



Y1523921

分类号 TP393
UDC _____

学号 03067012
密级 公开

工学博士学位论文

无线传感器网络点覆盖技术研究

博士生姓名	<u>张鼎兴</u>
学科专业	<u>计算机科学与技术</u>
研究方向	<u>移动计算技术</u>
指导教师	<u>徐明 教授</u>

国防科学技术大学研究生院

二〇〇八年十月

摘 要

无线传感器网络是由具备感知、数据处理、存储和数据传输的传感器节点自组织而形成的无线网络,在军用和民用领域有着广泛的应用前景。覆盖问题是无线传感器网络的重要问题之一,它是反映无线传感器网络服务质量的一项重要性能指标。由于受到能量的约束,覆盖问题的主要任务是找出较小的覆盖集,并使这些集合能独立完成所需要的任务,而其它节点则处于低功耗的休眠状态。

目前,点覆盖问题研究一般着重于设计集中式优化算法,这些算法对无线传感器网络的实际应用存在一定的局限性。本文以布尔感知模型为基础,通过发掘点覆盖问题中应用场景,研究分布式、集中式以及集中式算法的局部化处理算法。针对点覆盖问题中目标点分布对覆盖集连通性的制约,本文也系统地研究了覆盖连通问题,并提出了相应的连通算法。

本文首先讨论了无线传感器网络节点冗余覆盖调度机制,进而提出了一种分布式算法 SRCA。该算法通过检测网络中的覆盖冗余节点,让部分冗余节点休眠,从而降低网络的冗余覆盖程度。SRCA 算法在保证网络初始覆盖的同时,能够有效地降低系统能量消耗,延长网络的生存时间。

面向无线传感器网络的一种应用场景,本文提出了一种基于遗传算法的传感器节点调度的一种集中式近似算法—NSAGA 算法。该算法期望利用遗传算法的种群特性,求解无线传感器网络覆盖子集。该算法每次迭代可以同时产生多个满足条件的覆盖子集,还可以根据覆盖要求改变约束条件,满足不同的应用场景。

针对目前点覆盖调度的算法一般都将覆盖问题转化为数学规划模型,然后采用集中式算法近似算法求解。虽然这种求解方式比分布式算法的精度高,但不适合大规模无线传感器网络。为此,本文提出了一种将集中式算法进行局部化的思想,并提出了一种分布式 LCACA 算法。该算法首先选举局部中心节点将整个网络划分成多个规模较小的网络。然后在每个划分的网络中再运行集中式算法,完成传感器节点的调度。而且,该算法还可以根据不同子区域的目标的覆盖要求,通过划分网络后,在每个划分的网络上运行不同的调度算法。

一般点覆盖算法大多假设覆盖与连通是一致的,并没有考虑被监测的目标分布情况对网络连通性的影响。为此本文讨论了点覆盖与连通性的关系,提出了一种基于 Steiner 树集中式连通算法—CCAST 算法。该算法首先将所有连通簇看成一个虚拟节点,然后构建加权通信图并调用已有构造 Steiner 树的算法挑选出 Steiner 点,使得所有的覆盖节点保持连通。接着,本文又进一步提出了一种分布式连通算法 DCAVIS。该算法首先构造虚拟独立集,然后寻找使覆盖集成为连通集的中继节点。DCAVIS 算法为解决点覆盖连通问题可提供了一种分布式近似求解算法,

同时可以有效延长无线传感器网络的生存时间。

综上所述，本文以无线传感器网络点覆盖问题为主要目标，从分布式、集中式以及集中式算法的局部化三方面研究了点覆盖算法，其中，集中式算法的局部化还可做为以后的工作进一步研究。

主题词：无线传感器网络 点覆盖 遗传算法 分布式调度 局部化算法 连通性

ABSTRACT

Wireless sensor networks are such kind of integrated wireless networks which perform sensing, data processing, storage and delivering. There are wide applications for the networks in both military and civil affairs. The coverage problem is one of the important issues in wireless sensor networks and reflects the quality of service. With the energy constraint, one main requirement of the coverage problem is to find a small subset of sensor nodes such that this subset can independently perform the required tasks while other redundant nodes enter into the low energy sleep status.

At present, researchers stress the central optimize algorithm which restrict practicality use to solve the point coverage problems. By exploring new application scenario in the point coverage issues, this dissertation aims to the point coverage problems to research the relevant algorithms based on Boolean sensing model. Considering the restriction of the object distribution to the connectivity, this dissertation also studies the connected coverage problems to put forward a relevant algorithm.

This dissertation first discusses the redundant nodes scheduling mechanism for wireless sensor networks. Furthermore, a distributed algorithm SRCA is put forward in the dissertation. This algorithm checks the redundant nodes in the network to make the redundant nodes sleep such that the redundant nodes are reduced obviously in the network. SRCA can effectively reduce the energy expense so to prolong the network lifetime.

Toward an application scenario, this dissertation proposed a sensor node approximately scheduling algorithm NSAGA based on genetic algorithm. This algorithm utilizes the population characteristic to solve the coverage subset. NSAGA can not only produce a few of coverage subsets to satisfy the coverage requirements in an iterative but also suit to diverse application scenarios by changing the restriction according to the coverage requirement.

Nowadays, the algorithms translate the point coverage problems into mathematic programming models, and then solve the mathematic programming by the central approximate algorithm. Although the method is of better precision, it is unsuitable for the large wireless sensor network. Therefore, this dissertation proposed a method to localize central algorithms, then a distributed algorithm LCACA is put forward. LCACA first elects the local central nodes to divide the network into a few of small scale subnets. Then, the central algorithm can be performed in these subnets. The algorithm can perform different scheduling algorithms in wireless sensor networks according to the sub-area coverage requirement.

Since current point coverage algorithms which are proposed in case of coinciding the coverage with connectivity ignore the influence of the object distribution over

connectivity. A central algorithm CCAST based on Steiner tree is proposed. CCAST first translates a connected cluster into a virtual node. Then, it constructs a weighted graph and calls current algorithm to pick out nodes which connect the network. Furthermore, a distributed algorithm DCAVIS is proposed. The algorithm first constructs a virtual independent set, and then finds out relay nodes.

In summary, aiming at the point coverage issues, this dissertation discussed the point coverage problem in the distributed algorithm, central algorithm and the method to localize the central algorithm. The method to localize the central algorithm can be further exploited for future research.

Key Words: wireless sensor networks, point coverage, genetic algorithm, distributed scheduling, localized algorithm, connectivity

表 目 录

表 3.1 SRCA 算法伪码.....	36
表 3.2 传感器节点仿真参数设置.....	40
表 3.3 DSAMSN 算法描述.....	46
表 3.4 仿真实验参数.....	48
表 4.1 拥挤距离算法.....	58
表 4.2 Algorithm I.....	61
表 4.3 NSAGA 算法描述.....	63
表 4.4 执行算法 1 的并行结果.....	66
表 5.1 DCA 算法的伪码.....	76
表 5.2 筛选 L-node 算法的伪码.....	78
表 5.3 状态为 DISELECTED 的传感器节点的操作算法.....	80
表 5.4 L-node 节点的操作算法.....	81
表 5.5 DCA 算法与 CEL 算法的参数列表.....	85
表 6.1 图 6.4(a)的关联矩阵进行“ \vee ”运算的行列.....	95
表 6.2 图 6.4(a)的关联矩阵进行“ \vee ”运算后结果.....	95
表 6.3 CCAST 算法描述.....	96
表 6.4 VIS 算法.....	98
表 6.5 STVIS 算法.....	103

图 目 录

图 1.1 无线传感器网络系统框架	1
图 1.2 传感器节点的主要硬件系统框架	4
图 1.3 (a) 区域覆盖问题 (b) 等价的点覆盖问题	14
图 2.1 方向型感知模型	20
图 3.1 (a) PEAS 算法的节点状态转化图 (b) 引起节点能耗不均匀的图示	31
图 3.2 覆盖冗余节点图示	33
图 3.3 SRCA 算法执行的时间线	33
图 3.4 (a) 冗余覆盖节点在初始覆盖中图示 (b) 出现覆盖盲点图示	34
图 3.5 SRCA 算法与 PEAS 协议平均覆盖度比较	39
图 3.6 SRCA 算法与 PEAS 协议休眠节点数比较	39
图 3.7 SRCA 算法与 PEAS 协议盲点数比较	40
图 3.8 SRCA 算法与 PEAS 协议存活节点数比较	42
图 3.9 SRCA 算法与 PEAS 协议网络覆盖率比较	42
图 3.10 多功能传感器节点感知多类型物理量的目标的对应关系	44
图 3.11 网络生存时间与节点数的关系	48
图 4.1 覆盖子集轮流工作的时间线	51
图 4.2 Pareto 解示意图	56
图 4.3 二进制串染色体个体图示	57
图 4.4 计算拥挤距离图示	58
图 4.5 随着迭代进行个体向 Pareto 前沿进化图示	64
图 4.6 不同迭代次数下 Algorithm I 算法的收敛结果图示	65
图 4.7 NSAGA 算法在 $(\eta, \varphi) = (20, 20)$ 迭代次数与覆盖子集数关系	66
图 4.8 不同目标数下 NSAGA 算法与 MC-MIP 算法的覆盖子集数比较	67
图 4.9 两个算法覆盖子集数对感知半径的比较	67
图 5.1 L-node 将整个网络划分为 6 个子网	71
图 5.2 传感器节点的状态转换图	74
图 5.3 (a) 在 L-node 传输覆盖范围内 (b) 不在 L-node 传输覆盖范围内	83
图 5.4 L-node 节点数与预设值间的关系	85
图 5.5 L-node 节点的成员数均方差与节点数的关系	86
图 5.6 死亡节点数与网络运行时间的关系	87
图 6.1 定理 6.2 图示	90

图 6.2 目标 t_1 、 t_2 被随机部署的传感器节点覆盖情况.....	90
图 6.3 通过辅助节点连接 3 个连通簇.....	92
图 6.4 一个连通簇处理成一个虚拟节点图示.....	93
图 6.5 定理 6.7 的图示说明.....	99
图 6.6 维持连通加入的节点数与节点通信半径的关系.....	105
图 6.7 网络生存时间与节点通信半径的关系.....	106

独创性声明

本人声明所呈交的学位论文是我本人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表和撰写过的研究成果，也不包含为获得国防科学技术大学或其它教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献已在论文中作了明确的说明并表示谢意。

学位论文题目： 无线传感器网络点覆盖技术研究

学位论文作者签名： 张新兴 日期 2009年 1月 4日

学位论文版权使用授权书

本文完全了解国防科学技术大学有关保留、使用学位论文的规定。本文授权国防科学技术大学可以保留并向国家有关部门或机构送交论文的复印件和电子文档，允许论文被查阅和借阅；可以将学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存、汇编学位论文。

(保密学位论文在解密后适用本授权书)

学位论文题目： 无线传感器网络点覆盖技术研究

学位论文作者签名： 张新兴 日期 2009年 1月 4日

作者指导教师签名： 许响 日期 2009年 1月 4日

第一章 绪论

随着微电子技术、无线通信技术以及分布式计算技术的飞速发展，无线网络、移动通信以及便携设备同样获得了长足的进步。当前的信息技术已经能够实现信息的海量存储、高速传输以及快速处理等功能。这些技术通过与微传感器技术的结合，极大地推进了集信息采集、信息处理与无线传输等功能于一体的无线传感器网络的发展，它能广泛地应用在工农业生产、国防、城市管理、环境监测、医疗卫生以及抢险救灾等多方面，具有十分广阔的应用前景。无线传感器网络将逻辑世界和物理世界融合在一起，必将改变人类与物理世界的交互方式，并拓展人类认识物理世界的的能力。

1.1 无线传感器网络概述

1.1.1 无线传感器网络的基本概念

无线传感器网络是由许多布置于目标区域内的传感器节点通过自组织方式组成的一种面向具体任务的无线网络。这种网络综合了传感器技术、嵌入式计算技术、无线网络通信技术以及分布式信息处理技术等多种领域的技术，它经历了从单一化到智能化、从有线传输模式到无线传输模式的发展过程。无线传感器节点具有低成本、低功耗、能够感知探测对象、数据处理、数据存储和无线通信能力，可以对目标信息进行采集、处理和传输。图 1.1 是无线传感器网络的系统框架图。

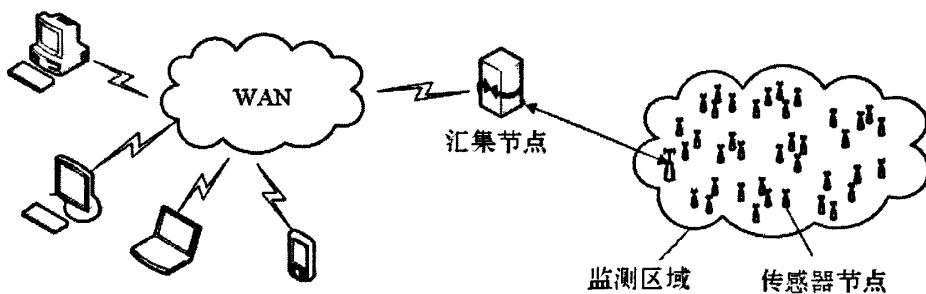


图 1.1 无线传感器网络系统框架

从图 1.1 可以看出，典型的无线传感器网络通常包括传感器节点、汇聚节点以及任务管理节点。在被监测区域中随机部署大量的传感器节点，并使这些传感器节点通过自组织的方式组成无线网络。现场节点探测到的信息通常由监测现场的节点

在本地进行初步处理,然后以多跳中继的方式传送至汇集节点,最后经卫星、互联网或者移动通信网络等途径到达数据管理中心。终端用户则经过数据管理中心对无线传感器网络进行配置管理、发布监测指令或是收集传回的数据。有些无线传感器网络,其中的部分或全部节点可以移动,因而拓扑结构也会随着节点的移动而不断地发生变化,这样,每个节点都可以充当路由器的角色,网络以 Ad Hoc 方式进行通信,并且每个节点都具备动态搜索、定位和恢复连接的能力。感知对象一般通过表示物理现象、化学现象或其它现象的数字量来表征,如温度、湿度、声频、振动和化学现象等。

汇集节点(Sink Node)通常具有较强的处理能力、存储能力和通信能力,它既可以是一个具有足够能量供给和更多内存资源与计算能力的增强型无线传感器网络节点,也可以是一个带有无线通信接口的特殊网关设备。汇集节点连接无线传感器网络与外部网络,通过协议转换实现管理节点与无线传感器网络之间的通信,把收集到的数据信息转发到外部网络上,同时发布管理节点提交的任务。

传感器节点通常是一个嵌入式系统,由于受到体积、价格和电源供给等因素的限制,它的处理能力、存储能力相对较弱。通信距离也很有限,通常只与自身通信范围内的邻居节点交换数据。要访问通信范围以外的节点,必须使用多跳路由。为了保证采集到的数据信息能够通过多跳送到汇集节点,节点的分布要相当密集。从网络功能上看,每个传感器节点都具有信息采集和路由的双重功能,除了进行本地信息收集和数据处理外,还要存储、管理和融合其它节点转发过来的数据,同时与其它节点协作完成一些特定任务。

1.1.2 无线传感器网络的基本特点

在过去的十几年间,无线网络技术已经取得了长足发展。不久的将来,无线传感器网络必将极大地扩展延伸 Internet 的应用范围。无线传感器网络作为一种不依赖于任何基础设施的网络,由于其应用特点和硬件特性,具有以下特点。

1. 与具体任务相关

无线传感器网络通过采集信息来感知客观物理世界,不同的任务势必具有相异的物理量或精度。不像 Internet 那样有统一的协议平台,不同的无线传感器网络虽然有一些共同的特性,但是,对于不同应用背景的无线传感器网络,在设计时必须让系统尽可能地接近应用,才能使无线传感器网络有效地发挥作用。因而,无线传感器网络必须针对具体应用来研究其硬件平台、软件系统和网络协议,这也是无线传感器网络有别于传统网络通信系统的一个显著特点。

2. 高密度且规模大

为了提高无线传感器网络的可靠性,除了增强单个节点的功能,很多应用还必

须依靠布置大量冗余节点来协同工作。由于节点通常利用电池供电，一旦耗尽电能，很难更换电池。而且，单个节点的感知范围也有限，因而，为了获得较好的覆盖质量就需要在目标区域部署较多的节点，有些任务甚至可能布置多达数千甚至上万个节点，这使得很多应用场景下的无线传感器网络具有高密度、大规模的特点。

3. 自组织

无线传感器网络往往会受到物理环境及网络本身一些难以预测因素的影响，如不能预先精确设定节点的位置、网络的拓扑结构、节点失效以及新加入网络中的节点等；此外，受监测的任务区自然环境影响，无线传感器网络所处的环境中的突发事件以及通信质量也是不可预测的；这些因素导致无线传感器网络的动态性较强。这样就要求节点具有自组织的能力，以适应节点的移动、加入和失效、节点电池的消耗以及节点信号传输范围的调节等。能够自动组网，并对网络进行自动配置和管理。

4. 分布式与抗毁性

无线传感器网络是一个对等式网络，没有严格意义上的控制中心，所有节点地位都是平等的，节点之间依靠分布式算法来协调各自的操作。此外，这种网络还可能随时加入或关闭一些节点，即使一些节点出现故障也不会影响整个网络的运行。因此，无线传感器网络具有很强的抗毁性。

5. 多跳路由

由于受节点硬件条件的限制，每个节点的发射功率较小，因而节点间的通信往往需要通过中间节点转发。由于没有专门的路由设备，这种多跳路由必须依靠普通节点协作完成。使得无线传感器网络中的每个节点既可以做为信息的发起者，同时也可以做为传递信息的中继者。

6. 拓扑变化

在无线传感器网络中，会出现节点移动、环境以及节点死亡的情况。此外，当进行节点调度时，每个节点还会在不同状态间进行切换。为了增强无线传感器网络监测的精度，往往还要补充新的传感器节点。这些因素都会引起无线传感器网络的拓扑结构发生变化，这种拓扑变化甚至难以预测。这就要求无线传感器网络能够适应拓扑变化，具有动态可重构网络的能力。

7. 以数据为中心的空间位置寻址

在无线传感器网络中，用户通常只关心监测区域内某个观测指标的数值，无需知道该数值来自哪一个节点，这就是无线传感器网络具有以数据为中心的特点。

传感器节点不必具有全球唯一的标识，不必采用因特网的 IP 寻址方式，因而节点的寻址不同于传统网络的寻址方式。用户使用无线传感器网络查询事件时，直接将需求通知给网络，而不是通告给某个确定编号的节点，只要网络能快速而有效

地组织各个节点的信息，然后融合提取出有用信息直接传送给用户即可。这种以数据做为查询线索的特点，使得无线传感器网络成为以数据为中心的网络。

8. 可靠性和安全性差

由于传感器节点往往在无人值守的状态下工作，在环境因素变化不可预知的情况下，使得网络的维护变得十分困难，甚至不太可能。此外，由于采用了无线信道、分布式控制等技术，网络更容易受到被动窃听、主动入侵等攻击。

最后，传感器节点具有的能量、计算能力、存储能力和通信能力等都受到其自身的硬件限制。无线传感器网络的上述特点，都给无线传感器网络的设计和开发带来了重要挑战。

1.1.3 传感器节点的基本结构和特点

传感器节点是一个微型的嵌入式系统，主要由传感器单元、处理器单元、存储器单元、通信单元和供电系统组成。图 1.2 所示为传感器节点的基本硬件系统框架。

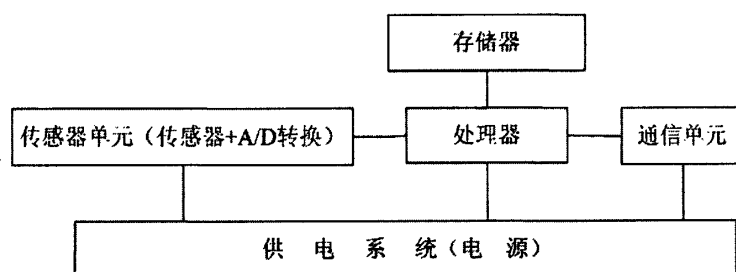


图 1.2 传感器节点的主要硬件系统框架

传感器单元包含不同类型的传感器，传感器节点通过这些传感器对监测环境进行主动或被动探测。如果没有传感器，无线传感器网络不过是空中楼阁。传感器单元主要包括采集物理信息的传感器和相应的 A/D 转换部件，与具体的无线传感器网络的监测任务密切相关。

处理器是传感器节点的核心部件，它执行相关的指令代码。处理器负责对传感器采集的数据进行处理，并协调节点各部分的工作如控制感知单元和电源的工作模式。目前在无线传感器节点中比较常见的处理器有 Atmel 处理器、Texas Instrument's MSP430 处理器等。

存储器用于存储数据和中间节点信息，存储器单元通常包括不同类型的存储器，以便存储程序和相关的信息。通信单元则是一个在无线信道上收发消息的无线通信器件。它负责与其它节点的通信，比如收发监测数据以及交换控制信息等。

供电系统为传感器提供正常工作所必需的能量。通常传感器节点采用直接供电方式是不太现实的,一般采用电池供电的方式。某些传感器的供电系统还配置了能量产生单元,可提取如太阳能之类的环境能源而实现电池的自我充电。此外,传感器节点还可以根据不同的监测任务添加其它硬件单元,如定位系统和移动部件。

在实现各种网络协议和应用系统时,由于传感器节点体积很微小,所以节点受到电池能量、通信能力以及计算和存储能力的限制。

1. 电源能量受限

传感器节点的微型化使得其电池能量有限,而且无线传感器网络的节点数量大,为节点更换电池带来困难。这样,节点的电池能量限制了无线传感器网络的工作寿命。如何高效使用有限能量尽量延长无线传感器网络的生存时间,这是无线传感器网络设计的关键技术之一。从传感器节点的基本组成可知,节点消耗能量的模块包括传感器模块、处理器模块和无线通信模块,其中绝大部分的能量消耗在无线通信模块上,通常 1 比特信息传输 100m 距离所需的能量大约相当于执行 3000 条计算指令所消耗的能量^[1]。

2. 通信能力有限

通常,无线通信的能耗 E 与通信距离 d 的关系为 $E = k \times d^n$, 其中, $2 < n < 4$ 。一般地,参数 n 的取值与很多因素有关。由于传感器节点体积小,发送端和接收端都贴近地面,障碍物多,干扰大, n 的取值要偏大。另外天线质量对信号发射质量的影响也很大。综合考虑这些因素,通常 n 取 3,即通信能耗与通信距离的 3 次方成正比。随着通信距离的增加,能耗会急剧增加。为节能起见,无线传感器网络应采用多跳路由的通信传输机制,尽量减少单跳通信的距离。

由于无线信道自身的物理特性,通常使得它所能提供的网络带宽相对有线信道要小得多。此外,节点能量的变化、周围地势地貌以及自然环境的影响,使得网络的无线通信性能也会经常变化,甚至通信有可能时断时续。由于受到硬件条件的限制,传感器节点的通信带宽也是很有限制的,通常传输速率仅达数百 kbps。

3. 计算和存储能力有限

传感器节点是一种微嵌入式设备,用于做为大规模部署,必须满足价格低能耗小的要求。因而传感器节点具有处理器能力弱、存储器容量小的特点,不能进行复杂的计算。对于无线传感器网络而言,传统的 Internet 网络上成熟的协议和算法开销太大,难以适用在传感器节点上运行。因此,要求传感器节点的协议和算法必须简单而有效,使之易于在微型传感器节点上实现。

1.1.4 无线传感器网络的应用背景

由于无线传感器网络集感知能力、计算能力以及通信能力为一体,它综合了传

感器技术、嵌入式技术、分布式技术以及无线通信技术,能够协作完成对物理世界的实时监测,并对所采集的数据进行处理和传输,满足用户的需求。无线传感器网络的这些特点,使得无线传感器网络的应用前景十分广阔,能够广泛地应用于军事、环境监测和预报、健康护理、智能家居、城市交通以及空间搜索等领域。随着无线传感器网络的更加深入研究和完善,无线传感器网络将逐渐深入到人类生活的各个领域。

1. 灾难预警与救助

无线传感器网络可应用于灾难预警与救助,当无线传感器网络用于监测意外火灾时,通过为传感器节点配备温度传感器,各个节点能够自动定位(相对地址或绝对地)火灾方位监测火情。如果在森林中安装这些传感器监测森林火警,消防员可利用 PDA 生成监测区域的“温度地图”或者确定高温区域的范围。类似的应用还包括化工厂的事故监控等。而无线传感器网络在军事领域和某些灾难预警与救助的应用有相似之处,只是传感器的监测对象是敌军而非意外火灾。美国国防部远景计划研究局资助的“智能尘埃(Smart Dust)”^[3]与美国军方提出的灵巧无线传感器网络(Smart Sensor Web)^[4]就是针对军事领域的应用而研究开发的无线传感器网络。

2. 环境控制和生物多样化勘测

生态环境的日益恶化,使得环境保护迫在眉睫。利用无线传感器网络实现对环境的监测,无线传感器网络不但能提供实时环境数据,而且可以避免人工数据采集给环节带来的负面影响。例如海洋监测,以便了解大陆架海域侵蚀状况;另外利用无线传感器网络可以研究生活在某一区域的动植物种类的数目(生物多样化勘测),这项应用主要利用了传感器能够长期自动无线工作的特点,该特点完全适合于观察目标^[5]。美国加州大学伯克利分校 Intel 实验室与大西洋大学联合,在“大鸭岛”部署了一个多层次结构的无线传感器网络系统监视海燕的生活习性^[6]。美国研制的 ALERT 系统^[7]也是用于监测降雨量、水位等环境条件,预测洪灾发生的可能性。

3. 智能楼宇

由于湿度、通风等因素的影响,使得空气调节设备的效率较低,往往导致楼宇会浪费大量的能源。而借助无线传感器网络对楼宇的温度、空气流通、湿度以及其它物理参数进行实时监控,可以显著地提高居住者的舒适程度,并减少能源消耗(据推测,仅美国,就能够节约 2×10^{24} 个单位的热量)^[8]。在提高能量效率的同时增加操作的方便性是“智能楼宇”的目标。为了实现该目标,目前的无线系统中(例如 BACnet、LonWorks, 或 KNX),有些正在改进,而有些已经投入使用。

4. 设备管理

无线传感器网络对管理多栋大楼中的设备也有用武之地,简单的例子包括智能

输入,利用进入大楼的人员所佩戴的徽章,由无线传感器网络判断哪一个人可以进入公司的哪个区域,这样,可以拓展到监视入侵者。此外,无线传感器网络用于监视在正常营业时间以外经过街道的车辆^{[9][10]},通过跟踪车辆的位置,向保安人员报警信息。无线传感器网络可以监视化工厂的设备化学药品的泄漏的情况等等。

利用无线传感器网络还可以对机器进行监视和维护。通过将传感器节点安装到机器里人很难触及的位置,根据探测到的振动模式判断是否有进行维护的必要。可以用于监视机器人以及火车的轮轴等。

5. 药品管理和卫生保健

无线传感器网络可以广泛地用于药品管理和卫生保健方面。如在术后护理和重病特别护理中,患者可以佩戴完成特殊任务的微型传感器,实现对病人特别是老人进行长期的监护。此外,还可以通过在药品库内投放传感器节点自动进行药品管理,可以大大降低给患者错误用药的概率。另外,利用医护人员和患者之间的跟踪系统可以及时地救治伤患^[11]。罗彻斯特大学利用无线传感器网络系统设计了一个智能医疗房间,使用智能尘埃测量居住者的体征,以掌握身体状况变化。Intel 公司研制出了基于无线传感器网络的家庭护理系统^[12],为老年人提供生活上的便利。

6. 其它应用

无线传感器网络还可用于其它很多领域,利用埋在街道或路边的传感器在较高分辨率下收集交通状况的信息^[13],可以监控车辆的行驶状况,这就是所谓的“智能马路”。它还可以为汽车提供如道路状况、危险警告以及前方是否交通堵塞之类的信息。无线传感器网络也可用于空间探索的监测,美国国家航空与航天局的 JPL 实验室(Jet Propulsion Laboratory)研制了 Sensor Webs 系统用于火星探测^[14]。

总之,无线传感器网络无论是在军事领域还是在民用领域都具有广阔的应用前景,必将对 21 世纪人类的生活产生重大的影响,无线传感网络研究的意义重大而深远。

1.2 无线传感器网络的研究内容和研究现状

1.2.1 研究内容

作为多学科交叉研究领域的新一代无线传感器网络,是目前信息领域的研究热点,许多关键技术有待研究人员去发现和解决。具体来说可以归结为硬件和软件两个方面的研究。对于传感器节点的硬件而言,主要包括如下研究内容:

- 低能耗节点的研究:传感器节点通常由电池供电,因此,研究体积小、能耗低的节点非常必要。具体来说,就是研究体积小、低功耗以及处理能力较强的芯片。研究体积小、容量大以及性能好的电池技术。

- 廉价的传感器节点：研究成本低廉的传感器节点很有必要。无线传感器网络往往是大规模的，单个传感器节点的成本是限制无线传感器网络推广使用的重要因素。
- 简单而鲁棒性好的信号调制和无线收发技术：研制传感器节点数字化数据的调制与解调问题、载波与天线技术等无线通信技术。传感器节点的调制方式和收发机的体系结构要求简单、成本低，并且要求的鲁棒性较好。
- 传感器技术：研究多功能微型化的传感器技术等。

对于传感器节点的软件系统而言，主要包括如下研究内容。

1. 自适应性及可扩展性

由于单个传感器节点无法完成目标监测任务，要求无线传感器网络能适应大规模应用场景，还要求各项技术能够有效地适用于大规模网络。目前的无线传感器网络研究中提出的很多网络协议面向具体的应用，不适应通用化的特点，研究具有自适应性的无线传感器网络协议体系结构很有必要。

2. MAC 协议的研究

无线传感器网络研究的核心问题之一是功耗管理，而射频模块是节点中最大的耗能部件，它是优化的主要目标。根据为减少数据碰撞和串音现象而采用的不同方法，将目前的 MAC 协议分为三类：(1)利用时分复用(TDMA)的方式为各节点分配独立固定的信道；(2)通过频分复用(FDMA)或者码分多址复用(CDMA)的方式，实现无冲突的强制信道分配；(3)通过竞争机制，保证节点随机使用信道并且不受其它节点的干扰。研究传感器节点的 MAC 协议直接控制射频模块，对节点功耗有重要影响。

3. 节能的路由协议

高效利用能量是无线传感器网络路由协议的重要特点。在无线传感器网络中，路由协议不仅关心单个节点的能耗，还要兼顾单个节点与整个网络的能耗均衡，这样才能有效延长网络的生存时间。与传统的路由协议不同，无线传感器网络是以数据为中心的网络，路由选择不是基于节点的编制进行，只要建立感兴趣的数据源节点到 Sink 节点间的链路即可。

4. 拓扑控制技术

拓扑控制主要的研究问题在保证网络的覆盖度和连通性的前提下，通过功率控制以及选择适当的节点充当骨干节点，删除节点间不必要的无线通信链路。从而减少无线信号冲突，降低无线传输能耗，延长网络生存时间，拓扑控制是无线传感器网络研究的核心问题之一。

5. 定位技术

无线传感器网络主要用来监测目标区域内发生的事件，传感器节点所采集的现

场数据必须与位置联系在一起,否则这些数据对用户毫无意义。因此尽可能准确地确定节点本身的位置十分重要。当前研究的主要问题是如何在有信标和无信标的情况下,利用多种测距方式或无需测距的方式实现节点自身的分布式定位。传感器节点的定位算法必须具有自组织、能量有效、分布式以及鲁棒性等特点。

6. 时钟同步技术

无线传感器网络在大多数情况下都需要时钟同步,时钟同步是无线传感器网络系统的关键技术之一。无论是节点调度还是目标跟踪都需要时钟同步,设计无线传感器网络的时钟同步机制,必须考虑能量高效、可扩展性、鲁棒性以及稳定性等要求。因此,必须针对无线传感器网络的特点设计专门的时钟同步协议。

7. 数据融合技术

由于无线传感器网络为了提高可靠性,而可能部署大量的冗余节点,这种节点冗余会引起数据冗余。为了减少网络中传输的冗余数据,以便有效地节省能量,传感器节点在收集数据的过程中,可利用节点自身的计算和存储能力进行数据融合,去掉冗余信息。但是数据融合技术在节省能量的同时,要以增加传输延迟以及降低鲁棒性为代价。如何根据特定应用的需求,研究数据的缓存和融合策略以及多传感器多数据类型环境下的数据融合方法是无线传感器网络的一个重要研究内容。

8. 数据管理技术

从数据存储的角度看,无线传感器网络就像一个分布式数据库系统。由于网络中的传感器节点数量多,产生的数据量很大,需要针对无线传感器网络的这种特点,设计节能的、健壮的数据管理方式。

9. 操作系统

传感器节点是一个微嵌入式系统,它只有非常有限的能量、计算和存储资源,所以,研究代码量尽可能小、复杂度尽可能低、具有较好的实时性和快速响应监测环境中发生的事件的无线传感器网络操作系统是无线传感器网络研究的一个主要任务。

10. 网络安全技术

由于现场数据必须从源节点传送至 Sink 节点,用户获取的数据必须是可靠的,因此,无线传感器网络需要配置必要的安全机制,如数据加密、消息认证以及安全管理等措施。由于无线传感器网络本身所具有的特点,使得研究它的安全技术的方法有别于传统网络的方法。

11. 网络的部署和覆盖问题

无线传感器网络必须根据具体的应用需求,在目标监测区域布置合适数量的传感器节点。对于无线传感器网络的部署,有很多方面值得研究,如在满足覆盖要求的前提下研究能量有效的部署技术。无线传感器网络的覆盖问题是通过节点的感

知能力建模,在某种感知模型下配置节点,使其能完整地获取监测对象的信息。在不影响网络覆盖性能条件下,研究冗余节点的特性,并利用这一特性合理调度节点以减少网络中活跃节点数量,并使得冗余节点轮换工作,达到延长网络寿命的目的,这是无线传感器网络服务质量(QoS)的基本要求。

1.2.2 无线传感器网络的研究现状

无线传感器网络起源于美国军方的需要,1978年DARPA在卡耐基-梅隆大学建立了分布式无线传感器网络工作组。进入21世纪后,由于无线传感器网络的应用前景非常广阔,已经引起了学术界以及工业界的极大关注。

美国国防部和各军事部门较早开始无线传感器网络的研究,设立了一系列军事无线传感器网络研究项目。美国陆军早在2001年以后陆续提出了一系列的无线传感器网络应用计划,如“灵巧无线传感器网络通信”计划,Smart Sensor Web^[4]、无人值守地面传感器群、传感器组网系统、网状传感器系统CEC^[15]、Intel公司在2002年基于微型无线传感器网络的新型计算发展规划等。Harvard大学的CodeBlue项目^[26]、Ohio大学的ExScal项目^[23]、Purdue大学的ESP项目^[24]、Yale大学的ENALAB实验室^[25]、Stony Brook大学的WINGS实验室^[27]、Colorado大学的MANTIS研究项目^[21]、USC大学的RESL实验室^[28]、USC/ISI的SCADDS项目^[29]、GIT大学的BWN实验室^[30]。此外,IBM^[31]、Intel^[32]以及Microso^[33]等公司也在从事无线传感器网络的相关研究。

目前,已经研制出一些实际的传感器节点产品。比较有代表性的产品包括美国Dust Networks和Crossbow Technologies等公司的“智能尘埃Smart Dust Mote”^[3]已进入应用阶段。无线传感器网络节点产品还有:UC Berkeley大学和Crossbow公司联合开发的MICA系列节点^[16],UC Berkeley大学BWRC研究中心开发的Pico Radio传感器节点PicoNode^[17],加州大学洛杉矶分校(University of California, Los Angeles,UCLA)开发的Medusa MK-2节点^[18],Intel公司开发的Intel Mote节点^[19]等。比较著名的无线传感器网络操作系统中有:UC Berkeley大学开发的TinyOS系统^[20],Colorado大学开发的MANTISIS系统^[21]以及UCLA大学开发的SOS系统^[22]等。

在国内,上海微系统研究所也于2001年开始了微传感器系统的研究。国内很多大专院校都设立了无线传感器网络相关研究课题。中科院计算机所、国防科技大学、哈尔滨工业大学、清华大学、浙江大学、上海交通大学、北京邮电大学等科研院所都较早的开展了无线传感器网络的研究。2004年3月,中国科学院与香港科技大学联合成立了无线传感器网络实验室。2006年初,国家973计划资助了无线传感器网络的基础理论研究,并成立了973无线传感器网络联合实验室,包括香港

科技大学、上海交通大学、国防科技大学在内的国内十余所高校联合开展了无线传感器网络的研究。在中国下一代互联网示范工程（CNGI）2006年研究开发、产业化已经应用项目中，也包含了无线传感器网络的研究项目。此外，国家自然科学基金和国家863高科技计划也都为无线传感器网络研究设立了专门资助项目。

1.3 无线传感器网络的点覆盖问题

对目标点/区域的覆盖问题是无线传感器网络研究领域的关键问题之一。在现有的研究成果当中，很多研究都是致力于解决无线传感器网络的区域覆盖与连接的关系等方面的问题。另外，也有一些研究致力于点覆盖问题，但这些方法大多采用传统的数学规划方法，并且在研究覆盖问题时很少涉及到连通性。尽管将点覆盖问题模型化为数学规划后，为理解无线传感器网络的点覆盖问题提供了理论背景。但由于无线传感器网络自身的特点，仅仅依赖数学规划问题的求解办法不能直接应用于无线传感器网络中。因此，很有必要更加深入地研究无线传感器网络的点覆盖技术。本节首先阐述了无线传感器网络的点覆盖技术的研究意义，然后讨论了点覆盖技术的挑战性，最后讨论了点覆盖问题与传统的区域覆盖问题间的关系。

1.3.1 无线传感器网络点覆盖技术的研究意义

跟区域覆盖问题一样，无线传感器网络点覆盖问题的目的是：通过调度部署在被监测区域内的节点，使其能感知被监测的目标点以达到节点获取这些目标点的信息的完整性和有效性。无线传感器网络的点覆盖问题在本质上与目标优化问题是一致的。由于节点的感知能力有限，常常需要大量的节点相互协作才能完成监测任务。实质上，传感器节点的覆盖问题就是在节点能量、通信带宽以及计算处理能力等条件受限的前提下，通过合理部署网络节点并采用节点调度技术，使无线传感器网络的资源得到有效而合理的利用，改善网络的服务质量。

从网络成本来说，配置合理的传感器节点的数量是非常重要的。目前的很多点覆盖问题归结为目标覆盖问题，然后模型化为数学规划问题求出优化解或近似优化解。但是无线传感器网络通常没有固定的基础设施，并且节点资源受限，即使节点没有移动性，随着节点的死亡，其拓扑也会发生变化。况且，对于大规模的无线传感器网络，也不可能采用集中式调度技术实现优化问题。因此，点覆盖问题必须通过分布式方式才能完成。然而，采用数学规划的方式只能通过集中式才能求出问题的优化解。

无线传感器网络是一种与具体应用密切相关的网络，应用的场景决定了覆盖的基本要求。如对于敌方区域或非常恶劣等人员不能到达的环境，只能通过随机部署（如空投撒播方式）足够多的传感器节点去监视目标。在这样的应用场景下，减少传

感器节点的无效工作时间,对那些共享感应目标和任务的传感器节点,可以关掉它们的电源以节省能量,从而可以延长网络的寿命。为此,必须确定关闭哪些传感器节点及如何调度分配节点的工作时间,以至于当关掉节点时,不至出现覆盖盲点而影响对目标的覆盖质量。

对于大规模密集型的无线传感器网络,从通信质量的角度看,关闭冗余节点也很有意义。由于无线传感器网络的节点间的通信通过由多跳共享方式实现,这种通信方式带来的一个问题是串音(overhearing)问题^[2]。当使用共享信道进行通信时,某个节点可能接收到不是发送给它的数据,从而造成“串音”。串音过程中的射频信号接收和解码过程会造成节点能量的浪费。虽然,“串音”问题是无线传感器网络的 MAC 协议考虑的,但是,减少“串音”源的数量可以降低 MAC 层的负担。另一方面,过多的节点发送信号,很容易造成信号间的干扰,影响信息的准确性和可靠性。况且,从物理上实现节点的休眠调度也是完全可行的。传感器节点无线通信模块通常具有发送(Tx)、接收(Rx)、空闲(Idle)和休眠(Sleep)四种工作状态,其能耗依次递减。其中,休眠状态的能耗远低于其它状态,因此为节能起见,通常希望节点尽可能处于休眠状态。可见,无线传感器网络的应用相关性对覆盖问题的影响是很显著的,研究无线传感器网络点覆盖问题是非常必要的。

点覆盖调度技术是保证信息获取的完整性和有效性的关键,没有覆盖调度技术做为基础,无线传感器网络的服务质量就得不到保障。点覆盖问题与区域覆盖问题一样,也是无线传感器网络研究领域中的关键技术的研究问题。

1.3.2 无线传感器网络点覆盖技术的研究关键点

无线传感器网络点覆盖的节点调度策略,不仅有助于网络节点能量的有效控制,而且可以提高信息获取的可靠性和延长网络的生存时间。但是,另一方面也会在通信、数据存储和计算等方面增加无线传感器网络的代价。与传统的分布式系统不同,特别是目标点的数量大,节点多且密集的无线传感器网络应用场景,由于无线传感器网络的节点资源受限,单个节点不可能存储全网的拓扑信息,网络拓扑动态性强,因此,无线传感器网络点覆盖问题存在很多有待解决的关键问题,大致可归纳为以下几点:

1. 无线传感器网络对目标的覆盖能力

覆盖能力是指无线传感器网络覆盖目标对象的程度,它直接反映了无线传感器网络感知目标的服务质量,也是衡量覆盖调度算法好坏的标准。由于单个节点的覆盖能力有限,在很多应用场景中,监测一个目标对象往往需要采集多个物理参量才能实现,需要多种类型的传感器节点。从这些特点可以看出,大规模节点的协调工作非常必要。但是,由于无线传感器网络的很多应用场景所处的环境条件复杂,通

常采用随机部署的方式, 这样, 会造成节点分布的随机性。因此, 如何合理调度节点相互协作是提高网络的覆盖能力的一项关键技术, 也是研究的难点之一。

2. 调度算法的通用性

无线传感器网络是与应用密切相关的无线网络, 但是, 为每种应用设计相关的调度算法, 不但会增加网络的成本, 而且不利于无线传感器网络产品的市场推广。因此, 如何从诸多的调度算法中发现问题的共性, 然后设计出问题的解决方案, 提高调度算法的通用性, 这是无线传感器网络调度中的又一个关键技术。

3. 合理使用无线传感器网络节点能量

由于单个传感器节点的资源受限, 并且网络规模大、网络的应用的地理环境条件复杂, 给每个节点置换电池是不现实的。因此, 合理分配节约节点的有效能耗, 降低节点的无效能耗, 在保证覆盖要求的前提下有效延长网络的生存时间, 是调度技术必须考虑的重要内容。

4. 调度算法的精确性

无线传感器网络的点覆盖问题大多采用单目标或多目标优化问题, 这些问题往往是 NP 难的, 只能求得问题的近似解, 如何减小问题解的误差, 提高调度算法的精度也是调度算法研究中的一项重要研究内容。

5. 调度算法实施的方式

无线传感器网络点覆盖算法的实施方式除了可以采用集中式、分布式的算法外, 还可以采取集中式与分布式二者相结合的混合式算法。其中, 集中式算法所得到的求解结果比较精确, 但 Sink 节点必须获取传感器网络的全局信息, 对于大规模无线传感器网络, 集中式算法很难实现。而分布式算法仅仅利用单个节点的本地邻居信息, 符合无线传感器网络的特征与应用要求, 但这种算法的精度比不上集中式算法, 而且很多点覆盖算法采用的数学规划模型无法利用分布式算法实现。为了兼顾集中式算法与分布式算法各自的优势, 将二者结合起来的混合式也是一种实施方式的一种尝试。总之, 对于大规模无线传感器网络, 设计精度高且节能的调度算法是无线传感器网络点覆盖调度的关键技术之一。

6. 点覆盖问题的连通性

在对无线传感器网络进行节点调度时, 无论采用直接还是通过多跳方式间接方式, 所有处于活跃的传感器节点必须能够彼此通信。因此, 除了要覆盖感知的目标点集之外, 通常需要配置另外的传感器节点作为中继节点, 使所有活跃节点能够形成一个连通的网络。对于无线传感器网络点覆盖问题连通性, 区域覆盖调度下的节点连通性条件不再适合点覆盖问题, 因此, 如何在保证调度节点覆盖目标点集的同时, 调度其余的节点, 使得这些工作节点是一个连通集合, 这也是点覆盖调度技术的一个关键问题。

1.3.3 无线传感器网络点覆盖与区域覆盖的关系

无线传感器网络的点覆盖与区域覆盖问题之间存在一定的关系，实际上，区域覆盖问题也可以转化为点覆盖问题^{[34][35]}。

如果做如下假设：如果某个监测区域被相同的传感器节点所覆盖，那么称这个监测区域为一个域。根据这种假设，无线传感器网络的区域覆盖中的被监测区域可以看成多个域的集合。由于每一个域都被相同的节点覆盖，因此，可以将这个域等同一个目标点，这样，区域覆盖问题就转化成为点覆盖问题。

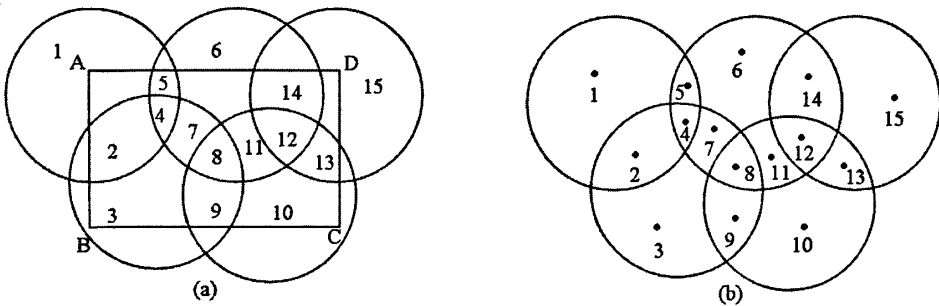


图 1.3 (a) 区域覆盖问题

(b) 等价的点覆盖问题

如图 1.3 (a) 所示矩形 ABCD 为被监测区域，根据前面关于域的假设，可以将矩形 ABCD 看成 15 个域，如果将每个域看成一个目标点，那么得到等价于这个区域覆盖的点覆盖图 1.3 (b)。文献^{[34][35]}提出了各自求解域的方法可以将区域覆盖问题转化为点覆盖问题。

1.4 本文研究内容

点覆盖问题是无线传感器网络的基本问题之一，它直接影响对周围物理世界的监测质量，也是无线传感器网络服务质量(QoS)的一项度量指标。在无线传感器网络应用中，考虑到监测质量和网络可靠性的要求，通常部署大量的节点组成大规模密集型网络，从而出现大量节点相互交迭地覆盖同一个目标的现象。从节能角度考虑，这种冗余覆盖必将导致冗余地采集和传输数据，不利于有效使用能量。但是，我们可以通过对节点进行合理调度，使冗余节点交替工作，减少节点的无效能耗，通过节点的多次轮换工作，有效地延长无线传感器网络的生存时间。因而，利用无线传感器网络这种节点冗余的客观特性，在不影响覆盖质量的前提下，研究延长网络生存时间的节点调度技术，是无线传感器网络研究的一个重要课题。本文以布尔

感知模型为基础，从以下 4 个方面系统地研究了点覆盖问题。

(1) 保持网络覆盖质量的节点调度问题

通过检测网络中的覆盖冗余节点，关闭部分冗余节点减少网络中的活跃节点数，从而降低网络的冗余覆盖度，延长网络的生存时间。

(2) 点覆盖问题转化为目标优化问题

当将点覆盖问题转化为目标优化问题时，如何以较快的速度得到问题的解？而提高算法的并行度减少算法的迭代次数不失一种理想的方式。利用遗传算法每次迭代可能产生多个符合要求的种群这一特点，可有效减少算法的迭代次数。

(3) 集中式与分布式相结合的混合算法问题

考虑到目前在点覆盖调度的很多算法都将覆盖问题转化为数学规划模型，然后采用近似最优的方法求解所构建的数学规划问题，只能用集中式算法近似算法求解。这些算法虽然这些算法由于它们采用集中式处理方式而不适合大规模无线传感器网络，但是这种求解方式比分布式算法的精度高，而且通过修改约束条件就可适合不同的应用场景，因而这些算法也具有其应用优势。对于大规模无线传感器网络，将集中式优化调度算法与分布式算法相结合的混合算法，既可以利用已有的集中式算法的优势，又可以利用分布式算法不需要全局信息的优点。

(4) 点覆盖节点的连通问题

由于大部分点覆盖算法都假设覆盖与连通是一致的，很少考虑点覆盖问题不仅与传感器节点的分布有关，而且与被监测的目标分布情况有关，因此，必须考虑到覆盖节点的连通问题。

1.5 本文的主要贡献

论文围绕无线传感器网络的点覆盖问题进行了深入研究，主要创新点包括：

(1) 提出了一种局部化、分布式无线传感器网络节点冗余覆盖调度机制 SRCA 算法。该算法首先检测网络中的覆盖冗余节点，同时能够在关闭部分冗余节点后保持无线传感器网络所要求的覆盖质量。并以节点剩余能量做为优先级，防止互为邻居的冗余节点同时关闭而导致网络中出现覆盖盲点。通过节点调度，SRCA 算法使冗余节点交替工作，可有效地减少冗余数据传输，节省系统能量。针对新的硬件技术的发展，多功能传感器的研制，研究了多功能传感器网络模型并针对多功能无线传感器网络的点覆盖问题，提出了调度覆盖算法的 DSAMSN 一种基于多功能节点调度覆盖算法的 DSAMSN。

(2) 提出了一种基于遗传算法的传感器节点调度的一种集中式近似算法 NSAGA 算法。该算法采用遗传算法的种群特性以解决无线传感器网络最小覆盖集问题。算法将网络的所有节点尽可能多地划分若干覆盖子网，减少活跃节点数量。

NSAGA 算法可以利用遗传算法每次迭代可能产生多个符合要求的种群这一特点, 可以提高算法的收敛速度。NSAGA 算法提供了一种点覆盖问题节点调度技术的框架, 在不同应用的覆盖要求下, 通过改变约束条件, 即可得到相应的目标优化, 构造覆盖节点子集。

(3) 针对不同的子区域具有不同的覆盖要求, 而且目前点覆盖调度的一般算法属于集中式算法, 提出了一种同一网络中的节点可以根据不同子区域运行相应的覆盖算法的思想。本文提出了一种将集中式优化调度算法本地化的覆盖算法—LCACA 算法。由于该算法并不要求网络的全局信息, 因此仅仅依赖每个节点的 m -hop 邻居信息。通过选举局部中心节点将整个网络划分成多个规模较小的网络, 然后在每个划分的网络中再运行集中式算法, 完成传感器节点的调度。LCACA 算法不要求节点的时钟同步, 适合应用在随机部署的大规模无线传感器网络中。

(4) 由于点覆盖问题不仅与传感器节点的分布有关, 而且与被监测的目标分布情况有关。针对这一特性, 本文提出了一种基于 Steiner 树的集中式连通算法—CCAST 算法。该算法首先将所有连通簇看成一个虚拟节点, 通过提出一种将连通簇转化为虚拟节点的算法, 在此基础上构建加权通信图, 然后调用已有构造 Steiner 树的算法, 挑选出 Steiner 点, 使得所有的覆盖节点保持连通。

(5) 为了适应大规模无线传感器网络的要求, 本文又进一步提出了一种分布式连通算法—DCAVIS 算法。该算法将连通簇看成虚拟节点, 以此为基础构造虚拟独立集, 再进一步寻找使所有覆盖节点能保持连通的中继节点。由于 DCAVIS 算法是一种分布是算法, 算法在不同的节点上进行, 因此, 寻找算法终止条件非常重要, 本文通过构造虚拟独立集的方式, 解决了算法终止条件问题。DCAVIS 算法为解决点覆盖连通问题提供了一种分布式近似求解算法。

1.6 论文结构

本文分为七章, 组织结构如下:

第一章是绪论。在这一章我们介绍无线传感器网络的基本概念、无线传感器网络的基本特点、研究内容和研究现状。分析了无线传感器网络中基本问题、面临的挑战以及现有工作的特点。

第二章描述了覆盖模型与相关问题。在这一章介绍了传感器节点的常见的感知模型, 覆盖问题研究的现状, 对无线传感器网络的覆盖问题系统地进行了分析。

第三章讨论了无线传感器节点自调度冗余覆盖问题。提出了基于节点冗余覆盖调度的 SRCA 算法, 该协议首先判别冗余覆盖方法检测网络中的覆盖冗余节点。通过使用节点剩余能量作为优先级, 防止互为邻居的冗余节点同时关闭导致网络中出现覆盖盲点。该方法关闭不必要的冗余节点以及尽可能地实现负载均衡, SRCP 机

制既可保持网络原始覆盖,又能有效地降低系统能量消耗。

第四章涉及基于遗传算法的传感器节点调度。我们的目的是通过限制覆盖每个被监视的目标对象的传感器节点数的上界,保证其被覆盖的冗余程度不至于太高,这样,在满足应用覆盖要求的前提下,可将整个无线传感器网络尽可能多地划分成若干覆盖子集,每一轮让这样的一个覆盖子集中的节点做为活跃节点执行监测任务。通过覆盖子集轮换工作,延长网络的生存时间。针对某种覆盖要求比较灵活的传感器应用,利用遗传算法的种群选择特性实现伪并行最小化传感器节点的覆盖算法。

第五章讨论将集中式优化调度算法局部化的覆盖问题。提出了一种基于集中式算法的本地化覆盖算法—LCACA 算法,这种算法并不要求网络的全局信息,仅仅依赖每个节点的 m -hop 邻居信息,借助相关的集中式算法即可完成传感器节点的调度。

第六章讨论分布式连通算法,首先将所有连通的活跃节点看成一个虚拟节点。在此基础上,提出了一种基于 Steiner 树连通集中式算法。为了适应大规模无线传感器网络的要求,本章进一步又提出了一个分布式连通算法—DCAVIS 算法。该算法将连同簇看成虚拟节点,以此为基础构造虚拟独立集,DCAVIS 算法为解决点覆盖连通问题提供了一种局部化与分布式的近似求解途径。

第七章对本文的工作进行总结并展望下一步的工作。

第二章 无线传感器网络覆盖问题的相关研究

无线传感器网络作为一种应用型无线网络，其节点不仅是传输数据的基本组织，更多地还必须承担协作监控物理环境的任务。覆盖问题的主要目标之一是延长网络的生存时间，它是无线传感器网络中非常重要的研究内容。目前，已经有很多研究者对无线网络中的覆盖调度技术进行了研究。本章主要讨论无线传感器网络领域中无线传感器网络覆盖的相关问题，包括介绍感知模型、覆盖问题的分类及相关工作和无线传感器网络的部署和初始化问题。

2.1 感知模型

无线传感器网络在承担监测任务时，只有当每个节点与事故发生点的距离足够近时，才能感知到事件的发生。如果给定了某一几何区域来部署无线传感器网络，那么每个节点只能感知离节点足够近的区域发生的事件，这种传感器所能感知到的目标区域范围称为传感器节点的感知范围。每个传感器节点的感知范围是由其自身所具有的特性决定的，而只能利用某个感知模型来近似描述同一类节点的感知范围，感知模型越接近现实问题，根据该感知模型研制的覆盖算法更精确。因此，传感器节点的感知模型直接决定了无线传感器网络的监测能力。

目前，传感器节点主要采用的感知模型主要有布尔感知模型(Boolean Sensing Model)、概率感知模型(Probabilistic Sensing Model)以及方向型感知模型(Directional Sensing Model)，下面将分别介绍这三种感知模型。

2.1.1 布尔感知模型

在二维空间或者三维空间中，传感器节点的感知能力是以节点为中心的，半径为 R 的圆面或球体， R 称为传感器节点的感知半径，传感器节点的感知半径是由它的感知单元的物理特性决定的。当目标对象位于某节点的感知圆面或球体内时，就认为这个传感器节点在该模型下能够检测到这个目标对象。假设在位置 p 处的传感器节点观察到位置 q 的事件，那么，传感器的输出信号的强度为：

$$C(p, q) = \begin{cases} 1 & |p - q| \leq R \\ 0 & |p - q| > R \end{cases} \quad (2-1)$$

(2-1) 式中， $|p - q|$ 是 p 、 q 两点之间的欧几里德距离。

根据布尔模型的特点，如果事件位于一个节点感知范围内，那么这个节点就能以概率为 1 可靠地感知该事件，而在范围之外的事件则完全是无法感知的。

布尔感知模型只是笼统地规定传感器节点的感知能力。事实上，每个传感器节点的感知能力还取决于该节点的剩余能量，随着节点在工作中不断消耗能量，它的感知能力也是不断变化的，即这个感知圆面或感知球体是在逐渐缩小的。Tian 和 Georganas^[36]在布尔感知模型的基础上，假设传感器具有不同的感知范围。在布尔传感模型中，假设同样形态(温度、湿度等)的所有传感器具有共同的感知范围 R 。目前，很多研究者使用的传感模型都采用这种具有全向感知能力，且没有随机变化的布尔感知模型。

2.1.2 概率感知模型

在实际应用环境中，由于受到噪声、信号强度的路径衰落等因素的影响，传感器节点的感知能力会表现出一定的不确定性。而采用概率感知模型则能反映这种不确定性因素。在概率感知模型中，传感器节点 s 检测到的感知区域内任意一目标点 p 的概率为

$$P(s, p) = e^{-\alpha|s-p|} \quad (2-2)$$

(2-2) 式中，参数 α 表示传感器节点的感知能力以及信号随着距离衰减的程度^{[37][38]}。这种感知模型粗略地反映了传感器节点的感知能力与被检测的信号到节点间的距离的变化规律。然而，对于某些特性的传感器而言，当被监测的目标对象到节点的距离超出一定范围后，传感器节点的感知能力才表现出不确定性的特征。为了表述这种特性，Zou^[39]将(2-2)式修改为

$$P(s, p) = \begin{cases} 0, & r+r_e \leq |s-p| \\ e^{-\lambda\alpha^{\beta}}, & r-r_e \leq |s-p| \leq r+r_e \\ 1, & r-r_e \geq |s-p| \end{cases} \quad (2-3)$$

(2-3)式中 r, r_e 分别表示传感器节的确定性感知范围与不确定性感知范围的量度，参数 $\alpha = |s-p| - (r-r_e)$ ，而参数 β 和 λ 则为节点 s 与目标 p 之间的距离处于 $[r-r_e, r+r_e]$ 内时，节点 s 感知 p 处的目标的能力。

当忽略邻近节点的影响时，J. Lu^[40]则定义传感器节点 s 检测到的感知区域内任意一目标点 p 的概率为

$$C_p = \begin{cases} \frac{1}{[1+\alpha|s-p|]^{\beta}} & |s-p| \leq R \\ 0 & |s-p| > R \end{cases} \quad (2-4)$$

(2-4) 式中 $|s-p|$ 表示传感器节点 s 到目标点 p 的距离， α, β 为与具体物理特性相关的参数， R 为传感器节点的覆盖半径。

由于传感器节点对事件的感知依靠其将物理环境的激励转化为电信号,这种激励信号的质量与传感器节点到事件发生地的距离密切相关。由于受到环境噪声的干扰,随着距离的增加,激励信号的强度也会随之衰减。N. Ahmed 等^[41]则认为来自目标对象的信号随着其传输距离的增加而衰减的,从而采用对数距离路径衰减模型描述传感器节点的感知能力。对数衰减模型是通信领域广泛采用的模型^[42],它由两部分组成,第一部分标示接收特定距离 d_0 处的目标对象的信号平均功率 $P_r(d_0)$,第二部分反映了接收到的信号强度是一个与距离有关的随机变量。如果用 dB 做为单位,可用 Gaussian 分布表示,这种感知模型具体描述如下

$$P_r(d) = P_r(d_0) - 10\beta \log(d/d_0) + X_\sigma \quad (2-5)$$

(2-5) 式中 β 为路径衰减指数, X_σ (dB)是服从 $(0, \sigma^2)$ 分布的正态随机变量,即 $X_\sigma \sim P_N(x) = \sqrt{1/2\pi\sigma^2} \exp(-x^2/2\sigma^2) dx$ 。这样,参数向量 (β, σ) 定义了信号强度的统计模型,它与实际环境有关。事实上,在现实世界里,由于这种衰减特性,在一定距离内接收的信号功率是一随机变量。因此,采用衰减模型更能接近传感器节点侦测目标对象的实际能力。

2.1.3 方向型感知模型

布尔感知模型与概率感知模型假定节点的传感器对各个方向具有相同的敏感度。然而,在实际应用环境中,由于传感器本身的构造(如摄像机等)以及传感器部署环境的特点,通常会出现传感器对某些方向具有更好的敏感度。例如,在某一方方向存在障碍物,一个节点的声音传感器可能它被挡住而影响感知能力。还有一种情况是在不同的时间里,传感器受到的环境温度和湿度也会有所变化,因而,同样的传感器对于同样的环境激励所产生的输出可能是不同的,比如传感线路暴露的环境的温度差异等因素的影响。

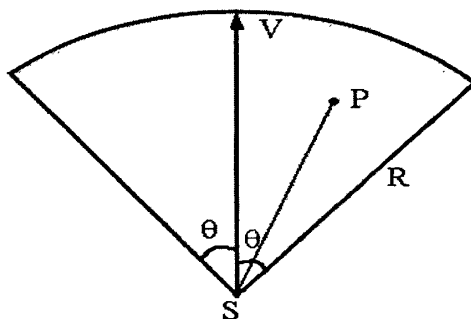


图 2.1 方向型感知模型

Ma^[43]针对视频传感器的特点提出了一种方向固定的感知模型,该感知模型将节点的感知范围描述为一个以节点为圆心,半径为其感知距离,且与视角相关的扇形区域。这种方向型感知模型不能用单以感知参量来描述传感器节点的感知特性。从图 2.1 可以看出,节点的有效视角 2θ 、最大感知距离 R 和方向矢量 \overline{SV} 一起描述传感器节点的感知特性。对于目标点 P ,当且仅当同时满足下面 2 个条件时,该目标的事件才能被传感器节点感知:

$$(1) |PS| \leq R;$$

$$(2) -\theta \leq (\overline{SP}, \overline{SV}) \leq \theta, \text{ 其中 } (\overline{SP}, \overline{SV}) \text{ 表示矢量 } \overline{SP} \text{ 和 } \overline{SV} \text{ 之间的夹角。}$$

在这种方向型感知模型中,方向矢量 \overline{SV} 可以是变化的,比如可以转动的视频头。不管怎样,这种节点能否感知某个目标的参量由 4 元组 $(\overline{SP}, \overline{SV}, \theta, R)$ 决定。

综上所述,布尔感知模型假定传感器节点感知发生的事件是确定的,即假设其覆盖区域有确定的大小。尽管布尔感知模型对于问题的描述存在一定的局限性,当前,绝大多数关于无线传感器网络覆盖问题的研究都基于布尔感知模型。本文采用布尔感知模型研究无线传感器网络的点覆盖问题。

2.2 覆盖问题的分类

覆盖作为无线传感器网络的一个重要研究问题,引起了国内外研究学者的广泛关注。目前,针对不同覆盖需求提出一系列特点各异的方法。

为了比较全面地认识无线传感器网络的覆盖问题,本节将从覆盖对象、部署方式以及传感器节点的移动性 3 个方面介绍无线传感器网络的覆盖问题。

2.2.1 按覆盖对象分类

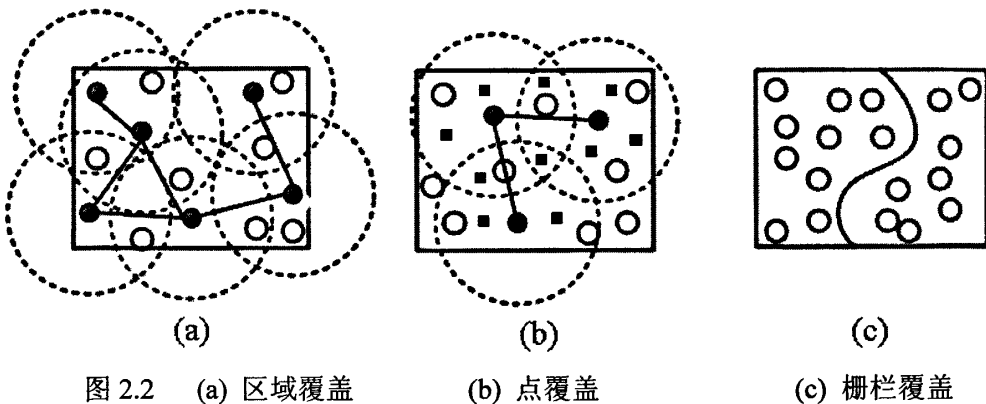
根据覆盖对象的不同,研究者将无线传感器网络的覆盖问题分为 3 类:区域覆盖(area coverage)、点覆盖(point coverage)和栅栏覆盖(barrier coverage)。

区域覆盖要求传感器节点覆盖目标区域中的每一个点。区域覆盖是无线传感器网络最常见的一种覆盖问题,主要用于研究对某个区域实施监测问题。如果区域内的每个点至少被 k 个节点所覆盖,那么称该区域是 k -覆盖。在理想情况下,要求目标区域内的每一点至少被一个传感器节点所覆盖,但其现实意义在于目标区域实现最大化覆盖,如图 2.2(a)所示。与区域覆盖相关的研究问题是满足分析满足覆盖要求下的节点密度以及感知半径大小分析、为了延长无线传感器网络的生存时间的节点调度算法等。

栅栏覆盖则关注的是对移动目的检测问题。在目标监测应用中,当某个移动目

标沿任意轨迹穿越节点的部署区域时,去研究该移动目标被监测到或没被监测到的概率,这就是栅栏覆盖问题,如图 2.2(c)所示。栅栏覆盖问题的目标是找出连接初始位置和目的位置的一条或多条路径,在不同模型定义下,使得找出的路径能提供对目标的不同感知质量。该问题的意义在于:一方面可以确定最佳网络部署,使得目标监测概率最大;另一方面,当穿越敌方的监控区域时,可以选择一条最安全的路径。

如果针对目标区域内的有限个离散目标实施覆盖时,那么,称为点覆盖问题。研究目的在于保证每个目标在任意时刻至少被一个传感器节点所覆盖,如图 2.2(b)所示。当大规模密集型传感器节点随机部署时,通过覆盖每个传感器位置来近似表示给定区域内所有点的覆盖。实际上,区域覆盖问题也可用点覆盖近似表示。因此,点覆盖研究方法 with 区域覆盖研究方法有很多类似的地方,不同点在于:点覆盖需要的是邻居节点和目标点的分布信息;而区域覆盖则还需要目标区域的几何特性等方面的数据。



2.2.2 按节点的部署方式分类

无线传感器网络的第一步是如何部署才能确保无线传感器网络高效执行任务,这也是无线传感器网络研究领域的一个基本问题。按照传感器节点部署方式分类,可以将无线传感器网络的覆盖问题分为确定性覆盖和非确定性覆盖或者随机覆盖。

1. 确定性覆盖

确定性部署通过机器人或者人工安装的方式将节点放置预定的位置,适合节点没有移动能力且监测区域安全的应用环境。确定性覆盖则指在无线传感器网络的状态相对固定或无线传感器网络环境已知的情况下,根据预先配置的传感器节点位置,确定网络拓扑情况或增加关键区域的传感器节点密度的一种覆盖问题。可以看

出,确定性部署的覆盖问题需要通过算法计算节点出预设位置,该类算法必须预先设计并在部署网络前提前运行的,数据按照预先确定的路径进行路由。因而,确定性部署的覆盖问题实质上就是一种特性的网络路径规划问题。

典型的确定性覆盖有确定性区域/点覆盖、基于网格的目标覆盖和确定性网络路径/目标覆盖三种类型。确定性区域/点覆盖是指已知节点位置的无线传感器网络要完成目标区域或目标点的覆盖,其中,与确定性区域/点覆盖相关的一个著名计算几何问题为艺术馆走廊监控问题。基于网格的目标覆盖是指当地理环境情况预先确定时,利用二维(或三维)的网格对网络进行建模,位于网格交叉点上的传感器节点实现区域/目标的覆盖。这是一种最为典型的均匀覆盖方式。确定性网络路径/目标覆盖特别考虑了如何对穿越网络的目标或其经过的路径上各点进行感应与追踪。

2. 随机性覆盖

由于无线传感器网络自身拓扑动态变化,导致确定性覆盖在实际应用中具有很大的局限性,不能适用于战场等危险或其它环境恶劣的场所。对于应用于危险监测区域内的无线传感器网络,只能采用随机方式部署的无线传感器网络。采用以随机的方式部署的无线传感器网络的覆盖问题称为非确定性覆盖或者随机覆盖,这种覆盖问题也需要设计算法来计算节点在监测区域内传感器节点的最小分布密度。由于随机的方式部署的无线传感器网络以 Ad hoc 模式而形成网络拓扑结构,因而,从能量有效和网络性能的角度分析,接收/发送器或者簇首的位置是至关重要的。

随机节点覆盖研究在无线传感器网络中节点随机分布且节点位置未知的条件下,网络完成对监测区域的覆盖任务。目前,无线传感器网络的随机性覆盖已成为无线传感器网络覆盖的一个热点问题,此类问题大致可分为随机节点覆盖和动态网络覆盖两类。目前关于此类问题的研究内容最为多见。有别于一般传感器节点一经部署位置固定不变,动态网络覆盖则考虑部分传感器节点具备一定运动能力,主要应用于节点密度稀疏或部分节点能量不足的监测场景中。该类网络可以动态完成相关覆盖任务,也称作移动辅助覆盖。

2.3 覆盖问题的相关研究现状

由于覆盖问题在无线传感器网络研究中的重要性,近年来,在覆盖问题方面,已有不少的研究工作积累了一批研究成果。下面从覆盖对象分类的角度对当前的相关研究成果进行归纳,分别介绍区域覆盖、栅栏覆盖和点覆盖相关的研究现状。

2.3.1 区域覆盖

目前,较为成熟的区域覆盖方法是从优化节点调度,延长网络寿命的角度出发,

采用轮换“活跃”和“休眠”节点的节能覆盖方案。而此节能覆盖方案的关键就是要在保证一定网络区域覆盖要求的条件下，最大化轮换节点集合数目。另外，考虑遗漏或薄弱区域的覆盖不充分问题，将移动节点引入无线传感器网络覆盖中，通过对原有的节点部署进行动态调整，实现动态适应实际需求的覆盖过程。

Slijepcevic^[44]提出将密集型无线传感器划分为互不相交的点集合，这样的每个节点集合可以完全覆盖目标区域。通过不同节点集合之间的交替工作，使得在任意时刻只有一个节点集合处于工作状态，而其它节点集合中的节点则处于低功耗的睡眠状态。显然，互不相交的集合越多，网络的生存时间越长。实质上，寻找最大数量的无交叉节点集问题可归结为 NP 完全的集合 K 覆盖问题，为此，Slijepcevic 设计了一种集中式启发式算法，这种算法先将网络覆盖区域划分为若干个区域 (field)，每个域中的点被相同的传感器节点覆盖。如果覆盖某个域的传感器节点数最少，则称之为临界区域。该算法的基本思想是尽量避免将覆盖临界区域的节点划分到同一个节点集合中。

Berman^[45]改进了 Slijepcevic 的这种思想，该算法不限制每个节点只能属于一个传感器节点集，通过将每个传感器节点分配给定的工作时间，只要不超过单个节点生存时间，使各节点集的工作时间的总和最大。由于某些应用只要求覆盖率达到预先设定的阈值即可，Berman 也进一步研究了在这种部分覆盖网络节点划分以及节点集的工作时间调度问题。

Abrams^[46]讨论了上述 SET-K COVER 问题的一个变种。其基本思想是将节点划分为若干互不相交的集合，但不要求每个集合必须完全覆盖整个目标区域，而是使得目标区域中多的区域被尽可能多的节点集合覆盖。Abrams 为此提出了随机法、集中式贪婪法和分布式贪婪法 3 种节点集的划分算法。

Fan Ye^{[47][48]}等提出了一种基于环境探测和自适应休眠 PEAS (Probing Environment and Adaptive Sleeping) 的算法。这种协议希望在恶劣的环境下工作的无线传感器网络保持恒定的覆盖度，并能尽可能节省能量以达到延长网络的寿命。PEAS 协议执行的前提是网络节点密度很大，由于传感器节点的计算和存储能力较低，对于复杂的协议，可能由于外界因素而导致节点频繁失效的现象出现。

PEAS 协议的基本思想假设节点最初处于睡眠状态，当一个随机时间间隔到达后，唤醒传感器节点。这个被唤醒的节点然后探测预设范围内是否存在其它已唤醒传感器节点，如果存在，则继续进入睡眠状态，否则进入活跃状态。而处于工作状态的传感器节点将持续工作至能量耗尽为止。通过调整探测范围和唤醒间隔，可以实现不同的网络覆盖质量，PEAS 算法不依赖节点的精确位置信息，计算开销小，但不能保证网络的覆盖质量。

PEAS 协议通过保留必要数量的工作节点，而其它节点则休眠从而使无线传感

器网络的寿命得以延长。在 PEAS 协议中, 休眠节点会不时地自动唤醒然后探测预设范围内的其它节点的工作状态, 然后根据预设的规则调整自己的状态, 以保证整个无线传感器网络正常工作; 节点休眠的时间长度依靠自适应调节来适应节点密度的变化。

Tian^{[49][50]}提出了一种本地化、分布式传感器节点调度算法。该算法的基础是必须知道邻居节点间的位置关系, 进而判断有哪些节点是冗余节点。但该算法在判别节点的冗余性时, 仅考虑了位于每个节点覆盖范围内的邻居节点, 其性能有待进一步提高。Tian 提出了一系列只能保证部分覆盖的节点调度算法: 基于邻居数量的调度算法、基于最近邻的调度算法以及基于概率的调度算法。

Yah^[51]则提出了一种可以为监测区域的不同点提供不同覆盖质量的节点调度算法。这算法将网络生存时间划分为若干等长的时间槽, 在每个时间槽内, 传感器节点首先随机产生一个位于该时间槽内的参考时间点。此外, 该算法将监测区域离散化成若干栅格 (grid) 点, 通过所有的栅格点都能是否能被充分覆盖来判断区域是否能被覆盖。对于落在传感器节点覆盖范围内的每个栅格点, 根据自身的参考时间点以及该栅格点被邻居节点覆盖的情况, 从而决定相对于该栅格点的工作周期。节点在每个时间槽内的工作周期为针对其覆盖范围内全部栅格点的工作周期的并集。在这个协议中, 每个传感器节点通过动态地决定自身的调度以保证确定的覆盖度。

A.Howard^[52]把势场技术应用到移动无线传感器网络环境中, 这种技术将每一个移动节点虚拟成带电粒子, 假设相邻节点之间以及节点与障碍物之间存在排斥力和吸引力两种相互作用力, 且每个节点都会受到“摩擦力”的影响。由于受势场力的作用, 节点会在受“力”作用下移动一定的距离, 使传感器节点迅速扩展开来, 直到全网所有节点都处于平衡状态充分地覆盖整个监测区域为止。

Zou Y^[53]提出了一种节点在执行算法时首先计算出其运动路径的部署方法, 当移动的最终位置确定后, 该节点直接移动到计算出的位置即可。虚拟力算法是基于势场方法的一种扩展, 无线传感器网络节点随机部署后能自动完善网络覆盖, 以保证无线传感器节点覆盖的范围最大。在算法执行的时并不移动传感器节点的地理位置, 通过算法计算出随机放置的传感器虚拟运动路径并传感器位置确定后, 才对相应节点进行一次移动操作, 实现网络覆盖范围最大化。

Heo N^[54]则根据节点间的相互虚拟作用力, 提出一种基于分布式节点自适应部署策略。该策略是一种基于受势场力和分簇结构的节点自部署算法, 执行算法时, 簇首节点根据传感器邻居节点间、节点与障碍物以及覆盖目标之间的引力和排斥力的合力情况, 计算本簇内其它成员节点的目的位置。所有的节点根据其周围邻居节点的分布密度以及它们的剩余的能量, 自适应地调整各自的工作模式, 从而使全网

能更有效地利用节点的能量。但这些部署方式的不足之处是无法保证达到预定的覆盖度。

G.Wang^[55]考虑了由静态节点和可移动节点组成的混合无线传感器网络中可移动节点的自部署问题。当静态节点检测到所在的区域存在覆盖盲区后,就向其邻近的可移动节点发出发现覆盖盲区的信息,要求可移动节点移动至相关位置消除覆盖盲区,由于一个可移动节点同时可能会时收多个不同静态节点所发出的信息,该文献提出了一种可移动节点向能产生最大覆盖位置移动的竞争协议。

G.Wang^[56]考虑了全部由可移动节点组成的无线传感器网络的自组织重部署问题,提出了一种分布式自部署协议。当传感器节点发现覆盖盲区后,这个协议试图通过计算其该移动到的目标位置,然后利用 Voronoi 图发现覆盖盲区。该文共提出了 3 种辅助移动传感器协议 VEC (VECTorbased)、VOR (VORonoi-based)和 Minimax 协议,这些协议试图将传感器节点从密集型区域移动至稀疏区域,最大程度地减少覆盖盲区的存在。

为减少部署过程中节点移动消耗的能量,G.Wang^[57]设计了一种基于代理的移动无线传感器网络的传感器部署协议,这种协议只有当移动传感器节点最终决定了其移动位置时才真正移动到所需的位置。J.Wang^[58]考虑了在移动无线传感器网络中如何通过节点移动来响应新事件或者消除失效节点的影响。提出了两步走的传感器节点重置方案,首先标示冗余传感器节点然后将它们重新部署到目标区域,为此提出了一种 Grid-Quorum 方法去快速决定最近的冗余传感器节点。

J.Wang^[59]提出了一种通过节点的移动来延长网络生存时间的集中式算法。在 SET-K Cover 问题中,网络的生存时间取决于互不相交节点集个数。该算法首先使用 Basagni S^[60]提出的方法将网络节点划分为若干无交节点集,再通过改变不属于任何节点集的可移动节点位置来将剩余节点组织成尽可能多的无交节点集。

B.Liu^[61]分析了节点移动导致的网络动态覆盖性能的变化。通过节点移动,原来不被覆盖的区域可以被覆盖,不能被检测到的目标可以被检测到。针对 Poisson 随机分布的节点,该文分析了在任意时刻部署区域的覆盖比例以及在某个时间间隔内目标区域被覆盖的比例,还分析了静止目标和移动目标的检测时间,提出了移动节点的最优移动策略。

2.3.2 栅栏覆盖

Meguerdichian^[62]首次提出了无线传感器网络的栅栏覆盖模型。给定目标穿越网络部署区域的起始点和终止点,文献试图确定网络的最优覆盖以及最差覆盖质量。目前主要的解决方法是,定义“最大突破路径”(Maximal Breach Path)和“最大支撑路径”(Maximal Support Path),分别使得路径上的点到周围最近传感器的最小距

离最大化以及最大距离最小化;进而采用 Voronoi 图和 Delaunay 三角测量技术来完成最大突破路径和最大支撑路径的构造和查找。该文定义了 MSP 和 MBP 路径。其中, MSP 路径上的每一点与最近节点的距离最小,而 MBP 路径上每一点与最近节点距离最大。MSP 和 MBP 路径分别对应了最优和最差网络覆盖质量。该文提出了一种基于 Voronoi 划分以及 Delaunay 三角剖分的集中式算法。MBP 路径必然位于 Voronoi 边上,而 MSP 路径则必然由 Delaunay 边组成。Li^[63]针对 MSP 问题提出了一种基于相对邻居图的分布式算法。

目标穿越网络时被检测到的概率不但与目标运动路径相关,还与目标在网络中所处的时间相关。目标在网络中所处时间越长,被检测到的概率越大。Meguerdichian^[64]提出了一种新的基于 exposure 的栅栏覆盖模型。使用概率检测模型,如果目标 p 与节点的距离 d ,则节点 s 对该目标的感知强度为 $S(s,p) = \lambda / [d(s,p)]^k$ 。如果目标沿路径 $p(t)$ 穿越网络的时间为 (t_1, t_2) ,则定义目标的 exposure 值为 $E(p(t), t_1, t_2) = \int_{t_1}^{t_2} I(F, p(t)) \left| \frac{dp(t)}{dt} \right| dt$ 。其中, $I(F, p(t))$ 为目标位于区域 F 中点 $p(t)$ 处时,最近节点在该点处的感知强度或者网络节点在该点处感知强度之和 $\sum_{i=1}^n S(s_i, p)$ 。文献提出了计算最小 exposure 穿越路径的数值近似算法。Meguerdichian^[65]提出了一种计算网络 exposure 值的分布式算法。Veltri 提出了另一种计算复杂性更低的局部化算法。

Adlakha^[66]假设移动目标以固定速度、沿一条直线穿越网络部署区域。每个节点有 2 个与检测能力相关的感知半径 r_1 和 r_2 。节点能够检测到任意与其距离小于 r_1 的目标,而不能检测到任意与节点距离大于 r_2 的目标。Adlakha 讨论了如何在概率感知模型和 exposure 模型下设置节点感知半径 r_1 以及必需的节点数。Adlakha 指出,若目标区域面积为 A ,当节点数为 $O(A/r_2^2)$ 时,网络的检测概率不低于 0.98。

Liu^[67]讨论了基于网格部署以及随机部署的无线传感器网络的覆盖以及目标检测问题,提出了一种基于渗流理论^[68]的临界节点密度。当节点密度小于该临界密度时,以概率 1 的方式存在一条不被发现的网络穿越路径。相反,当节点密度大于该临界密度时,无论目标如何移动,能够以概率 1 的方式保证检测到该目标。Liu^[69]分别讨论了二元感知模型以及概率感知模型下,节点分布满足 Poisson 点过程的随机无线传感器网络的覆盖比例、保持网络原始覆盖质量所能关闭的节点数以及网络对移动目标的检测能力与节点密度之间的关系。

2.3.3 点覆盖

对若干离散目标点的覆盖问题属于点覆盖考虑的范围;Cardei^[70]研究了监测对

象为离散目标点的场景下的节能覆盖问题。这种场景常常出现在军事领域中,将大量的节点随机部署在若干个离散目标点附近。如果将网络节点划分为若干个互不相交的节点集合且每个节点集合能够完全覆盖这些离散的目标点,那么,通过周期性地调度这些节点集合,在任何时刻只有一个节点集合处于活跃工作状态,其它节点全部处于睡眠状态,就可以有效延长整个网络的生存时间。显然,这样的节点集越多,网络生存时间越长。如何构造最大数量的不相交集是一个 NPC 问题,Cardei^[71]为此提出了一种基于混合整数规划的启发式算法。通过对上述问题进行扩展,不再限制每个节点只能加入一个节点集合,容许一个节点可以同时加入多个节点集。Cardei^[72]进一步讨论了当节点感知半径可变时的能量高效目标点的覆盖问题。

Liu^[73]讨论了如何通过调整节点的感知半径实现能量高效覆盖,同时通过构建虚拟骨干网络保证网络的连通性。

Kar^[74]讨论了当网络部署环境安全、可控时,如何使用最少数量的节点并确定节点位置来覆盖 n 个给定的目标点,并且保证这些节点组成的网络是连通的。假设网络同构,每个传感器节点的感知半径相等并且通信半径等于感知半径时,本文提出了一个基于最小生成树的多项式近似算法。

2.4 小结

无线传感器网络的覆盖问题就是通过节点调度以及路由选择等手段,优化分配无线传感器网络的各种资源,改善感知和通信服务质量。给定一个无线传感器网络,覆盖也可以一般性地认为是利用各个节点协作以达到对监视区域/目标的不同覆盖效果。区域覆盖关心的通过覆盖优化最大化覆盖区域,消除覆盖盲区,而点覆盖则关心的是离散目标的覆盖质量,避免出现盲点,二者的共同点都是通过节点调度等手段,使冗余节点交替工作,以延长整个网络的生存时间。栅栏覆盖关注的则是无线传感器网络对移动目标的检测能力,要求当某个目标沿着任意路径穿越部署无线传感器网络的区域时,不被该网络的节点所侦测,或者被至少一个节点所侦测。每个具体的覆盖问题适应不同的应用场景,对于更进一步研究无线传感器网络具有非常重要的指导意义。目前主要着重于区域覆盖的研究,没有充分考虑到大规模无线传感器网络的点覆盖问题的研究。尽管点覆盖问题与区域覆盖问题二者有很多相似之处,但区域覆盖以计算几何作为理论基础,这一点不适合点覆盖问题的研究;区域覆盖考虑的是连续区域的覆盖,而点覆盖问题研究的是静止且离散的目标,除了网络节点的分布,还必须考虑目标的分布情况,这使得用于研究区域覆盖问题的节点的通信半径与感知半径之间的关系也不适合点覆盖问题。因此,很有必要将点覆盖问题与区域覆盖问题分离开来单独进行研究。本文将以布尔感知模型为基础,侧重研究大规模无线传感器网络的点覆盖与连通问题以及相应的算法。

第三章 能量均衡的节点自调度覆盖算法

节点调度技术的动机是：首先，无线传感器网络是一种节点能量受限网络，通过节点调度优化能量管理是有效延长网络生存时间的重要方式之一。其次，在实际无线传感器网络中，节点规模常常庞大，集中式算法需要全局信息，实施算法时会加重网络负载，因而集中式算法的可扩展性和实际可操作性不强。最后，当无线传感器网络部署在敌区或者危险的区域后，手工配置传感器节点是很不方便甚至不可能的。

从上面分析可知，无线传感器网络作为一种基于数据资源的分布式网络，在固定目标监测应用中，研究无线传感器网络点覆盖问题的节点调度技术很有必要。一方面，本章结合现有关于节点调度方式的特点，针对点覆盖问题提出了一种基于能量均衡的节点自调度冗余覆盖分布式算法—SRCA(Self-scheduled Redundant Coverage Algorithm)。另一方面，目前的研究局限于每个节点的感知单元单一的情形，即每个节点只具有一种感知半径。随着硬件技术的发展，单个传感器节点可以配置多种类型的传感器，这样，每个节点能同时监测多种物理现象。本章将这种配置多种类型的传感器的节点称为多功能传感器节点(Multifunctional Sensor Node)；由多功能传感器节点组成的无线传感器网络称为多功能无线传感器网络。由于多功能传感器节点的研制，使得某些无线传感器网络应用场景中的节点可以同时感知多种物理对象。基于这种多感知功能节点的无线传感器网络点覆盖问题，本章也提出了一种基于多功能传感器节点能量的调度算法 DSAMSN (Distributed Scheduling Algorithm based on the Multifunctional Sensor Node)。然后分别对 SRCA 算法和 DSAMSN 算法进行了仿真实验，通过对该算法的有关性能进行了评价，验证该算法的有效性。

3.1 引言

为了满足实际应用需求，在部署无线传感器网络时，往往会存在许多冗余结点。如果将所有结点始终处于活动状态，不仅产生许多冗余数据，而且由于竞争无线通道，甚至可能导致网络拥阻，从而造成能量浪费。况且，对于这种随机部署在离散的目标点附近的无线传感器网络，很难接近这些目标且节点数量较大，电池更换是不可行的。那么如何合理使用节点的有限能量，尽可能延长全网的生存时间，这是无线传感器网络的一个重要的研究领域。而通过调度传感器节点的状态（活动状态或者睡眠状态），在保证覆盖要求的基础上减少活跃节点数量，则是延长无线传感器网络生存时间的一种重要方式。所谓节点调度就是在不降低网络覆盖要求的前提

下,将一些冗余结点暂时设置为休眠状态,当需要时再唤醒它们成为活跃节点,以延长网络的寿命。

假设节点具有活跃和非活跃两种基本状态,节点调度方式就是分配每个节点的工作状态。非活跃状态的节点处于低能耗的休眠状态而不参与感知任务。对于网络图 $G(V,E)$,通过节点调度后,节点 V 被分成两个部分: $V_{inactive}$ 与 V_{active} , 这样 $V = V_{active} \cup V_{inactive}$, 在每一轮工作时间内,由 V_{active} 承担感知和通信任务。

伴随着现代电子技术的发展,无线传感器节点采用了低功耗电路系统,由于节点具有多种离散工作状态,其微处理器通常也使用了多工作状态,如 Intel StrongARM 具有 3 种状态。在正常模式下,处理器的功耗达到 400mW;在空闲模式下,由于处理器的时钟控制器关闭,功耗则只有 100mW;而在休眠模式下,处理器只有实时时钟工作功耗只有 $50 \mu W$ 。微处理器的这些特点非常有利于节点的控制,为无线传感器节点的合理调度提供了硬件基础。

无线传感器网络的覆盖度是指覆盖监测区域的每个点/目标的平均传感器节点数。为了提高无线传感器网络的可靠性,往往在部署网络时抛撒大量的传感器节点,这种高节点密度的大规模无线传感器网络具有较高覆盖度。当目标点发生事件时,致使这些传感器节点产生大量的冗余数据,传感器节点在处理和传输这些数据时,会造成不必要的能量消耗,而且在传输数据时的无线信号也容易相互干扰,降低数据的可靠性。而利用传感器节点的高冗余特性,通过调度传感器节点的工作状态,关闭众多的冗余节点,可有效减少节点的能耗。由此可见,研究节点调度方式延长网络的生存时间完全有必要并且是可行的。

3.2 相关工作

关于无线传感器网络应用于点覆盖问题,现有的算法都是将它归纳为数学规划问题,然后由中心节点完成算法,这些算法大多假设目标点和传感器节点是均匀分布,从而采用构造不同的传感器集的方式。

M. Cardei 等^[70]和 S. Slijepcevic 等^[75]将无线传感器网络分成若干能独立完成覆盖任务的传感器节点集合,通过设计构造最大独立集的算法,调度各节点集工作以实现延长系统生存时间。但是,无论是文献^[70]还是文献^[75]提出的构造支配集的算法,假设基于每个支配集中的节点都均匀消耗能量。现实中的无线传感器节点,由于规模大,在使用这种集中式算法过程中,本身就得消耗大量的能量,往往还需其它的协议进行辅助才能完成,如节点分簇协议以及节能的路由协议等。

PEAS^[48]算法针对恶劣环境下无线传感器网络的应用,该算法是一种基于探测的密度控制算法。即每个睡眠节点在一定时间内、在其探测范围内发送广播消息

PRB, 任何收到消息的工作节点返回应答消息。若节点收到应答消息, 则进入睡眠状态, 工作节点继续工作, 直至其能量耗尽或物理失效; 否则睡眠节点立即进入工作状态。为了保证网络对监测区域的覆盖度, 这个协议通过仅保留必要数量的节点工作, 其它的传感器节点则休眠, 但是为了感知本地的网络环境, 休眠节点仍需不时地自动唤醒。如图 3.1 (a) 所示, PEAS 算法假设无线传感器网络的节点密度很大。该算法实现比较简单, 不需知道传感器节点精确的位置。由于节点状态的转化带有很大的随机性, 而且 PEAS 算法中的某些节点可能持续工作。导致其过早地死亡, 整个网络中的节点能耗不均匀, 从而使得 PEAS 算法很难保证覆盖质量。

D. Tian 和 N. D. Georganas^[76]提出了基于节点状态调度的分布式覆盖算法, 其中的节点休眠资格 (off-duty eligibility) 规则可以根据节点的物理位置信息或利用有向天线获得信号达到角度来计算节点与其邻居的覆盖关系, 选取工作节点。节点休眠资格规则只是针对区域覆盖的情形, 没有考虑无线传感器网络的点覆盖问题。

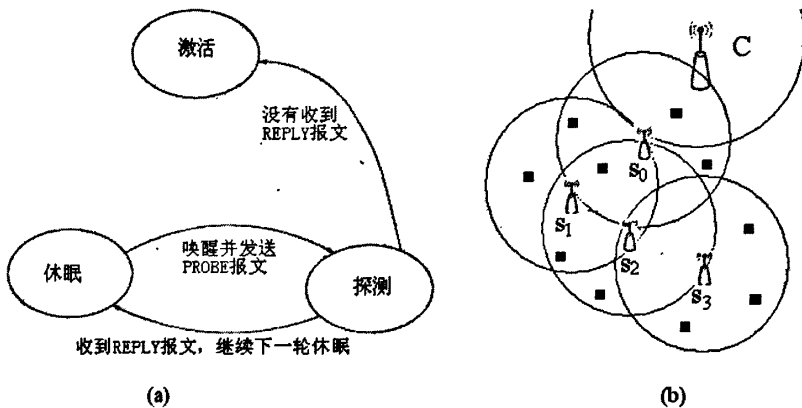


图 3.1 (a) PEAS 算法的节点状态转化图 (b) 引起节点能耗不均匀的图示

然而, 现实中的无线传感器网络与这种假设有很大的出入。由于传感器节点所处的位置不同, 在网络运行过程中, 不同的传感器节点可能承担的通信和监测目标的负荷也不同, 使得节点能耗不均匀。甚至一些节点还可能是承担更多的中继任务的主干节点, 它们的通信负荷比较大, 因而能耗更大, 导致即使其它的节点剩余足够的能量, 而这些中继节点却因能量耗尽而死亡, 使得网络的覆盖和数据传输质量可能降低。如图 3.1 (b) 所示, 传感器节点 s_0 与 s_2 除了必须监测目标的任务外, 还必须承担传感器节点 s_1 和 s_3 的数据中继任务, 这样, 传感器节点 s_0 与 s_2 比 s_1 和 s_3 能耗要高。当节点 s_0 或 s_2 的能量消耗至无法继续工作后, 节点 s_1 和 s_3 也将失效, 无法保证整个网络的服务质量。

从上面的分析也可以看出, 在无线传感器网络里, 传感器节点间的负载是很难

均衡的。但是,利用节点冗余特性,将整个网络按照工作时间划分为多个轮回(round)时间,通过网络节点的自组织将传感器节点分配不同的轮回工作。这样,即使在某一轮工作时间内,传感器节点间的负载可能是不均衡的,但从网络的整体工作时间内则可以降低节点间的负载不平衡的程度。

3.3 节点冗余覆盖自调度算法—SRCA 算法

根据点覆盖的特点,本节提出了一种分布式、局部化的冗余覆盖判别方法以及基于节点优先级的节点调度机制。首先提出 SRCA 算法网络模型,然后基于该模型提出了一种分布式节点调度算法,

3.3.1 网络模型

考虑 n 个传感器节点 $S=\{s_1, s_2, \dots, s_n\}$ 随机分布在目标区域 A 内,部署在目标附近监测该区域内的 m 个目标 $T=\{t_1, t_2, \dots, t_m\}$ 。关于网络模型的基本假设如下:

- 每个目标发出的事件至少能被一个传感器节点感知;
- 每个传感器节点都能监测到本感知域内发生的事件;
- 传感器节点采用全向天线的同构网络,即网络中每个节点拥有相同的感知半径、初始能量以及通信半径;
- 每个节点的位置信息已知。节点位置信息可以通过 GPS 或者其它定位服务获取。
- 网络中所有节点时钟同步。

定义 3.1 冗余覆盖节点

如果传感器节点的感知域中的目标能被至少一个异于自身的传感器节点所覆盖,该节点属于冗余节点。

如果要求每个目标能被至少 k 个节点覆盖,那么, k -覆盖冗余节点定义为:如果传感器节点的感知域中的目标能被至少 k 个异于自身的传感器节点所覆盖,该节点属于 k -覆盖冗余节点。

在图 3.2 中,目标 t_1, t_2, t_3 分别被节点 s_1, s_6, s_4 覆盖,但是,节点 s_2 同时覆盖这 3 个目标,节点 s_2 为 1-覆盖冗余节点。

定义 3.2 覆盖相关节点

传感器节点 s_1, s_2 所覆盖的目标的集合分别为 C_1 和 C_2 。如果 $C_1 \cap C_2 \neq \Phi$, 那么节点 s_1, s_2 覆盖相关。

例如:在图 3.2 中,节点 s_2 分别与节点 s_1, s_6, s_4 覆盖相关。

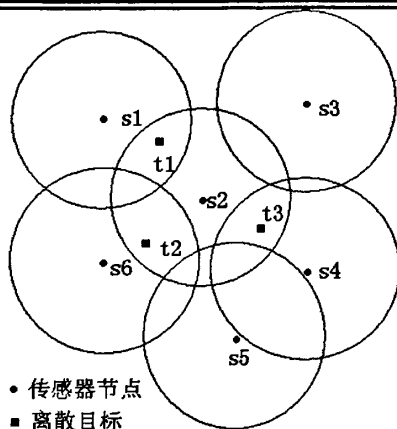


图 3.2 覆盖冗余节点图示

3.3.2 SRCA 算法

SRCA 算法将网络的生存时间划分为不同的时间段，每个时间段由节点调度阶段与活跃节点工作阶段两部分组成。如图 3.3 所示，为了减少能量开销，节点调度阶段的时间要比活跃节点工作阶段所占的时间短。这样设计 SRCA 算法执行的

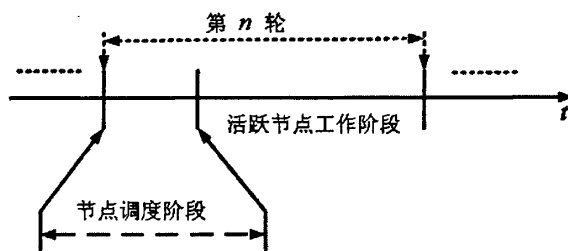


图 3.3 SRCA 算法执行的时间线

时间线就是为了尽可能延长每一轮的工作时间，减少传感器节点自调度阶段的能耗。节点调度阶段又包含邻居发现阶段和通过冗余节点发现的方式确定节点状态两步骤。SRCA 算法规定所有被选举为冗余覆盖节点的节点都休眠，其余的则为活跃节点。活跃节点在工作阶段执行监测目标对象的任务，并将采集的数据发送至 sink 节点。

3.3.2.1 邻居发现

在节点调度开始阶段，由于在节点的冗余覆盖判断时需要考虑所有覆盖邻居节点，每个节点通过分布式地收集邻居信息，获取每个节点的一跳邻居节点的 ID、

剩余能量以及当前位置信息。

每个节点通过向其邻居广播 PIM(Position Information Message)报文告知位置信息。PIM 报文包含了节点 ID、节点剩余能量以及当前位置信息。由于可能出现多个节点同时发送 PIM 消息的情况，为了避免导致信道冲突而引起的消息丢失，发送 PIM 报文前，每个节点必须预先随机等待一段时间 τ_r ，而设置邻居发现阶段等待接收 PIM 消息的时长为 τ_w 。同时为了控制这个阶段的能量开销保持最小，当传感器节点在发送 PIM 消息时，只需确保该报文的传输距离达到它的一跳邻居节点即可。这样，每个节点只能接收到其一跳邻居节点发送来的 PIM 报文。而且，在节点邻居信息列表中保存了第一次邻居发现时的节点信息，以后的 PIM 报文中没必要再包含传感器节点的位置信息。

3.3.2.2 确定节点状态

邻居发现之后，接着每个节点通过冗余覆盖判断，确定自身的状态。在这阶段，每个节点自主判断是否满足冗余覆盖的条件。

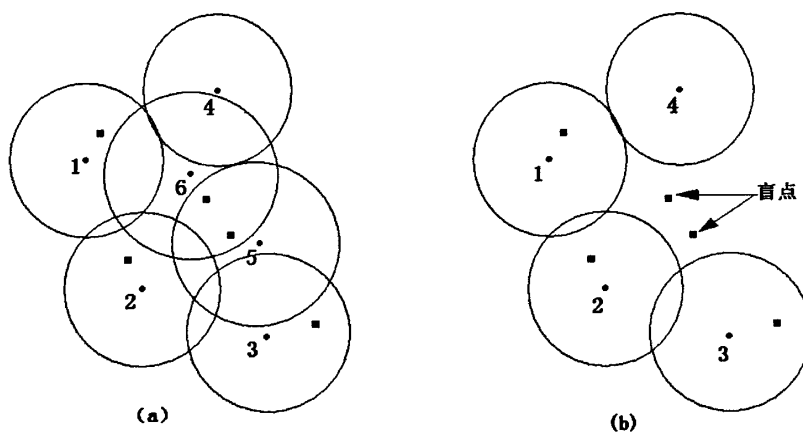


图 3.4 (a) 冗余覆盖节点在初始覆盖中图示 (b) 出现覆盖盲点图示

当传感器节点判断自己为冗余节点时，还必须判断自己是否真正有资格立即关闭其通信、感知模块进入睡眠状态。因为，这种情况有可能出现所有节点同时做出进入睡眠状态的决策，从而可能会导致出现覆盖盲点。如图 3.4 所示，传感器节点 5 和节点 6 都认为自己是冗余覆盖节点，节点 5 相信节点 6 是活跃工作状态，而节点 6 相信节点 5 是活跃工作状态。根据节点休眠的条件，传感器节点 5 和节点 6 都会使自己休眠，这样就会出现覆盖盲点。

为了避免上述情况出现，SRCA 算法解决这种问题的策略是：如果判断自己是

冗余覆盖节点，节点 s 是否休眠还必须等待一段时间 $T_{off}^{(s)}$ ，当定时 $T_{off}^{(s)}$ 结束时才将自己休眠。对于判断自己是冗余覆盖的节点，其策略将在下一小节详述。

3.3.2.3 确定传感器节点状态调度策略

SRCA 算法解决覆盖盲点问题的策略为节点剩余能量均衡优先调度机制。如果传感器节点 s 、 v 的剩余能量为 e_s, e_v ，它们的覆盖相关节点的平均剩余能量分别为 E_s, E_v ，那么，当它们判断自己是冗余覆盖节点时，分别设置定时器

$$T_{off}^{(s)} = K(1 - |e_s - E_s| / E_s) t_s \quad (3-3)$$

$$T_{off}^{(v)} = K(1 - |e_v - E_v| / E_v) t_v \quad (3-4)$$

在(3-3)(3-4)式中， $t_i (i = s, v)$ 为节点 i 的当前时间，而 K 是系统的调节参数，可根据实际应用设定。

节点休眠调度机制具体描述如下：

- 当 $T_{off}^{(v)} \neq T_{off}^{(s)}$ 时，节点休眠以定时器为准。
- 当 $T_{off}^{(v)} = T_{off}^{(s)}$ 时，以节点的 ID 号为优先休眠机制。

设置 $T_{off}^{(i)} = K(1 - |e_i - E_i| / E_i) t_i$ 的目的是尽量实现活跃节点间的剩余能量均衡，使活跃节点的剩余能量尽可能不低于平均节点的剩余能量，降低活跃节点在一个轮回里提前耗尽能量的概率。判断自己是可能的冗余覆盖的节点，其剩余能量与其相关节点的平均剩余能量越接近，休眠等待时间越长，从而休眠的概率越小。对于 $T_{off}^{(v)} = T_{off}^{(s)}$ 的情况，由于节点 s 、 v 的剩余能量相当，则根据它们的 ID 决定是否休眠，这样处理并不破坏上面所讨论的剩余能量均衡原则。

如果节点 s 准备休眠，必须在休眠前发送状态休眠报文 SSM (Status Sleeping Message) 通知其一跳邻居节点。如果在定时 $T_{off}^{(s)}$ 结束前收到来自其它节点的 SSM 报文，则立即设置为活跃节点。然后标记发送节点为休眠节点，并将该节点从它的活跃邻居节点列表中删除。

3.3.2.4 SRCA 算法

完成节点自我调度后，每个节点确定了属于休眠节点还是活跃节点，所有的活跃节点进入活跃节点工作阶段。如图 3.4 所示，活跃节点在保持一段时间后，睡眠节点被唤醒，全部节点重新开始新一轮的节点调度。

在 SRCA 算法中，每个可能的冗余覆盖节点覆盖相关节点的平均剩余能量，与相关节点的平均能量越接近，则休眠的概率也就越大；而邻居节点越多则休眠的概

率也就越大。

节点自调度算法的执行描述如下：

- (1) 发送 PIM 报文并收集邻居节点信息。
 - (2) 等待一断时间 τ_w ，使得传感器能充分收集邻居节点信息。
 - (3) 判断是否是可能的覆盖冗余节点，如果是则定时 $T_{off}^{(i)} = K(1 - |e_i - E_i|/E_i)t_i$ ，否则转换为活跃的工作状态。
 - (4) 对于可能的覆盖冗余节点，在定时 $T_{off}^{(i)} = K(1 - |e_i - E_i|/E_i)t_i$ 期间监听通信通道，如果接收到 SSM 报文，则标记发送节点为休眠状态并从活跃邻居节点列表中删去发送节点。重新判断自己是否仍属可能的覆盖冗余节点，如果是可能的覆盖冗余节点则继续监听通信通道，否则立即转入活跃的工作状态。
 - (5) 当定时 $T_{off}^{(i)} = K(1 - |e_i - E_i|/E_i)t_i$ 结束时，向相关邻居节点发送 SSM 报文并进入休眠准备期 τ ，继续监听通信通道，在此期间，如果接收到 SSM 报文，则标记发送节点为休眠状态并从活跃邻居节点列表中删去发送节点。重新判断自己是否仍属可能的覆盖冗余节点，如果则继续监听通信通道，否则立即转入活跃的工作状态。
 - (6) 当休眠准备期 τ 结束时，立即转入休眠状态。
- SRCA 算法伪码描述如表 3.1 所示。

表 3.1 SRCA 算法伪码

Each node performs v self-scheduling algorithm

1. Compute the coverage information C_v
2. $\Omega_v \leftarrow C_v$;
3. Generate a random delay time τ_r ;
4. Waiting for time τ_r ;
5. **If** $\tau_r = 0$ **then**
6. Node v broadcasts a PIM message;
7. **Endif**
8. **While** $\tau_w \neq 0$ **do**
9. Node v is receiving PIM messages;
10. **If** receiving a PIM message from node u **then**
11. Retrieve the information C_u, e_u, L_u from the sender;

12. $\Omega_v \leftarrow \Omega_v - C_u$;
13. **Endif**
14. **Endwhile**
15. **If** $\Omega_v = \Phi$ **then**
16. $\Lambda = \Phi$;
17. **While** $T_{off}^{(i)} > 0$ **do** // the node v is a potential redundancy node.
18. *listen to the channel*;
19. **If** *receiving the SSM messages from node u* **then**
20. *marks u as a sleeping node and delete u from its active neighbor list*;
21. $\Lambda \leftarrow \Lambda \cup C_u$;
22. **Endif**
23. **Endwhile**
24. **If** $\Lambda = \Phi$ **then**
25. *Broadcast a SSM messages to its related neighbor nodes*;
26. **While** $\tau > 0$ **do**
27. *listen to the channel*;
28. $\Delta = \Phi$
29. **If** *receiving the SSM messages from node u* **then**
30. *marks u as a sleeping node*;
31. *delete u from its active neighbor list*;
32. $\Delta \leftarrow \Delta \cup C_u$;
33. **Endif**
34. **Endwhile**
35. **If** $\Delta = \Phi$ **then**
36. *Set status of the node v as a sleeping node*;
37. **Else**
38. *Set status of the node v as an active node*;
39. **Endif**
40. **Else**
41. *Set status of the node v as an active node*;
42. **Endif**
43. **Else**
44. *Set status of the node v as an active node*;
45. **Endif**

注意：为了进一步提高网络的覆盖质量，当节点定时结束并发送 SSM 报文后，

该节点不能马上就休眠，而是继续等待一较短的时间 τ ，等待时间 τ 称为节点休眠准备期。如果节点在休眠准备期内收到 SSM 报文，必须重新判断发送 SSM 报文的节点休眠后自己是否还能休眠。如果是则休眠，否则立即设置为活跃节点。为了节省能耗，进入休眠准备期的所有节点，不管最终转换到哪种状态，都不再发送任何通报自己状态转换的报文。

3.4 性能评估

本节通过仿真实验对节点冗余覆盖自调度算法的性能进行评估。考虑部署在二维平面上的静态无线传感器网络，50 个离散目标分布在一个 $50\text{m} \times 50\text{m}$ 的矩形区域。网络中所有节点同构，节点的感知半径 $r_s = 10\text{m}$ ，无线通信半径 $r_c \geq 10\text{m}$ ，采用随机拓扑，网络节点随机分布在目标区域内。我们首先将 SRCA 算法与 PEAS 协议进行比较；假设均采用 GRAB^[77] 协议转发报文，考察分别使用节点冗余覆盖自调度算法和算法 PEAS 时，网络生存时间的变化。所有实验均使用 NS2 模拟平台。

3.4.1 SRCA 算法与 PEAS 协议的目标覆盖质量比较

在这一小节我们将 SRCA 算法与 PEAS 协议进行比较，我们让感知半径从 20m 到 60m 的取值，让每个节点的初始能量在 50J~60J 之间随机选择，让 SRCA 算法与 PEAS 协议在 50 个不同的随机网络拓扑下各自运行，通过统计在不同的感知半径下运行协议后所能关闭的节点数量、每个目标的平均覆盖度以及对网络覆盖盲点数，每个目标的平均覆盖度就是平均每个目标对象被传感器节点的个数，即第 i 个目标点覆盖的传感器节点数为 d_i ，对于 m 个离散目标，则每个目标的平均覆盖度为 $\sum_{i=1}^m d_i / m$ 。根据这个定义方式，仿真将通过比较 SRCA 算法和 PEAS 协议运行结果，以评价 SRCA 算法的性能。

PEAS 协议随机选择节点进行探测，直到网络中每个节点都确定其最终状态。而在执行 SRCA 算法时，假定每个节点的传输半径 $r_s \leq r_c \leq 2r_s$ ，每个节点接收来自范围为 r_c 的全部邻居节点的位置信息，然后进行节点冗余覆盖判别，根据每个节点剩余能量与其覆盖相关节点的剩余能量的平均值间的关系，确定已经判别自己为覆盖冗余的节点是否休眠。这样，在所有传感器节点都确定了工作状态后，再统计网络中的活跃的传感器节点数量和被休眠的传感器节点数以及每个离散的目标的覆盖度。另外，利用统计覆盖盲点的数目也从另一方面反映运行协议后网络的覆盖质量，显而易见，盲点数越多，那么网络的覆盖质量也越差。

图 3.5 给出了在不同网络节点数情况下，通过运行两协议后每个目标的平均覆

盖度的比较结果。从图中可以看出，SRCA 算法与 PEAS 协议的平均覆盖度相当。

图 3.6 给出了在不同网络节点数情况下，运行两协议后可休眠的传感器节点数比较结果。从图中可以看出，使用 SRCA 算法明显比使用 PEAS 协议所关闭的传感器节点数多。图 3.7 给出了在不同网络节点数下通过运行两协议后盲点数的比较结果。从图中可以清楚地看出使用 SRCA 算法与 PEAS 协议的盲点数要少，因而 SRCA 算法与 PEAS 协议在网络服务的性能上有所改善。

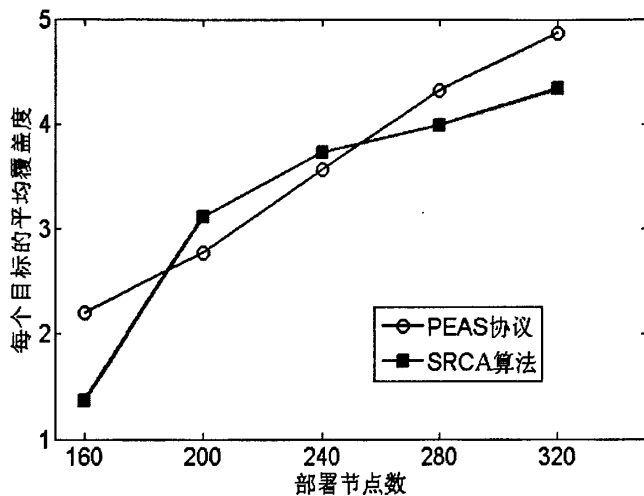


图 3.5 SRCA 算法与 PEAS 协议平均覆盖度比较

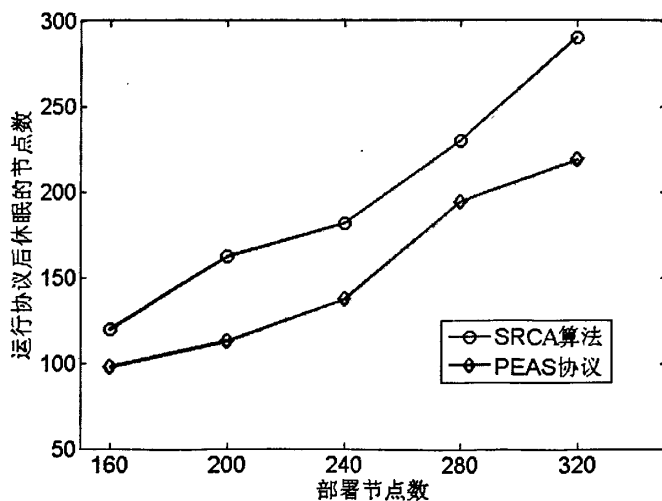


图 3.6 SRCA 算法与 PEAS 协议休眠节点数比较

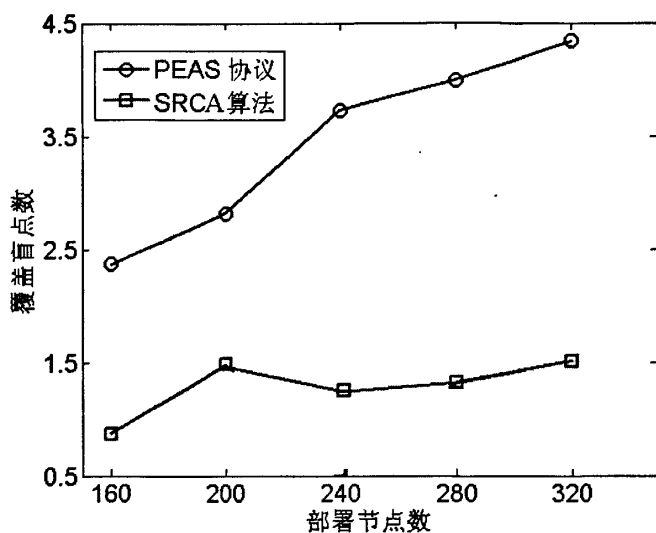


图 3.7 SRCA 算法与 PEAS 协议盲点数比较

3.4.2 网络的生存时间

在这一小节我们将 SRCA 算法与 PEAS 协议的网络生存时间进行比较, 每次由上一小节所述的过程决定所有传感器节点的工作状态后, 假设每个处于活跃状态的工作节点每 10 秒钟向 Sink 节点报告数据, 所有数据的转发采用 GRAB 协议完成, 具体的实验参数参见表 3.2 所示。

表 3.2 传感器节点仿真参数设置

休眠节点能耗	接收报文能耗	发送报文能耗	报文长度	传输速率
0.03mW	12mW	60mW	20Bytes	20Kbps

结合 GRAB 通信协议以及节点能耗模型, 考察 SRCA 算法与 PEAS 协议运行在网络生存时间上的性能比较。在这个仿真实验中, 我们假设 320 个传感器节点随机部署在 $50m \times 50m$ 的矩形区域, 主要考察运行 SRCA 算法与 PEAS 协议后, 在数据传输过程中出现第一个死亡的节点的时间以及网络覆盖率下降到 80% 的时间, 这里的网络覆盖率是指相对运行协议后的网络覆盖率而言。为了考察两者的性能比较, 我们在仿真时有意将网络的生存时间划分为等长的时间片, 这样, SRCA 算法与 PEAS 协议在这一点就统一了。考虑到新的拓扑生成后网络的连通性, 将离散目标采用均匀随机设置的方式且将每个节点的传输半径 $r_c = 15m$ 。

GRAB 协议将基于梯度场的路由协议以及受限广播协议结合起来, 在发送报文时 GRAB 协议在数据包头中插入如下字段:

1. 源节点赋予数据包的 credit 值 α : 每个节点需要计算保留多少 credit 值, 当数据包向 Sink 节点传送时 α 的大小不改变。
2. 源节点的梯度值 C_{source} : 这时一个从源节点发送一个数据包到 sink 节点的代价, 用于计算阈值。
3. 数据包从源节点传输到当前节点消耗的能量总和 $P_{consumed}$: 它用于网络初始时设置源节点广播数据包的代价。
4. 广播该数据包的邻居节点的梯度值 C_{sender} 。

在以下式子中, R_α 代表该节点和后续 hops 仍然可获得的部分 credit 值, R_{thresh} 指本节点到 Sink 节点的距离, 满足 $0 \leq R_\alpha, R_{thresh} \leq 1$ 。每个节点在收到邻居节点的数据广播之后, 将比较 R_α 和 R_{thresh} , 其中:

$$R_\alpha = \frac{REP \times \alpha - \alpha_{used}}{REP \times \alpha}, \quad (3-5)$$

$$R_{thresh} = \left(\frac{C_{receiver}}{REP \times C_{source}} \right)^2 \quad (3-6)$$

在(3-5)式中: $\alpha_{used} = PEP \times P_{consumed} + C_{receiver} - PEP \times C_{source}$

α_{used} 表示已经消耗的 credit 值, REP 表示已经接收到的 report 数据包,

在(3-6)式中, $C_{receiver}$ 表示节点的代价。 $PEP \times P_{consumed} + C_{receiver}$ 则表示节点经过任何路径转发数据包到 Sink 节点所需总的能量开销。

在 GRAB 协议执行时, 当 $R_\alpha > R_{thresh}$ 节点则通过控制发送功率将数据包转发给距离本节点最近的 3 个邻居节点; 而当 $R_\alpha < R_{thresh}$ 则首先利用梯度路由算法得到的单个最优邻居, 然后转发数据包发送给它。如果某个节点收到了同一个数据包的多份拷贝, 则只发送一份拷贝。GRAB 协议将广播范围收缩在从数据源节点到网关节点的一个条状区域中, 该协议既考虑了数据传送的可靠性同时还考虑了网络的能耗问题。但该协议的不足之处在于没有考虑到单个节点具有不同丢包率的特性, 由于将转发邻居数量设置为 3, 因而仅适用于节点丢包率较小的情况。

利用 SRCA 算法与 GRAB 协议相结合(记作 SRCA+GRAB) 与 PEAS 协议与 GRAB 协议相结合(记作 PEAS+GRAB), 以便比较 SRCA 算法和 PEAS 协议对生存时间的性能。SRCA 算法在节点调度阶段同样需要各传感器节点间的传送消息, 必须考虑调度阶段的能量消耗。对于实际无线传感器网络, 节能效果会受到节点感知半径以及初始部署节点数等多种因素的影响; 其次, 每一轮工作时间内数据数据产生和数据转发的次数、每个数据包的长度及节点的能耗模型也会影响节能效果。

由于节点接收数据的能耗与距离无关，只与数据量有关，这里假设每个数据包的大小为 20Bytes，在仿真中计算发送和接收数据包时，只考虑表 3.1 所列的能耗即可，而忽略节点的其它计算能量开销。在仿真中，假设 Sink 节点的坐标为(0, 0)，每个节点的初始能量为 2J。

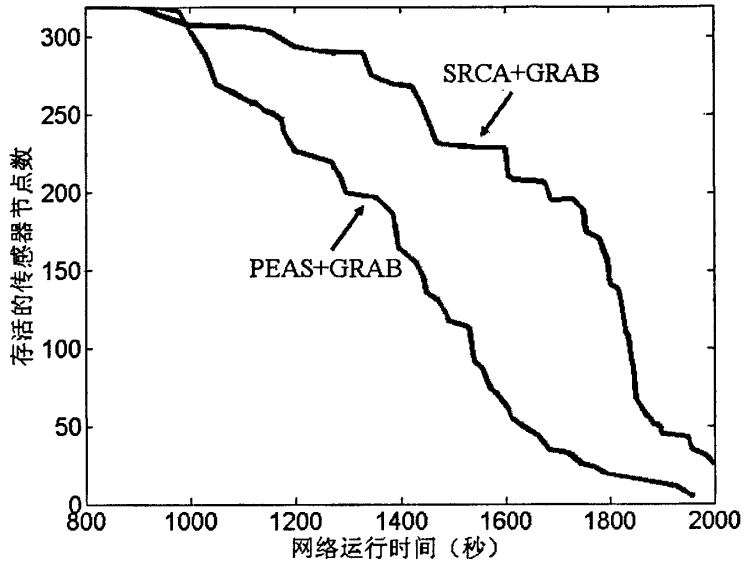


图 3.8 SRCA 算法与 PEAS 协议存活节点数比较

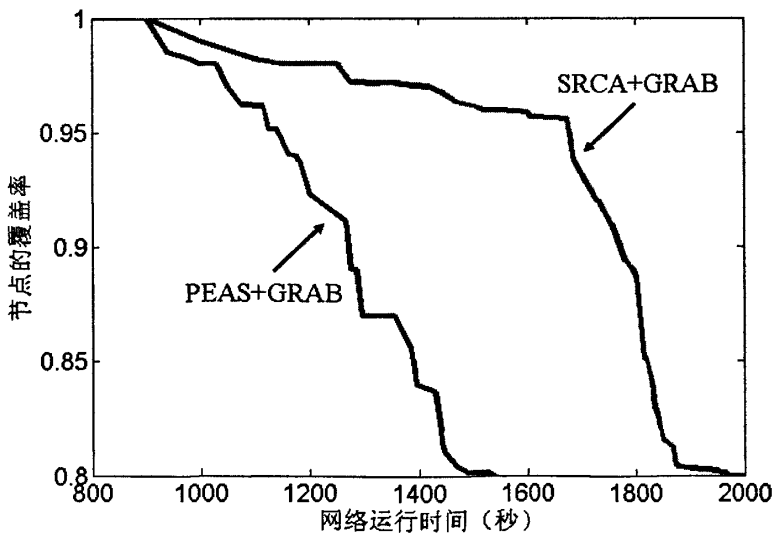


图 3.9 SRCA 算法与 PEAS 协议网络覆盖率比较

图 3.8 是网络中存活节点数量随时间变化的曲线中，从中可以看出，SRCA+GRAB 比 PEAS+GRAB 能量消耗更慢，相同的系统能量能够持续更长的时间。它体现了 SRCA+GRAB 对网络生存时间的有所改进。图 3.9 是网络覆盖比例随时间变化的曲线，它体现了 SRCA+GRAB 比 PEAS+GRAB 对点覆盖率下降到 80% 时各自所花的时间。显然，SRCA+GRAB 能更有效地比较长时间内保持比较好的覆盖质量。

3.5 多功能传感器节点覆盖调度

在很多应用场景目标的监测往往需要采集多种类型的物理量，如图像和声音。为了提高监测数据的可靠性，这种多功能无线传感器网络不仅需要各节点相互协作，而且有时需要各种节点的不同类型的传感器相互协作。这样，当对节点进行调度时，不仅需要调度节点是活跃还是休眠，而且需要控制活跃节点需要打开那些传感器单元。因此，多功能无线传感器网络覆盖问题就是通过算法合理调度各节点的状态以及所有活跃节点的各种传感器的工作，使得在满足覆盖要求的前提下，使网络资源得到合理利用，减少节点的能耗，延长网络的生存时间。

由于多功能传感器节点可以同时感知多种特性的物理对象，如果用布尔感知模型描述这种传感器节点，那么同一个节点具有多种不同的感知半径。针对这种多功能无线传感器网络的点覆盖问题，本节首先构造了这种节点覆盖特性的数学模型，提出了一种基于多功能节点能量的 DSAMSN 算法。

3.5.1 多功能无线传感器网络与优化模型

在多功能无线传感器网络中，每个节点配置了多种类型的传感器，因此，不能像单一传感器的那种网络去描述多功能无线传感器网络的模型。

定义 3.3 多功能传感器节点模型

用向量 $s_k = (s_{k1} \ s_{k2} \ \cdots \ s_{kr})$ 表示一个多功能传感器节点 s_k ，其中 $s_{ki} (i=1 \dots r)$ 表示节点 s_k 的第 i 种类型的传感器。

定义 3.4 多功能无线传感器网络模型

对于由 n 个传感器节点组成的多功能无线传感器网络，用 $S = (s_{ki})_{n \times r}$ 表示这个无线传感器网络。由于每个传感器元件具有不同的感知域，因此，多功能传感器节点的感知域不能用单个物理量来描述。

定义 3.5 多功能传感器节点的感知域

向量 $R = (R_1 \ R_2 \ \cdots \ R_r)$ 表示多功能传感器节点的感知域，其中 $R_i (i=1 \dots r)$ 表示第 i 种类型的传感器的感知半径。

定义 3.6 监测目标的模型

用 $T = (t_{ij})_{r \times m}$ 表示由 m 个监测目标组成的目标集, t_{ij} 表示第 j 个目标所需要监测的第 i 种物理量。

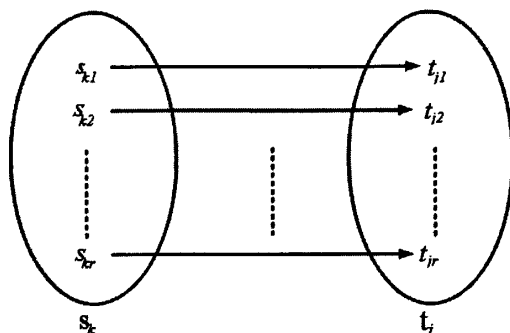


图 3.10 多功能传感器节点感知多类型物理量的目标的对应关系

如图 3.10 所示为多功能传感器节点感知多类型物理量的目标的对应关系, 多功能传感器节点 S_k 有 r 种类型的传感器 $\{s_{k1}, s_{k2}, \dots, s_{kr}\}$, 监测目标 t_j 含有 r 种类型的物理量 $\{t_{j1}, t_{j2}, \dots, t_{jr}\}$, 每种类型的传感器只能感知所对应类型的物理量。根据上面关于多功能无线传感器网络的模型, 下面定义它的优化模型。

对于多功能无线传感器网络 $S = (s_{ki})_{n \times r}$ 和监测的目标集 $T = (t_{ij})_{r \times m}$, 假设 s_{ki}, t_{ij} 均为布尔量, 当 $s_{ki} = 1$ 时表示第 k 个传感器节点配置了第 i 种传感器, 否则 $s_{ki} = 0$; 当 $t_{ij} = 1$ 时表示第 j 个监测目标 t_j 有第 i 种类型的物理量, 否则 $t_{ij} = 0$ 。

假设当前活跃的传感器节点集 $S_k = (x_1, x_2, \dots, x_n)$, 其中 $x_i (1 \dots n)$ 是布尔量, 当第 i 个传感器节点为活跃节点时, $x_i = 1$, 否则, $x_i = 0$ 。假设 c_{kij} 表示第 k 个节点是覆盖第 i 个监测目标的第 j 种类型的物理量, c_{kij} 是布尔变量。根据布尔感知模型的特点, 如果一个目标位于节点的某个传感器元件的监测范围, 那么这个节点就能监测到这个目标的这种类型的物理量, 即当 $\|x_k - t_i\| \leq R_j$ 时, $c_{kij} = 1$, 否则 $c_{kij} = 0$, 其中 x_k, t_i 分别表示第 k 个传感器节点和第 i 个目标的坐标

根据上述模型, 多功能无线传感器网络的覆盖的优化问题可形式化为如下的 ILP(Integer Linear Program)模型表示如下:

$$\text{Objective: Minimize } f(x_1, x_2, \dots, x_n) = \sum_{i=1}^n x_i$$

$$\begin{aligned} \text{Subject to } & \sum_{k=1}^n c_{kj} x_k s_{ki} t_{ij} \geq 1, \quad i=1 \dots m, \quad j=1 \dots r \\ & x_i \in \{0,1\}, \quad i=1 \dots n \end{aligned} \quad (3.7)$$

如果将节点的剩余电量做为参考物理量,可以得到基于节点剩余能量的优化问题的模型如下:

$$\begin{aligned} \text{Objective: Minimize } & f(x_1 \ x_2 \ \dots \ x_n) = \sum_{i=1}^n w_i x_i \\ \text{Subject to } & \sum_{k=1}^n c_{kj} x_k s_{ki} t_{ij} \geq 1, \quad i=1 \dots m, \quad j=1 \dots r \\ & x_i \in \{0,1\}, \quad i=1 \dots n \end{aligned} \quad (3.8)$$

在(3.8)式中, $w_i = 1/(e_i + e_0)$, e_i 是第 i 个节点当前的剩余电量, e_0 是一个小的常数,防止 $e_i = 0$ 时 w_i 无意义而设置的。式中 $\sum_{k=1}^n c_{kj} x_k s_{ki} t_{ij} \geq 1$ 保证所有目标点的属性至少能被一个的节点的传感器单元感知到。上述的目标函数是寻找总的剩余能量较大而节点数比较少的活跃节点集。

3.5.2 基于节点能量的分布式节点调度算法

在 3.5.1 小节提出了多功能传感器节点、多功能无线传感器网络以及相应的覆盖问题的优化模型。基于节点能量的覆盖问题的优化模型可以归结为 ILP 模型,但是无线传感器网络直接求解这样一个 ILP 问题显然不可行。下面将提出一种基于节点能量的分布式节点调度方法 DSAMSN 算法。与 SRCA 算法一样, DSAMSN 算法采用图 3.4 做为节点调度时间线。DSAMSN 算法的调度阶段分为邻居信息收集与指派节点状态两步。

定义 3.7 多功能传感器节点对目标的感知能力

由于不同类型传感器对的相应类型的物理量的感知半径的差异,一个多功能传感器节点在监测目标时,受到所处位置的影响,只能感知监测目标的部分物理量,多功能传感器节点这种能感知到的物理量的个数称为该节点的感知能力。

在节点调度开始前,每个节点首先确定自己的感知信息,并存入信息列表中。每个节点的感知信息在节点调度初始阶段通过 Neighbor 报文告知其邻居节点。节点调度初始化先收集邻居信息,为此,每个节点向其邻居广播 Neighbor 报文,该报文包括该节点的位置信息。然后,设置一个等待时间 *backoff* 接收来自邻居节点的 Neighbor 报文。

当等待时间 *backoff* 结束后,确定每个节点是否为冗余节点,如果是,则向其覆盖相关邻居节点发送 ReadySleeping 报文并进入休眠预备期 τ , 否则,确定需要

打开哪些类型的传感器。当节点的休眠预备期 τ 结束前，如果收到其它覆盖相关邻居节点的 ReadySleeping 报文，在除去发送者后，重新审查自己是否具备冗余节点的资格，如果是则继续等待直到结束休眠预备期，否则标志为活跃节点。当结束休眠预备期时，如果没有收到任何 ReadySleeping 报文，则标志为休眠节点。

DSAMSN 算法根据节点的剩余能量与覆盖能力来确定 *backoff*，这些参量值越大，则 *backoff* 越小，确定为冗余节点的可能性越小。*backoff* 的计算方式如下：

$$backoff \leftarrow \left(1 - \frac{r_i}{2r} - \frac{E_i}{2E_0} \right) T_0 \quad (3.9)$$

在(3.9)式中， E_i, E_0 分别表示第 i 个节点的当前剩余电量和初始电量， T_0 表示节点当前时钟， r 表示目标的物理量的种类， r_i 为第 i 个节点的覆盖能力。在算法中的其它参数的意义如下：

B_{ij} : 第 i 个节点能感知到的第 k 个目标的第 j 种类型的物理量；

C_v : 节点 v 的覆盖信息；

N_v^c : 节点 v 的邻居覆盖信息列表。

DSAMSN 算法具体算法描述如下：

表 3.3 DSAMSN 算法描述

DSAMSN Algorithm is described below:

1. Each node v Computes its coverage information C_v , where $C_v = \bigcup_{k=1}^m \bigcup_{j=1}^r B_{v,ij}$;
2. Broadcast the Neighbor message to its one-hop neighbor nodes.
3. $N_v^c = \Phi$;
4. **While** *backoff* > 0 **do**
5. **If** receiving a Neighbor message from the node u **Then**
6. $N_v^c \leftarrow N_v^c \cup C_u$;
7. **Endif**
8. **End while**
9. $\Omega_v \leftarrow C_v$;
10. $\Omega_v \leftarrow \Omega_v - N_v^c$; //The node v judges whether it is a potential redundant node.
11. **If** $\Omega_v \neq \Phi$ **then**
12. The node v sets the status as an active node immediately;

13. *Else*
14. *Broadcast the ReadySleeping message to its related neighbor nodes;*
15. *Generate a random delay time τ ;*
16. $\Theta_v = \Phi$;
17. *While $\tau > 0$ do*
18. *If receiving a ReadySleeping message from the node u then*
19. *Marks the node u as sleeping node;*
20. $\Theta_v \leftarrow \Theta_v \cup C_u$;
21. *Endif*
22. *End while*
23. *End if*
24. *If $\Theta_v = \Phi$ then*
25. *The node v sets the status as a sleeping node;*
26. *Else*
27. *The node v sets the status as an active node immediately;*
28. *End if*

3.5.3 DSAMSN 算法性能评估

本小节利用仿真实验在 Marlab7 平台下以模拟方式执行的。为了评价采用求解 (3-5) 式的集中式算法与 DSAMSN 算法两种算法的性能，假设所有的网络节点随机均匀分布在 $400m \times 400m$ 目标监测区域内，50 个目标分布在此目标区域内，每个目标有 3 种物理参数需要监测，而每个节点上的传感器单元能感知最多有 3 种物理参数。整个仿真的能量模型与 3.4 节的约定一致，忽略传感器采集物理数据时的能耗，且不考虑节点本身的硬件执行算法所消耗的能量。假设 Sink 节点位于目标区域的原点(0,0)，每个处于活跃状态的工作节点每 0.10 秒钟向 Sink 节点报告数据。

表 3.4 为仿真实验的参数设置，由于覆盖目标点的节点不一定构成连通网络，为此结合 GRAB 通信协议进行数据传送。对于采用 Marlab7 求解(3-5)式的集中式算法，假设该算法在 Sink 节点上执行，那么，Sink 节点首先必须收集网络全局信息，然后将调度结果通知每个节点。而分布式算法 DSAMSN 则完全在本地节点执行。所有的数据都通过 GRAB 通信协议的方式传送至 Sink 节点在实验中。

网络生存时间的定义为：从网络开始工作起，到发现有一个目标的一个属性点不能覆盖或者没有节点能将数据传送至 Sink 节点为止，这个时间长度称为网络的

生存时间。

图 3.11 说明了采用 Marlab7 求解 ILP 模型下的网络生存时间与这种分布式算法 DSAMSN 下的网络寿命的比较。从图中可以看出，并且随着节点数的增加，分布式算法 DSAMSN 网络寿命也明显比采用 Marlab7 求解 ILP 模型下的网络生存时间有比较好的改善。

表 3.4 仿真实验参数

目标区域	$400m \times 400m$
目标点数	50
节点数	200~700
传感器单元感知半径向量	$(R_1, R_2, R_3) = (50m, 60m, 70m)$
通信半径	100m
初始能量 E_0	$2J$

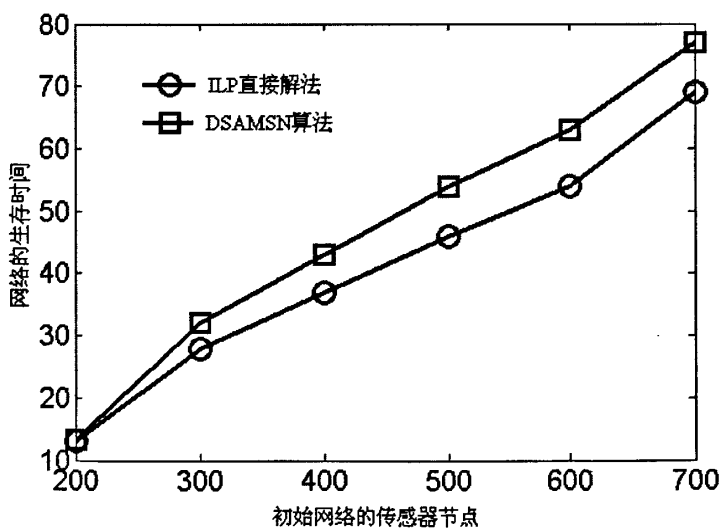


图 3.11 网络生存时间与节点数的关系

3.6 小结

本章首先分析了 PEAS 协议的不足，然后提出了基于节点冗余覆盖调度的 SRCA 算法，这种协议首先判别冗余覆盖方法检测网络中的覆盖冗余节点，并使用

节点剩余能量作为优先级防止互为邻居的冗余节点同时关闭导致网络中出现覆盖盲点。通过关闭不必要的冗余节点以及尽可能地实现负载均衡，SRCP 算法在保持网络原始覆盖的同时，能够有效地降低系统能量消耗。仿真实验表明，SRCP 算法在网络平均覆盖度、活跃工作节点数等方面都比 PEAS 协议有所改进。

接着，本章还提出了多功能无线传感器网络的节点模型，并且针对多功能无线传感器网络的点覆盖问题，归结了一定的约束条件下的点覆盖优化的 ILP 模型。然后通过设计 DSAMSN 算法求得问题的解，该算法按照能量和节点的覆盖能力做为优先级来决定节点是否是冗余，其目的为了尽量使节点的能耗均衡，延长网络的生存时间。

第四章 基于遗传算法的点覆盖算法

第三章讨论了无线传感器网络的节点分布式调度方案,面向的是保证覆盖要求而不希望有任何覆盖损失。本章研究利用整数数学规划(Integer Mathematic Program)IMP方法求解无线传感器网络点覆盖问题,并着重研究了一种无线传感器网络的具体应用—冗余度上界受限的点覆盖问题,这种应用可以不必完全保持初始网络的覆盖。但是,在IMP模型下,一种具体的应用往往只是约束条件的差异,因而,IMP模型就像一个点覆盖优化问题的处理框架,在这个框架下,不同的应用对应不同的约束条件,通过改变IMP的约束条件,即可得到不同应用要求的点覆盖问题的优化。

针对上述提出的IMP问题,本章首先将无线传感器网络的一种点覆盖归结为多目标优化问题,然后提出了一种基于遗传算法(Genetic Algorithm)的无线传感器网络节点调度技术。这样,可将整个无线传感器网络尽可能多地划分若干子集,这些子集通过轮换工作,实现无线传感器网络的覆盖节点的调度任务,减少活跃节点数量,延长网络的生存时间。

4.1 引言

无线传感器网络常常用于侦察敌方区域以及环境监测,为了提高覆盖质量的要求,往往在监视的区域撒播大量的节点而形成密集型节点的网络。通过节点调度的覆盖技术,可以有效减少实际工作节点数量;甚至可以在不影响覆盖要求的前提下降低网络的节点密度,延长网络的使用寿命。而将网络划分为多个轮换工作的子网,则是密集型无线传感器网络进行覆盖的常用方式^{[78][79]}。

对于森林环境监测之类的无线传感器网络应用,这种数据采集的时间要求不是很严格的无线传感器网络的应用场景,采集每个目标的数据间隔比较长,节点可以在不同的时间段采集目标的信息,只要合理划分每个时间线中活跃节点的时间间隔,而不必要求同时获取所有目标的信息。因此,在划分子网时,可以进一步挖掘这种密集型网络的潜力来延长网络的生存时间。

根据上述特点,本节提出了一个目标点的至多 k -覆盖问题(At Most k -Coverage Problem, AM k -CP),简称为AM k -CP问题。该问题要求每个目标至多被 k 个节点覆盖,再通过优化目标尽量减少出现覆盖盲点,而不要求一定达到网络初始覆盖质量。AM k -CP问题的提出的动机是:(1)轮换调度机制下的覆盖盲点不是静态的。在某个覆盖子集工作时期内,即使出现少数的覆盖盲点,但通过调度不同的覆盖子集轮流工作,使得在某个覆盖子集下的盲点在另一个覆盖子集下不再是盲点。

从总体网络服务来说,通过多个覆盖子集轮流工作,在时间交错的意义上说,这种覆盖机制也可以保证网络的覆盖质量。(2)现实世界里的目标不是一个抽象的质点,可以将同一个目标看成几个质点或目标点。这种方式相当设置冗余目标点,这样,即使某目标点是覆盖盲点,被监测的目标不一定是覆盖盲点。由此可见,在实际应用中,利用现实世界的特点,也可有助于无线传感器网络点覆盖问题的解决。下面给出 AM k -CP 问题的定义。

定义 4.1 AM k -CP 问题

给定无线传感器网络 $S=\{s_1 s_2 \cdots s_n\}$, 目标集 $T=\{t_1 t_2 \cdots t_m\}$, 为了使活跃节点数尽可能少,在覆盖盲点尽可能少的前提下,每个目标至多被 k 个传感器节点覆盖。这样一个无线传感器网络的点覆盖问题称为 AM k -CP 问题。

不同于以前点覆盖问题中的其它工作^[80], AM k -CP 问题考虑到有效减少覆盖子集的冗余度,只限制了覆盖每个目标对象的传感器节点数的上界,要求每个目标点至多被 k 个节点覆盖。前面已经提出, AM k -CP 问题在某些无线传感器网络的应用领域。这样的场景所监测的目标并不需要每时每刻去监测,也不需要统一的时间内收集它们的信息,而只要在一定的时间间隔内将采集的信息传送 Sink 节点即可。如图 4.1 所示为覆盖子集间轮流工作的时间线,图中所示有 n 个覆盖子集,每个子集按照一定顺序工作,当 n 个覆盖子集全部完成一轮的调度后,从下一轮开始按相同的顺序进行工作。

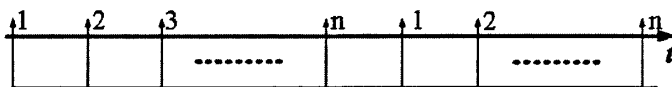


图 4.1 覆盖子集轮流工作的时间线

由此可见,不同的应用场景对无线传感器网络的点覆盖问题会提出不同的要求,针对应用场景的具体解决方案有利于优化点覆盖问题。因而,研究不同应用场景的点覆盖问题是很有必要的。

4.2 相关工作

Chakrabarty^[81]等最先采用整数线性规划(Integer Linear Programming, ILP)模型研究按栅格点布置传感器节点的感知问题,通过求解最优解或近似最优解而获得问题的解集。

无线传感器网络密度控制常用的手段是将网络分成多个不相交的节点子集,每个节点子集是完全覆盖目标区域。Cardei 和 Du^[82]提出了一种旨在尽可能多地将无

线传感器网络分割成分互不相交的分离集的算法。不同的分离集轮流工作，独立完成全网的监测任务。他们证明了分离集的覆盖问题是 NP 完全的，并将这个这种分离覆盖集的构造归结为最大流问题，通过构造了一种基于节点能量控制的调度技术，并将这样的点覆盖问题归结为一种混合整数规划数学模型来求得问题的解。

Cardei^[83]等通过进一步松弛独立覆盖集的约束，对上述算法进行改进，容许一个节点可同时属于不同的非相交覆盖集。Slijepcevic 与 Porkonjak^{[34][35]}则将区域覆盖问题划分为域 (Field) 的集合，指出一个域 F 是一个点集，如果两个点属于同一个域，当且仅当它们能被同一个节点集覆盖。无线传感器网络初始化后，每个节点保存其覆盖的所有域的列表。通过选择覆盖关键域 (被最少的节点覆盖的域) 的节点，然后通过选择覆盖域最多的节点加入覆盖集，从而达到减少冗余节点的目的。

本章从另一个角度提出无线传感器网络应用的问题—AM k -CP 问题，该问题只限制覆盖每个目标对象的节点个数的上界，因而分割的传感器子集也并不要求覆盖所有被监测的目标。该问题只是应用于具体的应用场景下，要求目标点有较低的覆盖冗余。本章在求解 AM k -CP 问题时有两个特点：(1) 同一个节点可以加入一个或多个活跃节点集；(2) 除了限制了目标覆盖的冗余度的上界，并没有对下界进行限制。显而易见，这种方法生成的覆盖子集可能产生覆盖盲点。当然，在处理过程中我们尽可能使盲点数最小化。

4.3 最小化传感器节点数学模型及其优化

基于上两节的分析，本节将 AM k -CP 问题形式化为整数规划问题；给定无线传感器网络 $S = \{s_1 \ s_2 \ \dots \ s_n\}$ ，目标集 $T = \{t_1 \ t_2 \ \dots \ t_m\}$ ，将 AM k -CP 问题模型化为一个 0-1 非线性规划问题。该模型包含 $n+m$ 个变量和 $2m$ 个约束，但其非线性部分只出现在约束条件中，并且其中的 m 个变量可以利用这些非线性约束求出。这样，很大程度上简化了问题的求解。

4.3.1 AM k -CP 问题的多目标 0-1 规划形式化

假设关系矩阵 $C = (c_{ij})_{n \times m}$ 表示节点与目标点的覆盖关系，其中 c_{ij} 是一个布尔变量 ($i=1 \dots n, j=1 \dots m$)。规定如果目标 t_j 被节点 s_i 覆盖，那么 $c_{ij}=1$ 否则 $c_{ij}=0$ 。同时定义布尔变量 x_k ($k=1 \dots n$) 为：对于某个覆盖子集 A ，如果节点 $s_k \in A$ ，那么 $x_k=1$ ，否则 $x_k=0$ 。

定义 4.2 目标点的重复覆盖度

目标 t_j 的目标点的重复覆盖度 TOC (Total Overlapping Coverage of Single Target) 定义为一个覆盖子集 A 中覆盖目标 t_j 的节点数，用代数式表示为：

$$\sum_{i=1}^n c_{ij} x_i \quad (j=1 \dots m) \quad (4-1)$$

对于一个覆盖子集, 比较理想的覆盖应该是每个目标点的 TOC 不要高 k (k 的取值视具体覆盖要求而定) 太多。对于高密集型的无线传感器网络, 每个目标点的 TOC 非常大, 必须采用有效的处理机制调节以便减少该值。

为了方便起见, 我们约定布尔变量 y_p ($p=1 \dots m$) 表示目标 t_p 是否为覆盖盲点, 如果 $y_p=1$ 那么目标点 t_p 是覆盖盲点, 如果 $y_p=0$, 那么目标点 t_p 能被相应的覆盖子集所覆盖。根据这样的约定, 在一个给定的覆盖子集中, 总的覆盖盲点数可形式化为:

$$\sum_{p=1}^m v_p y_p \quad (4-2)$$

在 (4-2) 式中, v_p 表示目标对象 p 的权重, 表示每个目标的重要程度的量化指标。显而易见, 如果一个给定的覆盖子集总的覆盖盲点数为 0, 那么该覆盖子集全覆盖给定的目标集中的所有目标。根据上述定义, 我们进一步形式化定义 AM k -CP 问题如下。

定义 4.3 AM k -CP 问题

给定无线传感器网络 $S = \{s_1 \ s_2 \ \dots \ s_n\}$, 目标集 $T = \{t_1 \ t_2 \ \dots \ t_m\}$, 寻找一族覆盖子集 $\Omega = \{\Omega_1 \ \Omega_2 \ \dots \ \Omega_r\}$, 在满足给定的覆盖要求下要求: (1) 对于 $\forall \Omega_i \in \Omega$, 使得 $|\Omega_i| = \text{Min} \left\{ \sum_{k=1}^n x_k \right\}$, 其中 $|\Omega_i|$ 表示集合 Ω_i 的元素个数; (2) 产生的覆盖盲点数 $\sum_{p=1}^m v_p y_p$ 最少; (3) $\forall t_j \in T$, $\sum_{i=1}^n c_{ij} x_i \leq k$, $j=1 \dots m$, 其中 k 是一个预先给定的正整数。

根据 AM k -CP 问题的形式化定义, 进一步上述将该问题归结为以下多目标优化问题。

$$\begin{aligned} \text{Objective: Minimize } f_1(x_1 \ x_2 \ \dots \ x_n) &= \sum_{i=1}^n w_i x_i \\ \text{Minimize } f_2(y_1 \ y_2 \ \dots \ y_m) &= \sum_{j=1}^m v_j y_j \\ \text{Subject to } y_j + \sum_{i=1}^n c_{ij} x_i &\geq 1, j=1 \dots m \\ y_j \sum_{i=1}^n c_{ij} x_i &= 0, j=1 \dots m \\ \sum_{i=1}^n c_{ij} x_i &\leq k, j=1 \dots m \end{aligned} \quad (4-3)$$

$$x_i, y_j \in \{0, 1\}, i = 1 \dots n, j = 1 \dots m$$

在(4-3)式中, w_i, v_k 分别表示第 i 个节点和第 k 个目标对象的权重。约束条件 $y_j + \sum_{i=1}^n c_{ij} x_i \geq 1, j = 1 \dots m$ 和 $y_j \sum_{i=1}^n c_{ij} x_i = 0, j = 1 \dots m$, 可以保证当 $y_j = 0$ 时, 则 $\sum_{i=1}^n c_{ij} x_i \neq 0$; 并且当 $\sum_{i=1}^n c_{ij} x_i = 0$ 时, 必有 $y_j \neq 0$ 。约束 $\sum_{i=1}^n x_i c_{ij} \leq k (j=1 \dots m)$ 保证在任意一个覆盖子集中, 每个目标对象至多被 k 个节点覆盖。为了获得更多的覆盖子集, 在(4-3)式中没有限制 TOC 的值的下界。

(4-3)式只给出了 AM k -CP 问题的 IMP 模型, 对于不同覆盖要求的无线传感器网络的应用场景, 可以通过改变(4-3)式中的约束条件而获得相应问题的 IMP 模型, 比如: 当将约束条件 $\sum_{i=1}^n x_i c_{ij} \leq k$ 该为 $k_1 \leq \sum_{i=1}^n x_i c_{ij} \leq k_2, (k_1 < k_2)$ 时, 则可得到每个目标至少被 k_1 个节点覆盖, 但知道只能被 k_2 个节点覆盖。因此, 无线传感器网络的点覆盖问题的 IMP 模型就像一个框架, 不同的应用场景可以通过改变约束条件得到问题的优化模型。

此外, (4-3)式包含了 $3m$ 个约束条件, 其中 $2m$ 是线性的, 而另外 m 个则是非线性的。但是, 根据约束 $y_j + \sum_{i=1}^n c_{ij} x_i \geq 1, j = 1 \dots m$ 与 $y_j \sum_{i=1}^n c_{ij} x_i = 0, j = 1 \dots m$, 的特点, (4-3)式中的 $(m+n)$ 个变量不是完全独立的。如果将每个解向量分解成向量 $(x_1 \ x_2 \ \dots \ x_n)$ 和向量 $(y_1 \ y_2 \ \dots \ y_m)$ 两个分量。根据约束条件可知, 如果获得了向量 $\mathbf{x} = (x_1 \ x_2 \ \dots \ x_n)$, 即可相应获得解向量 $\mathbf{y} = (y_1 \ y_2 \ \dots \ y_m)$ 。这样, 通过对约束 $y_j \sum_{i=1}^n c_{ij} x_i = 0$ 和 $y_j + \sum_{i=1}^n c_{ij} x_i \geq 1, j = 1 \dots m$ 这两个约束条件, 可以预先对(4-3)式进行处理, 这样, 就能有效减少(4-3)式中约束条件的个数, 简化求解(4-3)式的复杂性。

根据(4-3)式, 可定义解向量 $\mathbf{x} = (x_1 \ x_2 \ \dots \ x_n)$ 的可行函数为:

$$feasible(\mathbf{x}) = \max \left\{ \sum_{i=1}^n c_{ij} x_i - k, j = 1 \dots m \right\} \quad (4-4)$$

那么, 如果解向量 \mathbf{x} 为不可行解, 当且仅当 $feasible(\mathbf{x}) \leq 0$, 反之亦然。进一步, 定义不可行解 $\mathbf{x} = (x_1 \ x_2 \ \dots \ x_n)$ 对第 j 个目标点 t_j 的不可行度为:

$$violate(x, t_j) = \begin{cases} \sum_{i=1}^n c_{ij} x_i - k, & \text{if } \sum_{i=1}^n c_{ij} x_i - k > 0 \\ 0, & \text{otherwise} \end{cases} \quad (4-5)$$

根据(4-5)式, 不可行解 \mathbf{x} 对所有目标对象的累积不可行度定义为不可行度的总

和为：

$$\sum_{j=1}^m \text{violate}(x, t_j) \quad (4-6)$$

AM k -CP 问题没有限制每个目标对象被重迭覆盖的下限，那么该算法由此产生的覆盖子集在工作时可能会出现覆盖盲点（没有被某覆盖子集中的任何一个传感器节点覆盖的目标对象）。在这个问题中，一方面，我们希望每个覆盖子集的传感器节点数尽可能少；另一方面，又希望覆盖盲点出现的个数尽可能少，两者本是矛盾的。为了解决这个问题，只有同时考虑最小化覆盖集节点数与覆盖盲点二者的联合优化问题。如果一个无线传感器网络有 n 节点，目标对象为 m 个，这个问题的数学模型就是一包含 $(n+m)$ 个变量， $2m$ 个约束的整数非线性规划。

4.3.2 基于 Pareto 优化的多约束多目标优化问题

在上小节，我们分析了将问题转化成数学规划模型，并对该模型的约束条件进行了分析，事实上，多目标优化是一个 Pareto 优化问题^[84]。由于多目标优化的规划问题的解的个数不单一，而遗传算法能通过多个种群可产生它的 Pareto 最优集。按照判断多目标条件，这个问题的优化解集应该是 Pareto 最优化的^[84]。考虑到多个 Pareto 最优解，虽然很难从理论上说哪一个解是绝对最优的，但是，一旦找到多个 Pareto 最优解，就能根据现实问题的评价规则从中挑选更合适的优化解。

定义 4.4 Pareto 支配

给定解向量 x 和解向量 y ，它们的目标函数值向量分别为 $u=(f_1(x) \ f_2(x) \ \dots \ f_k(x))$ 和 $v=(f_1(y) \ f_2(y) \ \dots \ f_k(y))$ ，如果 $\forall i \in \{1 \dots k\}$ ，均有 $f_i(x) \leq f_i(y) \wedge \exists j \in \{1 \dots k\} : f_j(x) < f_j(y)$ 。则称解向量 x 的目标函数值向量 $u=(f_1(x) \ f_2(x) \ \dots \ f_k(x))$ 支配解向量 y 的目标函数值向量 $v=(f_1(y) \ f_2(y) \ \dots \ f_k(y))$ ，记为 $u \prec v$ 。相应地，称解向量 x 是 Pareto 支配解向量 y 的，记作 $x \prec y$ 。

“ \prec ”这是一种偏序关系，对于受约束的多目标优化问题，如果满足下列条件之一，就称解向量 x 是 Pareto 约束支配解向量 y 的：（1）解向量 x 是可行解，而解向量 y 是不可行解；（2）解向量 x 和解向量 y 都是不可行解，但解向量 x 的不可行度小于解向量 y 的不可行度；（3）解向量 x 和解向量 y 都是可行的，但解向量 x 是 Pareto 支配解向量 y 的。

定义 4.5 Pareto 最优集

如果解向量 $x \in \Omega$ 是可行域 Ω 上的 Pareto 优化解，当且仅当不存在任何解向量 y 是 Pareto 支配解 x ，所有 Pareto 优化解 x 所构成的集合称为 Pareto 最优集。

定义 4.6 Pareto 最优集的 Pareto 前沿

对于一个多目标优化问题 $F(x)$ 和一个 Pareto 最优集 P^* , Pareto 前沿定义为: 对于向量 $x \in P^*$, $PF^* := \{u = F(x) = (f_1(x), \dots, f_k(x))\}$, PF^* 叫做 Pareto 前沿(Pareto front)。

对于多目标优化问题的搜索空间中的多个 Pareto 优化解, 我们不能认为这些解中的任何一个都是最优解, 因而只能根据应用需求从获得的 Pareto 优化解中找到更适合的解。图 4.2 是一个两目标优化问题的说明, 图中考虑了保护目标函数 1 和目标函数 2 的情形。从图 4.2 可以看出, 点 A 表示更靠近最小化目标函数 2 的一个解, 但离最小化目标函数 1 较远, 同样, 点 E 表示更靠近最小化目标函数 1 的一个解。在实际应用中, 如果二者都比较重要, 那么, 我们不能认为解点 A 比解 E 好, 反之亦然。事实上, 如果没有更进一步的实际应用方面的需求的话, 解 B, C, D 都是 Pareto 优化解。

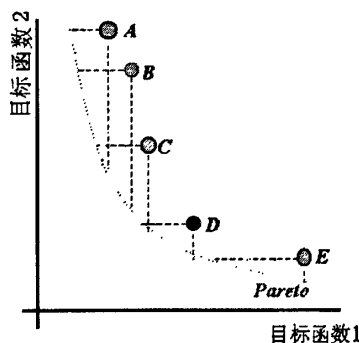


图 4.2 Pareto 解示意图

为了求解非线性规划问题 (4-3), 首先利用遗传算法找出 Pareto 优化解; 然后利用无线传感器网络的实际应用的需求再从中挑选出优化解。由于遗传算法非常适合处理包含多个体的种群优化问题, 因此, 首先设计了一种基于这个模型求解无线传感器网络覆盖子集的遗传算法。在该算法中, 只将解向量 $\mathbf{x} = (x_1, x_2, \dots, x_n)$ 定义为种群中的个体, 根据约束条件 $y_j + \sum_{i=1}^n c_{ij} x_i \geq 1, j=1..m$ 和 $y_j \sum_{i=1}^n c_{ij} x_i = 0, j=1..m$ 求出解向量 $\mathbf{y} = (y_1, y_2, \dots, y_m)$, 这样, 在求解上述多目标优化问题时可进一步减少约束条件的个数, 降低解向量求解的难度。

4.4 基于遗传算法的多约束多目标优化

基于遗传算法的多约束多目标优化的主要思想是同步搜索多个 Pareto 优化

解，以便从 Pareto 最优集中找到最适合当前应用的解集。本节利用基于 K. Deb 等^[85]提出的方法处理非线性规划问题 (4-3)，然后找出接近现实问题的解。使用这种方法，遗传算法的种群组成了包括非可行个体在内的 Pareto 优化前沿 (Pareto optimal fronts)

4.4.1 个体表示

既然非线性规划问题 (4-3) 中的解向量 (\mathbf{x}, \mathbf{y}) 的分量 \mathbf{y} 可以由分量 \mathbf{x} 决定，那么我们仅仅将分量 \mathbf{x} 作为种群中的个体，因而只需对分量 \mathbf{x} 编码即可。设计遗传算法的首要问题是染色体合适的表示方式，对于 0-1 整数规划优化问题采用 0-1 二进制串显而易见可表示一个染色体。对于由 n 个节点组成的无线传感器网络，我们可以用一个长为 n 位的二进制串标志一个覆盖集，如果第 i 位的值为 1，意味传感器节点 s_i 属于这个覆盖子集，图 4.3 说明了一个染色体的二进制串表示。

1 0 0 0 1 0 0 1 1 1 1 0 ... 0 0 0 1 1 1

图 4.3 二进制串染色体个体图示

在这里必须说明一点，虽然公式(4)的解向量是 (\mathbf{x}, \mathbf{y}) ，这里 $\mathbf{x}=(x_1, x_2, \dots, x_n) \wedge x_i \in \{0, 1\}(i=1 \dots n)$, $\mathbf{y}=(y_1, \dots, y_m) \wedge y_j \in \{0, 1\}(j=1 \dots m)$ 。由于解向量 \mathbf{y} 由解向量 \mathbf{x} 决定，所以，我们只是定义二进制向量 \mathbf{x} 种群的个体。

但是，这种表示方法也会存在一个问题，在应用遗传算法子进行操作时，产生的可能不是可行解。文献^[85]采用惩罚函数方式处理这个问题，但是确定一个性能比较好的惩罚函数却并不容易；我们采用了一种启发式操作算子将不可行解直接转化为可行解的方式解决上述问题。

4.4.2 种群个体分级处理

在应用遗传算法子进行操作时，产生的可能不是可行解，那么，如何选择适当的种群作为下一次迭代就很重要了。在设计算法时，我们希望不可行个体在迭代过程中朝可行性方向进化，而在选择下一次迭代的个体时，也希望选中可行个体的概率大于选中不可行个体的概率。

为了解决上述问题，我们将按照 Pareto 支配的情况对所有可行个体进行分类，而对不可行的个体 \mathbf{x} 按照可行函数 $feasible(\mathbf{x})$ 的值从小到大进行排序。首先，将所有个体分成可行的个体 X 和不可行 Y 的个体两部分，然后，找出所有可行解的 Pareto 最优集 X_1 ，我们称 X_1 为第 1 级 Pareto 最优集，然后，找出 $(X - X_1)$ 的 Pareto

最优集 X_2 ，我们称 X_2 为第 2 级 Pareto 最优集，依次类推，直到将所有可行个体分类完毕，得到第 m 级 Pareto 最优集 X_m 。对于不可行的个体，我们将它们放在可行个体的级别之后，根据 (4-6) 式计算所有不可行个体每个不可行个体对所有目标对象的累积不可行度，并按照累积不可行度进行分级。

在对所有个体分成不同级别的 Pareto 最优集之后，然后按照从小到大的顺序分成第 $(m+1)$ 级 Pareto 优化前沿、第 $(m+2)$ 级 Pareto 优化前沿直到第 $(m+r)$ 级 Pareto 优化前沿。基于^[85]的思想，根据非支配的级别进行 Pareto 最优集分类，由于有 2 个优化目标，因此，分类出各级 Pareto 最优集的计算复杂性为 $O(2N^2)$ 。

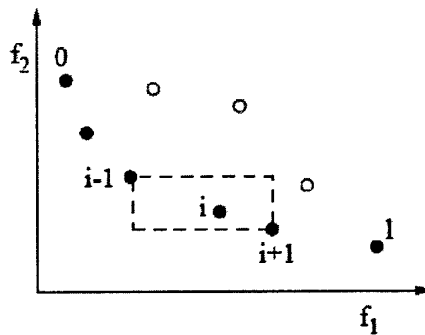


图 4.4 计算拥挤距离图示

如图 4.4 所示，为了估计种群的个体间的围绕种群中一个特定点（个体）的密度，我们引用文献^[80]关于拥挤距离的定义。所谓拥挤距离是该点（个体）沿着每个目标函数的边的两个点（个体）间的平均距离，数量指标 i -distance 看做是围绕这个点（个体） i 的最大长方形大小的估计，该长方形内除了点 i 外不包含种群中的其它任何点（个体），称这个长方形的大小为拥挤距离。对于二维的情况，拥挤距离是长方形两条边的平均长度。在集合 S 中，每个点的拥挤距离计算方法^[85]如下所示。

表 4.1 拥挤距离算法

1. $l=|S|$;
2. **for** each I , set $I[i]$ distance=0
3. **for** each objective function f
4. $S=\text{sort}(S, f)$;
5. $I[1]$ distance= $I[1]$;
6. distance= ∞ ;

```

7.      For  $i=2$  to  $(l-1)$ 
8.       $\Pi[i].distance = \Pi[i]$ ;
9.       $distance += (\Pi[i+1].f - \Pi[i-1].f)$ ;
10.     End for
11.     End for
12.     End for

```

上述式子中, $\Pi[i].f$ 表示集合 S 中的第 i 个个体目标函数值 f , 从表 4.1 所示的拥挤距离计算方法可以看出, 该算法的计算复杂性实际上由分类算法 $sort(S, f)$ 来决定。在最坏的情况下, 所有的个体在同一个 Pareto 优化前沿, 此时, 分类函数的复杂性为 $O(2N \log N)$, 其中 N 表示集合 S 中个体数。

4.4.3 个体的选择、交叉和变异

上面我们分析了多目标优化问题的解的特性、Pareto 最优集以及同一级别的 Pareto 最优集内解与解之间拥挤距离的概念, 对于种群中的每个个体 i 一定具有下面两个属性:

1. 个体 i 属于第 k 级 Pareto 优化前沿, 记为: $i_{rank}=k$
2. 个体 i 在其所处的第 k 级 Pareto 优化前沿中的拥挤距离 $i_{distance}$

这小节描述种群的个体选择过程, 为此引入个体的拥挤比较算子 \succeq_n 。

定义 4.6 个体的拥挤比较算子 \succeq_n

如果 $i \succeq_n j$, 当且仅当 $(i_{rank} < j_{rank})$ or $((i_{rank} = j_{rank})$ and $(i_{distance} > j_{distance}))$ 。

偏序 \succeq_n 标志算法收敛至 Pareto 优化前沿的不同策略, 如果两个解可能属于不同级 Pareto 优化前沿, 那么选择低级别的解, 如果两个解可能属于同级 Pareto 优化前沿, 那么依靠拥挤距离小者。

遗传算法的关键是个体的选择、交叉和遗传变异, 在算法中, 我们采用 Tournament 选择方式选择进行交叉的种群, 采用 SBC (Simulated Binary Crossover)^[86] 做为交叉算子。SBC 用于模拟实变量单点二进制串交叉操作, 而我们采用的是二进制变量而非实变量。在我们提出的算法中, 也采用 SBC 交叉算子和多项式变异算子处理二进制串的个体进化操作问题, 只是修改了遗传交叉算子和变异算子。具体的种群个体的选择、交叉和变异步骤如下:

1. 利用 Tournament 选择进行交叉的种群

Tournament 选择是通过随机选择 n 个个体, n 等于遍历 (Tour) 大小, 从这些个体中仅选择一个个体放入交叉池 (mating pool), 然后依据该个体的偏序关系 \succeq_n

选择个体。也就是说评价某个体是否能被选择到，首要条件是根据个体所在的 Pareto 优化前沿级别，Pareto 优化前沿级别最低者将优先被选中。其次，如果有两个或两个以上的个体处在相同的 Pareto 优化前沿级别，那么进一步根据拥挤距离来选择，拥挤距离较大者将优先被选中。

2. 重组与选择

子代种群采用当前代种群进行组合，而进行选择为了设置下一代个体，种群根据 Pareto 优化前沿进行分类，新一代种群的个体来自每个前沿直到种群中的个体数超过当前种群的个体数，如果加入前沿 F_j 中的种群中的个体数超出了 N ，那么，前沿 F_j 中的个体将根据它们的拥挤距离按递减的顺序进行选择，直到种群大小是 N 为止。迭代执行这个过程产生后代。

在该遗传算法中，新的个体的重组过程则按照以下步骤进行：

- (a) 产生一个均匀分布在 $(0, 1)$ 间的随机数 μ ，然后，根据下面的式子求出样本 β 的值。

$$\int_0^{\beta} p(x) dx = \mu \quad (4-7)$$

在(4-7)式中

$$p(x) = \begin{cases} 0.5(\eta+1)x^{\eta} & \text{if } 0 \leq x \leq 1 \\ 0.5(\eta+1)x^{-(\eta+2)} & \text{otherwise} \end{cases} \quad (4-8)$$

在(4-8)式中， η 是一个给定的交叉分布指数，从上述概率分布函数求出 β 。

- (b) 对于进行交叉遗传的两个种群个体 $p^{(1)} = (p_i^{(1)})_{b \times n}$ 、 $p^{(2)} = (p_i^{(2)})_{b \times n}$ ，假设交叉后所产生的两个后代个体中分别为 $c^{(1)} = (c_i^{(1)})_{b \times n}$ 、 $c^{(2)} = (c_i^{(2)})_{b \times n}$ ，那么 $c^{(1)}$ 、 $c^{(2)}$ 根据如下方式求得

$$c_i^{(1)} = \begin{cases} 1, & \text{if } t_i^{(1)} \geq 1 \\ 0, & \text{otherwise} \end{cases}, \quad c_i^{(2)} = \begin{cases} 1, & \text{if } t_i^{(2)} \geq 1 \\ 0, & \text{otherwise} \end{cases} \quad (4-9)$$

在(4-9)式中

$$t_i^{(1)} = 0.5[(1+\beta)p_i^{(1)} + (1-\beta)p_i^{(2)}], \quad t_i^{(2)} = 0.5[(1-\beta)p_i^{(1)} + (1+\beta)p_i^{(2)}]$$

- (c) 根据下列等式确定样本 δ

$$\int_0^{\infty} p(x)dx = \mu \quad (4-10)$$

在(4-10)中, $p(x) = 0.5(\varphi+1)(1-x^\varphi)$, φ 是一个预先给定的变异分布指数。

(d) 根据式(4-10)求出 δ 后, 再按照下列的等式对后代个体 $c_i = (c_{ij})_{1 \times n}$ 进行重新组合得到 c_i 变异后的个体 d_i , d_i 的求法如下:

$$d_{ij} = \begin{cases} 1, & \text{if } y_{ij} \geq 1 \\ 0, & \text{otherwise} \end{cases}, \text{ 其中 } y_{ij} = c_{ij} + \delta$$

上述过程是遗传算法的主体。通过多次迭代, 最后得到满足给定条件的个体。在这样有必要指出的是, 步骤(a)、(b)、(c)、(d)可以是预先估计所列的方向进行设定, 以减少传感器节点的计算能量开销。事实上, 对于给定的一种遗传算法, 参数 δ 是预先确定的, 不必在每次执行遗传算法时计算这个值, 因而, 参数 δ 的求取不会影响算法的计算复杂度。下面将上述算法如表 4.2 的伪码描述, 为了便于后面引用此算法, 将它记做 Algorithm I。

表 4.2 Algorithm I

Input: maximal iterative times *max_gen*

1. *Generate an initial population P of size M; //every individual is chosen uniformly at random from the decision space $X = \{0, 1\}^n$.*
2. *For gen=1 to max_gen*
3. *Sort the population P according to the Parto frontier;*
4. *Select the individuals in the population P according to Tournament selection technique;*
5. *Make use of (b)(d) for crossover and polynomial mutation to obtain offsprings;*
6. *gen ← gen+1;*
7. *End for*
8. *Return Pareto Optimal Set.*

4.5 基于遗传算法的传感器节点调度算法

本节我们进一步描述如何使用遗传算法实现无线传感器网络点覆盖问题的节点的调度,这种基于遗传算法的传感器节点调度技术(Node Schedul Algorithm based on GA) 简称为 NSAGA 算法。NSAGA 算法做为一种传感器节点调度算法, NSAGA 算法运行的结果得到多个近似解, 每个近似解则对应一个覆盖子集。

4.5.1 NSAGA 算法应用前提

NSAGA 算法执行的根本点是必须知道每个节点当前覆盖目标的准确信息, 因此, 每个传感器节点必须具备两个基本条件。第一, 每个传感器节点知道自己的空间位置, 相关工作在^{[87][88][89]}等文献中已经提出了; 第二, 每个传感器节点能获取或存贮了离散目标的位置信息, 现在的存贮技术完全能很好地支持。

NSAGA 算法是一种集中式算法, 每个节点将自己掌握的信息发给中心节点或 Sink 节点, 因而, NSAGA 算法必须在无线传感器网络完成初始化并且建立了完整的拓扑的基础上。

NSAGA 算法必须约定各目标点的位置是静态的, 因为这种情况允许集中式算法在整个网络工作期只须运行一次, 所以 NSAGA 算法并只适合监视静态目标的应用场合。由于每个节点将自己掌握的信息发给中心处理节点或 Sink 节点, 必然要花费相当的能量代价, 因而, NSAGA 算法不适合大规模无线传感器网络, 而只能适应规模较小的密集型无线传感器网络。

4.5.2 NSAGA 算法及其复杂性分析

在无线传感器网络中, 当节点在确定自己的位置后, 即可建立一个覆盖目标对象信息表 list。如果 list 非空, 则将此表发送至中心节点或 Sink 节点。中心节点或 Sink 节点收集完毕来自各节点发来的 list 表后, 建立这个目标对象的覆盖信息表 all-list。

根据覆盖信息表 all-list 调用 Algorithm I, 从而获得相应的 Pareto 最优集, Pareto 最优集的每个元素对应一个覆盖子集, 例如, 给定 10 个节点的无线传感器网络, Pareto 最优集中的元素 $X=(1,1,0,0,0,1,0,1,0,0)$ 表示可传感器节点 s_1, s_2, s_6, s_8 在同一个覆盖子集。根据 Algorithm 1 的特点, Pareto 最优集中会包含多个 Pareto 最优解的情况, 可以根据应用要求对这些解进一步筛选, 从中挑选合适的元素。最后, 算法必须运行至所有的节点都被指派给合适的覆盖子集为止。

当覆盖子集划分完毕后, 中心节点将每个节点所属覆盖子集的序号以及调度策

略的信息发送给每个传感器节点。从 NSAGA 算法的执行过程可以看出，由 Algorithm I 产生的 Pareto 最优解中，可能会出现某些节点出现在不同的覆盖子集中频率较高，但可根据具体的应用要求规定同一个节点出现在不同的覆盖子集中的频率。

在每次在获得新的覆盖子集后，就从传感器集合中去掉已指派到覆盖子集中的节点。然后，改变目标函数的参数个数，具体方式为：如果算法在第 k 次划分进入新的覆盖子集的节点数为 n_k ，而在第 $k-1$ 次划分后剩余的节点数为 n_p ，那么修改式 (4-3) 中的目标函数的参数 $n = n_p - n_k$ 。同时，种群的每个个体的维数也改变为 $n = n_p - n_k$ 。上述处理方式是为了避免让同一传感器节点划分到太多的覆盖子集中。

假设传感器集合 S ， $|S|=n$ ，表 4.3 给出了 NSAGA 算法的伪码描述。

表 4.3 NSAGA 算法描述

<p>Input: all-list</p> <ol style="list-style-type: none"> 1. <i>Initialize cover set index, i.e., cover_index=1;</i> 2. While ($S \neq \Phi$) do 3. <i>Call Algorithm I to obtain Pareto Optimal set $P = \{x_1 \dots x_p\}$;</i> 4. <i>Choose a better active node subset $P_s = \{x_{s1} \dots x_{sr}\}$ from Pareto Optimal set P based on the current application;</i> 5. For $i=1$ to r 6. <i>each x_{si} determines an active node subset $S(\text{cover_index})$;</i> 7. $S \leftarrow S - S(\text{cover_index})$; 8. $\text{cover_index} \leftarrow \text{cover_index} + 1$; 9. $i \leftarrow i + 1$; 10. End for 11. <i>Modify the dimension of the individuals in the population P;</i> 12. <i>Modify the parameter n in the object function (4-3);</i> 13. End while 14. Return $\{S_1 \dots S_{\text{cover_index}}\}$.

定理 4.1 NSAGA 算法的计算复杂度 $\leq O(2krnN^2)$ 。

证明 对于 n 个传感器节点的无线传感器网络， N 是种群的规模，那么将种群的所有个体进行进行 Pareto 前沿分级的复杂度为 $O(2N^2)$ [80]。对于同级的个体，计算拥挤距离的的复杂性为 $O(2N \log N)$ 。而对于两个个体进行算子 \geq_n 比较的复

复杂度是 $O(2N \log(2N))$ ，那么完成这些过程的计算复杂度是 $O(2N^2) + O(2M \log N) + O(2N \log(2N)) = O(2N^2)$ 。如果 Algorithm I 的迭代次数为 k ，那么 Algorithm I 的计算复杂度为 $O(2kN^2)$ 。对于 NSAGA 算法，最坏的情况是每个覆盖子集只包含一个传感器节点，这样，需要划分 n 次才能完成，而每次完成迭代获得的 Pareto 最优解的元素个数为 $r, r < N$ ，因而 NSAGA 算法的计算复杂度的上限是 $O(2krnN^2)$ 。

4.6 性能评价

为了评估算法的性能，我们在 Matlab7 下对 Algorithm I 算法进行了测试，在这一节从两个方面对我们提出的算法进行评价。由于收敛性能好的遗传算法能快速求得问题的近似解，首先评价 Algorithm I 算法的收敛性能；其次，针对无线传感器网络对点覆盖问题，我们也考察了使用 Algorithm I 算法划分覆盖子集的并行度，这里的并行度是指 Algorithm I 完成每次迭代后所产生的有效覆盖子集的个数。同时，我们通过仿真实验考察了在不同的迭代次数下，NSAGA 算法对划分的覆盖子集数量的影响；最后，为了说明与相关算法的性能比较，我们比较了 NSAGA 算法与 MC-MIP 算法在相同初始条件下所产生的覆盖子集个数的比较结果。

4.6.1 Algorithm I 算法的收敛性能分析

通过实验的平均结果验证修改后的算法的有效性。在所有仿真实验中，交叉分布指数 η 和变异分布指数 φ 设置为 20。假设初始种群的大小为 50，最大迭代次数为 200，交叉概率为 0.9，通过运行 50 次后得到的分布在迭代进行至 5 次、20 次、30 次以及 50 次时的平价结果。

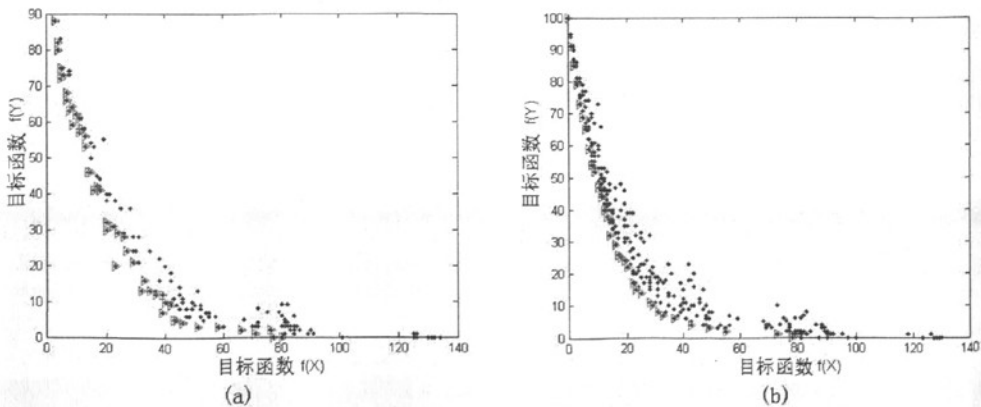


图 4.5 随着迭代进行个体向 Pareto 前沿进化图示

图 4.5(a)(b)说明 Algorithm I 算法随着迭代的进行,个体向 Pareto 前沿进化的情况,可以看出, Pareto 前沿随着迭代的进行趋于明显。

图 4.6 表明了 Algorithm I 算法对于不同的迭代次数是有效的,随着迭代次数的增加, Pareto 前沿朝着 Pareto 优化前沿收敛。但是,从图中可以看出,迭代 30 次后, Pareto 前沿改善程度不是很明显了。实验结果也表明 Algorithm I 算法在搜索可行区域的解的鲁棒性能比较好,因而, Algorithm I 算法在迭代过程中能较好地使个体朝着可行的 Pareto 优化前沿进化。

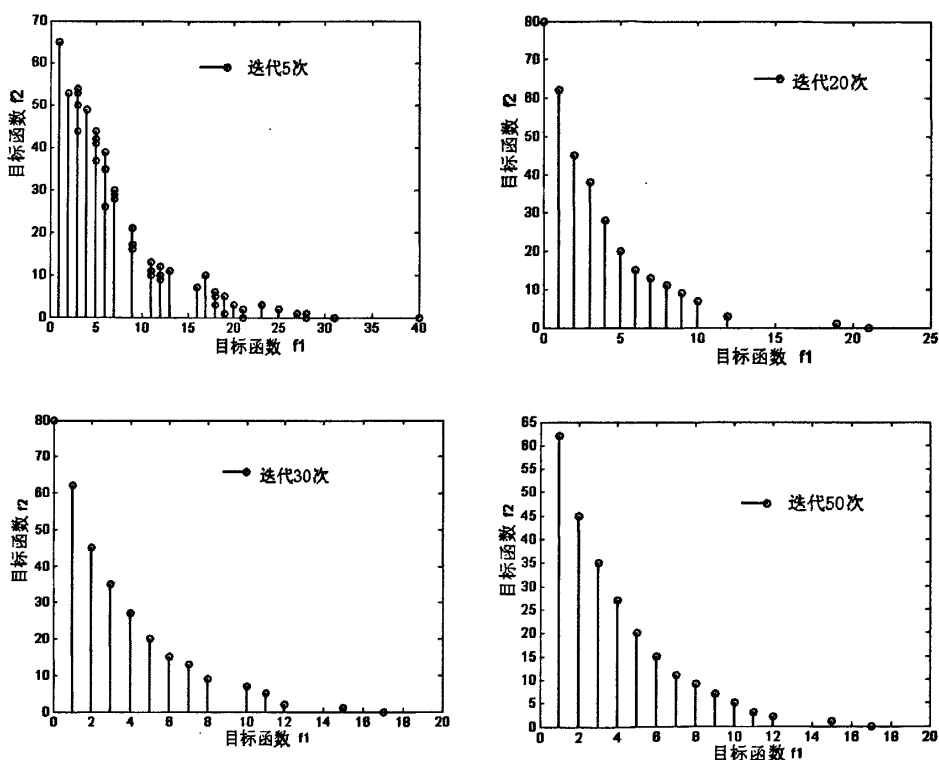


图 4.6 不同迭代次数下 Algorithm I 算法的收敛结果图示

在上一节我们已经提到 Algorithm I 算法是一种伪并行算法,可以提高寻找到适合问题的覆盖子集速度,为此,我们假设分别 80 个目标均匀分布在 $400m \times 400m$ 的区域内,对传感器节点数分别为 200、300、400、500 以及 600 的前提下,限定每个目标至多被 4 个传感器节点覆盖,每个传感器节点的感知域设定 $80m$,其它参数与前面的仿真设置相同。

表 4.1 是将 Algorithm I 运行 50 次的平均结果,当传感器节点数为 200 时,平

均迭代 3~4 次, 在 Algorithm I 的 50 次运行中, 一次运行能产生最大的覆盖子集数为 5, 最小为 2, 50 次运行产生的平均覆盖子集数 3.27。随着节点数增加, 相应的并行度也增加, 对于密集型无线传感器网络, Algorithm I 算法显示了其优势。从仿真可以看出 Algorithm I 算法能快速划分满足问题的覆盖子集。

表 4.4 执行算法 1 的并行结果

传感器节点数	迭代次数	最大并行度	最小并行度	平均并行度
200	3.75	5	2	3.27
300	5.13	7	2	4.61
400	6.2	7	2	5.3
500	6.71	8	2	5.72
600	9.31	11	2	5.47

4.6.1 NSAGA 算法的收敛性能分析

我们研究了不同的情况下算法的有效性, 为了验证修改后的算法的有效性, 我们设置了不同的情况对算法进行了研究, 然后得到这些实验的平均结果。

在第一个实验里, 我们考虑 15 个目标对象随机分布在 $500\text{m} \times 500\text{m}$ 区域内, 同时随机布置了 150、200 以及 250 个传感器节点, 每个节点的感知域为 250m , 我们限制了每个目标对象最多被 2 个传感器节点覆盖, 并且每个目标对象的覆盖盲点设置为 0。设置交叉指数与变异分布指数对 (η, φ) 为 $(20, 20)$, 然后通过设置不同的迭代次数, 考察 NSAGA 算法对划分的覆盖子集数量的影响。

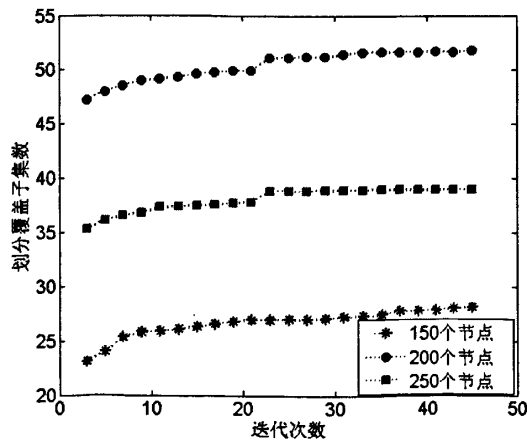
图 4.7 NSAGA 算法在 $(\eta, \varphi) = (20, 20)$ 迭代次数与覆盖子集数关系

图 4.7 说明 NSAGA 算法的迭代次数与划分的覆盖子集数的关系。可以看出，当迭代次数达到一定值时，所生成的覆盖子集并没有太多变化，这说明 NSAGA 算法可以预先设定比较合理的迭代次数，因此，该算法具有良好的鲁棒性。

下面的实验设置交叉指数与变异分布指数的分布指数 $\eta = 20$ 和 $\varphi = 20$ ，在所有实验中，交叉分布指数 $\eta = 20$ 和变异分布指数 $\varphi = 20$ 。所有的参数与 4.4.4 相同，我们考虑了在 $500\text{m} \times 500\text{m}$ 区域内，有 10~50 个目标对象随机的分布在这个区域，同时随机布置了 90 个传感器节点，每个节点的感知域为 250m，我们限制了每个目标对象最多被 2 个传感器节点覆盖，并且每个目标对象的覆盖盲点设置为 0。

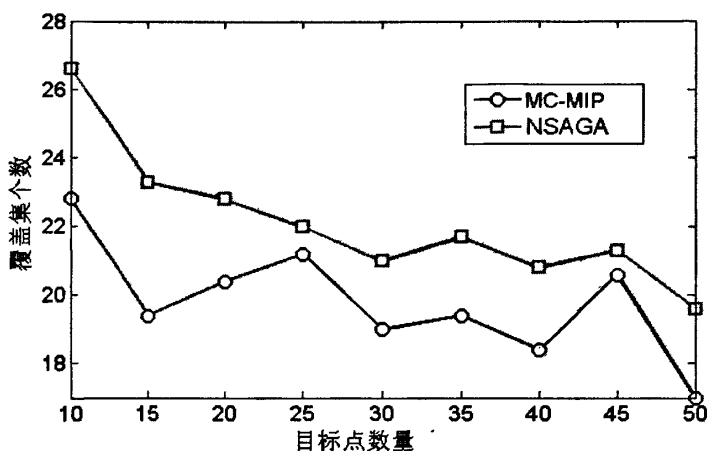


图 4.8 不同目标数下 NSAGA 算法与 MC-MIP 算法的覆盖子集数比较

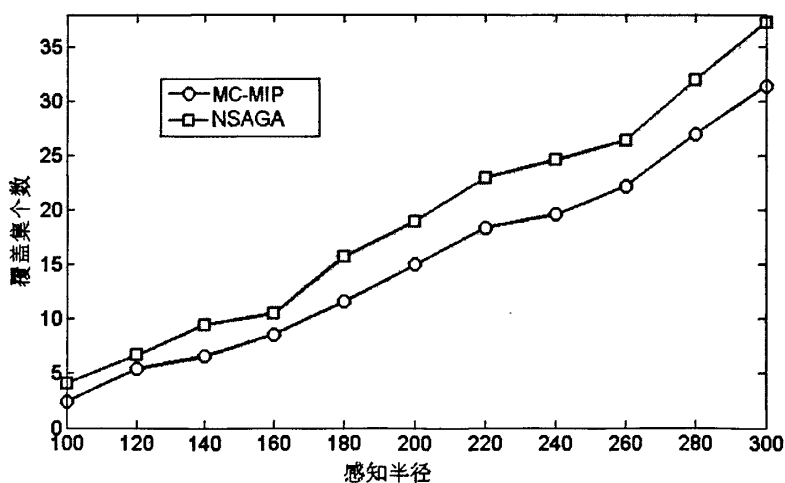


图 4.9 两个算法覆盖子集数对感知半径的比较

图 4.8 是比较 NSAGA 算法与文献^[76]中所提出的 MC-MIP 算法所产生的覆盖子集个数的图示。

图 4.9 显示了 NSAGA 算法与文献^[76]中所提出的 MC-MIP 算法所产生的覆盖子集个数的比较；我们设置每个传感器节点的感知域在 100m~300m 之间以增量为 20m 的变化规律改变。不同的样本值的结果显示 NSAGA 算法比文献^[76]中所提出的 MC-MIP 算法能产生更多的覆盖子集。

4.7 小结

无线传感器网络依靠部署冗余节点可以改善网络的服务质量，然后，利用节点调度技术来提高目标的可覆盖性和延长网络使用寿命。如果调度不合理，节点由于大量转发不必要的报文而消耗过多的能量，甚至可能引起网络冲突。因而只要，只要不影响覆盖要求，在网络中保持一部分活跃节点工作，有利于密集型网络延长它的生存时间。本章针对上述特点提出了密集型无线传感器网络的 AM k -CP 问题，并将此问题形式化为一个多目标的非线性规划问题。通过对问题的分析，提出了一种基于遗传算法的解决方法。在此基础上，本章分别设计了划分覆盖子集的一种集中式算法—NSAGA 算法。最后，我们对所设计的算法进行了仿真，结果表明 NSAGA 算法的有效性。

第五章 点覆盖问题的局部化算法

第三、四章研究了无线传感器网络点覆盖问题, 针对密集型无线传感器网络提出了分布式节点调度算法与集中式算法。本章研究点覆盖问题的局部化技术。所谓点覆盖问题的局部化, 是指将大规模无线传感器网络按照一定的算法将其分割为多个规模较小的网络, 然后在每个划分后的网络上运行点覆盖算法, 实现无线传感器网络的节点调度, 延长网络的生存时间。研究点覆盖问题的局部化技术的动机是:

(1) 目前, 已经有一些关于点覆盖问题的研究成果。但是, 大部分研究成果往往局限于集中式调度算法, 譬如采用的是数学规划的优化算法。求解这些算法必须由中心处理节点才能完成算法的执行。由于需要知道网络的全局覆盖信息, 集中式算法并不适合大规模的无线传感器网络; (2) 在实际应用场景中, 不同监测区域的目标其重要程度往往也会参差不齐。有一些区域的目标不要求精度很高的物理数据, 而另一些则相反; 有一些区域的目标不需要时刻监测, 而另一些区域的目标则监测周期很短, 需要时刻收集实时信息。

针对无线传感器网络的上述特点, 我们提出了一种基于集中式算法的局部化覆盖算法 LCACA (Localized Coverage Algorithm based on Central Algorithm)。该算法首先选举若干个局部中心节点, 通过节点自由加入的原则是每个节点找到所属的局部中心节点, 从而将大规模无线传感器网络划分为多个规模较小的网络。然后通过每个规模较小的网络上运行集中式算法而实现点覆盖, 这种局部化的集中式算法不再需要网络的全局信息, 避免了大规模无线传感器网络执行集中式算法的不足之处。同时, 对于实际应用场景, 还可以根据每个规模较小的网络所在的区域, 针对不同区域的目标的重要程度, 实现不同区域的目标之间有差别的点覆盖, 此时, 不同区域的每个子网既可以执行不同的集中式算法亦可执行相应的分布式算法。

由此可见, LCACA 算法不仅能达到集中式算法局部化的目的, 而且能根据实际应用, 实现同一个无线传感器网络不同区域内的节点可运行多种覆盖算法。

5.1 引言

无线传感器网络的生存时间是无线传感器网络研究中的一个基本问题。网络的生存时间通常可以定义为第一个节点或最后一个节点能量耗尽所经历的时间。传感器节点消耗的能量为有用的和浪费的两个方面能量消耗。有用的能耗是指当传输和接收数据, 处理感知任务以及将采集的数据转发至其邻近节点而必须消耗的能量。当传感器节点处于 idle 状态去监听媒介层状态、处理由于数据包的冲突而重发以及产生或处理控制报文时, 这些能量可归结为浪费的能耗。

为了尽量减少浪费的能耗,研究者目前提出了几种 MAC 层算法^[90-93],这些算法是为了减少浪费的能耗。而另一方面,减少传感器节点完成给定的任务所需要的有用能耗也是必要的。这些算法大体上可以分为 3 类:第一类算法是通过控制每个节点的传输能耗,以增加网络容量且保证网络是连通的^{[94][95]},第二类是基于能量优化的路由算法^[97-100],第 3 类算法是通过网络拓扑控制,调度每个节点在网络中是保持工作状态还是睡眠状态^{[101][102]}。对于大规模密集型的无线传感器网络,在保证覆盖质量的前提下,合理地调度节点的活跃或休眠则是这一类算法中的一个研究热点。

在点覆盖方面,虽然已经有研究者提出了很多优化覆盖的算法,但是,这些算法绝大多数采用将覆盖问题模型化为各种数学规划的优化解,然后采用集中式算法获得问题的近似解^[103-106]。集中式算法必须依赖中心节点才能组织全局网络中的传感器节点的调度,当从各传感器节点收集信息并将调度信息发送给各传感器节点时,必须传输或转发大量数据报文;同时,众多的节点发射无线信号,也容易产生信号间的串扰,使得数据出错概率增加。总之,集中式算法不利于节省有用能耗也会提高浪费的能耗。

我们在上面已经分析了传感器节点能耗的特点,集中式算法不适合于大规模无线传感器网络的一个主要原因是网络直径大,如果减小网络直径,则可以避免这种缺陷。我们在下面将提出的 LCACA 算法,就是从大规模无线传感器网络上上述特点出发,将网络划分成若干较小规模的子网的一种算法。这种算法的基本思想在每个规模较小的子网内运行集中式算法去完成传感器节点的调度,将发送或转发数据报文限制在这个较小规模的子网内,从一定程度上降低网络的能耗。原有的集中式算法中只由一个中心处理节点承担传感器节点的调度任务。LCACA 算法则通过多个子网的局部中心处理节点分布式地承担传感器节点的调度任务,这些局部中心处理节点分布在网络的不同位置,从逻辑上将全网划分成多个规模较小的网络。为了描述方便,我们将这种局部中心处理节点记做 L-node 节点。如图 5.1 所示,传感器节点 1、2、3、4、5 和 6 作为 L-node 节点,图中的虚圆圈为每个 L-node 节点的通信范围,这 6 个 L-node 节点间可以不存在直接通信链路的。这 6 个节点将整个网络划分成 6 个部分,属于同一划分内的传感器节点只接受所属的 L-node 节点分配的调度任务。

问题的关键是如何从众多的传感器节点中选出这些 L-node 节点。为此,本章也提出了一种分布式算法产生 L-node 节点,L-node 节点不同于通常的分簇协议产生的簇头节点。所谓簇,就是具有某种关联的网络节点集合,每个簇由一个簇头节点和多个簇内成员组成。簇头节点的作用是承担本簇的成员节点与其它簇头节点的通信,传送或接受所有簇成员的信息、信息融合、以及传送接收融合过后的信息到

信息收集中心(Sink),簇头节点必须完成所有传感器节点到 Sink 节点的连通,因此,选举簇头节点的关键是除了簇头节点必须与本簇内的成员节点连通外,还必须保证所有的簇头节点是一个连通域。而 L-node 节点只负责完成集中式节点调度算法的任务,如果不加入其它的非 L-node 节点,纯粹的 L-node 节点可以不属于一个连通域。

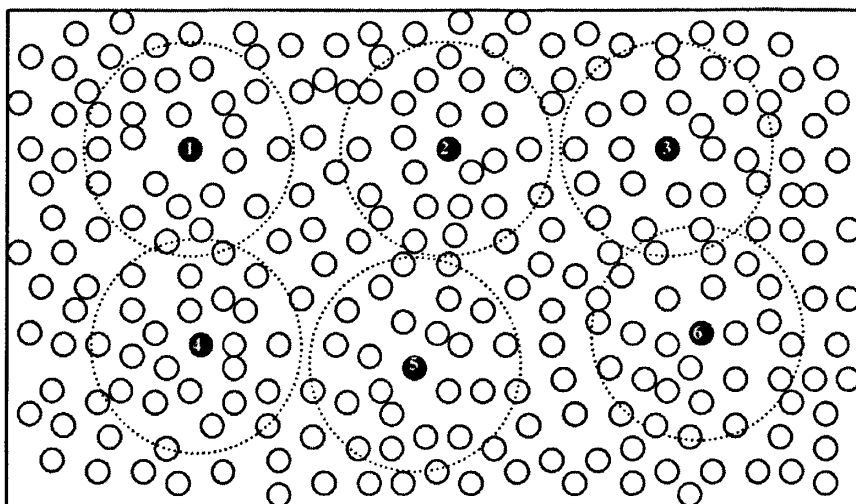


图 5.1 L-node 将整个网络划分为 6 个子网

另一个问题是如果算法选举产生过多的 L-node 节点,不利于集中式节点调度算法的优势,甚至会使得集中式节点调度算法变得毫无意义,因此,必须在算法中考虑 L-node 节点的稀疏性。L-node 节点不同于通常的簇头节点,它只负责执行本地的集中式算法,并不承担 L-node 节点间的通信任务,一旦完成本地传感器节点的调度,L-node 节点就可释放作为 L-node 节点的角色,转而成为一个普通的传感器节点。

5.2 相关工作

LCACA 算法与分簇算法有很多相似之处,分簇算法必须选举本簇的簇头,而 LCACA 算法则必须选举每个子网的 L-node 节点。

Deng J 等^[107]基于每个簇的休眠调度,提出了一种基于线性距离的调度算法 LDS。该算法是一种按距离的休眠调度方式,基本思想是传感器节点离簇头的距离越远,那么该节点休眠的概率越大。但是,对于无线传感器网络其主要任务是对目标对象的监测,因此必须考虑到所有的目标对象能否被覆盖这一基本要求,因此,

LDS 算法不适合无线传感器网络应用。

W. Heinzelman 等^[108]提出的 LEACH 是一个很有代表性的分簇算法,该算法是第一个针对无线传感器网络而提出的分簇算法。该算法的基本思想为:每个节点首先产生一个 $0 \sim 1$ 之间的随机数,如果这个数小于阈值 $T(n)$,该节点则向周围节点广播它是簇头的消息。 $T(n)$ 的计算公式为:

$$T(n) = \begin{cases} \frac{p}{1 - p \times [r \bmod (1/p)]}, & n \in G \\ 0, & \text{其它} \end{cases}$$

其中: p 是簇头节点当选簇头的概率, r 是当前分簇的轮合(round)数, G 是最近 $1/p$ 轮合中还未当选过簇头的节点集合。根据 $T(n)$ 计算方法可以看出,当选过簇头的节点在接下来的 $1/p$ 轮合中将不能成为簇头,剩余节点当选簇头的阈值 $T(n)$ 增大,节点产生小于 $T(n)$ 的随机数的概率随着增大,所以该节点当选簇头的概率增大。执行 LEACH 算法的每一轮可以归结为产生簇头、形成簇然后发现簇路由 3 个步骤。

在 LEACH 算法中, P 的值决定了每轮产生的簇头数量,在实际应用中,很难确定最佳的 P 值,这个值与网络的规模和节点的密度有关。此外,在 $T(n)$ 计算中,并没有考虑节点的剩余能量,因而,LEACH 算法必须基于两个前提才能达到每个节点平均消耗能量的预期目标:(1) 每个节点初始能量必须均等;(2) 每个节点担任簇头期间消耗的能量均等。然而,由于每个簇的大小以及簇头到基站(BS)的距离不一样,前提(2)很难符合现实的无线传感器网络。

另外,与 LEACH 算法相关的 LEACH-C^[109]是一种集中式的簇头选择算法,而 LEACH 算法则由每个节点根据随机数“自主决定”是否当选为簇头,每一轮产生的簇头的数量和位置是不确定的。LEACH-C 算法的思想是:首先,每个节点将每个节点的位置和当前剩余能量报告给基站,基站根据所有节点提供的这些信息计算平均剩余能量,剩余能量低于平均剩余能量的节点则不能成为候选簇头。利用这种方式虽然可以有效解决 LEACH 算法处理剩余能量的不足,但该算法必须根据全局信息由基站选择簇头,并且簇头集合和簇的结构必须通过基站广播出去。对于大规模无线传感器网络,这种算法也是不符合实际需要。

O. Younis 等^[110]根据延长网络的生存时间、可扩展性和负载平衡这 3 个原则,提出了一种能量有效的分布式分簇算法 HEED。HEED 算法选择簇头的基本原则主要依据剩余能量,剩余能量较多的节点将有较大的概率成为候选簇头,该节点最终能否被确定为簇头则取决于它的剩余能量是否比其周围邻居节点多很多。其次,在

确定簇成员节点时则依赖于簇内通信代价,根据通信代价最小化的原则将节点加入本簇,然后以 AMRP(簇内成员平均可达能量)做为衡量簇内通信代价的标准,从而将网络的能量消耗均衡地分布到全网。HEED 的簇头选择算法采用分布式的簇头产生方式;该在有限次迭代内完成;最小化控制报文开销;簇头分布均衡。HEED 的主要改进是在簇头选择中考虑了节点的剩余能量,并以主从关系引入了多个约束条件作用于簇头的选择过程。HEED 在簇头选择标准以及簇头竞争机制上都与 LEACH 不同。

I.Gupta 等^[111]根据 Mamdani 模糊逻辑方法提出了一种簇头选举算法 CEFL 算法,该算法将节点剩余能量、节点所在位置周围节点的密度以及节点靠近簇中心的程度做为输入变量,然后将这些输入变量量化为 3 个等级的隶属度,利用模糊判决规则得到 very small、small、rather small、medium、rather large、large 以及 very large 这 7 个输出。CEFL 算法采用重心法做为模糊判决规则,从模糊输出隶属函数中找出一个最能代表模糊集合的精确量,但该算法只能适合中等规模网络。

此外,与 LEACH 分簇路由协议相关的分簇算法还包括 TEEN 算法^[112], HEED 算法^[110]等。从上面的分析我们可以看出能量有效的问题在分簇算法中的地位,我们的工作不同于传感器节点的分簇问题,所有的簇头节点必须保持连通,因而要求担任簇头的节点具有比较大的发送和接收能力。而我们选举的 L-node 节点的作用是为了完成网络的重新规划,利用 L-node 节点充当中心处理节点的角色。

5.3 基于 L-node 节点的局部化点覆盖算法

基于 L-node 节点的本地化点覆盖算法,简称为 PCLA(Point Coverage Local Algorithm based on L-node Node)算法。该算法由选举 L-node 节点和基于 L-node 节点覆盖节点的调度两个主要的步骤完成。基本思想是:首先选举产生本地 L-node 节点;然后所有的非 L-node 节点自愿加入 L-node 形成以 L-node 节点为局部中心处理节点的组;最后,由 L-node 节点执行离散目标优化覆盖的集中式算法,从而完成节点的调度。LCACA 算法具有以下几个显著的特点。

(1)本地化算法

LCACA 算法是一种本地化算法,通过选出的 L-node 节点来充当集中式算法的中心节点角色。该算法仅仅依赖于本地或者局部节点的信息。事实上,要求任意一个节点掌握全局信息的通信量很大,容易导致节点能耗的大量消耗,因此,基于全局的集中式算法是不适合大规模无线传感器网络的。

(2)适合运行多种点覆盖的优化算法

LCACA 算法可以为多种面向全网的集中式算法提供局部化的解决方案,很多点覆盖的优化算法都可以运用到 LCACA 算法中。

(3)面向大规模密集型无线传感器网络，在保证覆盖目标的同时，兼顾了延长网络的生存时间

LCACA 算法能保证选举出的 L-node 节点稀疏地分布在网络中，这样可以充分地发挥集中式覆盖优化算法在延长网络生存时间处理方案，其次，在 L-node 节点的选举时，尽可能地减少报文传输的数量，有利于降低无线传感器网络在节点调度阶段的能耗。

(4)选举 L-node 节点无需时间同步

LCACA 算法开始运行时，由于网络延迟与时钟差异，导致每个节点运行算法时的时间差别，而这种特性不会影响算法的执行。

5.3.1 选举 L-node 节点

选举 L-node 节点分为 3 个步骤，即

- 节点状态约定
- 决定 DISELECTED 节点和 CELECTED 节点
- 从 CELECTED 节点中筛选出 L-node 节点。

5.3.1.1 传感器节点状态约定

假设每个传感器节点有以下 4 种状态之一：“INITIAL”、“CELECTED”、“ELECTED”以及“DISELECTED”。如图 5.2 所示为节点的状态转换图。

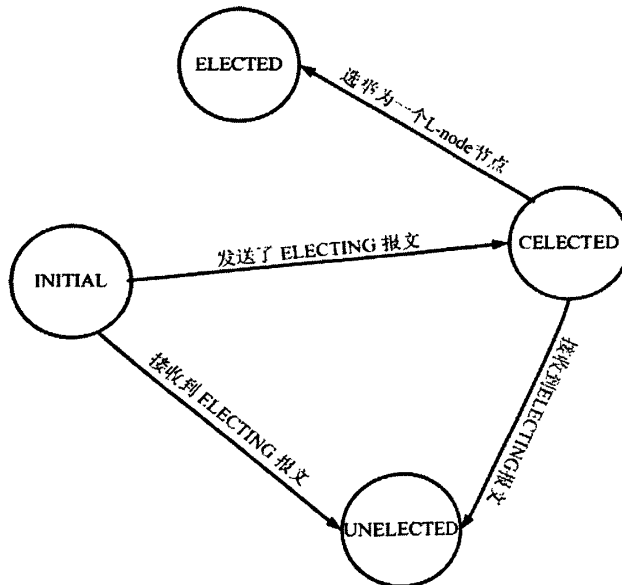


图 5.2 传感器节点的状态转换图

为了描述方便，我4种节点定义如下：

- (1) INITIAL 节点：状态为“INITIAL”，表示节点的初始状态。
- (2) CELECTED 节点：状态为“CELECTED”，表示节点成功发送了 ELECTING 报文，CELECTED 节点是候选 L-node 节点，它可能成为 L-node 节点，也可能成为 DISELECTED 节点。
- (3) DISELECTED 节点：状态“DISELECTED”，表明该节点接收到 Electing 报文但没有发送 Electing 报文，DISELECTED 节点是节点的最终状态，不可能再成为其它类型的节点。
- (4) L-node 节点：状态“ELECTED”，表示节点被选举为局部中心节点，它由部分 CELECTED 节点转换而来。

5.3.1.2 DISELECTED 节点和 CELECTED 节点的确定算法

下面我们介绍 DISELECTED 节点和 CELECTED 节点的确定算法。为了描述方便，我们将该算法简记为 DCA 算法。当 DCA 算法决定 DISELECTED 节点和 CELECTED 节点时，利用随机方式让部分传感器节点向其 k 跳邻居发送 ELECTING 报文。这样可以有效减少发送报文的开销， k 的取值根据具体的网络部署的密度而定。每个 ELECTING 报文的基本信息为 (v_{id}, L_v, TTL) ，其中的符号分别为传感器节点 v 的序列号、节点 v 的位置以及报文的 TTL (Time-to-Live)。ELECTING 报文表示发送这个报文的节点可能成为 L-node 节点。

假设每个节点在执行 DCA 算法前都处于“INITIAL”状态。在算法的第(1)~(3)部分，为了减少同一邻居的节点同时发送 ELECTING 报文的可能性，在随机发送 ELECTING 报文前，每个传感器节点等待一个随机延迟的时间 T 。在延迟的时间 T 内，如果接收到了来自其它节点的 ELECTING 报文，那么该节点转换为 DISELECTED 节点，不再发送 ELECTING 报文，并将这个报文存入 ELECTING 报文列表 $Elist$ 中。由于 DISELECTED 节点可能会接收多个 ELECTING 报文，以下我们用 $m_u(TTL)$ 表示报文 m_u 的 TTL ， $Elist$ 表示接收 ELECTING 报文的列表，CELECTED 节点将接收的 ELECTING 报文存入 $Elist$ 中。如果 $m_u(TTL) \neq 0$ ，那么它继续向其邻居转发这个 ELECTING 报文。

在算法的第(4)部分，对于 INITIAL 节点以概率 P_0 向 k 跳邻居发送 ELECTING 报文。当 INITIAL 节点成功地发送了 ELECTING 报文后，该节点转换为 CELECTED 节点，然后等待一常数时间。

考虑到不需要时间同步，每个节点以概率 P_0 发送 Electing 报文，可以将此事件看做在某一时间范围内的泊松事件流。假设 t 为从运行算法开始所经历的时间，概率 P_0 则按(5-1)式选取

$$P_0 = P\{v(t) = m\} = \frac{(\lambda t)^m}{m!} e^{-\lambda t} \quad (5-1)$$

在部署无线传感器网络时，可以预先设定 L-node 节点的数量，即确定了部署区域内 L-node 节点密度 λ 的取值。假设节点向其 k 跳邻居发送 Electing 报文，节点的传输半径为 R ，那么 m 按下面的方式选取： $m = \lceil k\pi\lambda R^2 \rceil$ 。

在等待时间 T_0 内，如果接收到其它节点的 Electing 报文，则将这个报文存入 Electing 报文列表 $Elist$ 中。由于 INITIAL 节点以概率 P_0 向 k 跳邻居发送 Electing 报文，有可能发送了也有可能没有发送。对于这两种情况，如果接收到其它节点的 Electing 报文，我们在算法的(4)部分还区别处理如下：

1. 如果没有发送 Electing 报文，那么该节点可能是 INITIAL 节点，也可能是 DISELECTED 节点。处理过程与算法的第(3)部分相同，但 DISELECTED 节点无论 $m_u(TTL)$ 是否为 0，它都不向其邻居转发这个 Electing 报文。
2. 如果节点成功发送了 Electing 报文，那么该节点转换为 CELECTED 节点，在等待时间结束时不管它是否接收到来自其它节点的 Electing 报文，其状态都不会改变。CELECTED 节点无论 $m_u(TTL)$ 是否为 0，它都不向其邻居转发这个 Electing 报文。

算法的第(5)部分处理那些既没有发送 Electing 报文又没有接收到 Electing 报文的 INITIAL 节点，在这种情况下，该节点将发送 Electing 报文的概率增加一个很小的常数值 $\delta(0 < \delta < 1)$ ，即 $P_0 \leftarrow \text{Min}(P_0 + \delta, 1)$ ，以便提高发送 Electing 报文的概率，然后，重新执行一次算法的(3)-(5)。表 5.1 描述了 DCA 算法的伪码。

表 5.1 DCA 算法的伪码

```

//for each sensor node v perform the following program
1. Input: initial energy  $E_0$ , residual energy  $E_i$ ;
2. compute the probability  $P_0$ ;
3. set a random delay period  $T$ ;
4. while( delay time  $T \neq 0$ ) do
5.     if receive an ELECTING message  $m_u$  from other sensor  $u$  then
6.         status  $\leftarrow$  DISELECTED;
7.         tabulate  $m_u$  in its ELECTING message list  $Elist$ ;
8.         if  $m_u(TTL) \neq 0$  then
9.             forward the message  $m_u$ ;
10.        Endif

```

```

11.      Endif
12.  Endwhile
13.  if status = INITIAL then
14.      broadcast a ELECTING message to its Neighbors with the probability
           $P_0$ ;
15.      if successfully transmits a ELECTING message then
16.          status ← CELECTED;
17.      Endif
18.      set a delay period  $T_0$ ;
19.      while(delay time  $T_0 \neq 0$ ) do
20.          if hear a ELECTING message  $m_u$  from other sensor u then
21.              if status = INITIAL then
22.                  status ← DISELECTED;
23.                  tabulate  $m_u$  in its ELECTING message list Elist;
24.                  if  $m_u(\text{TTL}) \neq 0$  then
25.                      forward the message  $m_u$ ;
26.                  Endif
27.              Endif
28.              if status = DISELECTED then
29.                  tabulate  $m_u$  in its ELECTING message list Elist;
30.              Endif
31.              if status = CELECTED then
32.                  tabulate  $m_u$  in its ELECTING message list Elist;
33.              Endif
34.          Endif
35.      Endwhile
36.  Endif
37.  if (status = INITIAL) then
38.       $P_0 \leftarrow \text{Min}(P_0 + \delta, 1)$ ;
39.      go to (3)
40.  Endif

```

5.3.1.3 从 CELECTED 节点中筛选出 L-node 节点

根据 DCA 算法,所有节点要么是 DISELECTED 节点要么是 CELECTED 节点,

下面从 CELECTED 节点中进一步筛选 L-node 节点，该算法简记为 CEL 算法。

为了从 CELECTED 节点中筛选出 L-node 节点，将 CELECTED 节点分为 2 种情况来考虑。一种情况是 CELECTED 节点没有接收到任何其它的 Electing 报文，即确定为 L-node 节点。另一种情况是 CELECTED 节点接收到了其它的 Electing 报文，即这样的 CELECTED 节点的 k 跳邻居内还有其它的 CELECTED 节点。这类 CELECTED 节点要成为 L-node 节点，当且仅当 $Thresh \geq K$ ，其中 K 为一个给定范围内的随机数。否则，将该节点降为 DISELECTED 节点。按照下列方式设置阈值 $Thresh$ ：

$$Thresh = (rat + \mu N_h) 2^{-\omega} \quad (5-2)$$

在 (5-2) 式中： ω ：接收到的 Electing 报文中的最小跳数。

N_h ：接收到的 Electing 报文总数。

μ ：一个权值。

$rat = \frac{E_v}{E_0}$ ：节点的剩余能量与初始能量的比值。

从 (5-2) 式可以看出，每个 CELECTED 节点根据自己接收的 Electing 报文数、剩余能量以及随机数 K 来确定自己能否当选为 L-node 节点。所有的 L-node 节点都必须向其 k 跳邻居节点发送 ISL-node 报文，表 5.2 描述了筛选 L-node 算法的伪码。

表 5.2 筛选 L-node 算法的伪码

```

//for each CELECTED node v perform the following program
1. Input: Thresh, Elist;
2. if Elist =  $\Phi$  then
3.   status  $\leftarrow$  ELECTED;
4.   transmits an ISL-node message to its k-hop Neighbors;
5. else
6.   generate a random number K;
7.   if Thresh  $\geq$  K then
8.     status  $\leftarrow$  ELECTED;
9.     transmits an ISL-node message to its k-hop Neighbors;
10.  else
11.    status  $\leftarrow$  DISELECTED;
12.  endif
13.  endif

```

5.3.2 基于 L-node 节点的覆盖节点调度算法

基于 L-node 节点的覆盖节点的调度算法由 2 类不同的传感器节点独立完成, 即状态为 DISELECTED 的节点操作算法和状态为 ELECTED 的节点操作算法。

5.3.2.1 状态为 DISELECTED 的节点操作算法

上一小节已经提出了选举 L-node 的算法, 为了完成传感器节点的调度, 这小节进一步提出传感器节点的工作调度算法。该调度算法不需要网络的全局信息, 而只在 L-node 及其成员节点间进行调度。在完成 L-node 选举后, 所有的传感器节点要么是 ELECTED 状态要么是 DISELECTED 状态。下面我们将详细描述状态为 DISELECTED 的传感器节点的操作算法。

每个转变为 DISELECTED 节点都延迟 T_d , 等待来自 L-node 的 ISL-node 报文。ISL-node 报文至少包含该 L-node 的序列号, T_d 的取值可以根据 ISL-node 报文要求传输的跳数而定。如果要求向其 k 跳邻居传输, 一般设置 $T_d > mT_t$, 保证有足够的时间到达目的节点并接收来自其它节点的 JOINING 报文。

由于 DISELECTED 节点可能会接收多个 ISL-node 报文。每个 DISELECTED 节点保存与其最近的 L-node 节点信息列表, 以下用 $Closest(L-node)$ 表示与它最近的 L-node 节点, $m_u(TTL)$ 表示报文 m_u 的 TTL。每当接收多个 ISL-node 报文, DISELECTED 节点计算自己到发送该报文的 L-node 节点的距离, 决定是否修改 L-node 节点信息列表中的 $Closest(L-node)$ 。ISLlist 表示接收 ISL-node 报文的列表, DISELECTED 节点将接收的 ISL-node 报文存入 ISLlist 中。如果 $m_u(TTL) \neq 0$, 那么它继续向其邻居转发这个 ISL-node 报文。

在结束 ISL-node 报文的接收后, 对于任意一个 DISELECTED 节点 v , 选择 $Closest(L-node)$ 做为它的 L-node 节点, 并将这个距离最小的 ISL-node 报文所经过的路由加入发送 JOINING 报文的路由列表。然后沿此路由由发送 JOINING 报文给它的 L-node 节点。在此时间内, 如果这个 DISELECTED 节点必须是其它 DISELECTED 节点发送 JOINING 报文中继节点, 那么它还得将所有接收到的 JOINING 报文保存在 $Jlist$ 列表中。然后, 将所有 $Jlist$ 列表中的 JOINING 报文与自己待发送的装配成一个 JOINING 报文, 最后发送此报文。

根据上述规则, 每个节点选择好 L-node 以后, 所发送的 JOINING 报文至少包含四个方面的内容: (1) 该节点的序列号, (2) 该节点的位置, (3) 该节点所覆盖的目标对象的信息, (4) 它所属的 L-node 的序列号。为了方面描述, 我们定义该报文的有效信息部分的格式为: (v_{id}, vID, L_v, C_v) , 其中的符号分别为传感器节点

v 的序列号、节点 v 所属的 L-node 的序列号、节点 v 的位置以及覆盖信息。表 5.2 描述了状态为 DISELECTED 的传感器节点的操作算法伪码， $Closest(L-node) = \infty$ 表示 $Closest(L-node)$ 为无穷远节点。

表 5.3 状态为 DISELECTED 的传感器节点的操作算法

```

//for each DISELECTED node v perform the following program
1.  Input: delay time  $T_d$ ,  $Closest(L-node) = \infty$ 
2.  while(delay time  $T_d \neq 0$ ) do
3.      if hear an ISL-node message  $m_u$  then
4.          if  $Dist(u, v) < Closest(L-node)$  then
5.               $Closest(L-node) \leftarrow u$ ;
6.              tabulate  $m_u$  in its ISL-node message list  $ISLlist$ ;
7.              if  $m_u(TTL) \neq 0$  then
8.                  forward the message  $m_u$ ;
9.              endif
10.         endif
11.     endif
12.     if hear a JOINING  $m_s$  then
13.         tabulate  $m_s$  in its JOINING message list  $Jlist$ ;
14.     endif
15. endwhile
16. assemble all JOINING message in  $Jlist$  into a JOINING
    message;
17. transmits a JOINING message to its  $Closest(L-node)$ .

```

5.3.2.2 L-node 节点的操作算法

当一个传感器节点被选举为 L-node 后，它设置一个等待时间 T_w 以便有足够的时间接收来自其成员节点的 JOINING 报文。虽然 T_w 的选择可能会影响网络的延迟，但对于无线传感器网络来说，算法执行的性质导致这种选择的次数不是频繁的，因而，不会对网络的快捷性造成太大的影响。设置定时时间 T_w 时必须考虑到是否能保证 L-node 接收到来自其成员节点的所有 JOINING 报文。

当 L-node 接收到来自其它节点的 JOINING 报文时，必须将发送该 JOINING

报文中的节点所属的 L-node 的序列号与自己的序列号进行比较。如果不一致则丢弃这个 JOINING 报文；否则，将该节点序列号、位置信息以及覆盖信息分别保存到其组成员列表和本地覆盖信息列表，并将发送 JOINING 报文的节点路由保存到相应的路由列表，当 L-node 以后发送 Group 报文时，将沿此路由发送至该节点。当 $T_w = 0$ 时，就认为所有传感器节点成员都已经加入，结束接收 JOINING 报文。

当 L-node 完成 JOINING 报文的接收任务后，获得了本地传感器节点的覆盖信息，接下来的工作是完成传感器节点的调度。L-node 执行通常的集中式算法后，将所有成员划分为相应的覆盖子集，并将每个覆盖子集分配不同的序列号。L-node 结束执行相应的集中式算法后，向它的所有成员发送 Group 报文。Group 报文至少包含(1)接收到 Group 报文的传感器节点所在的覆盖子集的序列号 CoverID。如果集中式算法容许同一个传感器节点可以属于多个覆盖子集的话，那么有些传感器节点可能不只一个这样的 CoverID，说明该节点不只属于一个覆盖子集。(2)该传感器节点所在分组号 GroupID，用于标识该传感器节点属于哪个 L-node，一般以所属的 L-node 的序列号作为 GroupID 号。

上述算法结束后，每个节点知道自己所在的覆盖子集 CoverID 以及相应的组 GroupID，每个节点根据 CoverID 和 GroupID 决定何时休眠何时活跃。特别地，每个 L-node 都有一个唯一的 ID 号，而采用集中式算法算法对所有的节点分组后，每个 L-node 的成员节点都有相应的分组 GroupID 对应。这样，可以用一个 3 元组表明一个传感器节点的调度策略。例如，传感器节点 v 的调度可以表示为 $(v_{id}, CoverID, GroupID)$ 。由于一些集中式算法算法允许一个节点同时属于不同的覆盖子集，因而，某些传感器节点可以不只具有一个 CoverID。L-node 节点的操作算法伪码如表 5.4 所示。

表 5.4 L-node 节点的操作算法

- | |
|--|
| <ol style="list-style-type: none"> 1. <i>while</i>($T_w \neq 0$)<i>do</i> 2. <i>receive the JOINING message from other node u</i> 3. <i>retrieval the information from the JOINING message from the node u ;</i> 4. <i>Endwhile</i> 5. <i>run a central schedual algorithm;</i> 6. <i>transmite the Group message to its group.</i> |
|--|

5.3.3 Group 报文的发送策略

前几节描述了执行 LCACA 算法的全部过程，每个的节点必须发送相应的报文

决定自己的角色。而发送和转发报文都要消耗节点的能量，特别是 L-node 节点发送的 Group 报文，Group 报文要到达目标节点，可能需要多个中间转发，因而，合理设计 Group 报文的发送策略非常重要。发送和转发 Group 报文主要的能耗来自两个方面，第一方面是发送的每个报文的长度；另一方面是 L-node 发送的 Group 报文数量，数量越多，传感器节点中继转发的报文可能越多，在传输和转发报文的过过程中可能会消耗大量的能量，不利于改善网络的生存时间。

当传感器节点发送报文时，必须启动无线收发模块电路，其开销比较大；因而增加每次发送报文的长度，可以减少无线收发模块电路的启动次数，将启动能量开销平摊到更多的数据上。基于 IEEE802.11^[102]的硬件发送数据包测试表明，无线设备在发送数据时的能耗，相对于总的发送功率而言，发送 1Mb/s 的数据包与发送 11Mb/s 的数据包之间的能耗差别<10%。况且，L-node 发送的 Group 报文要求到达的距离极其有限，将发送至多个传感器节点的 Group 报文按一定的规则融合一起，反而可以降低无线收发模块电路的启动次数，这有益于 L-node 在这个阶段节省能量。因此，在执行算法时可以忽略 Group 报文的长度对能耗的影响。

显然，节点发送和转发 Group 报文的能耗主要受第一个因素的影响。L-node 每发送一个 Group 报文，其传输覆盖范围内的节点可能接收到此报文。如果不运行任何报文转发策略，即使接收到的 Group 报文不属于自己的，它也会继续转发此报文，很容易造成局部广播风暴。因此，在执行前面的算法时，需要慎重考虑 Group 报文的发送策略。

策略 1: 在 L-node 覆盖范围内的节点，可以将多个节点的调度策略融合在同一个 Group 报文中，并在 Group 报文中标示传输的跳数为 1。当所有这些节点接收到这种报文时，如果包含自己的调度策略，则从中取出，否则丢弃该报文。利用该方法，可以有效减少 Group 报文发送的数量并防止节点进一步转发。如图 5.3(a)所示，G1、G2 与 G3 内的所有节点均 L-nodeC 的传输覆盖范围内，可以分别将 G1、G2 与 G3 内的所有节点的调度策略融合在一个 Group 报文中，这样可有效减少 Group 报文数量。

策略 2: 如果 Group 报文的节点不在 L-node 覆盖范围内，L-node 接收到 JOINING 报文时，必须登记哪些发送 JOINING 报文的节点属于这种情况，然后登记该报文到达 L-node 的最佳路径，有哪些 JOINING 报文是经过同一中继节点转发的。这样，当 L-node 发送的 Group 报文可以经同一中继节点转发，并且在同一个 Group 报文中融合该中继节点与所有这些节点的调度策略。当节点接收到这样的 Group 报文时，首先检查自己是否为该报文中继节点，如果是则从中取出属于自己的调度策略，并转发此报文；否则不再转发而直接丢弃该报文。如图 5.3(b)所示，传感器节点 s1、s2、s3、s4 以及 s5 都不在 L-nodeC 的传输覆盖范围内，它们必须

经过中继节点 s_0 转发才能接收到 Group 报文, L-node 则将 s_1 、 s_2 、 s_3 、 s_4 、 s_5 以及 s_0 的调度策略融合在同一个 Group 报文中。当节点 u 、 v 和 w 接收到这个 Group 报文时, 发现自己不是中继节点, 因而不转发而直接丢弃此报文即可。

为了执行 Group 报文发送策略, 当 L-node 接收到 JOINING 报文时, 必须登记哪些发送 JOINING 报文的节点所经过的路由信息, 建立传输相应的 Group 报文的最佳路径列表。如果 Group 报文的目的地不在 L-node 覆盖范围内, 那么, 还必须在 L-node 与相关的中继节点执行 Group 报文转发策略。

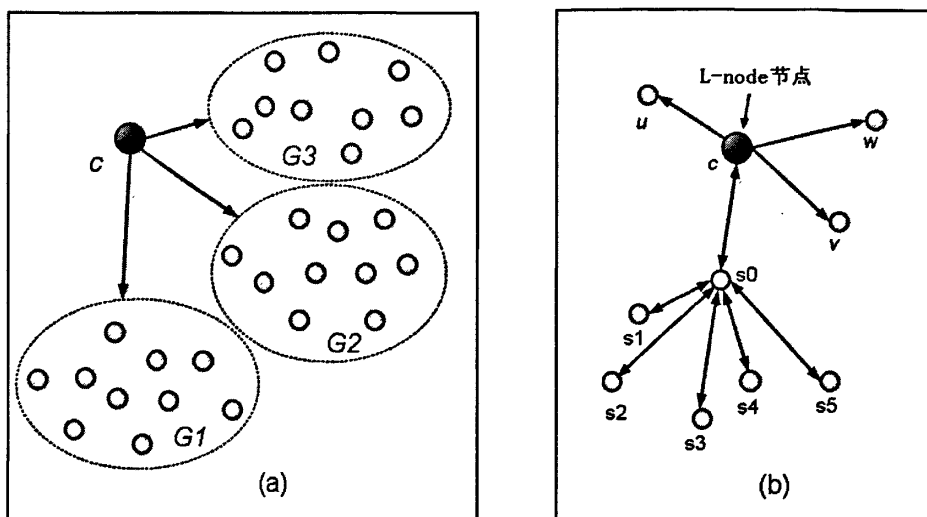


图 5.3 (a) 在 L-node 传输覆盖范围内

(b) 不在 L-node 传输覆盖范围内

5.4 LCACA 算法分析

上面详细描述了 LCACA 算法, 这小节分析 LCACA 算法的一些性质, 说明算法执行的效果。LCACA 算法的时间复杂性由两个方面决定: 选举 L-node 节点的时间复杂性与所采用的集中式覆盖优化算法的时间复杂性。

定理 5.1: 假设算法产生的 L-node 节点数为 s , 那么每个 DISELECTED 节点发送报文个数至多有为 $O(s)$ 。

证明 在 CDA 算法中, 每个 L-node 节点发送一个 ISL-node 报文, 最坏的情况下, 当 k 足够大时, 这个非 L-node 节点能接收到所有的 ISL-node 报文, 并且必须转发这些报文, 因而转发报文的个数最多为 s 。如果这个 DISELECTED 由 CELECTED 节点转换的, 那么它还发送一个 Electing 报文; 由于每个 DISELECTED 按照一定的路由单播 JOINING 报文一次, 在 k 跳范围内至多转发 Electing 报文一

次, 这样, 每个 DISELECTED 节点多发送 $(s+3)$ 个报文。因此, 每个 DISELECTED 节点发送报文个数至多为 $O(s)$ 。

引理 5.1 假设节点的传输半径为 R , 对于任意节点 DISELECTED 节点 s , 其 k 邻居中的节点中至少有一个节点发送过 Electing 报文。

证明 假设节点 s 的 k 邻居 $N_k(s)$ 内没有一个节点发送过 Electing 报文, 根据算法可知, 节点 s 不可能接收到 Electing 报文, 因而, s 的状态只能是 INITIAL 节点, 这时 s 将发送概率增加 δ 继续运行算法, 在有限次反复后, s 要么接收到 $N_k(s)$ 内的其它节点发送的 Electing 报文, 要么成功发送了 Electing 报文。

定理 5.2: CDA 算法是分布式的并且收敛。

证明 在 CDA 算法中, 所有的决定完全只依赖本地节点信息, 显然是分布式的。由引理 5.1 知, 所有的节点总能决定自己是 CELECTED 节点还是 DISELECTED 节点, 因而 CDA 算法收敛。

5.5 LCACA 算法性能评价

在本节从两个方面评价 LCACA 算法的性能: (1) 评价 L-node 节点选举的有效性; (2) 评价 LCACA 算法与相关的集中式算法活跃节点数以及网络的生存时间。假设在运行集中式算法和本地化算法后, 所有处于活跃状态的节点是连通的, 对于不连通的情况将在第 6 章进行讨论。

5.5.1 L-node 节点选举的有效性

仿真工作在 Matlab7 平台上, 仿真场景是一个 $500\text{m} \times 500\text{m}$ 的区域, 所有的节点均随机放置。

5.5.1.1 性能指标

在比较各种分簇算法的性能时, 主要考虑以下性能指标:

(1) 网络中的 L-node 节点数 L , 这个指标反映选举 L-node 节点后与实际制定的 L-node 节点数的差别。L 的选取以满足系统要求与减少能量开销为原则。

(2) L-node 节点成员数的标准差, 这个指标反映每个 L-node 节点所拥有成员的平衡性, 这个值越小, 说明每个 L-node 节点所拥有成员数越均匀。

5.5.1.2 参数设定与能耗模型

根据 L-node 节点选举算法的特点, 选举 L-node 节点算法包括 2 个步骤: 1. 决定 DISELECTED 节点和 CELECTED 节点的 DCA 算法; 2. 从 CELECTED 节点中筛选出 L-node 节点的 CEL 算法。我们将仿真中的参数设置如表所示 L-node 节点密

度 λ 的取值, 节点发送 ELECTING 报文的跳数 k , 即 ELECTING 报文的 TTL 值, 我们考察了 500~1000 个传感器节点均匀地分布在 $500m \times 500m$ 目标监测区域内, 假设所有节点的传输距离为 20m, 选举 L-node 节点时, ELECTING 报文发送至 3 跳邻居节点, 假设需要选举产生 20 个 L-node 节点。所有节点有相同的初始能量 $E_0=12J$, 剩余能量 E_i 则为均匀分布在 0~12J 内的随机数。 K 的取值为(0~1.5)间的随机数。

我们采用 W. R. Heinzelman 等^[113]提出的能量消耗模型做为无线信号模型。信号接收和发射阶段的能量消耗模型为: 接收和发射信号的能耗为 $E_e = 50nJ/bit$, 发射信号时用于放大电路消耗的能量为 $\zeta_{amp} = 100pJ/bit/m^2$ 。那么传感器节点发送一个 k 比特的报文的能耗为: $E_t(k, d) = E_e \times k + \zeta_{amp} \times k \times d^2$, 接收一个 k 比特的报文的能耗为: $E_r(k, d) = E_e \times k$ 。

表 5.5 DCA 算法与 CEL 算法的参数列表

DCA 算法参数							CEL 算法参数	
E_0	E_i	T	T_0	k	λ	δ	μ	K
12J	(0~12J)	200ms	200ms	3	20	0.05	0.75	(0~1.5)

5.5.1.3 结果与分析

图 5.4 描述了不同规模网络下 L-node 节点数与预定的 L-node 节点数之间的关系。从图中可以看出, 每个网络预设好 L-node 节点数后, 由于产生 CELECTED 节点的数量受 DCA 算法控制, CEL 算法则进一步限制了 k 跳邻居内 L-node 节点的数量, 因此, DCA 算法+CEL 算法能够产生非常接近预设值的 L-node 节点数。

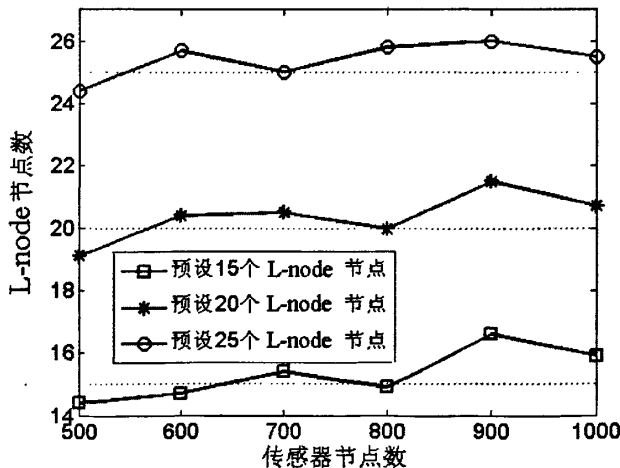


图 5.4 L-node 节点数与预设值间的关系

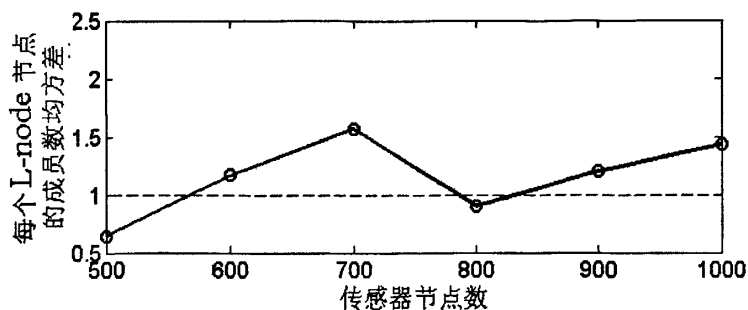


图 5.5 L-node 节点的成员数均方差与节点数的关系

图 5.5 描述了不同规模网络下 L-node 的成员节点数的均方差, 从图中可以看出, L-node 节点的成员节点数均方差 $0 \leq \theta < 2$, 反映了算法运行后每个 L-node 节点的成员节点数的差距很小, 节点分配比较均匀。

由上述仿真结果分析可知, L-node 节点选举算法表现出比较理想的性能, 所得 L-node 节点的分布比较合理, 表现为当预设的 L-node 节点时, 算法既能比较理想地满足预设要求, 而且各 L-node 节点的成员数趋于一致, 整体负载平衡性较好。

5.5.2 LCACA 算法的性能改善

我们假设基站位置在(0,0)点, 1000 个节点分布在大小为 $500\text{m} \times 500\text{m}$ 监测区域 A 内, 同时, 为了保证覆盖所有目标的传感器节点能保持连通, 我们假设有 600 个目标均匀分布在监测区域 A 内。

为了考察采用 LCACA 算法对网络生存时间的改善, 可取的方式是将 LCACA 算法中的集中式算法采用 M.Cardei 等^[104]提出的两种集中式算法, 它们分别为 LP-MSc Heuristic 算法和 Greedy-MSc Heuristic 算法, 然后将 LCACA 算法下的网络生存时间与这两个算法进行比较。在仿真中我们规定, 当节点 v 的剩余能量 $E_v \leq 0.2J$ 时, 认为该节点为死亡节点。

如图 5.6 所示为节点死亡数量与网络工作时间(Round)之间的关系。图 5.6 中可以看到, 当运行至第 1000 轮回(round)时, LP-MSc Heuristic 算法与 Greedy-MSc Heuristic 算法死亡的节点数都达到或将近 900 个, 而 LCACA 算法+LP-MSc Heuristic 算法与 LCACA 算法+Greedy-MSc Heuristic 算法死亡的节点数不够 700 个节点。因此表示无论是 LCACA 算法+LP-MSc Heuristic 算法还是 LCACA 算法+Greedy-MSc Heuristic 算法, 所获得的网络生存时间都比单纯的集中式 LP-MSc Heuristic 算法或 Greedy-MSc Heuristic 算法有所改善。

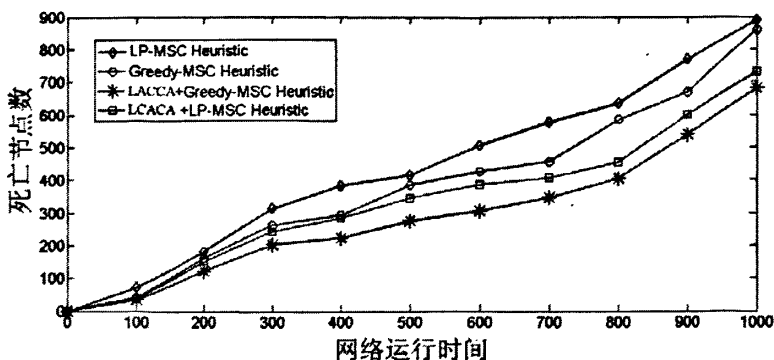


图 5.6 死亡节点数与网络运行时间的关系

5.6 小结

本章针对目前大规模无线传感器网络点覆盖问题研究实际情况，结合实际情况，提出了一种集中式算法局部化的思想，其目的是为了利用集中式算法在点覆盖精度比较高的优点，同时避免大规模无线传感器网络执行集中式算法的局限性。针对有些应用场景中被监测目标的重要性的差异，提出了对同一应用中不同监测区域的节点执行不同的调度算法的思想。为了实现上述构想，提出了一种点覆盖的局部化调度算法—LCACA。该算法包括节点分组与节点调度两个步骤。所谓分组就是通过选举 L-node 节点，然后将网络的节点指派给各自的 L-node 节点。而节点调度则是按实际应用需要，通过运行某种集中式节点调度算法或分布式节点调度算法，以达到点覆盖的目的。

最后，针对 LCACA 算法的有效性进行了模拟实验。仿真表明，选举 L-node 节点的算法可以获得更加合理的 L-node 节点分布，能够基本满足预设的 L-node 节点数的要求。同时，采用 LCACA 算法后的在同一时刻死亡的节点数比单纯的集中式算法要少得多，表明 LCACA 算法的性能比单纯的集中式算法有显著提高，并且这种局部化算法更适合无线传感器网络。

第六章 基于覆盖集的分布式连通算法

前几章针对密集型无线传感器网络的特性,提出了无线传感器网络的点覆盖算法。但是,前几章并没有考虑到最终的网络是否连通,而目前很多关于点覆盖的节点调度算法都假设覆盖与连通是统一的。实际上,对于点覆盖问题,即使所构造的覆盖集能完全覆盖所有离散目标,也很难保证这些活跃节点是连通的。

为此,本章研究点覆盖集的连通问题,首先将所有连通的活跃节点看成一个虚拟节点,在此基础上提出了一种基于 Steiner 树连通集中式算法。为了适应大规模无线传感器网络,本章又进一步提出了一种分布式连通算法,该算法将连通簇(指承担监测任务且相互连通的活跃节点)看成虚拟节点,以此为基础构造虚拟独立集。然后,利用在虚拟独立集构造连通的活跃节点集,该算法为解决点覆盖的连通问题提供了一种分布式求解方法。

6.1 引言

目前,主要有两种机制面向网络连通性问题的研究:(1)功率控制,即在保证网络连通的前提下,动态调整每个节点的发射功率,减少节点能量消耗,依靠节点之间以多跳方式实现数据转发^[114]。(2)通过节点状态调度,选取骨干节点进行数据路由从而构建连通域^[115-121],保证网络连通性。

近年来,如何保证网络覆盖且连通是一个热门问题。Zhang 等^[122]研究了提供高质量网络覆盖与连通的充分必要条件,证明当传感器节点的通信传输范围 R_t 与感知域 R_s 满足条件 $R_t \geq 2R_s$ 时,如果当前的工作节点能够完全覆盖整个监测区域,那么这些工作节点之间一定是连通的。Zhang 等将这种连通覆盖间的充分必要条件归结为如下的定理 1。

定理 6.1 对于密度有限的无线传感器网络,如果这个网络能完全覆盖监测区域,那么网络是连通的充分必要条件为 $R_t \geq 2R_s$ 。

X.Wang 等^[123]进一步证明了当 $R_t \geq 2R_s$ 时 K -覆盖(每个点至少被 k 个传感器覆盖)与 K -连通的一致性,即 K -覆盖的网络一定是 K -连通的,并提出了覆盖配置协议 CCP 算法。并且证明了当 $R_t \geq 2R_s$, 如果网络对一个凸监测区域是 K -覆盖的,那么这个网络一定是 K -连通的。

而对于 $R_t < 2R_s$ 的情况,通过将 CCP 算法与 SPAN 算法^[124]相结合,所产生的覆盖工作节点集也是连通的。根据传感器节点的位置信息,当 $R_t = 2R_s$ 时,SCR-CADS 算法^[125]则利用多跳转支配集 MPR-DS 算法^[115],通过调度传感器节

点的状态而实现了网络的连通覆盖。Gupta 等人^[126] 针对 Sink 节点所发送的查询监测区域的消息这种应用, 通过选择最小的连通节点集, 并且这个节点集是充分覆盖监测区域的, Gupta 等人^[126] 为此提出了 Self-Organization of Sensor Networks for Efficient Query Execution 的集中式与分布式两方面提出了相应的贪婪算法, 目的是通过选择节点连通路径以便达到最大的网络覆盖效果。Zhou 等人^[127] 则基于 Voronoi 图和 RNG 图, 从另一方面提出了在可调节通信传输范围 R_t 与感知域 R_s 的情况下, 保证网络 K 重覆盖和 K 度连通的算法 VRKCKC 算法; 该算法的基本思想是每个传感器节点根据本地的 K 阶 Voronoi Cell 设置它的 R_s 的大小, 而根据 K 阶 RNG 图来设置其 R_t 的大小, VRKCKC 算法也能实现无线传感器网络的 K 重覆盖与 K 度连通。此外, COWPOW^[128] 的基本思想是所有节点使用一致的发射功率, 在保证网络连通的前提下, 节点使其功率最小化。LINT^[129] 是基于节点度数的算法。Li 和 Hou 提出了基于邻近图理论的算法 DRNG^[130]; 上述无线传感器网络覆盖与连通协议都依赖于节点的物理位置信息进行状态调度, 维持网络连通性覆盖。

其实, 定理 6.1 只是给出了传感器节点的感知半径与通信半径关系的下界, 并没有给出上界, 因此, 如何根据传感器节点的感知半径来设置通信半径仍然是一个不确定的问题, 而且, 这种关系也只是面向传感器节点的区域覆盖领域。假设所部署传感器节点用于覆盖离散的目标, 当所有的活跃节点能完全覆盖目标对象时, 其情形又会如何呢? 下面我们先引入定理 6.2 分析一种两个离散目标对象相距为 R_s 时的情况。

定理 6.2 如果传感器节点 s_1, s_2 分别覆盖目标 t_1, t_2 , 传感器节点的通信传输半径和感知域分别为 R_t 与 R_s , 且 $|t_1 t_2| = R_s$, 那么, 节点 s_1, s_2 连通的条件 R_t 上界为 $3R_s$ 。

证明: 如图 6.1 所示, 如果传感器节点能覆盖目标 t_1 和 t_2 , 当且仅当这些传感器节点分别部署在半径为 R_s 的圆面 t_1 和 t_2 上, 假设 A, B 分别是的圆 t_1 和 t_2 上任意两点, 显而易见, $|AB| \leq |s_1 s_2| = 3R_s$, 这样, 只要传感器节点部署在圆面 t_1 和 t_2 上任意位置, 两传感器间的距离 $d \leq 3R_s$ 时, 所以, 节点 s_1, s_2 连通时 R_t 的上界为 $3R_s$ 。

从定理 6.1 可以看出, 由于点覆盖问题受到这些离散目标分布情况的制约, 当 $2R_s < R_t < 3R_s$ 时, 也不能使覆盖这两个目标对象的活跃节点连通。更进一步, 如果离散目标的分布不够集中, 那么保持活跃节点连通所需要的 R_t 就更加难以确定。同样, 我们在定理 6.2 中不能确定 R_t 的下界。其实, 当两目标的距离小于 $0 < |t_1 t_2| \leq 2R_s$ 时, 由于部署传感器节点的随机性, 可能一个传感器节点就将它们完全覆盖了, 也可能需要 2 个传感器节点才能覆盖它们。

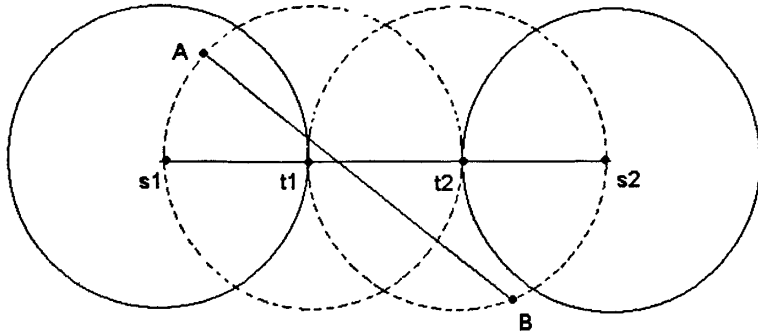
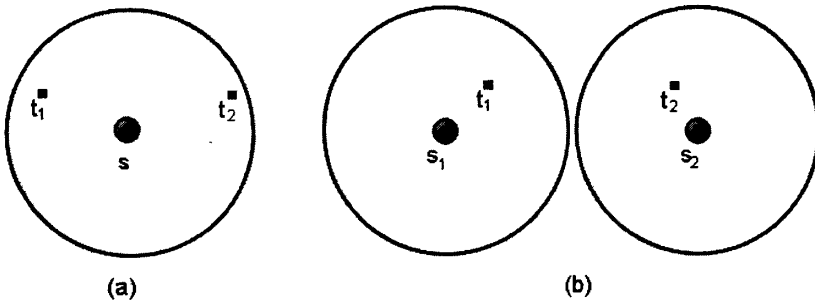


图 6.1 定理 6.2 图示

如图 6.2 (a) 所示, 目标 t_1 、 t_2 被随机部署的传感器节点 s 覆盖, 而图 6.2 (b) 所示由于目标 t_1 、 t_2 之间不存在如图 6.2 (a) 所示的传感器节点, 它们必须被随机部署的传感器节点 s_1 、 s_2 才能覆盖。从图 6.2 (b) 可以看出, 传感器节点 s_1 、 s_2 如果能相互通信的话, 它们的通信传输范围 $R_c = 2R_s$ 是不够的, 况且, 从节能的角度考虑, 过大的 R_c 也是很不利于有效使用节点的能量。可以看出, 对于点覆盖, 如果目标对象不够集中, 覆盖与连通往往是不一致的。

图 6.2 目标 t_1 、 t_2 被随机部署的传感器节点覆盖情况

针对上述分析, 本章将相互连通的活跃的节点看成一个虚拟节点, 在此基础上设计连通算法使这些满足一定覆盖要求的活动节点保持连通。并分别提出了基于已有覆盖集的集中式算法与分布式算法。这些算法在保证所有担任覆盖任务的传感器节点连通的前提下, 为了有效降低通信和管理的代价, 尽可能地减少承担中继任务的传感器节点。本章采用无向图 $G=(V, E)$ 来模拟无线传感器网络, 假设所有的传感器节点分布在 2 维平面, 每个节点的发送距离都相同, 并且接收距离与发送距离

相等。另外，每个传感器节点都有唯一的 ID，信号传输由节点 MAC 层负责。

6.2 构造 Steiner 树的集中式连通算法

无线传感器网络执行点覆盖算法后，承担监测任务的部分活跃节点相互连通，有些活跃节点之间则是不连通的。那么，活跃节点的连通问题可以抽象为构造 Steiner 树的问题，而这些活跃节点则抽象为 Steiner 点。据此，本节先给出将连通覆盖节点虚拟化方法，然后形式化定义连通问题，最后提出基于 Steiner 树的集中式连通算法 CCAST (Central Connected Algorithm based on Steiner Tree)，该算法为承担覆盖任务的节点选择中继节点。

6.2.1 基于 Steiner 树的连通问题

基于点覆盖问题在构造覆盖集时，并没有考虑节点间是否是连通的问题。点覆盖不同于区域覆盖，即使传感器节点完全覆盖了所有的目标，也不能保证所构造的覆盖子集是连通的。因此，还必须配置中继节点以保持所有活跃节点的连通，为了方便描述，下面给出 Steiner 树的及其相关的定义。

定义 6.1 Steiner 树

加权图 $G=(V,E)$, $V_{sub} \subset V$, $Stei_T$ 为 G 的子树，如果 $V_{sub} \subset Vertex(Stei_T)$ ，那么称 $Stei_T$ 是子集 V_{sub} 在图 G 上的 Steiner 树，点集 $Vertex(Stei_T)-V_{sub}$ 称为这棵 Steiner 树的 Steiner 点。

相应地，最小 Steiner 树问题则可以简单地叙述为：给定平面上若干个节点，如何将它们连接起来使得连线的总长最短。求给定节点集的 Steiner 树，最理想的解是找到相应的最小 Steiner 树，但是，求解最小 Steiner 树问题是 NP-难解的^{[131][132]}。除非 $P=NP$ ^[133]，得到最小 Steiner 树需要指数复杂度的算法^[134]。但在实际应用中，求解最小 Steiner 树通常是不可接受的，因而设计求解 Steiner 树的快速启发式算法具有非常重要的实际意义。

实际上，图的 Steiner 树与最小 Steiner 树往往统一称为一棵连接所有给定节点子集的子树，因而，Steiner 树问题就转化求出具有最小费用的 Steiner 树^[135]。求解 Steiner 树问题关键在于 Steiner 点的选取方法。如何有效构造 Steiner 树至今仍然被人们所关注。常见的如穷举法、动态规划等，但最小 Steiner 树精确解算法仅在小规模图问题上有效。在具体应用场景中，如果图 G 规模较大，那么寻求近似最小 Steiner 树就成为可行选择。近似最小 Steiner 树的重点是考察算法时间和算法的 Steiner Ratio 值(即近似最小 Steiner 树与最小 Steiner 树之间最大的路径代价比)，二者都是越小越好。但是，若限定每一种类型的最小 Steiner 树在某个时间(例如多

项式时间)内求解, 则近似方法的最佳 Steiner Ratio 界限值到底是多少仍然是一个有重大意义的理论研究课题。

最小 Steiner 树问题在无线通信网络和计算机网络等实际问题中有着广泛的应用, 许多实际通信问题都可以抽象为求解最小 Steiner 树问题。本节将无线传感器网络的连通覆盖问题转化为最小 Steiner 树问题, 并对这个问题展开研究。

如果将所有任意两个直接连通的节点间长度看成 1, 无线传感器网络的连通覆盖问题就是在已有覆盖节点集基础上寻找若干个节点, 使得这些覆盖节点能够连通。假设用带权的连通图 $G=(V, E, \omega)$ 表示给定的无线传感器网络, 其中 V 是节点集, E 是边集, $\omega: E \rightarrow R^+$ 为 E 中的每条边赋一个费用值。 V 中的每个元素表示一个传感器节点, 对于节点 $v_i, v_j \in V, e_{ij}=(v_i, v_j) \in E$ 当且仅当传感器节点 v_i, v_j 能直接通信。由于受到能量的限制, 并非任意的两个传感器节点之间都能进行直接通信的, 通常每个传感器节点只能与其临近的几个节点直接通信, 因而, 图 G 很多节点间不存在直接边, 图 G 并不是完全图。下面引入无线传感器网络的 Steiner 树问题。

定义 6.2 无线传感器网络 Steiner 树

给定无线传感器网络 $G=(V, E, \omega)$ 与节点集 $C \in V$ 为覆盖目标点/目标区域的传感器节点, 挑选传感器节点集合 R_c , 使得 $C \cup R_c$ 为连通集。

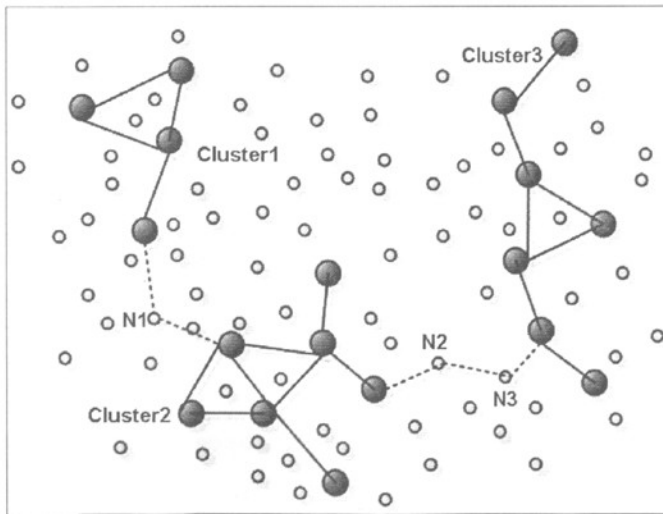


图 6.3 通过辅助节点连接 3 个连通簇

从定义 6.2 可以看出, 离散目标的连通覆盖问题, 就是在已构造的覆盖节点集的基础上, 加入尽可能少的节点, 使得所有这些节点是连通的。在这些承担覆盖任

务的节点中,可能存在部分节点之间相互连通的现象,也就是说当前担任覆盖任务的活跃节点由 k 个不连通的部分组成,将每个连通部分称之为连通簇。

如图 6.3 所示,每个连通簇由担任覆盖任务的节点以及这些节点间的通信边组成。那么,任意两个连通簇必须增加之外的辅助的通信链路才能连通,这种辅助的通信链路的两个端节点是这两个连通簇中的节点,其余节点则为两个连通簇之外的节点。如果我们将每个连通簇看成一个虚拟节点,那么,构造这些连通簇的最少节点的连接算法问题,实际上就是挑选新的节点做为 Steiner 点连同所有虚拟节点一起构成一棵最小 Steiner 树。

6.2.2 连通簇的虚拟节点化算法

上一小节提出了将一个连通簇看成一个虚拟节点,下面将详细描述如何将一个连通簇处理成一个虚拟节点的过程。

如图 6.4(a)所示,节点 s_1 、 s_2 及 s_3 这 3 个节点组成一个连通簇,边 e_8 、 e_{11} 、 e_{12} 及 e_{13} 与 s_1 相连、边 e_5 、 e_6 、 e_9 、 e_{10} 及 e_{11} 与 s_2 相连,而边 e_{10} 、 e_{14} 及 e_{15} 则与 s_3 相连,其中,边 e_{10} 及 e_{11} 为这个连通簇的内部边,其余的边的另一端点不属于这个连通簇。如图 6.4(b)所示当这个连通簇处理成一个虚拟节点 s 后,内部边 e_{10} 及 e_{11} 则消失了,其余的边不属于这个连通簇的端点保持不变,而属于这个连通簇的端点则处理成虚拟节点 s 。

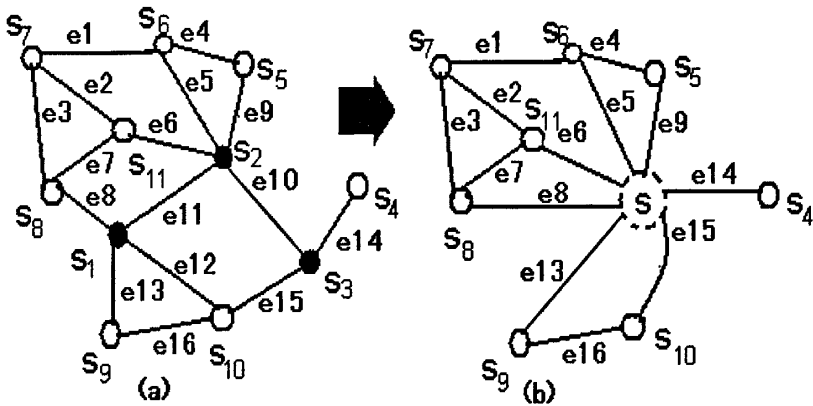


图 6.4 一个连通簇处理成一个虚拟节点图示

下面将介绍这个虚拟过程的形式化算子“ \vee ”的定义。一个通信图 $G=(V,E,W)$ 的节点间的关系可以描述为节点间的关系矩阵,首先定义矩阵 $A=A=(a_{ij})_{n \times m}$ 的列向量间的“ \vee ”运算。

定义 6.3 如果列向量 $\alpha = (a_1, a_2, \dots, a_n)^T, \beta = (b_1, b_2, \dots, b_n)^T$ 为布尔向量, 即 $a_i, b_i = 0$ 或 1 , 那么 $\alpha \vee \beta = (a_i \vee b_i)_{n \times 1}$, $a_i \vee b_i = 0$ 当且仅当 $a_i = b_i = 0$, 否则 $a_i \vee b_i = 1$ 。类似地, 行向量的“ \vee ”运算与之相同。

定义 6.4 对于矩阵 $A = (\alpha_1, \alpha_2, \dots, \alpha_m) = (a_{ij})_{n \times m}$, 不失一般性, 假设 $i < j$, 如果将它的第 i 列与第 j 列进行“ \vee ”运算后得到新的矩阵为:

$$B = (\beta_1, \beta_2, \dots, \beta_{m-1}) = (b_{ij})_{n \times (m-1)}$$

其中列向量 $\beta_i = \alpha_i \vee \alpha_j$, 当 $k \geq j$ 时, $\beta_k = \alpha_{k+1}$, 当 $i < k < j$ 及 $1 \leq k < i$ 时, $\beta_k = \alpha_k$ 。类似地, 矩阵的两行“ \vee ”运算后得到的新矩阵定义如下:

定义 6.5 对于矩阵 $A = \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_n \end{pmatrix} = (a_{ij})_{n \times m}$, 其中 $\alpha_i (i=1 \dots n)$ 为行向量, 不失一般性,

假设 $i < j$, 如果将它的第 i 行与第 j 行进行“ \vee ”运算后得到新的矩阵为:

$$B = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_n \end{pmatrix} = (b_{ij})_{(n-1) \times m}$$

其中行向量 $\beta_i = \alpha_i \vee \alpha_j$, 当 $k \geq j$ 时, $\beta_k = \alpha_{k+1}$, 当 $i < k < j$ 及 $1 \leq k < i$ 时, $\beta_k = \alpha_k$ 。

根据上述定义, 可以很容易得到处理虚拟节点的形式化过程, 具体过程的描述见定理 6.4。

定理 6.3 如果通信图 $G=(V, E)$ 所对应的关联矩阵为 A , 将节点 s_i 与节点 s_j 处理成一个虚拟节点 s 得到新的通信图 $G_1=(V_1, E_1)$, 那么 $G_1=(V_1, E_1)$ 的关联矩阵 B 为将矩阵 A 的第 i 列与第 j 列进行一次“ \vee ”运算, 然后再将第 i 行与第 j 行进行一次“ \vee ”运算后得到的矩阵。

一般地, 如果将 3 个或 3 个以上的节点虚拟成一个节点, 只需重复定理 6.4 的过程, 进行 2 次或 2 次以上的行列的“ \vee ”运算即可。

为了说明定理 6.4, 仍然以图 6.4 为例说明将连通簇处理成一个虚拟节点 s 的形式化过程。图 6.4(a) 对应的关联矩阵见表 6.1 所示, 将节点 s_1, s_2 及 s_3 这 3 个节点组成的连通簇虚拟成一个节点 s 得到图 6.4(b), 所对应的关联矩阵实际上将表 6.1 所示的矩阵的第 1、2、3 行分别进行行向量的“ \vee ”运算, 然后第 1、2、3 列分别进行列向量的“ \vee ”运算的结果, 如表 6.2 所示为图 6.4(b) 的关联矩阵。

表 6.1 图 6.4(a)的关联矩阵进行“ \vee ”运算的行列

		S										
		S ₁	S ₂	S ₃	S ₄	S ₅	S ₆	S ₇	S ₈	S ₉	S ₁₀	S ₁₁
S	S ₁	1	1	0	0	0	0	0	1	1	1	0
	S ₂	1	1	1	0	1	0	0	0	0	0	1
	S ₃	0	1	1	1	0	0	0	0	0	1	0
	S ₄	0	0	1	1	0	0	0	0	0	0	0
	S ₅	0	1	0	0	1	1	0	0	0	0	0
	S ₆	0	1	0	0	1	1	1	0	0	0	0
	S ₇	0	0	0	0	0	1	1	1	0	0	1
	S ₈	1	0	0	0	0	0	1	1	0	0	1
	S ₉	1	0	0	0	0	0	0	1	1	1	0
	S ₁₀	1	0	1	0	0	0	0	0	1	1	0
	S ₁₁	0	1	0	0	0	0	1	1	0	0	1

表 6.2 图 6.4(a)的关联矩阵进行“ \vee ”运算后结果

	S	S ₄	S ₅	S ₆	S ₇	S ₈	S ₉	S ₁₀	S ₁₁
S	1	1	1	0	0	1	1	1	1
S ₄	1	1	0	0	0	0	0	0	0
S ₅	1	0	1	1	0	0	0	0	0
S ₆	1	0	1	1	1	0	0	0	0
S ₇	0	0	0	1	1	1	0	0	1
S ₈	1	0	0	0	1	1	0	0	1
S ₉	1	0	0	0	0	1	1	1	0
S ₁₀	1	0	0	0	0	0	1	1	0
S ₁₁	1	0	0	0	1	1	0	0	1

定理 6.4 如果有 n 个节点中包含 k 个连通簇，其中节点数最多的连通簇为 C_p ，用 $|C_p|$ 表示 C_p 的节点数，那么，将连通簇处理成虚拟节点的时间复杂性为 $O(2k|C_p|)$ 。

证明 由于两个行向量或列向量间的“ \vee ”运算只是两个二进制数的“或”运算，对于一个有 r 个节点的连通簇，这样的行向量或列向量间的“ \vee ”运算次数为 $2(r-1)$ 。对于节点数最大的连通簇为 C_p ，行向量或列向量间的“ \vee ”运算次数为 $2(|C_p|-1)$ ，如果 n 个节点中包含的 k 个连通簇，那么处理虚拟节点的计算复杂性不大于 $O(2k|C_p|)$ 。

6.2.3 基于虚拟节点构造 Steiner 树的集中式算法

为了求出基于虚拟节点的最小 Steiner 树，将连通簇处理成虚拟节点后，接下

来建立加权图，我们采取的加权规则是由如下定义规定的。

定义 6.6 所有的虚拟节点的权为 0，其余节点的权则可以根据具体需要来确定。比如节点的剩余能量等都可以做为一个节点的权中的一个参数。

定义 6.7 路径 $Path = \langle v_1, v_2, \dots, v_k \rangle$ 的权等于 $\sum_{i=1}^k w_i$ ，其中 w_i 表示第 i 个节点的权。相应地我们规定树的权即其所有顶点的权之和。

根据上述加权规则，可以创建图 $G_1=(V_1, E_1)$ 的加权图 $G_w=(V_1, E_1, W)$ ，然后调用构造最小 Steiner 树算法得到 Steiner 树。虽然最小 Steiner 树问题是 NP-难解的，但是目前已经有很多的多项式时间近似算法^{[136][137]}，CCAST 算法采用 Robins 等^[142]Steiner 树算法去构造 Steiner 树。最后可以得到确保这些虚拟节点连通的辅助节点。该算法的时间复杂度决定于所选用的最小 Steiner 树算法。如果 C 表示担任覆盖任务的当前节点集，下面我们给出 CCAST 算法的具体描述。

表 6.3 CCAST 算法描述

1. **Input:** $G=(V, E)$, C .
2. **Output:** *Auxiliary node set A.*
3. **Begin**
4. *Find out all of connected clusters;*
5. *Each connected cluster is constructed a virtual node;*
6. *Construct Weighted Graphs $G_w=(V_1, E_1, W)$;*
7. *Construct a minimal Steiner tree of the graph $G_w=(V_1, E_1, W)$;*
8. *Return all of Steiner vertexes are the auxiliary node set A.*
9. **End**

定理 6.5 如果通信图 $G=(V, E)$ 处理虚拟节点后得到的通信图 $G=(V_1, E_1)$ ，CCAST 算法的最坏时间复杂性小于 $O(k^2(|V_1| + |E_1|))$ 。其中 k 为虚拟节点的个数。

证明 构建加权图 $G_w=(V_1, E_1, W)$ 可以使用宽度优先搜索算法 (BFS) 得到，而计算任意两个虚拟节点间的最短加权连通路的时间复杂性为 $O(|V_1| + |E_1|)$ 。那么得到加权图 $G_w=(V_1, E_1, W)$ 的时间复杂性为 $O(k^2(|V_1| + |E_1|))$ 。结合定理 6.4，那么完成 CCAST 算法中(1)、(2)、(3)步的时间复杂性为 $O(k^2(|V_1| + |E_1|))$ 。构建 Steiner 树算法可以在不大于 $O(k^2|V_1|)$ 时间内完成。从而 CCAST 算法的时间复杂性小于 $O(k^2(|V_1| + |E_1|))$ 。

6.3 基于构造虚拟独立集的分布式连通算法

在上一节,我们提出了一种基于 Steiner 树的集中式连通算法 CCAST 算法,集中式算法 CCAST 需要中心处理节点收集所有节点的信息,可扩展性不强。是很难得到实际应用的。对于大规模且资源受限的无线传感器网络,设计分布式连通算法非常重要。分布式算法中,每个节点根据本地获取的邻居信息,决定它与周边邻居的连通性质。因此,设计这种分布是算法的一个难点是:在一个节点独立决策它的活跃状态后,怎样保证所有的活跃节点能一起组成一个连通集?另一个难点是怎样减少活跃节点的数量?为了解决这些问题,本节提出一种虚拟独立集的思想。所谓虚拟独立集,是指图 $G=(V,E)$ 的一个顶点集 S 本不是一个独立集,但是,如果将 S 中存在边相连的所有节点看成一个虚拟节点,那么, S 就是一个独立集,但 S 并不是真正的独立集,因此,称 S 为虚拟独立集 VIS (Virtual Independent Set)。然后,在此基础上提出一种基于构造虚拟独立集 VIS 的分布式连通算法 DCAVIS (Distributed Connected Algorithm based on Virtual Independent Set)。

DCAVIS 算法由 2 个主要步骤完成:

1. 采用分布式算法求得网络图的虚拟独立集 VIS, 简称为 VIS 算法。
2. 在 VIS 基础求出生成树算法(Spanning Tree based on VIS), 简称为 STVIS 算法。

6.3.1 VIS 算法

由于 DCAVIS 算法是一个分布式算法,由于每一步都在单个节点进行,如果覆盖节点集为 C , 在加入其它非覆盖节点 C_c 后,问题是如何判断这些 $C \cup C_c$ 是连通的?实际上,这是一个确定算法终止条件的问题。理想的终止条件是每个节点必须将其所连接的覆盖节点列表告知其它节点,每个节点都应知道全部节点的连通情况。但这样将产生大量的报文,不利于节能,显然也是不可行的。

为了使所有的活跃节点最终能形成一个连通集,DCAVIS 算法通过构造一种特殊的节点集 VIS,然后根据这种 VIS 的特性判断算法是否终止。为了便于描述算法,假设除去承担覆盖任务的节点外,有足够多的节点保持这些覆盖节点的连通。用 black 节点表示虚拟独立集的元素,其它颜色的节点表示非虚拟独立集的元素,VIS 算法就是确定 black 节点的过程。

所有承担覆盖任务的节点标示为 black 节点,其余的节点标示为 white 节点。对于一个 white 节点,如果它的邻居节点中没有任何 black 节点,那么将这个 white 节点着色成 black 节点,并将它的所有邻居节点着色成 grey 节点;当所有的 white

节点被着色成 black 节点或者 grey 节点后, 算法终止。具体的算法的形式化描述如表 6.4 所示。

表 6.4 VIS 算法

Input: *A undirected graph $G=(V,E)$, A cover set C*

Output: *A Independent Set M*

1. $M=C, W = V - C$;
2. *All nodes in M are black, others are white;*
3. *While the White node set $W \neq \Phi$ do*
4. *Find a white node $v \in W$ such that $Neighbor(v) \cap M = \Phi$;*
5. *All in $Neighbor(v)$ are grey, the node v is colored black;*
6. $M \leftarrow M \cup \{v\}$;
7. $W \leftarrow W - \{v\} - Neighbor(v)$;
8. *End while*
9. *Return M ;*

6.3.2 VIS 算法性质

在上一小节已经提出了 VIS 算法, 执行该算法的目的是为了构造一个虚拟独立集 VIS, 为进一步构造连通集做准备, 那么 VIS 究竟具有什么特性, 为了方便能清楚地表达 DCAVIS 算法的基本思想, 下面分析 VIS 的性质。

定理 6.6 VIS 算法最后得到所有 black 节点构成一个独立集。如果存在担任覆盖任务的连通簇, 那么将这样的连通簇看成一个虚拟节点后, 这样的一个虚拟节点就是的一个 black 节点, VIS 算法最后得到所有 black 节点连同虚拟的 black 节点一起构成一个虚拟独立集 VIS。

证明 假设算法最后生成 black 节点集合 M , 那么, M 只能由两种不同的元素通过算法获得, 一种是担任覆盖任务的节点, 另一种是由 white 节点转换来的。 $\forall u \in M$, 以及 $\forall v \in M (v \neq u)$, 如果 u, v 都是担任覆盖任务的节点, 由上面的约定, 它们不属于同一个连通簇, 所以不可能相邻。如果 u, v 中至少有一个是由 white 节点转换来的节点, 不妨设 v 是由 white 节点转换来的, 根据算法 $Neighbor(v) \cap M = \Phi$, 即 v 的所有邻居节点都是 grey 节点, 因此, $u \notin Neighbor(v)$, 这表明所有的 black 节点是不相邻的, 从而所有 black 节点构成一个独立集。考虑

到有可能存在连通簇而出现虚拟 black 节点，所以由所有 black 节点构成的独立集可能是一个虚拟独立集 VIS。

定义 6.8 PBG 路径

如果路径 T 的两端点为不相邻的两个 black 节点，中间节点由 grey 节点组成，则称路径 T 为一条 PBG (Path from a Black node to Grey node) 路径。如果这条 PBG 路径的 grey 节点数不多于 2 个，则称这样的 PBG 路径为邻接 PBG 路径，否则称为非邻接 PBG 路径。

定理 6.7 在 VIS 算法生成的独立集 M 中，任何节点总可以经由一条或者数条邻接 PBG 路径到达 M 中的任一个节点。如果将担任覆盖任务的连通簇看成一个 black 节点，那么，虚拟独立集 M 中的任何节点总可以经由一条或者数条邻接 PBG 路径到达 M 中的任一个节点。

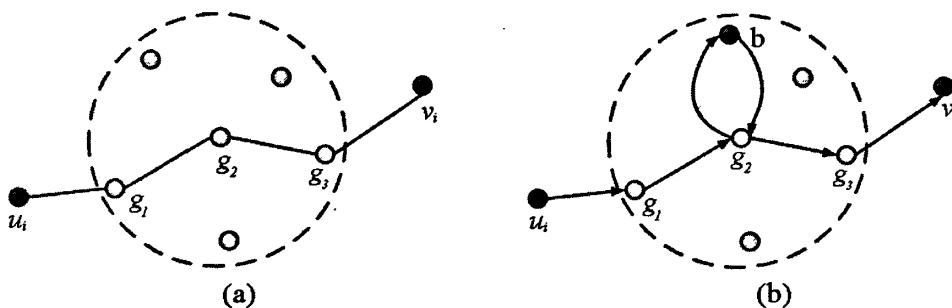


图 6.5 定理 6.7 的图示说明

证明 $\forall u, v \in M$ ，反设 u 到 v 的 PBG 路径中包含一条非邻接 PBG 路径，那么这条非邻接 PBG 路径至少包含 3 个 grey 节点。不失一般性，不妨设这条非邻接 PBG 路径包含 3 个 grey 节点，用 $\langle u_i, g_1, g_2, g_3, v_i \rangle$ 表示这条非邻接 PBG 路径， g_1, g_2, g_3 为 grey 节点。如果 g_2 的邻居节点集 $Neighbor(g_2) \cap M = \Phi$ ，如图 6.5(a) 所示，根据算法， g_2 不可能是 grey 节点，而只能是一个 black 节点，那么 $\langle u_i, g_1, g_2, g_3, v_i \rangle$ 不是一条 PBG 路径。另一方面，假如 g_2 的邻居节点集 $Neighbor(g_2) \cap M \neq \Phi$ ，如图 6.5(b) 所示存在 black 节点 $b \in Neighbor(g_2)$ ，那么，可以找到邻接 PBG 路径 $\langle u_i, g_1, g_2, b \rangle$ 和邻接 PBG 路径 $\langle b, g_2, g_3, v_i \rangle$ 连通 u_i, v_i ，总之，无论是哪种情形，都与假设矛盾的。这样就证明了定理 6.7。

6.3.3 VIS 算法分布式执行

上一小节已经提出了 VIS 算法，根据无线传感器网络应用特点，在这小节将提

出这种算法的分布式执行方式, VIS 的分布式版 DVVIS (Distributed Version of VIS Algorithm) 算法。为了保证算法运行后能保证所有的 black 节点连通, 假设整个网络的时钟同步, 目前已经有很多研究解决无线传感器网络的时钟同步问题, 目前已经提出很多比较经典的时钟同步算法, 如 TPSN 算法^[138]、FTSP 算法^[139]、LTS 算法^[140]、GCS 算法^[141]等。并且每个节点发出的报文能够被其 1 跳邻居正确接收。每个节点维护一张 degree 表, 用于保存 blue 节点的信息, DVVIS 算法的描述如下:

- 1) 当前担任覆盖任务的节点着色为 black 节点, 其余的节点着色为 white 节点。
- 2) 每个 black 节点广播一个 BLACK 报文, 每个 BLACK 报文的生存时间只有一跳。
- 3) 当 black 节点接收到 BLACK 报文时, 直接丢弃此报文。
- 4) 当 white 节点 v 接收 BLACK 报文时, 表明它的邻居节点集 $Neighbor(v)$ 包含 black 节点, 则将自己着色为 grey 节点。
- 5) 为了寻找可以转换为 black 节点的 white 节点, 所有的 grey 节点广播一个生存时间只有一跳的 GREY 报文。
- 6) 当 black 节点或 grey 节点接收到 GREY 报文时, 不进行任何操作, 直接丢弃此报文。
- 7) 当 white 节点接收到 GREY 报文时, 则将自己着色为一个 blue 节点, 然后向其一跳邻居节点广播一个生存时间只有一跳的 BLUE 报文。
- 8) 等待一延迟时间 τ_w 后, 如果 blue 节点没有接收到任何 BLUE 报文, 则将自己着色为 black 节点。
- 9) 在延迟时间 τ_w 内, 当 blue 节点收到 BLUE 报文时, 该节点的邻居 blue 节点数 $degree+1$, 并记录发送者的 ID; 其它节点收到 BLUE 报文时则不进行任何操作。
- 10) 延迟时间 τ_w 后, 每个 blue 节点统计所接收到的 BLUE 报文个数, 然后发送 Degree 报文给其邻居 blue 节点。
- 11) 每个发送过 Degree 的 blue 节点等待时间 τ_d 接收 Degree 报文。
- 12) 当 τ_d 结束后, 判断自己是否为 degree 最小的节点。如果是则将自己着色为 black 节点。如果其所有邻居 blue 节点的 degree 都相同, 则按 ID 的值确定是否着色为 black 节点, 可以取 ID 最小者为 black 节点。
- 13) 新着色的 black 节点向其一跳邻居节点广播一个 BLACK 报文。
- 14) 所有 blue 节点继续等待时间 τ_d 接收 BLACK 报文。如果接收到此报文, 则将自己着色为 grey 节点, 否则将不做任何操作。
- 15) 每个新着色的 grey 节点向其一跳邻居节点广播一个 GREY 报文。

16) 当 blue 节点接收到此报文时, 则从 degree 列表中删除发送者, 并将该节点的邻居 blue 节点数 degree-1。

17) 然后重复过程 6)-16), 直到没有 blue 节点为止。

定理 6.8 在 DVVIS 算法中, 每个 blue 节点都能决定自己是 grey 节点还是 black 节点。

证明 根据 DVVIS 算法的第 7) 步可知, 所有的 blue 节点都由 white 节点着色来的, 即当 white 节点 u 接收到 GREY 报文时, 节点 u 将自己着色为一个 blue 节点, 那么 u 的邻居节点集 $Neighbor(u)$ 中不可能包含 black 节点。由算法的第 8) 步知, 如果一个 blue 节点 v 的 $Neighbor(v)$ 中不包含任何 blue 节点, 那么 v 着色为 black 节点。从算法的第 9) 到 16) 步实际上是处理连通的 blue 节点, 当执行完这些步骤后, 至少有一个 blue 节点被着色为 black 节点。所以, 在迭代若干次后, 最后所有的 blue 节点都能决定自己是 grey 节点还是 black 节点。

定理 6.9 如果将连通的覆盖节点看成一个 black 节点, 那么, 运行 DVVIS 算法后, 所有的 black 节点形成一个 IS (Independent Set), 并且, 任何两个 black 节点间一定由若干条邻接 PBG 路径相连。

证明 如果将连通的覆盖节点看成一个 black 节点, 那么算法运行后, 显然, 任何两个 black 节点不可能相邻, 从而所有的 black 节点组成一个独立集。

如果两个 black 节点 u, v 间相隔一个或二个 grey 节点, 那么结论成立。如果两个 black 节点 u, v 间相隔三个或三个以上的 grey 节点, 不失一般性, 假设两个 black 节点 u, v 间只相隔三个 grey 节点, 如果是图 6.5(a) 所示的情形, 根据算法可知, 必与假设矛盾。如果是 6.5(b) 所示的情形, 则存在两条邻接 PBG 路径连接节点 u, v 。同理可以证明两个 black 节点 u, v 间相隔三个以上的 grey 节点的情形。

定理 6.10 DVVIS 算法总的报文数至多为 $O(n^2)$ 个, 算法时间复杂度为 $O(n^2)$ 。

证明 如果将连通的覆盖节点看成一个 black 节点, 假设开始时有 k 个 black 节点, 那么在算法的第 2) 步, 每个 black 节点发送一个 BLACK 报文, 一共发送 k 个 BLACK 报文。这样, 至多存在 $(n-k)$ 个 grey 节点, 每个 grey 节点发送一个 GREY 报文, 那么在算法的第 5) 步至多发送 $(n-k)$ 个 GREY 报文。同时, k 个 black 节点至少产生 $\log k$ 个 grey 节点, 那么, 由此产生的 blue 节点数至多为 $(n-k-\log k)$ 个。由于每个 blue 节点发送一个 BLUE 报文和一个 Degree 报文, 因而至多发送 $2(n-k-\log k)$ 个报文。DVVIS 算法最坏的情况是每次迭代只有一个 blue 节点着色为 black 节点, 然后每个新着色的 black 节点发送一个 BLACK 报文, 这样至多迭代 $(n-k-\log k-1)$ 次。综上所述, 最后, DVVIS 算法总的报文数至多为 $O(n^2)$ 个。而且, 很容易得到算法时间复杂度为 $O(n^2)$ 。

6.3.4 STVIS 算法

本节将在前一节的基础上构建连通的覆盖集，这个连通集包括了所有的 black 节点和部分 grey 节点。

每个 black 节点首先发送一个 INVITATION 报文给所有 3 跳内的 black 节点，当 black 节点接收到这个报文时，查询该报文经过的 grey 节点并将其记录在 $G(v)$ 列表里，将 black 节点记录在 $B(v)$ 列表里，这个过程一直到 3 跳内所有的 black 节点连通为止。当 grey 节点接收到这个报文时，如果这个报文经过的跳数没有超过 2 跳，就继续转发此报文，

该算法利用两种类型的报文：INVITATION 报文和 JOIN 报文。INVITATION 报文由 black 节点发出并由 gray 节点转发给其它 black 节点或 gray 节点。INVITATION 报文由两个基本部分组成：节点的 ID 以及该报文的有效跳数，我们用 $\langle vID, hop \rangle$ 描述这个报文。当 black 节点发出 INVITATION 报文时 $hop = 3$ ，以后，每转发一次减去 1。

JOIN 报文是 black 节点为了应答接收到 INVITATION 报文而发出的。JOIN 报文沿着原来传送 INVITATION 报文路径返回到原 black 节点，因而 JOIN 报文属于单播报文。JOIN 报文由两个基本部分组成：发送 INVITATION 报文的 black 节点 ID、发送 JOIN 报文的 black 节点 ID 以及所经过的路径上的所有 grey 节点列表。

算法开始时，每个 black 节点发送一个 INVITATION 报文。当 grey 节点接收到 INVITATION 报文时，检查 hop 的值是否小于 1，如果是则丢弃此报文，否则，将自己加入 INVITATION 报文中的路径，同时 $hop = hop - 1$ ，然后转发经过这些修改后的 INVITATION 报文。当 black 节点接收到 INVITATION 报文时，发送 JOIN 报文，这个 JOIN 报文以单播形式沿着 INVITATION 报文返回到 black 节点，每个 JOIN 报文中都包括路径上的所有 grey 节点的列表。

当 grey 节点接收到 JOIN 报文时，检查自己是否在 JOIN 报文中的 grey 节点列表里，如果是将自己标记为一个活跃节点并转发此报文，否则，丢弃此报文。当 black 节点接收到 JOIN 报文时，将发送这个 JOIN 报文的 black 节点加入 black 节点列表，这个分布式算法的进一步描述在表 6.5 中。

在前面我们将一个连通簇看成一个 black 节点，没有考虑相邻的 black 节点的情形，因此，上述算法在实际应用中，必须稍做修改，当 black 节点接收到 INVITATION 报文，如果这个报文来自相邻的 black 节点，在不考虑时钟是否同步的情况下，当该节点没有发送过 INVITATION 报文时，则将自己的 ID 加入报文，并转发这个 INVITATION 报文，不改变报文中的 hop 值。如果以及发送过

INVITATION 报文，则直接丢弃此报文即可。

表 6.5 STVIS 算法

<p>Input: <i>A black node Set M, a grey node set G</i></p> <p>Output: <i>A Connected Cover Set</i></p> <ol style="list-style-type: none"> 1. <i>Each black node broadcast a INVITATION message;</i> 2. <i>Upon receiving the INVITATION message, grey node v does as follow:</i> 3. If <i>hop > 1 then</i> 4. <i>hop ← hop - 1;</i> 5. <i>Appending v to the path in the INVITATION message;</i> 6. <i>Forwarding the new INVITATION message;</i> 7. Else 8. <i>Grey node v does nothing;</i> 9. End if 10. <i>Upon receiving the INVITATION message, black node v sends a JOIN message;</i> 11. <i>Upon receiving the JOIN message, black v appends the black node which sent the JOIN message to its black list;</i> 12. <i>Upon receiving the JOIN message, grey node v does as follow:</i> 13. If <i>the path in the JOIN message includes grey v then</i> 14. <i>Grey v colors itself blue;</i> 15. else 16. <i>Grey node v does nothing;</i> 17. End if

定理 6.11 DCAVIS 算法运行后,所有的 black 节点连同 blue 节点构成连通集。

证明 如果存在 black 节点是分离的,不失一般性,不妨设 black 节点 v_i, v_j 是分离的,根据定理 6.9 可知, v_i, v_j 总可以经过一条或数条邻接 PBG 路径连通,即可以经过其它的邻接 PBG 路径使之连通。由于每个 INVITATION 报文最多能被传送到 3 跳远的节点,即每个 3 跳内的 black 节点都能接收到这个 INVITATION 报文,因此,算法运行后,加入了 blue 节点,所有的 black 节点连同相关联的 blue 节点一起构成邻接 PBG 路径。由上述算法可知每个 black 节点都通过邻接 PBG 路径与其 3 跳内的所有 black 节点连通,因而 v_i, v_j 不可能是分离的。定理的正确性得以证明。

6.4 性能评价

为了评估算法性能,本仿真在 Matlab7 平台下设计了基于虚拟节点连通算法的仿真程序。在 $400m \times 400m$ 目标区域中,随机部署了 1 000 个节点。全部仿真采用表 3.2 所示的能耗模型。为了测试 CCAST 与 DCAVIS 算法两种算法的性能,假设有 10 个连通簇,每个连通簇的节点数为 5 个,10 个孤立的节点,所有这 60 个节点共同承担覆盖任务,并且随机分别在这个区域中,假设 Sink 节点位于区域的原点(0,0),每个处于活跃状态的工作节点每 10 秒钟向 Sink 节点报告数据。

对于 DCAVIS 算法,首先将每个连通簇看成一个虚拟节点后的节点数,在此基础上调用 VIS 算法构造虚拟独立集,然后调用 STVIS 算法求出连通节点。对于集中式算法 CCAST,则每个轮回开始时,每个节点必须向 Sink 节点报告它们的邻居信息,这样处理的目的是相当于 Sink 节点了解哪些节点已经死亡,以便构造新的通信图。仿真实验从以下两方面考察算法的性能。

- (1) 维持覆盖集连通而增加的节点个数。连通算法致力于在已选择了若干个节点维持覆盖的条件下激活最少的节点,以确保这些节点与覆盖集构成的网络是连通的。如果新增加的节点较多,网络中在一个轮回调度工作周期内持续工作的节点较多,因此会缩短整个网络的生存时间。
- (2) 网络生存时间。在确保所有目标点完全被覆盖和以及网络连通的条件下,考虑网络生存时间。可以从两个方面说明:(a)如果网络按照随机调度的方式担任覆盖任务,每一轮的活跃节点依照时间线的方式依次轮流工作,那么,网络生存时间是指从网络开始工作到网络中出现一定数量的失效节点所经历的时间长度。(b)如果将网络中的传感器节点分成若干个覆盖子集,每次让一个覆盖子集作为活跃节点担任覆盖任务,那么,每个活跃子集的生存时间是指,从该活跃子集开始工作到出现一定数量的失效节点所经历的时间长度,而网络生存时间则为所有处于覆盖子集依次轮流工作时总的生存时间。

6.4.1 维持分组连通时额外加入到分组内的节点个数

图 6.6 为节点在不同的通信半径下,执行 DCAVIS 算法和 CCAST 算法额外增加的节点个数,图中结果为执行 50 次实验的平均值。从图中可以看出,若采用 CCAST 算法,当节点半径从 $20m$ 增加到 $30m$ 时,增加的节点个数从 52 个左右减少到 25 个左右,而执行 DCAVIS 算法则从 83 个左右减少到 50 个左右。随着节点通信半径的逐渐增加,两种算法维持连通而需要增加的节点个数递减的程度逐渐降

低。在相同的节点通信半径的条件下，DCAVIS 算法比 CCAST 算法需要加入的节点数要多。这是因为 DCAVIS 算法中必须构造虚拟独立集，并且采用的是分布式算法，而 CCAST 算法则采用的是集中式算法，两者的精度显然不一致的结果。因此，在执行连通算法后，DCAVIS 算法增加的节点个数也较多。

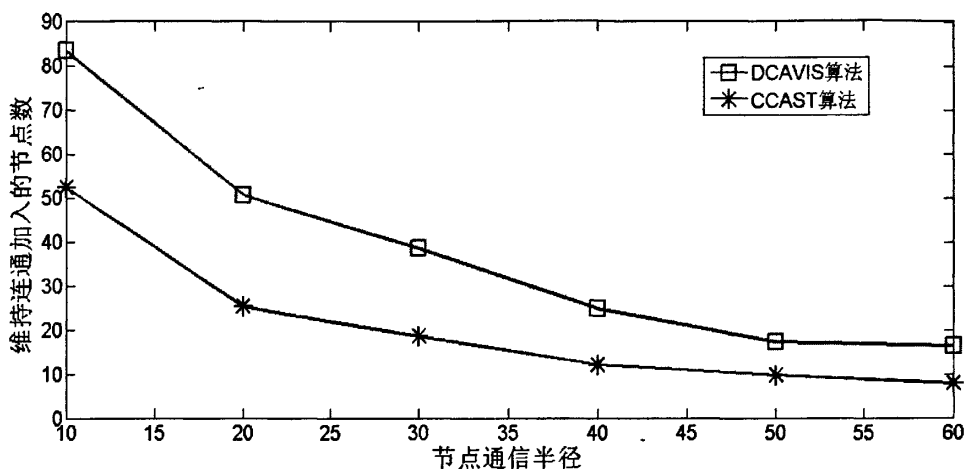


图 6.6 维持连通加入的节点数与节点通信半径的关系

6.4.2 网络生存时间

本节的仿真实验兼顾这两种情形，而只考虑一个活跃节点集承担覆盖任务的情况，定义当这个活跃子集中出现第一个节点失效所经历的时间长度为网络的生存时间。在仿真实验中，不考虑节点执行调度协议的能耗，并且假设忽略每个承担覆盖任务的节点执行监测任务的能耗。这样做的目的是，每次保持同一个节点集担任覆盖任务，而不考虑所采用的覆盖调度算法。当出现中继节点死亡而造成网络部连通时，重新加入新的节点构造新的连通集。周而复始，直到担任覆盖任务的节点中出现第一个节点失效，或者网络中的节点不足以维持网络连通为止。

图 6.7 给出了 DCAVIS 算法和 CCAST 算法中节点的通信半径与网络生存时间之间的关系。从图中可以看出，当节点通信半径从 20m 逐渐增加到 60m 时，若采用 CCAST 算法，网络生存时间从 585 增加到 1772。而采用执行 DCAVIS 算法，网络生存时间从 765 逐渐增加到 2332。

在 6.4.1 小节中已经看出，虽然 DCAVIS 算法比 CCAST 算法需要加入的节点数要多，但是，从图中可以看出，执行 DCAVIS 算法的网络生存时间反而比 CCAST 算法长。在执行 CCAST 算法时，每次当出现新加入的节点死亡时，Sink 节点必须重新收集各节点的信息，从而节点需要消耗大量的能量。因而出现执行 DCAVIS

算法的网络生存时间比 CCAST 算法反而长, 说明在传感器网络中采用分布式算法比集中式算法更利于延长网络的生存时间。

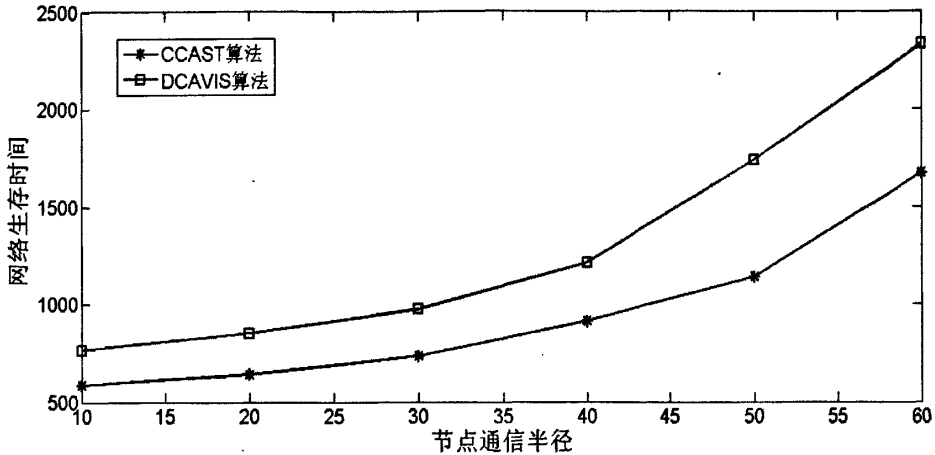


图 6.7 网络生存时间与节点通信半径的关系

6.5 小结

本章讨论了离散目标的无线传感器网络中的覆盖连通问题。首先提出了一种基于 Steiner 树的集中式近似算法 CCAST 算法, 该算法的基本思想是在覆盖集的基础上寻找 Steiner 点, 使得这些 Steiner 点连同覆盖集构成 Steiner 树, 以确保该覆盖集的连通性。但是, 求解最小 Steiner 树问题是 NP-难解的, 但是已有相当多的近似算法去求解近似最小 Steiner 树问题。为此, CCAST 算法将覆盖集中的连通簇虚拟化, 通过构造虚拟化后的加权图, 然后调用现有的近似算法求解近似最小 Steiner 树算法, 挑选出 Steiner 点, 用于计算确保该覆盖集连通所需的中继节点。

针对集中式算法的缺陷, 本章进一步又提出了一个分布式连通算法—DCAVIS 算法。在 DCAVIS 算法中, 将连同簇看成虚拟节点, 以此为基础构造虚拟独立集, DCAVIS 算法为解决点覆盖连通问题可提供了一种分布式近似求解算法。仿真实验表明, 虽然 CCAST 算法的精度比 DCAVIS 算法高, 但是, 对于大规模无线传感器网络, 集中式算法不利于延长网络的生存时间。

第七章 结 束 语

本章对全文进行总结，并对进一步的工作进行展望。

7.1 工作总结

本文研究的覆盖问题属于点覆盖与连通问题。在很多无线传感器网络应用中，如军事上，往往需要对某些离散的目标点进行监控。由于受到能量的约束，覆盖问题的主要任务是找出较小的覆盖集，并使这些集合能独立完成所需要的任务，而其它节点则处于低功耗的休眠状态。由于在点覆盖问题中，完全覆盖所有目标的覆盖集不一定是连通覆盖集，因而用尽可能少的节点构造的连通覆盖集是一个值得深入研究的问题。本文围绕这两个问题进行了深入分析和研究，本文主要对无线传感器网络的以下几个重要问题进行了深入研究。

1. 保持网络覆盖质量并降低冗余度的能量高效的节点调度问题

在随机部署的大规模无线传感器网络中，传感器节点是能量受限的无线设备，节点数量多，密度大，不可避免地存在大量冗余节点。必须对这些冗余节点的工作周期进行精心设计和调度，否则不利于延长无线传感器网络的生存时间。因而，如何合理利用大规模密集型无线传感器网络的特点，延长网络的生存时间非常关键。而且，在对节点工作周期进行调度时，必须保证关闭部分节点后仍能保持网络的覆盖要求。

针对上述问题，本文研究了保持网络覆盖质量并降低冗余度的能量高效的节点调度问题，提出了一种分布式无线传感器网络节点冗余覆盖调度机制 SRCA 算法。该算法通过检测网络中的覆盖冗余节点，并使用节点剩余能量做为优先级，防止互为邻居的冗余节点同时关闭而导致网络中出现覆盖盲点，并在关闭部分冗余节点后保持无线传感器网络的所要求的覆盖质量。SRCA 算法能够有效地减少冗余数据传输，节省系统能量。仿真实验表明，SRCA 算法降低网络的冗余覆盖度，显著地提高了网络的生存时间。

在第四章，我们又提出了一种基于遗传算法的集中式节点调度控制 NSAGA 算法，这种算法将无线传感器网络划分为若干个覆盖子集时，利用遗传算法每次迭代可能产生多个符合要求的种群这一特点，可有效减少算法的迭代次数。该算法首先考虑了一种无线传感器网络的应用场景，限制覆盖每个被监视的目标对象的传感器节点数的上界，保证其被覆盖的冗余程度不至于太高，从而将网络尽可能多地划分为若干覆盖子网，减少活跃节点的冗余度。同时，NSAGA 算法也能提供一种点覆盖问题的框架，不同应用的覆盖要求只要改变约束条件即可。

2. 集中式算法的局部化

目前,点覆盖问题研究一般该问题转化为数学规划模型,然后采用近似最优的方法求解。这种数学规划方式只能用集中式算法才能求解,不适合大规模无线传感器网络应用。但是,这种求解方式比分布式算法的精度高,而且通过修改约束条件还可适合不同的应用场景,因而,这些算法也具有较高的应用价值。

在本文的第五章,提出了一种将集中式优化覆盖调度算法局部化的 LCACA 算法。这种算法仅依赖每个节点的 m -hop 邻居信息,通过选举局部中心节点,采用节点自愿加入一个局部中心节点的原则,将整个网络划分成多个规模较小的网络,然后在每个划分的网络中再运行集中式算法,完成传感器节点的调度。而且, LCACA 算法不要求节点的时钟同步,适合应用在随机部署的大规模无线传感器网络中。LCACA 算法可使无线传感器网络根据不同应用场景,在同一个网络上既可以运行集中式算法,也可以运行分布式算法,或者二者兼而有之,为点覆盖问题的集中式算法提供局部化的基础。

3. 点覆盖节点的连通

由于点覆盖与被监测的目标分布情况有关,因此,考虑覆盖节点的连通行非常必要。针对点覆盖问题的这一特性,本文在第六章提出了一种基于 Steiner 树集中式连通算法 CCAST 算法。CCAST 算法首先将所有连通簇看成一个虚拟节点,然后提出了一种将连通簇转化为虚拟节点的算法,在此基础上构建加权通信图,然后调用已有构造 Steiner 树的算法,挑选出 Steiner 点,使得所有的覆盖节点保持连通。

接着,我们又进一步提出了一种分布式连通算法—DCAVIS 算法。DCAVIS 算法将连通簇看成虚拟节点,然后构造虚拟独立集,再进一步寻找使所有覆盖节点能保持连通的中继节点。DCAVIS 算法为解决点覆盖连通问题可提供了一种分布式近似求解算法。

7.2 课题研究展望

覆盖问题是无线传感器网络的基本问题,已经受到越来越多研究者的关注。本文围绕无线传感器网络的点覆盖问题开展研究。在本文研究的基础上,需要进一步研究的课题包括:

1. 异构无线传感器网络的点覆盖

在设计覆盖算法时,当前的很多研究是在同构网络下进行的,同构网络的所有节点具有相同的感知半径、相同的能量以及相同的通信半径。在无线传感器网络的实际应用中,可能存在以下几种情况: 1. 存在多种类型的传感器节点。由于被监测的物理世界存在多种类型的物理量,很多事件的发生也必须综合不同的物理量才能确定,在这种情况下无线传感器网络应用,必须在被监测的区域内部署多种类

型的传感器节点。2. 存在网络中的节点功能相异, 如有些节点的电池容量大, 数据处理能力比较强, 感知和通信的范围比较大等。事实上, 即使初始网络绝对同构, 当网络运营后, 由于节点能耗的不平衡性, 就会打破网络的同构性, 因此, 同构网络只是一种理想网络。此外, 节点剩余能量也制约节点的通信半径和感知半径, 还有传感器节点的通信半径和感知半径也可能是可调的, 这种无线传感器网络的节点可能根据剩余能量来调整自己的感知半径或通信半径。总之, 异构网络才接近实际网络的特性, 因为, 所有的覆盖算法需要面向异构网络去设计, 在这种前提下考虑如何合理调度节点的工作周期, 将更利于降低整个网络的能耗, 延长网络生存的时间。

2. 概率感知模型下点覆盖

本文点覆盖问题的讨论采用布尔感知模型描述传感器节点的感知能力。与大部分当前研究工作类似, 考虑到被监测信号衰减以及环境噪声等因素影响, 势必造成传感器节点检测到的监测目标的信号在一定程度上表现出不确定性, 而使用概率感知模型则能描述这种不确定性因素, 比如, 采用模型 $P_r(d) = P_r(d_0) - 10\beta \log(d/d_0) + X_\sigma$, 其中 X_σ (单位为 dB) 为满足 $(0, \sigma^2)$ 的高斯分布, $P_r(d_0)$ 则是一定应用下节点对距离为 d_0 目标点的检测概率, 可以用常数来描述。在概率感知模型下网络的覆盖质量描述, 以及相应的节点进行调度算法的设计还需要深入研究。

3. 无需要位置信息的点覆盖

大规模无线传感器网络的定位问题目前仍然是一个比较困难的问题。如果能设计不依赖节点的任何位置信息的调度机制则很具有现实性。如何在不依赖节点位置信息的条件下, 设计能够同时保证网络覆盖质量和网络连通性的能量高效节点调度机制, 这也是我们下一步工作要考虑的问题。

此外, 当前的研究工作很少考虑到由于覆盖而导致的节点状态变化, 这种变化无疑会影响到相关的路由协议。因此, 如何协调点覆盖与网络路由协议的关系也是值得研究。

参考文献

- [1] Iyer R, Kleinrock L. QoS control for sensor communication network protocols in deep space[C]. In Proc.of the IEEE International Conference on Communications, 2003, 1: 517-521.
- [2] Bettstetter C, Hartmann C. Connectivity of wireless multihop networks in a shadow fading environment [C]. The 8th International Workshop on Modeling Analysis and Simulation of Wireless and Mobile Systems. San Diego: MSWiM, 2003:28-32.
- [3] M. Roussopoulos, M. Baker. Controlled Update Propagation in Peer-to-Peer Networks. Proceedings of USENIX Annual Technical Conference. 2003.
- [4] S.H.L.Liang, V.Tao, A.Croitoru. The Design and Prototype of a Distributed Geospatial Infrastructure for Smart Sensor Web. Proceedings of the 6th AGILE Conference on Geographic Information Science.2003
- [5] A. Cerpa, J. Elson, M. Hamilton, and 1. Zhao, Habitat Monitoring: Application Driver for Wireless Communications Technology. First ACM SIGCOMM Workshop on Data Communication in Latin America and the Caribbean, April 2001.
- [6] J.R. Polastre, D. Culler. Design and Implementation of Wireless Sensor Networks for Habitat Monitoring, University of California at Berkeley, 2003.
- [7] P.Bonnet, J.Gehrke, P.Seshadri. Querying the Physical World. IEEE Personal Communications, 2000, 7.
- [8] Rabaey J M, Ammer M J, da Silva J L Jr. Patel D, Roundy S. PicoRadio supports ad hoc ultra-low power wireless networking[J]. IEEE Computer Magazine 2000:42-48
- [9] D. Estrin, R. Govindan, J. Heidemann, et al.. Next century challenges:scalable coordination in sensor networks.ACM MOBICOM, Seattle, WA, August 1999.
- [10] N. Bulusu, D. Estrin, L. Girod, J. Heidemann, Scalable coordination for wireless sensor networks: self-configuring localization systems, International Symposium on Communication Theory and Applications (ISCTA 2001), Ambleside, UK, July 2001.
- [11] P. Bauer, M. Sichertiu, R. Istepanian and K. Premaratne. The Mobile Patient: Wireless Distributed Sensor Networks for patient monitoring and Care. In Proc. IEEE EMBS International Conference on Information Technology Applications in Biomedicine, Arlington, pp.17-21, 2000.
- [12] <http://www.agingtech.org>.
- [13] <http://www.rfidjournal.com/article/view/3625/>.
- [14] I.F.Akyildiz,W.Su, Y.Sankarasubramaniam, E.Cayirci. A Survey on Sensor Networks. IEEE Communications Magazine, 2002,40(8):102-114.
- [15] Conrad J G.CEC:Sensor Netting with Integrated Fire Control[J].Johns Hopkins APL Technical Digest,2002,23(2):2-3.

-
-
- [16] http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/MICA.pdf
- [17] <http://bwrc.eecs.berkeley.edu>
- [18] UCLA. Medusa MK-2 Programming Resource [EB/OL]. <http://nesl.ee.ucla.edu/projects/ahlos/mk2/>, 2003-10.
- [19] <http://www.intel.com/research/exploratory/motes.htm>.
- [20] UC Berkeley. TinyOS: An Open-source Operating System Designed for Wireless Embedded Sensor Networks [EB/OL]. <http://www.tinyos.net/>, 2003-10.
- [21] <http://mantis.cs.colorado.edu/index.php/tiki-index.php>.
- [22] <http://nesl.ee.ucla.edu/projects/sos/>.
- [23] <http://www.cast.cse.ohio-state.edu/exscal/>.
- [24] <http://projects.cerias.urdue.edu/esp/>.
- [25] <http://www.eng.yale.edu/enalab>
- [26] <http://www.eecs.harvard.edu/~mdw/proj/codeblue/>.
- [27] <http://wings.cs.sunysb.edu>.
- [28] <http://www.robotics.usc.edu/embedded/>.
- [29] <http://www.isi.edu/scadds/>.
- [30] <http://www.ece.gatech.edu/researchAabs/bwn/index.html>.
- [31] <http://www.zurich.ibm.com/sys/communication/sensors.html>.
- [32] http://www.intel.com/research/exploratory/wireless_sensors.htm.
- [33] <http://research.microsoft.com/nec/>.
- [34] P.Berman,G.Calinescu, C.Shah and A.Zelikovsky.Power Efficient Monitoring Management in Sensor Networks,Proceedings of WCNC'04,March 2004.
- [35] S. Slijepcevic and M. Potkonjak. Power Efficient Organization of Wireless Sensor Networks, IEEE International Conference on Communications, June 2001.
- [36] J. Jiang and W. Dou. A Coverage-Preserving Density Control Algorithm for Wireless Sensor Networks. In International Conference on AD-HOC Networks & Wireless (ADHOC-NOW), pp.42-55, July 2004.
- [37] Dhillon S S,Chakrabarty K,Iyengar S S.Sensor placement for grid coverage under imprecise detections[C].Proceeding of International Conference on Information Fusion,2002:158 -1587.
- [38] Dhillon S S,Chakrabarty K.Sensor Placement for Effective Coverage and Surveillance in Distributed Sensor Networks[C] .Proceeding of IEEE WCNC 2003.New Orleans,Louisiana,USA,2003:1609-1614.
- [39] Zou Y, Chakrabarty K. Sensor deployment and target localization in distributed sensor networks. ACM Trans. on Embedded Computing Systems, 2004,3(1):61-91.
- [40] J. Lu, et al.. Coverage-aware Self-scheduling in Sensor Networks. In Proc. of CCW 2003, Oct. 2003, pp. 117-123.
- [41] Nadeem Ahmed, Salil S. Kanhere, Sanjay Jha. Probabilistic Coverage in Wireless Sensor Networks. Fifth International IEEE Workshop on Wireless Local

Networks(WLN 2005), pp. 672-681,2005.

[42] T. S. Rappaport, *Wireless communications,Principles and Practice*, Prentice Hall, New Jersey, 1996.

[43] H.Ma,Y.Liu.On coverage problems of directional sensor networks.Proceedings of the International Conference on Mobile Ad-Hoc and Sensor Networks.2005.

[44] Slijepcevic S, Potkonjak M. Power efficient organization of wireless sensor networks. In: Glisic S, ed. Proc. of the IEEE Int'l Conf. on Communications (ICC). Helsinki: IEEE Press, 2001, 472-476.

[45] P. Berman, G. Calinescu,C.Shah, et.al.. Power Efficient Monitoring Management in wireless sensor networks. Proceedings of IEEE Wireless Communication and Networking Conference(WCNC'04), Allanta, USA, 2004.

[46] Abrams Z, Goel A, Plotkin S. Set k-cover algorithms for energy efficient monitoring in wireless sensor networks. In: Ramchandran K, Sztipanovits J, eds. Proc. of the 3rd Int'l Conf. on Information Processing in Sensor Networks. Berkeley: ACM Press, 2004. 424-432.

[47] F. Ye, G. Zhong, S. Lu, and L. Zhang, Energy. Efficient Robust Sensing Coverage in Large Sensor Networks. Technical Report UCLA (2002).

[48] Ye F, Zhong G, Lu S, Zhang L. PEAS: A robust energy conserving protocol for long-lived sensor networks. In: Stankovic J, Zhao W, eds. Proc. of the Int'l Conf. on Distributed Computing Systems (ICDCS). Providence: IEEE Press, 2003. 28-37.

[49] Tian D, Georganas N. A coverage-preserving node scheduling scheme for large wireless sensor networks. In: Raghavendra CS, Sivalingam K, eds. Proc. of the 1st Int'l Workshop on Wireless Sensor Networks and Applications (WSNA 2002). Atlanta: ACM Press, 2002, 32-41.

[50] Tian, D., and Georganas, N. D. A node scheduling scheme for energy conservation in large sensor networks. *Journal of Wireless Communications and Mobile Computing* 3 (2003), 271{290. Special issue on Algorithmic, Geometric, Graph, Combinatorial, and Vector Aspects of Wireless Networks and Mobile Computing.

[51] Yah T, He T, Stankovic J. Differentiated surveillance service for sensor networks. In:Akyildiz IF,Estion D,eds.Proc.of the 1st Int'l Conf.on Embedded Networked Sensor Systems.Los Angels:ACM Press,2003.51-63.

[52] Howard A,Mataric M J, Sukhatme G S. Mobile Sensor Network Deployment using Potential Fields: a distributed, Scalable Solution to the Area Coverage Problem[C]. In: Proceeding of International Conference on Distributed Autonomous Robotic Systems,2002:299-308

[53] Zou Y, Chakrabarty K.Sensor Deployment and Target Localization based on Virtual Forces[C]. In: Proceeding of IEEE INFOCOM, 2003:1293-1303

[54] Heo N, Varshney P K. An Intelligent Deployment and Clustering Algorithm for a Distributed Mobile Sensor Network[C]. In Proceeding of IEEE International Conference

on Systems, Man and Cybernetics, 2003: 4576- 4581

[55] G.Wang, G.Cao, T.La Porta. A Bidding Protocol for Deploying Mobile Sensors. Proceedings of the 11th IEEE International Conference on Network Protocols(ICNP'03),2003

[56] G.Wang, G.Cao, T.La Porta, Movement-Assisted Sensor Deployment, Proceedings of IEEE INFOCOM'04, HongKong, 2004

[57] G.Wang, G.Cao, T.La Porta, Proxy-based Sensor Deployment for Mobile Sensor Networks, Proceedings of the 1st IEEE International Conference on Mobile Ad-hoc and Sensor Systems(MASS'04),Fort Lauderdale, Florida, 2004

[58] G.Wang, G.Cao, T.Porta, W.Zhang. Sensor Relocation in Mobile Sensor Networks. Proceedings of IEEE INFOCOM'05, Miami, Florida,2005

[59] J.Wang, D.Du, E.Kusmierek. Energy Efficient Oragnization of Mobile Sensor Networks, Proceedings of the 2004 International Conference on Parallel Processing Workshops(ICPPW'04), Montreal,Quebec,Canada,2004

[60] Basagni S. Distributed Clustering for Adoc hoc Networks[C]. In: Proceeding of IEEE International symposium on Parallel Architectures, Algorithms, and Networks, 1999: 310-315

[61] B.Liu,P.Brass,O.Dousse,P.Nain,D.Towsley.Mobility Improves Coverage of Sensor Network,Proceedings of ACM Mobihoc'05, Urbana-Champaign,IL,2005

[62] Meguerdichian S, Koushanfar F, Potkonjak M, Srivastava MB. Coverage problems in wireless ad-hoc sensor network. In: Sengupta B, ed. Proc. of the IEEE INFOCOM. Anchorage: IEEE Press, 2001. 1380-1387.

[63] Xiang-Yang Li, Peng-Jun Wan, Ophir Frieder. Coverage in Wireless Ad-Hoc Sensor Networks. Proc. IEEE Trans. Computers, 2003, vol.52(6): 753-763.

[64] Meguerdichian S, Koushanfar F, Qu G, Potkonjak M. Exposure in wireless ad-hoc sensor networks. In: Rose C, ed. Proc. of the ACM Int'l Conf. on Mobile Computing and Networking (MobiCom). New York: ACM Press, 2001. 139-150.

[65] Cheng M X,Ruan L,Wu W.Achieving Minimum Coverage Breach Under Bandwidth Constraints in Wireless Sensor Networks[C].Proc of the 24th Conf of the IEEE Communications Society, 2005.

[66] Adlakha S, Srivastava M. Critical density thresholds for coverage in wireless sensor networks. In: Tachikawa K, ed. Proc. of the IEEE Wireless Communications and Networking (WCNC). New Orleans: IEEE Press, 2003. 1615-1620.

[67] B. Liu, D. Towaley. On the Coverage and Detectability of Large-scale. Wireless Sensor Networks. Proc. of the Modeling and Optimization in Mobile, Ad hoc and Wireless Networks, INRIA Sophia-Antipolis, France, 2003

[68] R. Meester and R. Roy. Continuum percolation, Cambridge University Press, Cambridge, 1996. 21.

- [69] Liu B, Towsley D. A study of the coverage of large-scale sensor networks. In: Proc. of the ACM Mobile Ad-Hoc and Sensor Systems (MASS'04). New York: ACM Press, 2004. 475-483
- [70] Cardei M, Du DZ. Improving wireless sensor network lifetime through power aware organization. *Wireless Networks*, 2005, 11(3): 333-340.
- [71] Cardei M, Thai MT, Li Y. Energy-Efficient target coverage in wireless sensor networks. In: Raton B, ed. Proc. of the IEEE INFOCOM. Miami: IEEE Computer Society, 2005. 1976-1984.
- [72] Chang JH, Tassiulas L. Maximum lifetime routing in wireless sensor networks. *IEEE/ACM Trans. on Networking*, 2004, 12(4): 609-619.
- [73] M. Lu, J. Wu, M. Cardei and M. Li. Energy-efficient connected coverage of discrete targets in wireless sensor networks. *IEEE International Conference on Volume* , Issue , 19-21 June 2006 Page(s): 94 – 101
- [74] Kar K, Banerjee S. Node placement for connected coverage in sensor networks. In: Crowcroft J, ed. Proc. of the Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks. Sophia-Antipolis: IEEE Press, 2003. 50-52.
- [75] S. Slijepcevic and M. Potkonjak. Power efficient organization of wireless sensor networks. In: IEEE International Conference on Communications, Helsinki, Finland, June 2001.
- [76] D. Tian and N. D. Georganas. A coverage-preserving node scheduling scheme for large wireless sensor network, in Proceeding of 1st ACM Workshop on Wireless Sensor Networks and Applications (WSNA 02), Atlanta, Georgia, 2002.
- [77] Ye F, Zhong G, Lu S, Zhang L. Gradient broadcast: A Robust Data Delivery Protocol for Large Scale Sensor Networks. *ACM Wireless Networks*, 2005, 11(3):285-298.
- [78] M. Cardei, J. Wu, M. Lu, and M. O. Pervaiz. Maximum network lifetime in wireless sensor networks with adjustable sensing ranges. Proc. of IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob) (2005)
- [79] F. Dai and J. Wu. On constructing k-connected k-dominating set in wireless networks. Proc. of IPDPS (2005)
- [80] Abrams Z, Goel A, Plotkin S. Set k-cover algorithms for energy efficient monitoring in wireless sensor networks. In: Ramchandran K, Sztipanovits J, eds. Proc. of the 3rd Int'l Conf. on Information Processing in Sensor Networks. Berkeley: ACM Press(2004)424-432
- [81] K. Chakrabarty, S. S. Iyengar, H. Qi, and E. Cho. Grid coverage for surveillance and target location in distributed sensor networks. *IEEE Trans. on Comput.*, 51(12)(2002) : 1448–1453
- [82] M. Cardei, D.Z. Du. Improving wireless sensor network lifetime through power

- aware organization. *ACM Wireless Networks* (2005), 11 (3)
- [83] Cardei M, Thai MT, Li Y, Wu W. Energy-efficient target coverage in wireless sensor networks. In: *Proceedings of 24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, Vol. 3 (2005) 1976–1984
- [84] Steuer, R. E. *Multiple criteria optimization: Theory, computation, and application*. New York : Wiley(1986)
- [85] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan: A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II. In: *Proceedings of Parallel Problem Solving from Nature -PPSN VI*. Springer (2000) 849–858
- [86] Kalyanmoy Deb and R. B. Agarwal. Simulated Binary Crossover for Continuous Search Space. *Complex Systems* (1995) (9)115–148
- [87] Bulusu N, Heidemann J, Estrin D. GPS-Less low cost outdoor localization for very small devices. *IEEE Personal Communications Magazine*, 2000.7(5): 28–34.
- [88] He H, Huang C, Blum BM, Stankovic JA, Abdelzaher TF. Range-Free localization schemes in large scale sensor networks. In: Johnson DB, ed. *Proc. of the ACM MobiCom 2003*. San Diego: ACM Press, 2003. 81–95.
- [89] Romer K, Zurich E. The lighthouse location system for smart dust. In: Siewiorek D, ed. *Proc. of the 1st Int'l Conf. on Mobile Systems, Applications, and Services*. San Francisco: ACM Press, 2004. 15–30.
- [90] S. Singh and C. Raghavendra. PAMAS: Power Aware Multi-Access protocol with Signalling for Ad Hoc Networks. *ACM Computer Communication Review*, vol. 28, no. 3, pp. 5–26, July 1998.
- [91] K. Sohrabi, J. Gao, V. Ailawadhi, and G. J. Pottie. Protocols for Self-Organization of a Wireless Sensor Network. *IEEE Personal Comm. Mag.*, vol. 7, no. 5, Oct. 2000.
- [92] Z. Haas and S. Tabrizi. On Some Challenges and Design Choices in Ad-Hoc Communications. In: *IEEE MILCOM'98*, October 1998.
- [93] W. Ye, J. Heidenmann, and D. Estrin. An Energy-Efficient MAC Protocol for Wireless Sensor Networks. In: *Proceedings of IEEE INFOCOM*, New York, NY, June 2002.
- [94] V. Kawadia and P. R. Kumar. Power Control and Clustering in Ad Hoc Networks. In: *Proceedings of IEEE INFOCOM*, April 2003.
- [95] S. Narayanaswamy, V. Kawadia, R. S. Sreenivas, and P. R. Kumar. Power Control in Ad-Hoc Networks: Theory, Architecture, Algorithm and Implementation of the COMPOW protocol. In: *Proceedings of European Wireless 2002. Next Generation Wireless Networks: Technologies, Protocols, Services and Applications*, February 2002, pp. 156–162.
- [96] C. Intanagonwiwat, R. Govindan, and D. Estrin. Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks. In: *Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking*

(MOBICOM), 2000.

[97] J. Kulik, W. R. Heinzelman, and H. Balakrishnan. Negotiation-Based Protocols for Disseminating Information in Wireless Sensor Networks. *ACM Wireless Networks*, vol. 8, no. 2-3, pp. 169–185, 2002.

[98] J.-H. Chang and L. Tassiulas. Energy conserving routing in wireless ad-hoc networks. In: *Proceedings of IEEE INFOCOM*, Tel Aviv, Israel, Mar. 2000.

[99] W.R. Heinzelman, A. Chandrakasan, and H. Balakrishnan. An Application-Specific Protocol Architecture for Wireless Microsensor Networks. *IEEE Transactions on Wireless Communications*, vol. 1, no. 4, pp. 660–670, October 2002.

[100] A. Cerpa and D. Estrin. ASCENT: Adaptive Self-Configuring Sensor Networks Topologies. In: *Proceedings of IEEE INFOCOM*, New York, NY, June 2002.

[101] Y. Xu, J. Heidemann, and D. Estrin. Geography-Informed Energy Conservation for Ad Hoc Routing. In: *Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM)*, Rome, Italy, July 2001, pp. 70–84.

[102] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris. Span: an Energy-Efficient Coordination Algorithm for Topology Maintenance in Ad Hoc Wireless Networks. *ACM Wireless Networks*, vol. 8, no. 5, September 2002.

[103] J. Wang, Ning Zhong. Efficient point coverage in wireless sensor networks[J]. *Journal of Combinatorial Optimization*, 2006, Volume 11, Number 3, 291-304.

[104] Cardei M, Thai MT, Li Y. Energy-Efficient target coverage in wireless sensor networks. In: Raton B, ed. *Proc. of the IEEE INFOCOM*. Miami: IEEE Computer Society, 2005. 1976-1984.

[105] Cheng M X, Ruan L, Wu W. Achieving Minimum Coverage Breach Under Bandwidth Constraints in Wireless Sensor Networks[C]. *Proc of the 24th Conf of the IEEE Communications Society*, 2005.

[106] Chen Wang, My T. Thai, et al. Minimum Coverage Breach and Maximum Network Lifetime in Wireless Sensor Networks, *IEEE GLOBECOM 2007*, Washington, DC, November 26-30, 2007.

[107] Deng J, Han YS, Heinzelman WB, Varshney PK. Scheduling sleeping nodes in high density networks. *ACM/Kluwer Mobile Networks and Applications (MONET)*, 2005, 10(6): 825-835.

[108] W. Heinzelman, A. Chandrakasan and H. Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In: *Proc. of the 33rd Annual Hawaii International Conference on System Sciences (HICSS)*, Maui, HI, Jan. 2000, pp. 3005-3014.

[109] W. Heinzelman, A. Chandrakasan and H. Balakrishnan. An application-specific protocol architecture for wireless microsensor networks [J]. *IEEE Transactions on Wireless Communications*, 2002, 1(4):660-670.

-
- [110] Younis O, Fahmy S. HEED: A hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. *IEEE Trans. on Mobile Computing*, 2004, 3(4): 660-669.
- [111] Gupta I, Riordan D, Sampalli S. Cluster-Head election using fuzzy logic for wireless sensor networks. In: *Proc. of the 3rd Annual Communication Networks and Services Research Conf.* Halifax: IEEE Computer Society, 2005. 255-260.
- [112] Manjeshwar A, Grawal DP. TEEN: A protocol for enhanced efficiency in wireless sensor networks. In: *Proc. of the 15th Parallel and Distributed Processing Symp.* San Francisco: IEEE Computer Society, 2001. 2009-2015.
- [113] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In: *Proceedings of the 33rd Hawaii International Conference on System Sciences*, 2000, pp. 1-10.
- [114] Kubisch M, Karl H, Wolisz A, Zhong LC, Rabaey J. Distributed algorithms for transmission power control in wireless sensor networks. In: Taga T, ed. *Proc. of the IEEE Wireless Communications and Networking Conf. (WCNC)*. New Orleans: IEEE Press, 2003. 558-563.
- [115] Qayyum A, Viennot L, Laouiti L. Multipoint relaying: An efficient technique for flooding in mobile wireless networks. Technical Report, 3898, INRIA-Rapport de Recherche, 2000.
- [116] Wan PJ, Alzoubi KM, Frieder O. Distributed construction of connected dominating set in wireless ad hoc networks. In: Kermani P, ed. *Proc. of the IEEE Infocom*. New York: IEEE Press, 2002. 1597-1604.
- [117] Das B, Sivakumar R, Bhargavan V. Routing in ad hoc networks using a spine. In: Makki K, ed. *Proc. of the Int'l Conf. on Computer Communications and Networks (ICCCN)*. Los Alamitos: IEEE Press, 1997. 34-41.
- [118] Alzoubi KM, Wan PJ, Frieder O. Message-Optimal connected dominating sets in mobile ad hoc networks. In: Hubaux JP, ed. *Proc. of the ACM Int'l Symp. on Mobile Ad Hoc Networking and Computing (MobiHoc)*. New York: ACM Press, 2002. 157-164.
- [119] Chen Y, Liestman A. Approximating minimum size weakly-connected dominating sets for clustering mobile ad hoc networks. In: Hubaux JP, ed. *Proc. of the ACM Int'l Symp. on Mobile Ad Hoc Networking and Computing (MobiHoc)*. New York: ACM Press, 2002. 165-172.
- [120] Wu J, Dai F. Broadcasting in ad hoc networks based on self-pruning. In: Bauer F, ed. *Proc. of the IEEE Infocom*. San Francisco: IEEE Press, 2003. 2240-2250.
- [121] Dai F, Wu J. An extended localized algorithm for connected dominating set formation in ad hoc wireless networks. *IEEE Trans. on Parallel and Distributed Systems*, 2004, 15(10):908-920.
- [122] Zhang H, Hou JC. Maintaining sensing coverage and connectivity in large sensor networks. *Ad Hoc & Wireless Networks*, 2005, 1(1): 89-124.
-

- [123] Wang X, Xing G, Zhang Y, Lu C, Pless R, Gill C. Integrated coverage and connectivity configuration in wireless sensor networks. In: Akyildiz IF, Estion D, eds. Proc. of the ACM Int'l Conf. on Embedded Networked Sensor Systems (SenSys). New York: ACM Press, 2003. 28–39.
- [124] Chen B, Jamieson K, Balakrishnan H, Morris R. SPAN: An energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks. In: Rose C, ed. Proc. of the 7th ACM Int'l Conf. on Mobile Computing and Networking (MobiCom). New York: ACM Press, 2001. 85–96.
- [125] Carle J, Gallais A, Simplot-Ryl D. Preserving area coverage in wireless sensor networks by using surface coverage relay dominating sets. In: Ammar R, Saracco R, eds. Proc. of the 10th IEEE Symp. on Computers and Communications (ISCC). Cartagena: IEEE Press, 2005. 347–352.
- [126] Gupta H, Das SR, Gu Q. Connected sensor cover: Self-Organization of sensor networks for efficient query execution. In: Gerla M, ed. Proc. of the ACM Int'l Symp. on Mobile Ad Hoc Networking and Computing (MobiHoc). New York: ACM Press, 2003. 189–200.
- [127] Zhou Z, Das SR, Gupta H. Fault tolerant connected sensor cover with variable sensing and transmission ranges. In: Garcia-Luna-Aceves J, ed. Proc. of the 2nd Annual IEEE Communications Society Conf. on Sensor and Ad Hoc Communications and Networks (SECON). Santa Clara: IEEE Press, 2005. 594–604.
- [128] Narayanswamy S, Kawadia V, Sreenivas RS, Kumar PR. Power control in ad-hoc networks: Theory, architecture, algorithm and implementation of the COMPOW protocol. In: Lenzini L, ed. Proc. of the European Wireless Conf.—Next Generation Wireless Networks: Technologies, Protocols, Services and Applications. Pisa: IEEE Press, 2002. 156–162.
- [129] Ramanathan R, Rosales-Hain R. Topology control of multihop wireless networks using transmit power adjustments. In: Sidi M, ed. Proc. of the IEEE INFOCOM. Tel Aviv: IEEE Press, 2000. 404–413.
- [130] Li N, Hou JC. Topology control in heterogeneous wireless networks: Problems and solutions. In: Li VOK, ed. Proc. of the IEEE INFOCOM. Hong Kong: IEEE Press, 2004. 232–243.
- [131] Karp RM. Reducibility among combinatorial problems. In: Miller RE, Thatcher JW, eds. Proc. of the Complexity of Computer Computations. New York: Plenum Press, 1972. 85–103.
- [132] Garey MR, Graham RL, Johnson DS. The complexity of computing Steiner minimal trees. *SIAM Journal on Applied Mathematics*, 1977, 32(4):835–859.
- [133] Garey M R and Johnson D S. *Computers and intractability: A guide to* New York: Freeman, 1979.
- [134] Boyce W M. An improved program for the full Steiner tree problem. *ACM*

Transactions on Mathematical Software, 1997, 3: 359-385.

[135] Gilbert E N and Pollak H O. Steiner minimal trees. SIAM Journal on Applied Mathematics, 1996, 16: 1-29.

[136] P. Berman and V. Ramaiyer. Improved approximations for the Steiner tree problem. In: Proceedings of the Third Symposium on Discrete Algorithms, pages 325-334, 1992.

[137] P. Berman and V. Ramaiyer. Improved approximations for the Steiner tree problem. Journal of Algorithms, 17: 381-408, 1994.

[138] S. Ganeriwal, R. Kumar, and M. Srivastava. Timing-Sync Protocol for Sensor Networks. In: Proceedings of the 1st International Conference on Embedded Networked Sensor Systems, pp. 138-149, November 2003.

[139] M. Maróti, B. Kusy, G. Simon, and Á. Lédeczi. The Flooding Time Synchronization Protocol. In: Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems, pp. 39-49, November 2004.]

[140] J. V. Greunen and J. Rabaey. Lightweight Time Synchronization for Sensor Networks. In: Proceedings of the 2nd ACM International Conference on Wireless Sensor Networks and Applications, pp. 11-19, September 2003.

[141] Q. Li and D. Rus, Global Clock Synchronization in Sensor Networks, INFOCOM'04, March 2004.

[142] G. Robins and A. Zelikovsky. Improved Steiner tree approximation in graphs. In Proceedings of the 11th Annual ACM-SIAM Symposium on Discrete Algorithms, pages 770-779, 2000.

攻读博士学位期间撰写的主要论文

- [1] Dingxing Zhang, Ming Xu, Shulin Wang, Boyun Zhang. Minimization of the redundant coverage for dense wireless sensor networks. The 2007 International Conference on Embedded Software and Systems (ICESS-07), Daegu, South Korea, Lecture Notes in Computer Sciences (LNCS), vol.4523 pp.751-760, 2007 (EI 检索号: 080611083062).
- [2] Dingxing Zhang, Ming Xu, Yingwen Chen, Shulin Wang. Probabilistic coverage configuration for wireless sensor networks. The 2006 International Conference on Wireless Communications, Networking and Mobile Computing(WiCOM 2006)(EI 检索号: 072510660525)
- [3]Dingxing Zhang, Ming Xu, Wei Xiao, Junwen Gao, Wenshen Tang. Minimization of the redundant sensor nodes in dense wireless sensor networks. The 7th International Conference on Evolvable Systems: From Biology to Hardware(ICES 2007), pp. 355-367, 2007(LNCS) (EI 检索号: 080311024487) .
- [4] 张鼎兴, 张波云, 伍春华等. 基于Web的IP网络管理实现. 计算机工程, 第31卷, 第11期,pp.108-110,2005.6. (EI检索号: 05269184962)
- [5] 张鼎兴,徐明,刘爱心. 一种节能多属性目标监测的感知网络覆盖算法. 计算机工程与应用, 第43卷,第18期,pp.7-10,2007.6.
- [6] 张鼎兴,徐明,高俊文,刘爱心.一种多属性目标监测的无线感知网络覆盖算法.计算机工程与科学,第30卷,第4期,pp.98-101,2008.
- [7] 张鼎兴,张波云,唐文胜,王树林.一种网元管理系统的设计 湛江海洋大学学报 第24卷,第3期,pp.56-60,2004.6.
- [8] 张鼎兴,徐明,唐文胜.无线传感器网络节点自调度冗余覆盖算法.传感器与微系统(已录用,待发表)
- [9] 张鼎兴,张波云,唐文胜. 无线传感器网络分布式连通算法.计算机工程与应用(已录用,待发表)
- [10]Shulin Wang, Huowang Chen, Shutao Li and Dingxing Zhang.Feature extraction from tumor gene expression profiles using DCT and DFT.The 13th Portuguese Conference on Artificial Intelligence, Portugal, Lecture Notes in Computer Sciences (LNCS), vol. 4874 (EI 检索号: 080411056819).
- [11] Shulin Wang, Huowang Chen, Ji Wang, Dingxing Zhang, Shutao Li. Molecular diagnosis of tumor based on independent component analysis and support vector machines. Proceedings of the 2006 International Conference on Computational Intelligence and Security (CIS2006), Lecture Notes in Artificial Intelligence (LNAI), Guangzhou, vol. 4456 (EI 检索号: 080511069912).
- [12] Boyun Zhang, Jianping Yin, Jingbo Hao, Dingxing Zhang, and Shulin Wang. Malicious codes detection based on ensemble learning. The 4th International Conference

on Autonomic and Trusted Computing (ATC-07), Lecture Notes in Computer Science (LNCS), Springer-Verlag, Hongkong, vol.4610 (EI 检索号: 080311022214).

[13] Boyun Zhang ,Jianping Yin, Jingbo Hao, Shulin Wang, Dingxing Zhang .New malicious code detection based on N-gram analysis and rough set theory. Computational Intelligence and Security International Conference(CIS 2006) (LNCS) (EI 检索号: 080411055413)

[14] 张波云,殷建平,蒿敬波,张鼎兴. 基于多重朴素贝叶斯算法的未知病毒检测. 计算机工程, 2006, (10): 18-20 (EI 检索号: 06249938873)

[15] 王树林, 王戟, 陈火旺, 张鼎兴.k-长DNA子序列频数分布研究. 生物物理学报, 第22卷,第3期, 2006.

[16] Shulin Wang, Huowang Chen, Faren Li, Dingxing Zhang. Gene selection with rough sets for the molecular diagnosing of tumor based on support vector machines. International Computer Symposium 2006, Taiwan.

[17] 张波云, 殷建平, 张鼎兴, 蒿敬波. 基于K-最近邻算法的未知病毒检测. 计算机工程与应用, 第6期, 2005.

[18] 王树林, 王戟, 陈火旺, 张鼎兴.k-长DNA子序列计数算法研究. 计算机工程, 第33卷,第9期, 2007.

[19] 张波云, 殷建平, 张鼎兴, 蒿敬波. 基于支持向量机的计算机病毒检测系统. 计算机工程与科学. 第7期, 2007.

[20] 张波云,殷建平,张鼎兴, 蒿敬波等. 基于集成神经网络的计算机病毒检测方法. 计算机工程与应用, 第43卷, 第13期, 2007.

致 谢

来到国防科学技术大学计算机学院这个学习知识、砥砺品行的地方求学，倍感荣幸。

首先，谨向我的导师徐明教授致以衷心的感谢。徐明老师严谨求实的治学态度、敏锐的学术洞察力、渊博的专业知识、忘我的工作热情、一丝不苟的工作作风给我留下了很深的印象，并深深地影响着我工作态度。在攻读博士学位期间，徐老师自始至终从学业、科研等多方面给予我很多启发和帮助。本文是在徐明老师的悉心指导下完成的，论文的字里行间中都凝聚着他的很多心血，让我终身难忘。徐明老师对科学事业孜孜以求、废寝忘食的崇高精神，鞭策我奋发努力、锐意进取。徐明老师的严格要求让我在感激之余，深感受益匪浅和唯恐有负厚望。在本文即将结束之际，谨向导师徐明教授致以深深的敬意与诚挚的谢意。

感谢国防科大计算机学院博士生队的所有队领导，他们在我读博的几年里给我提供很多生活和学习上的便利。

感谢所有同门师弟师妹们在我博士论文期间提供的非常宝贵的帮助。

感谢朝夕相处多年的同室的王树林副教授、唐文胜副教授和张波云副教授，他们与我一起渡过那段难熬的非典岁月；几年的共同学习生活不仅让我感受到友情的力量，使我忘掉生活和工作中的烦恼，让我感觉到了生活的愉快，这种友谊是赐予我的非常宝贵的财富。

感谢珠海电信的伍春华博士在论文撰写期间对我兄弟般的鼓励和帮助。

感谢广东省电信的弟弟张贻锋硕士在我读博期间挑起赡养父母的担子。

感谢我的儿子张沛然在我读博期间完成了初中阶段的学习，并以优异的成绩考上了省重点中学，为我减轻了精神上的负担。

衷心感谢所有支持、理解和帮助过我的师长、同学、同事、亲人和朋友们。最后，谨向百忙中抽出宝贵时间评阅本论文的各位专家、学者致以最诚挚的谢意！

作者: [张鼎兴](#)
学位授予单位: [国防科学技术大学](#)

相似文献(4条)

1. 学位论文 [嵇玮玮 无线传感器网络的节点定位与覆盖技术研究](#) 2008

无线传感器网络是随着微机电技术、传感器技术、嵌入式计算技术、现代网络、无线通信技术以及分布式信息处理技术等的发展建立起来的分布式信息感知与处理系统,在军事、医疗、商业以及环境保护等领域具有广泛的应用前景。

无线传感器网络是由具有感知、计算和通信能力的无线传感器节点组成。在其各种各样的应用研究中,节点定位与网络覆盖是无线传感器网络应用的两个主要支撑技术。节点定位是实现无线传感器网络众多应用的前提,其定位精度是无线传感器网络的关键性能指标之一;而网络覆盖则决定了无线传感器网络所能提供的服务范围,也在很大程度上影响了网络的成本和各种具体应用的性能,是网络设计时必须考虑的首要问题。

本文主要针对无线传感器网络中的节点定位与网络覆盖技术进行深入的研究与探讨。在节点定位技术方面,分别研究了基于距离的定位算法,距离无关的定位算法以及递增式定位算法;在网络覆盖技术方面,分别对静态网络覆盖与动态网络覆盖进行了研究。

本文的主要工作可以归纳如下:

1. 基于距离的定位算法与距离无关的定位算法研究1)研究了基于距离的定位算法,提出基于UWB的TOA测距方法。UWB信号具有信号带宽大、测距精度高的特点,但在节点定位应用中,UWB直达信号难以精确检测。本文提出通过对首次到达信号时间和最强信号时间进行加权来得到直达信号到达时间,并采用模糊逻辑技术计算加权系数。实测数据仿真试验表明,基于UWB的定位技术可极大地提高定位精度。

2)研究了距离无关的定位算法,并对其中的典型DV-Hop算法提出了改进。DV-Hop算法本身是一种传统的距离无关定位算法,在各向同性的密集网络中,DV-Hop可以得到比较合理的定位精度。然而,在随机分布的网络中,节点定位误差较大。本文根据DV-Hop算法定位过程,在平均每跳距离估计、未知节点到各参考节点之间距离的计算和节点位置估计方法等3个方面进行了改进,分析和仿真了不同改进措施和综合改进的定位性能。这些改进有效的提高了传感器节点的定位精度,且无需增加网络中节点的通讯开销与硬件复杂度。

2. 递增式定位算法研究1)在对递增式定位算法进行详细分析的基础上,指出了递增式定位算法存在的累积误差和无效节点问题。

2)在对累积误差特点进行分析的基础上,提出将未知节点与其参考节点中定位精度较高的节点间的距离作为约束条件来减少定位过程中的累积误差。仿真结果表明,本文方法能够显著减小传感器节点定位过程中的传播误差,提高节点定位精度,改善了方法的适用性。3)分析了无效节点的存在特征并对其进行分类,提出利用运动目标的位置信息以及无效节点与已知节点的连接信息对无效节点进行定位,显著提高了无线传感器网络对目标的跟踪性能。

3. 网络覆盖技术研究1)研究了静态网络覆盖,提出了新的网络静态覆盖与动态覆盖方式。在业已开展的网络静态覆盖研究中,网络1-覆盖得到深入研究。本文针对网络静态覆盖中的确定性覆盖和随机性覆盖,分别提出了新的传感器节点部署策略和调整传感器节点感知半径的方式,实现了无线传感器网络的k($k \geq 3$)覆盖。针对网络动态覆盖,提出了能量有效的本地节点选择方案,节点根据相邻时刻接收到的目标信号能量变化独立决定其工作状态。理论分析和计算机仿真表明该方案可大大减少与相邻节点及中心节点的信息交换,减少网络的能量消耗。

2)研究了动态网络覆盖,提出一种最佳节点分布几何结构,提高了网络对目标位置的估计性能。

2. 学位论文 [刘存祥 无线传感器网络覆盖技术研究](#) 2008

无线传感器网络是由低成本、低功耗、具备感知、数据处理、存储和无线通信能力的微型传感器节点通过自组织方式形成的网络。为了保障网络的可靠性,提高网络的容错能力和数据采集的精度,大部分无线传感器网络采用高冗余的设计原则,从而使得网络中的节点数量众多。如何在单个节点能量受限、生存时间较短的情况下节省节点的能量从而延长整个网络系统的生存时间,是无线传感器网络覆盖协议设计中的一个重要挑战。

本文是无线传感器网络覆盖技术研究,为达到有效探测目标区域的目的,只使部分传感器节点工作,而把剩余传感器节点设置为休眠状态。在工作状态的部分传感器节点能量耗尽后,激活休眠状态的节点,以保证无线传感器网络继续有效探测目标区域。

本文主要研究成果包括以下两个方面:

(1)对基于2-Voronoi图的覆盖控制算法进行了详细阐述,针对网络中出现失效节点的情况,提出邻居信息重建方案。

(2)对Di Tian和Nicolas D. Georganas提出的覆盖算法CPNSS (Coverage preserving node scheduling scheme)进行了详细阐述,然后分析了CPNSS的不足,并在其基础上进行了改进,形成了完整的覆盖控制算法ECPNSS (Extended coverage preserving node scheduling scheme),并通过仿真对比两者之间的性能差异。

3. 学位论文 [刘华峰 传感器网络三维拓扑组织及分簇算法研究](#) 2007

无线传感器网络是由低成本、低功耗、具备感知、数据处理、存储和无线通信能力的微型传感器节点通过自组织方式形成的网络系统。拓扑组织和能量消耗是无线传感器网络的两个核心问题。无线传感器网络的拓扑结构决定了覆盖和连通性等网络基本特性,而能量消耗则决定了无线传感器网络的生存时间。本文针对无线传感器网络在三维空间的拓扑组织问题和传感器节点的高能效分簇问题进行了深入的研究。

目前大多数无线传感器网络研究采用的二维平面假设,并不能完全满足实际应用的需求。在现实世界中存在着大量三维无线传感器网络应用,例如空间、水下和地下传感器网络系统。然而,由于三维空间中求解问题的复杂性,现有的二维无线传感器网络的研究成果大部分不能直接应用于三维无线传感器网络。三维无线传感器网络的发展要求新的理论和方法。

本文首先讨论了随机三维无线传感器网络中的相变现象。针对经典概率分析工具过于复杂的问题,利用箱覆盖技术,推导了随机三维无线传感器网络的临界感知半径。

接着本文将代数理论引入无线传感器网络的三维拓扑结构研究。首先面向理想空间环境,提出三种规则对称的立方格结构无线传感器网络,根据格的生成矩阵给出节点部署位置的计算方法,并分析了网络覆盖和连通的条件。这种确定部署的规则空间结构无线传感器网络具有很多优点,如可以保证对目标空间的完全覆盖,易于形成连通的稀疏网络,节点数一定的条件下可以最大化覆盖范围,减少节点间的通信量等,适用于空间、地下和建筑物内部等网络应用。

确定部署的规则空间结构无线传感器网络虽然特点突出,但是对于大规模的无线传感器网络,确定部署会带来很大的开销,而且在很多应用领域,由于环境的限制,可能无法进行确定部署。所以本文针对随机部署的三维无线传感器网络,提出一种基于虚拟立方格Voronoi单元的三维无线传感器网络空间组织策略。空间中随机分布的传感器节点通过计算距离自己最近的虚拟立方格格点,确定自身所在虚拟Voronoi单元的ID。同一单元内的节点可以通过一定的选举算法,周期性的选举一个活动节点代表单元工作,从而实现在满足覆盖和连通要求的同时,减少节点能耗,延长网络寿命的目的。

水下传感器网络是一种典型的三维网络应用系统。本文针对水下监测应用的特点,提出一种新颖的三维水下传感器网络拓扑生成算法ETG。算法通过控制传感器节点在垂直方向上的移动,由随机部署在海平面上的传感器节点自动生成近似体心立方格结构的三维传感器网络。ETG算法将节点的部署和调度相结合,在高节点密度的条件下,可以快速形成一个由活动节点构成的覆盖目标空间的三维监视网络。仿真实验表明,与现有的完全随机部署策略和基于立方体覆盖的深度调节策略相比,ETG算法不但可以提高网络覆盖效率,而且能够通过少量节点的移动覆盖较大的空间,从而有效减少网络初始建立阶段由于节点移动带来的系统能耗开销。

最后本文研究了两种类型的无线传感器网络高能效分簇算法。第一类算法基于虚拟单元划分网络覆盖空间,一个单元一旦确定,在网络生存期内成员将不再变化,所以称为静态分簇算法。本文分析了现有单元内随机簇首选举算法存在的缺陷,提出一种自适应的随机簇首选举算法SARLE,通过大量的仿真实验对SARLE的性能进行了评估。第二类是动态分簇算法,网络随机选举簇首并周期性的重新选举,通过簇首和簇成员间角色的不断变化,将能量开销分配到整个网络。LEACH是最著名的无线传感器网络动态分簇算法之一。本文利用随机过程的方法指出LEACH协议中存在簇个数不稳定的问题,提出一种改进的分簇方案I-LEACH,并通过大量的仿真实验说明I-LEACH的有效性。

4. 学位论文 [余金 基于多跳的定位机制与节能覆盖技术研究](#) 2008

传感器、微机电系统(MEMS)、集成电路和低功耗无线通信等技术的飞速发展,使得低成本、低功耗、多功能的微型无线传感器网络的大规模应用成为可能。布局和覆盖、节点定位、网络通信协议是传感器网络研究中的几个基本问题。其中,节点定位问题是传感器网络进行目标识别、监控、跟踪等众多应用的前提,也是传感器网络研究中的热点问题之一。

本文对传感器网络的定位技术进行了较为深入的研究。由于传感器节点受体积、成本和能量等因素的限制,且通常工作在人类难以或不接近的场合,因此定位技术的基本途径是:部署少量已知坐标的参考节点,其他未知坐标的节点通过测量与参考节点的距离、角度,或依据相对位置关系、网络连通性进行一定的计算得出自身的坐标。

DV-Hop定位算法将参考节点的位置信息在网络中进行传播,并将未知节点到参考节点之间的距离用网络平均每跳距离和两者之间最短路径跳数乘积表示,使用三边计算获得节点位置信息。本文研究适用于煤矿环境的无线传感器网络节点定位技术,对其中的几种主要算法进行了分析与比较;讨论基于多跳的定位机制,为实现煤矿环境下应用对DV-Hop算法进行了改进,仿真结果表明,改进算法比传统的DV-Hop算法定位精度约提高了10%。研究了定位覆盖问题,对Node Self-Scheduling覆盖控制算法和暴露穿越覆盖两种方案进行了研究、分析和比较,并基于这些优缺点比较,根据煤矿监测系统的特点,提出一种

基于Self-Scheduling覆盖算法的节能覆盖方法，仿真分析表明，采用本文的节能覆盖算法，在网络生命期上明显优于贪婪算法，并且本文的节能覆盖算法敏感于网络节点数量，在大量网络节点的情况下，优势更加明显。本文提出的优化覆盖方案，具有较长的网络寿命，是一种适合于煤矿环境监测系统的无线传感器网络覆盖方案。

本文链接：http://d.wanfangdata.com.cn/Thesis_Y1523921.aspx

授权使用：于晶晶(wfdwj)，授权号：201ba92d-a6ec-4ec5-8627-9e88016a188d

下载时间：2011年2月12日