点时钟保持严格的时间同步,也就是说这些节点均准确地依次轮流执行传感任务,任意时刻簇内存在而且只存在一个活跃节点.由于簇内节点的传感任务可相互代替完成,因此簇之间节点的时钟异步对于该算法的覆盖性能没有影响.由于实现簇内节点时钟同步较为容易(通过簇内某个节点广播一个同步消息,即可实现时钟同步),且簇之间的时钟异步并不影响算法性能,因此该算法适用于大型的传感器网络.

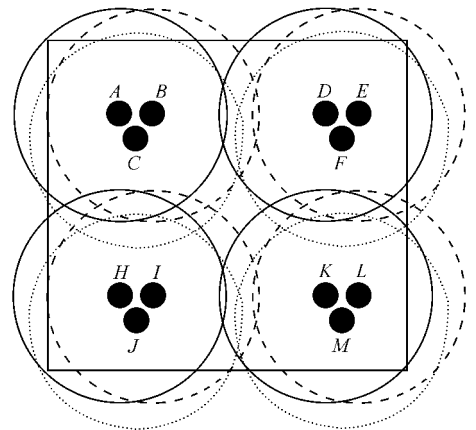


Fig. 4 A scenario of clusters.

图4 簇的示意图

4 性能评价

为了评估算法性能,我们用Java语言在Windows XP操作系统的平台下设计了基于分组的分布式节点调度覆盖算法的仿真程序,并实现了文献[7]提出的随机调度覆盖算法.在仿真实验中,本文提出的

算法与随机调度覆盖算法在覆盖率上进行了比较,并验证了在不同的  $K$  值下,簇之间的时钟异步对于覆盖率的影响.在传感器网络应用中,通常情况下传感器节点被放置在没有基础结构的地方.因此,在节点随机分布的条件下,各种算法的优劣关系到算法的实际应用范围.本模拟实验考虑节点随机分布在目标区域的情况.表 2 是该仿真实验中采用的实验参数:

Table 2 The Parameters in the Simulations

表 2 仿真实验的实验参数

Parameter	Value
Target Area ( $m^2$ )	$300 \times 300$
The Number ( $N$ ) of Nodes	2000
The Number ( $K$ ) of Subsets	1~8
Sensing Range of Nodes (m)	10

#### 4.1 覆盖率

当  $K$  取不同的值时,随机调度覆盖算法与基于分组的分布式节点调度覆盖算法得到的平均覆盖率如图 5 所示.从图中可以看出,当  $K \geq 2$  时,本文设计的算法都要优于随机调度覆盖算法,且当  $K = 4$  时随机调度覆盖算法与  $K = 6$  时基于分组的分布式节点调度覆盖算法相比,得到的平均覆盖率大致相等,此时本文设计的算法获得的网络生存时间是前者的 1.5 倍.从图中还可以看出,本文设计的算法得到的平均覆盖率接近于理论上界值.

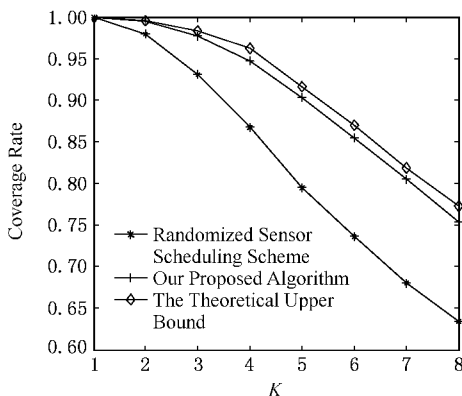


Fig. 5  $K$  vs. coverage rate.

图 5  $K$  与覆盖率之间的关系示意图

#### 4.2 簇之间节点的时间异步对于覆盖率的影响

为了评估簇之间节点的时钟异步对于覆盖率的影响,在模拟实验中,我们给每个簇内的所有节点增加了一个相同的随机偏移时间.图 6 反映了在不同的  $K$  值下,簇之间节点的时钟异步对于覆盖率的影响.

从图中可以看出,当  $K$  值较小,簇之间节点是异步的与簇之间节点是同步的相比,得到的覆盖率大致相等,证实了第 3 节的分析结果.当  $K$  值较大,簇之间节点是异步时,得到的覆盖率小于簇之间节点是同步的情况.这是因为在节点密度一定时簇内的这  $K$  个节点之间距离较大,簇内节点的传感任务不可相互代替完成.

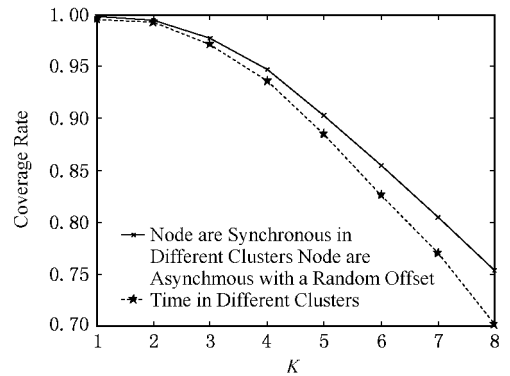


Fig. 6  $K$  vs. coverage rate.

图 6  $K$  与覆盖率之间的关系示意图

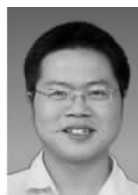
## 5 结 论

节点调度是延长网络生存时间的一种有效方法.当节点随机分布在目标区域时,现有的基于分组的节点调度算法不能保证各个组内的节点均匀地分布在目标区域.本文从使位置非常临近的传感器节点平均分配到不同组这一目的考虑,建立了极大相似分布的数学模型,提出了一种近似的求解算法:基于分组的节点调度覆盖算法,并给出了基于分组调度时平均覆盖率的理论值上界.当节点密度较大时,对于一定范围内的  $K$  值,该算法仅需簇内的节点保持时钟同步,簇之间节点的时钟异步对于算法的覆盖性能的影响可以忽略,适应于大型传感器网络难以维持整个网络保持时钟同步的要求.算法分析和仿真实验表明,本文提出的这种算法使得各个分组内的传感器节点较为均匀地分布在目标区域,获得的平均覆盖率接近于理论上界值,优于随机调度覆盖算法.

## 参 考 文 献

- [1] J Agre, L Clare. An integrated architecture for cooperative sensing networks [J]. IEEE Trans on Computers, 2000, 33 (5): 106-108

- [2] Ren Yan, Zhang Sidong, Zhang Hongke. Theories and algorithms of coverage control for wireless sensor networks[J]. Journal of Software, 2006, 17(3):422-433 (in Chinese) (任彦,张思东,张宏科. 无线传感器网络中覆盖控制理论与算法[J]. 软件学报, 2006, 17(3):422-433)
- [3] F Ye, G Zhong, S Lu, et al. PEAS: A robust energy conserving protocol for long-lived sensor networks [C]. The 23rd Int'l Conf on Distributed Computing Systems (ICDCS), Providence, Rhodeisland, USA, 2003
- [4] Mao Yingchi, Liu Ming, Chen Lijun, et al. A distributed energy-efficient location-independent coverage protocol in wireless sensor networks[J]. Journal of Computer Research and Development, 2006, 43(2):187-195 (in Chinese) (毛莺池,刘明,陈力军,等. 一种高效节能的与节点位置无关的传感器网络覆盖协议[J]. 计算机研究与发展, 2006, 43(2):187-195)
- [5] Romit Roy Choudhury, Robin Kravets. Location-Independent Coverage in Wireless Sensor Networks [OL]. <http://www.crhc.uiuc.edu/~croy/pubs>, 2004
- [6] R Zheng, G He, X Liu. Location-free coverage maintenance in wireless sensor networks [R]. Department of Computer Science, University of Houston, Tech Rep: UH-CS-05-15, 2005
- [7] C Liu, K Wu, V King. Randomized coverage-preserving scheduling schemes for wireless sensor networks[C]. In: Proc of the 4th IFIP Int'l Conf on Networking. Berlin: Springer-Verlag, 2005. 956-967
- [8] D Tian, N D Georganas. A coverage-preserving node scheduling scheme for large wireless sensor networks[C]. In: Proc of the 3rd ACM Int'l Workshop on Wireless Sensor Networks and Applications. New York: ACM Press, 2002. 32-41



**Li Xiaolong**, born in 1981. Since 2005, he has been Ph. D. candidate in computing science from Hunan University. His current research interests include sensor networks. 李小龙,1981年生,博士研究生,主要研究方向为传感器网络。



**Lin Yaping**, born in 1955. Professor and Ph. D. supervisor in the Computer and Communication College of Hunan University. His main research interests include communication network and machine learning. 林亚平,1955年生,教授,博士生导师,主要研究方向为通信网络与机器学习(yplin@hnu.cn)。



**Hu Yupeng**, born in 1981. Since 2005, He has been Ph. D. candidate in computing science from Hunan University. His current research interests include sensor networks and software engineering. 胡玉鹏,1981年生,博士研究生,主要研究领域为传感器网络、软件工程。



**Liu Yonghe**, born in 1974. Assistant professor at the Department of Computer Science and Engineering, UT-Arlington. His main research interests include wireless LANs, and sensor networks. 刘永和,1974年生,助理教授,主要研究方向为无线局域网络与传感器网络。

## Research Background

Sensor nodes have very limited energy supply due to their extremely small dimension, and it is in general impossible to recharge or replace the batteries of sensor nodes. How to utilize the limited energy of nodes and prolong the lifetime of sensor network become a paramount concern under the condition of meeting service requirements. Recent research has found that management of node duty circles, termed sensor scheduling, is able to save significant energy and prolong the network lifetime. Sensor scheduling schemes could be categorized into two types in terms of scheduling features: round-based scheduling scheme and subset-based scheduling scheme respectively. Compared with round-based scheduling scheme, subset-based scheduling scheme is an effective solution because it is only carried out once after deployment. However, existing group-based scheduling algorithms can not guarantee that any subset of sensor nodes is uniformly distributed over the target area. A scenario in which nodes are dense in some sub-regions of the target area incurs communication collision and data redundancy, while the opposite scenario in which nodes are sparse in other sub-regions induces that the sub-area area isn't detected effectively. Motivated by these reasons, firstly we set up the mathematic model of maximum similarity distribution, and then propose an approximation algorithm: the subset-based coverage-preserving distributed scheduling algorithm. The algorithm is robust to clock asynchrony of nodes in different clusters, hence it is applicable to large-scale sensor networks, for which precise time synchronization is very hard. In addition, this paper presents the theoretical upper bound for average coverage rate while nodes are randomly distributed over the target region. The analytical results and experimental simulation demonstrate that this algorithm has the ability that sensor nodes in each subset are rather uniformly distributed over the target area, and available coverage rate approaches the upper bound. This work is supported partly by the National High-Tech Research and Development Plan of China under grant No. 2006AA01Z227, the Natural Science Foundation of Hunan Province under grant No. 06JJ20049 and the Scientific Research Fund of Hunan Provincial Education Department under grant No. 06B047.