



Y1851322

**Dissertation Submitted to Zhejiang University of Technology
for the Degree of Master**

**RESEARCH ON THE HIERARCHICAL ROUTING
PROTOCOL OF WIRELESS SENSOR NETWORK
BASED ON SWARM INTELLIGENCE ALGORITHM**

Candidate: Hou Jiabin

Advisor: Wang Wanliang Fan Xinggang

College of Computer Science and Technology

Zhejiang University of Technology

Nov 2010

浙江工业大学

学位论文原创性声明

本人郑重声明：所提交的学位论文是本人在导师的指导下，独立进行研究工作所取得的研究成果。除文中已经加以标注引用的内容外，本论文不包含其他个人或集体已经发表或撰写过的研究成果，也不含为获得浙江工业大学或其它教育机构的学位证书而使用过的材料。对本文的研究作出重要贡献的个人和集体，均已在文中以明确方式标明。本人承担本声明的法律责任。

作者签名：侯佳斌

日期：2010年12月29日

学位论文版权使用授权书

本学位论文作者完全了解学校有关保留、使用学位论文的规定，同意学校保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权浙江工业大学可以将本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和汇编本学位论文。

本学位论文属于

1、保密 ，在 _____ 年解密后适用本授权书。

2、不保密

(请在以上相应方框内打“√”)

作者签名：侯佳斌

日期：2010年12月29日

导师签名：王万

日期：2010年12月29日

基于群智能算法的无线传感网络分簇路由协议的研究

摘 要

设计一种节能高效的无线传感网络路由协议始终是无线传感网络研究领域的一个重要课题。无线传感网络路由协议—LEACH 协议(Low Energy Adaptive Clustering Hierarchy)是一种经典的分簇路由协议, LEACH 协议相比其他平面路由协议生命周期提高 15%以上。然而 LEACH 协议还是存在不少使能耗过大的缺陷, 为改进 LEACH 协议这些缺陷本文提出了一种节能高效的基于群智能算法的智能分簇路由协议 SI-CRP 协议(Swarm Intelligence—Clustering Routing Protocol)。

本文提出的 SI-CRP 协议主要内容和创新点如下:

1. LEACH 协议随机选举簇首的机制带来分簇不均匀的问题, 这导致节点在发送数据时耗费过多能量, 针对这个问题 SI-CRP 协议提出基于离散粒子群算法直接计算全局最优位置的簇首, 从而使分簇均匀。

2. LEACH 协议簇首与基站直接单跳通信的机制使簇首在发送数据时消耗大量能量, 针对这个问题 SI-CRP 协议提出基于蚁群算法以节点通信能耗为权值的簇首间多跳路径算法, 从而找到从某个簇首出发经过由其他簇首到基站的最小能耗多跳路径;

3. LEACH 协议周期性重新全局分簇的机制也会消耗大量的能量, 针对这个问题 SI-CRP 协议提出基于能量自感应的局部簇首更新机制, 从而大大减少 LEACH 协议重新全局分簇消耗的能量。

4. 本文用 MATLAB 对 SI-CRP 协议和 LEACH 协议进行仿真比较。分簇仿真实验的结果表明, 对于相同的传感器节点位置分布, SI-CRP 协议的分簇效果要比 LEACH 协议好的多, 簇结构均匀簇首位置合理。协议运行仿真实验的结果表明, 在相同的节点位置分布和网络工作模式下, SI-CRP 协议的节点生命周期比 LEACH 协议延长 1 倍多, 网络能耗也比 LEACH 协议节省一半多。

本文提出的 SI-CRP 协议以改进 LEACH 协议缺陷为目标, 结合群智能算法的研究成果, 将改造后的群智能算法应用于无线传感网络分簇路由协议, 降低了网络工作能耗, 延长了网络生命周期。

关键词：LEACH 协议，SI-CRP 协议，离散粒子群算法，蚁群算法，最小能耗多跳路径，局部簇首更新

RESEARCH ON THE HIERARCHICAL ROUTING PROTOCOL OF WIRELESS SENSOR NETWORK BASED ON SWARM INTELLIGENCE ALGORITHM

ABSTRACT

To design a power-efficient routing protocol is always an important subject in WSN research area. LEACH(Low Energy Adaptive Clustering Hierarchy) is a classical clustering routing protocol in WSN routing protocols area. In comparison with other flat routing protocols, LEACH can prolong network lifetime by 15%. While there are some shortcomings in LEACH which waste a lot of energy of sensor nodes, therefore an intelligent routing protocol—SI-CRP(Swarm Intelligence—Clustering Routing Protocol) based on swarm intelligence algorithms is proposed in this dissertaton to solve the shortcomings in LEACH.

The main content and contribution of SI-CRP proposed by this paper are as follows:

1. The uneven clustering problem caused by the random selecting cluster heads mechanism of LEACH leads to much energy consumption. To solve this problem, SI-CRP proposes an algorithm based on DPSO(Discrete Particle Swarm Optimization) to directly calculate cluster heads with global best position to make clustering result even.

2. To solve the energy waste problem caused by single-hop communication between cluster heads and base station, SI-CRP proposes an algorithm based on ACO(Ant Colony Optimization) to calculate multi-hops path between cluster heads and base station using pathweight as communication energy consuming. SI-CRP gets the minimum energy consumption communication path starting from certain cluster head to base station using this algorithm.

3. Periodical global re-clustering mechanism of LEACH aslo leads to much unnecessary energy consumption, to solve this problem SI-CRP proposes local cluster heads update mechanism based on energy self-awareness which reduces a lot of energy consumption brought by global re-clustering mechanism of LEACH.

4. SI-CRP is compared with LEACH in this dissertation through MATLAB simulation. The result of clustering simulation shows for the same position distribution of sensor nodes SI-CRP gets much better clustering result in comparison with LEACH where the cluster structure is more even and the positions of cluter heads are more reasonable. The result of protocol working simulation shows for the same nodes position distribution and netwok working mode lifetime of

SI-CRP is more than 1 time longer than that of LEACH and the average network energy consumption of SI-CRP is half of that of LEACH.

SI-CRP of this paper which aims to solve the defects in LEACH reduces the network energy consumption and prolong the network lifetime by researching the swarm intelligence algorithms and the related research result and apply them to WSN routing protocol.

Key words: LEACH, SI-CRP, DPSO, ACO, minimum energy consumption communication path, local cluster heads update mechanism

目录

摘要	i
第 1 章 绪论	1
1.1 无线传感器网络概述	1
1.1.1 无线传感网络的概念	1
1.1.2 无线传感器网络的体系结构	1
1.1.3 无线传感器网络协议栈	3
1.1.4 无线传感器网络的关键技术	4
1.2 无线传感网络分层路由协议	5
1.2.1 无线传感网络路由协议概述	5
1.2.2 无线传感器网络分层路由协议介绍	9
1.2.3 LEACH 协议的改进	13
1.3 本文的研究内容和组织结构	18
第 2 章 群智能算法	20
2.1 粒子群优化算法	20
2.1.1 标准粒子群优化算法	21
2.1.2 粒子群算法惯性权重	23
2.1.3 离散粒子群算法	26
2.2 蚁群优化算法	28
2.2.1 蚁群优化算法原理	28
2.2.2 信息素浓度增量计算模型	29
2.2.3 蚁群算法参数设置	30
2.3 本章小结	31
第 3 章 基于群智能算法无线传感网络分簇路由协议	32
3.1 基于离散粒子群算法的分簇策略	33
3.1.1 适应度目标函数	33
3.1.2 分簇问题的离散粒子群改造	34
3.1.3 混合惯性权重调整和粒子变异策略	36
3.1.4 启发式算法	37
3.1.5 选择全局最优簇首流程	38
3.2 基于改进蚁群算法的簇首间多跳策略	40
3.2.1 概率选择公式期望函数改进	41
3.2.2 信息素浓度增量改进	41
3.2.3 基于无线传感网络路由的蚁群算法机制改进	42
3.2.4 基于蚁群算法簇首间多跳路径计算流程	42
3.3 基于能量自感应的局部簇首更新机制	44
3.4 本章小结	46
第 4 章 SI-CRP 协议的实现与仿真	48
4.1 SI-CRP 协议的实现	48

4.1.1 网络簇结构生成阶段流程.....	48
4.1.2 稳定数据传输阶段流程.....	50
4.2 分簇实验仿真分析.....	52
4.3 协议运行仿真分析.....	56
4.4 本章小结.....	59
第5章 总结与展望.....	61
5.1 总结.....	61
5.2 展望.....	62
参考文献.....	64
致谢.....	67
攻读学位期间参加的科研项目和成果.....	68

第 1 章 绪 论

1.1 无线传感器网络概述

1.1.1 无线传感网络的概念

随着传感器技术、无线通信技术、集成电路等技术的飞速发展和日益成熟,传感器领域已经开始向微型化、集成化和网络化方向发展,于是无线传感器网络(Wireless Sensor Network, 简称 WSN)这个概念就应运而生了。

无线传感网络是由大量部署在监测区域中具有数据处理和无线通信能力的微型传感器节点形成的多跳的自组织网络系统,是一种全新的信息获取和处理技术。无线传感网络的产生带来了一场信息技术的革命,它能够在无人值守的监测区域实时监测、感知和采集人们感兴趣的环境信息数据(如光、温度、声音等),并通过传感器节点间的无线射频将采集到的数据传送到基站(Base Station),最后通过 Internet 或者其他网络通讯方式将监测信息传送给管理控制中心^[1]。

无线传感器网络的出现引起了全世界范围的广泛关注。无线传感器网络是继Internet之后,会对人类生活方式产生重大影响的热点技术。如果说Internet改变了人与人之间沟通的方式,那么无线传感器网络则将信息处理技术与真实物理世界融合在一起,这将产生一种人类和自然全新的交互的方式。无线传感器网络能广泛地应用于军事、环境检测和预报、健康护理、智能家居、建筑物状态监控、城市交通、空间探索等诸多领域,美国商业周刊和MIT技术评论在预测未来技术发展的报告中,分别将无线传感器网络列为21世纪最有影响的21项技术和改变世界的10大技术之一^[2]。2009年,美国IBM公司和中国总理温家宝分别就物联网技术提出了“智慧地球”和“感知中国”,而无线传感器网络正是物联网的最核心技术。可以预计,无线传感器网络的发展和广泛应用,将对人们的社会生活和产业变革带来极大的影响和产生巨大的推动。

1.1.2 无线传感器网络的体系结构

一个典型的无线传感器网络包括具有无线射频功能的传感器节点(Nodes)、基站(Base Station)或者汇聚节点(Sink)、基础设施网络(Internet或卫星)以及无线传感器网络任务管理节点,如图1-1所示。无线传感器节点通过人工部署或者飞机布撒的方式大量布置在被监测区

域内并自组织构成网络。传感器节点采集数据后通过多跳中继的方式传送给基站，然后基站再通过互联网或者卫星将数据传送到任务管理节点进行集中处理。其中普通传感器节点由电池供电，能量有限而且计算存储能力都比较弱。而基站能量充足，且有计算和存储能力较强的设备。

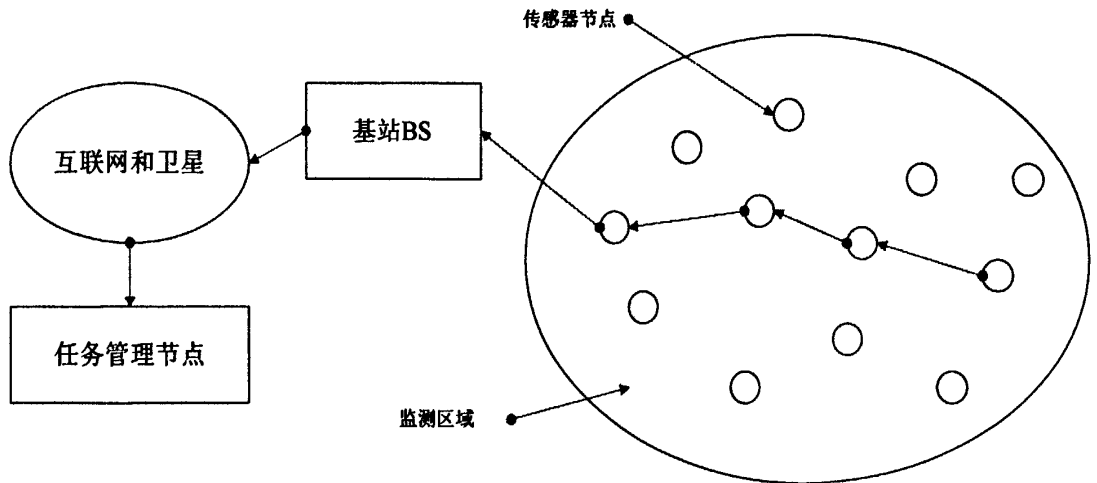


图1-1 无线传感器网络的体系结构

传感器节点包括四个基本模块：数据采集模块(由传感器和数模转换器组成)、数据处理模块(由微处理器和存储器组成)、无线通信模块(通常是无线收发器)和能量供应模块(通常采用能量很有限的电池)，如图1-2所示。首先由传感器采集到感知对象的模拟信号，然后通过数模转换器将采集的模拟信号转换为数字信号。然后节点将该数字信号传递给数据处理模块对数据进行存储或处理。节点判断该数据如果是需要传送的，则节点通过无线射频将数据传送给另外的节点。而能量供应模块则提供以上三个模块工作所需要的能量，通常采用微型电池供电。有些应用情况可能需要传感器节点获得其精确的位置信息，因此必须配具有位置搜索功能的模块。军事应用中可能要求传感器节点具有移动功能，这时候传感器节点就需要机械装置支持传感器节点的移动功能。

Sink节点与普通传感器节点有很大不同，它不但具有很强的无线通信能力，而且具有较强的计算和存储能力，它还可以通过有线线路源源不断地获取电能。它主要用于发布各种控制指令，控制监测区域各传感器节点的行为，监测网络运行情况，处理收集传感器节点采集的数据。同时它还能作为互联网的连接终端，将数据通过互联网传送给控制中心。

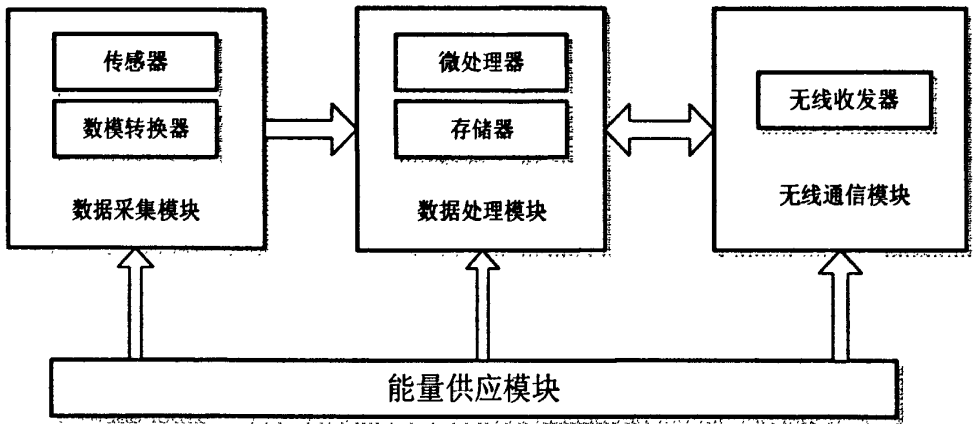


图1-2 无线传感器网络节点结构

1.1.3 无线传感器网络协议栈

无线传感器网络协议栈^[3]包括物理层、数据链路层、网络层、传输层、应用层以及能量管理平台、移动管理平台和任务管理平台，如图1-3所示。

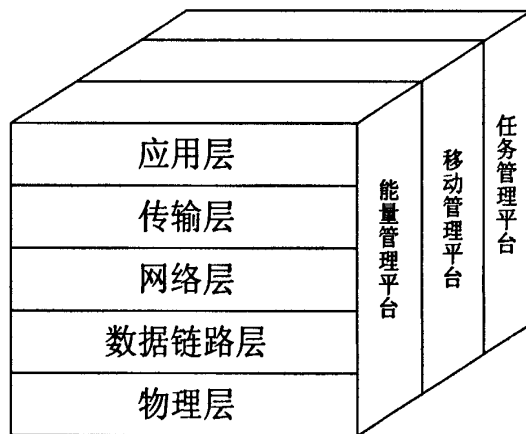


图1-3 无线传感器网络协议体系结构

物理层主要负责信号的调制和数据的收发，目前无线传感器网络主要是通过无线电进行通信，也有些应用场合可能会使用红外线和光波。无线电通信中主要解决无线通信的频段选择、调制和扩频技术等。物理层是决定传感器节点体积、功率和成本的关键。

数据链路层主要用于建立可靠的点到点或点到多点的通信链路，主要技术是介质访问控制MAC协议(Media Access Control)。MAC协议主要是使传感器节点能够以共享的方式使用传输介质。MAC有基于调度和基于随机访问两种方式。根据通信资源分配方式的不同，基于调度机制的方式可以分为时分多址、频分多址和码分多址三种。基于随机访问机制可分为时隙和非时隙协议，以及载波多路访问协议等。MAC协议的性能指标包括带宽要求、

功耗和冲突避免等。

网络层负责路由建立、路由选择和路由维护，主要就是要为网络层设计高效、节能、简单的路由协议。由于传感器节点能量有限，网络层的路由协议将决定整个无线传感器网络的拓扑结构，而网络的拓扑结构的合理与否将决定整个网络的能量损耗，因此路由协议必须以高效节能为第一目标。而传感器节点计算能力也很有限，而且复杂的路由建立和维护也将消耗大量的能量，因此无线传感网络路由协议必须简单。无线传感器网络路由协议按照层次结构可以分为平面路由协议和分层路由协议。本文提出的基于群智能算法的 SI-CRP协议就是一种分层结构的路由协议。

传输层负责数据流的传输控制，是保证通信服务质量的重要部分。无线传感网络传输层主要解决如何降低传输层技术能耗，如何保证数据传输的可靠性，如何提高网络容错性和可扩展性等问题。目前针对无线传感网络传输层主要有PSFQ(Pump Slowly Fetch Quickly)协议和ESRT(Event-to-Sink Reliable Transport)协议。

应用层提供传感器节点与网络的服务接口。应用层通过协调控制整个网络，优化现有的网络资源，以获得最大的网络利用率和单个任务的最少消耗量，同时提供可操作的人机交互界面。目前已经提出的一些应用层协议是传感器管理协议(SMP, Sensor Management Protocol), 任务分配和数据广告协议(TADAP, Task Assignment and Data Advertisement Protocol)和传感器询问和数据分发协议(SQDDP, Sensor Query and Data Dissemination Protocol)。

1.1.4 无线传感器网络的关键技术

无线传感器网络的关键技术包括能量控制管理、时间同步、定位、数据融合和安全等方面。

能量控制管理技术：由于无线传感器网络的节点都是由能量有限的电池供电，如果电池电量耗尽，该节点就成为失效节点无法继续工作。若传感器节点被部署在军事战场或环境恶劣的地方，那么更换电池也是不可能的。因此研究能量控制管理策略可以有效地提高能量的使用效率，从而延长网络寿命。能量控制管理策略需要与节点的具体应用、路由协议、MAC 协议和物理层等各种要素结合在一起研究。

时间同步技术：很多无线传感器网络的协议和应用都要求各节点的时钟保持同步，比如数据融合算法和低能耗的 MAC 协议等。结合无线传感器网络的特点，面向无线传感网络的时间同步研究需要考虑以下因素：能源有效性、可扩展性、可靠性、自适应性和自配置性。目前无线传感器网络的同步研究包括多模可调节的自适应同步调节机制以及面向领

域的不同步机制探索。

定位技术: 某些应用需要传感器节点具有准确定位的功能比如网络管理、目标跟踪等场合都需要精确的节点位置信息。无线传感器网络的定位技术的目标是以最小的通信开销或者硬件代价实现节点定位。获得节点位置的一个直接方法是通过全球定位系统(GPS)来实现,但由于使用 GPS 系统来获得节点位置信息受到价格、体积、功耗等方面的限制,存在着一些困难。还有一种选择就是利用少量已知位置的节点来获得其他节点的定位算法。目前无线传感器网络节点自身的定位算法可以分为基于协作的无线传感器网络定位算法和基于非协作的无线传感器网络定位算法。目前的研究成果来看,基于协作的定位算法相对成熟一点,基于非协作的定位算法由于受到一些因素的制约,需要进一步完善。

数据融合技术: 在大规模网络应用中,由于每个传感器节点的物理因素限制,在布置传感器节点时,需要重叠节点的监测覆盖范围以增强整个网络的鲁棒性和准确性。那么,整个网络中的采集的数据就会具有一定的空间相关性,即相隔不远的节点所采集的数据具有一定的相似性。在传统的数据传输模式下,每个节点都将传输全部的监测信息,这其中就包含了大量的相似性很高的信息,这些相似信息是无用的而且还造成不必要的能量浪费。因此,在大规模无线传感器网络中,使各个节点多跳传输监测数据到基站前,先对数据进行融合处理是非常有必要的,数据融合技术应运而生。

安全技术: 无线传感器网络在实际应用中往往是被部署后就处于无人值守的状态,或者在军事应用中直接暴露在敌方区域,因此安全问题是无线传感器网络不得不面对的一个严峻问题。在无线传感器网络的研究中,针对物理层主要研究高效的加密算法以及通过扩频通信减少电磁干扰;数据链路层研究安全的 MAC 协议;网络层主要研究安全的路由协议;应用层主要研究密钥管理和安全组播机制。考虑到无线传感器网络的特点,一些传统网络上复杂的安全协议和机制不能适用于无线传感器网络,因此无线传感器网络安全技术是一个研究热点。当前无线传感器网络安全方面的研究主要集中在加密与认证消息机制,密钥管理,安全路由这几个方面。

1.2 无线传感网络分层路由协议

1.2.1 无线传感网络路由协议概述

由于无线传感器网络传感器节点的能量有限,而路由协议的好坏直接决定传感器节点在发送接收数据耗费的能量,因此设计节能简单高效的无线传感器网络路由协议一直是无线传感器网络领域的研究热点。

无线局域网、无线自组网等传统无线网络的首要目标是提高服务质量的网络并能高效公平地利用网络资源。这类网络路由协议的主要任务是寻找源节点到目的节点的通信延迟最小路径，避免产生拥塞现象均衡网络流量。而无线传感器网络节点由电池供电，能量有限，路由协议的好坏将直接决定无线传感器网络的生命周期。因此无线传感器网络路由协议的首要目标是如何高效地利用能量。除了高效利用能量的首要目标以外，与传统的无线网络和Ad hoc网络的路由协议相比无线传感器网络的路由协议还具有以下特点：第一，无线传感器网络是以数据为中心的，与传统网络中以地址为中心的路由方式是不同的；第二，由于传感器节点的计算能力和存储能力都很有限，因此无线传感器网络的路由协议应尽可能简单，既不能有大量的路由信息，也不能有复杂的路由计算；第三，由于相邻传感器节点采集的数据具有相似性，存在冗余信息，因此需路由协议要消除冗余，来提高能量和带宽利用率；第四，无线传感器网络随应用需求而变化，很难设计通用性强的路由协议。

现在无线传感器网络路由协议的研究正在全世界范围内如火如荼地开展。美国美国的麻省理工学院、康奈尔大学、南加州大学等很多大学开展了无线传感器网络路由协议的研究，先后提出了几种新的路由协议，包括SPIN^[8]、Directed Diffusion^[6]、LEACH^[10]、TEEN^[12]、APTEEN^[12]、PEGASIS^[11]等。

网络数据传输离不开路由协议，它是整个网络工作的基础。无线传感器网络的特点对路由协议的设计提出了更高的要求，因此开展对无线传感器网络路由协议的研究是非常有意义的。虽然现在已经提出了很多经典的路由协议，但是这些路由协议都有各自的优点和缺点，国内外学者就针对上述经典的无线传感器网络路由协议提出了各种改进算法。本文正是以无线传感网络路由协议为研究对象，针对无线传感器网络现有分层路由协议存在的问题，提出了一种新的能量高效、简单的分簇路由协议。

无线传感器网络路由协议的任务是将源节点采集的数据经过网络多跳传输到目的节点，这主要包含两个功能：确定源节点到目的节点的优化路径；将数据通过该确定的优化路径正确转发。在无线传感器网络中，当一个源节点不能直接将数据分组发送到目的节点时，需要通过中继节点传输数据分组。

随着无线传感器网络的研究与发展，国内外已经有很多经典的无线传感器网络路由协议被提出。根据不同的划分角度，可以将无线传感器网络路由协议分成不同的类别。从网络拓扑结构的角度划分，可以将无线传感器网络路由协议分为平面路由协议和分层路由协议；根据路由发现策略的角度，可以将无线传感器网络路由协议分为主动路由协议和被动路由协议；根据是否以地理位置来标识目的地，路由计算中是否利用地理位置信息，可以

将无线传感器网络路由协议分为地理定位路由辅助路由协议和无地理定位路由辅助路由协议；从路由选择是否考虑服务质量QoS(Quality of Service)的角度，可以将无线传感器网络路由协议分为保证QoS路由协议和不保证QoS的路由协议^[4]。

本文主要按照网络拓扑角度的划分来研究两类无线传感器网络路由协议：平面路由协议和分层路由协议。

1) 平面路由协议

在平面路由协议中，所有传感器节点的地位都是平等的。在路由优化路径生成阶段，传感器节点之间通过相互的信息反馈和局部交互来生成路由路径。当某个传感器节点采集到数据并要传输给基站时，可以通过其他传感器节点作为中间节点来转发数据，经过一系列中间节点的转发，数据最后被传输到基站，如图1-4所示。从图中可以看出，每个传感器节点的地位都是一样的，每个节点既可能作为源节点采集数据，也可能作为中间节点转发数据。一般来说，基站附近的传感器节点担当中间节点进行转发的概率要远大于远离基站的传感器节点，因此基站附近的传感器节点可能由于大量的转发任务而很快耗尽电能。平面路由协议一般由基站发送查询任务到特定的监测区域，监测区域的节点收到查询任务后就采集数据，然后将数据按照路由表发送给路由路径的下一跳传感器节点。

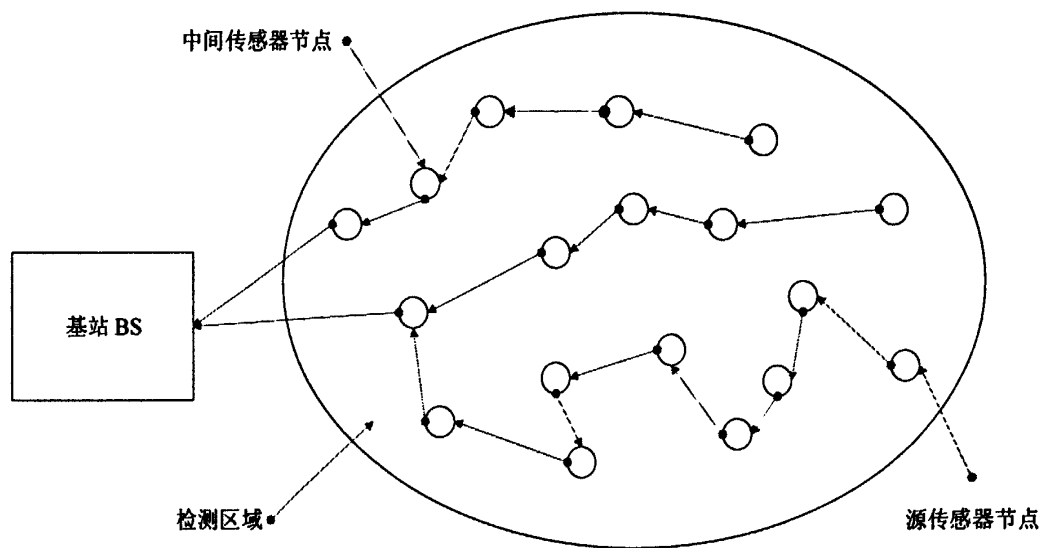


图1-4 无线传感器网络平面路由协议拓扑结构

平面路由协议的优点是协议简单，易扩展，无须进行任何结构维护，具有良好的健壮性。缺点是建立维护路由的开销比较大，网络中无管理节点缺乏对资源的优化管理，对网络结构的动态变化反应速度比较慢，另外平面路由结构中数据从源节点传输到目的节点一般要经过很多中间节点的转发，因此传输时延比较大，适合规模比较小的网络。

典型的无线传感器网络平面路由协议有Flooding协议, Gossiping协议^[5], DD协议^[6](Directed Diffusion), SAR协议^[7](Sequential Assignment Routing), SPIN协议^[8](Sensor Protocols for Information via Negotiation), Romor协议^[9]等。

2) 分层路由协议

分层路由协议又称为分簇路由协议, 整个无线传感器网络被划分成簇(cluster)。在这种分簇的拓扑结构中, 无线传感器网络中的节点被分为簇首CH(Cluster Head)和簇内节点两类, 因此分簇路由协议中节点的地位和职责是不平等的。每个簇都有一个簇首和若干个簇内节点, 簇内节点专门负责采集数据, 然后将采集的数据发送给它的簇首, 而簇首节点负责收集本簇内所有簇内节点采集的数据, 并经过数据融合处理以减少数据传递量, 然后将融合后的数据发送给更高级的簇首或者基站。最简单的分簇结构即为两层, 整个网络分成若干个簇, 每个簇的簇首构成高层网络, 而每个簇的簇内节点为低层网络, 如图1-5所示。复杂的分簇结构可以是多层的结构, 即低一层网络的簇首是高一层网络中的簇内成员, 低一层网络的簇首与同层的簇首形成高一层的簇与该层的簇首通信, 最高层网络簇首直接与基站通信。整个分层路由协议分成两个阶段: 第一个阶段由一定的策略将整个网络划分成若干个簇, 并确定簇首和簇内节点; 第二个阶段在形成的簇结构按照簇内成员到簇首的路由路径传送数据。

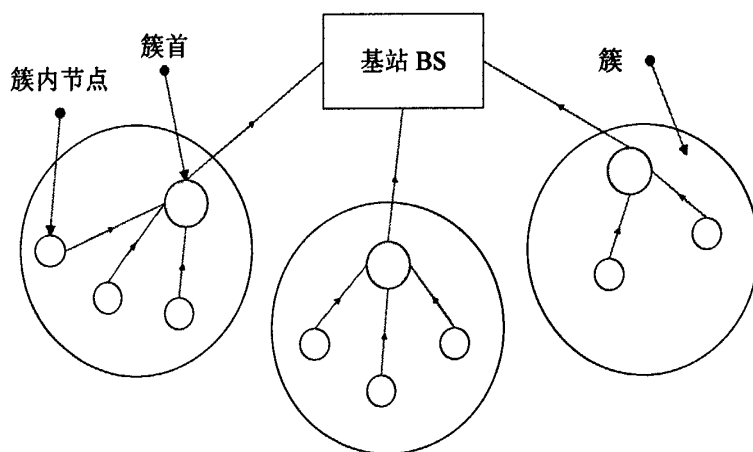


图1-5 无线传感器网络分层路由协议拓扑结构

无线传感器网络分层路由协议相比平面路由协议的优点是: 成员节点的功能简单, 无须维护复杂的路由信息, 大大减少了路由控制维护的开销; 分簇的结构便于管理, 有较好的扩展性, 适合大规模的网络应用; 簇首经过数据融合处理后再将数据传送给基站, 减少了数据传输量; 分层路由协议能量消耗分布比较均匀, 能有效地延长网络寿命, 平衡网络负载; 在一般层次的分层路由结构中, 从最低层簇内节点将数据传到基站的平均跳数比较

少, 因此传输时延比平面路由协议要小。无线传感器网络分层路由协议的缺点是簇首由于负担较多的数据传输任务, 其能量消耗较快, 一旦簇首能量耗尽或失效, 该区域网络就失效。

典型的无线传感器网络分层路由协议有LEACH协议^[10](Low-Energy Adaptive Clustering Hierarchy), PEGASIS协议^[11](Power-Efficient Gathering in Sensor Information Systems), TEEN协议^[12](Threshold Sensitive Energy Efficient Sensor Network Protocol), HEED协议^[13](Hybrid Energy Efficient Distributed Clustering)以及这些的改进协议等。

1.2.2 无线传感器网络分层路由协议介绍

1. LEACH协议(Low-Energy Adaptive Clustering Hierarchy)^[10]

LEACH协议全称是低功耗自适应分簇聚类路由协议, 它是无线传感器网络中提出的第一个分簇路由协议。LEACH协议将网络中所有传感器节点分成若干个簇, 每个簇内有一个簇首和若干个簇内节点, 这是一种经典的无线传感器网络分层结构, 如图1-5所示。

对于传感器节点和网络模型, LEACH协议作了如下假设:

- 1) 传感器节点和基站Sink节点都是固定的, 每个节点都可以直接和基站通信。
- 2) 传感器节点可以根据与通信节点的距离调整射频发送功率, 射频发射消耗的能量与两节点之间的距离成正比。
- 3) 网络中相邻节点采集的数据都有一定相似性, 可以进行一定的数据融合达到减少数据传输量的目的。

LEACH协议中定义了“轮”的概念, 其操作是按“轮”来进行的, 每轮包含簇的建立和稳定运行两个阶段。为了节省能量开销, 一般稳定运行阶段时间要长于簇建立阶段。

在簇建立的阶段, 首先通过随机机制选举出若干个簇首节点。具体为每个传感器节点产生一个0-1的随机数, 如果该随机数小于阈值 $T(n)$, 则该节点就成为簇首节点。阈值 $T(n)$ 的公式如下:

$$T(n) = \begin{cases} \frac{K}{N - K \lceil r \bmod (N/K) \rceil} & n \in Gr \\ 0 & otherwise \end{cases} \quad (1-1)$$

其中 n 表示某个传感器节点的编号, N 是传感器节点的个数, K 是期望产生的簇头个数, r 是选举簇首的轮数, Gr 是最近的 N/K 轮中还没当选过簇首的节点集合。由公式1-1可以看出, 刚当选过簇首的节点在接下来的 N/K 轮都没有机会再当选节点, 因为其阈值 $T(n)$ 等于0。而

没有当选过簇首的节点的阈值 $T(n)$ 在增大, 因此没有当选过簇首的节点成为簇首的概率在增加。

当节点当选为簇首节点后, 该节点会向网络中所有传感器节点广播自己当选为簇首的消息。其余没有当选簇首的节点接收该广播信息, 每个节点都可能收到多个簇首的广播信息, 节点根据收到各个簇首广播信息的信号强度也就是与簇首的距离决定加入哪个簇, 每个节点选取广播信号强度大的簇首作为自己的簇首, 将自己加入的信息发送给相应的簇首, 簇首再发送信息加以确认, 从而将整个网络分成簇成员个数不相等的若干个簇。

在LEACH协议的稳定运行阶段, 每个簇首统计自己的簇内节点个数, 并为它们确定数据传输的TDMA时隙, 规定每个簇内节点传输数据的时间。在数据传输时, 每个簇内节点按照分配好的时间单跳将采集的数据发送给簇首, 簇首经过数据融合达到减少数据通信量的目的, 然后将融合的数据单跳传输给基站。稳定运行阶段持续一段时间后, 继续下一轮节点选举簇首的过程。

LEACH协议优点如下^[14,15]: 通过随机选举簇首的机制使网络能耗均匀地分布于整个网络的传感器节点之间, 有效延长网络生命周期; 协议算法简单, 比较容易实现; 节点不需要维护太多路由信息, 簇内节点只需要记录簇首节点信息即可; 网络扩展性较好, 有节点加入或退出网络只需要通知相应的簇首节点即可; 簇首经过数据融合再传输给基站, 有效减少了数据传输量。

但LEACH协议也有很多的缺陷, 本文就是以LEACH的分簇结构为网络模型, 针对LEACH的主要缺陷进行改进, 从而提出一种新的无线传感器网络分簇路由协议。LEACH协议有以下几个缺点^[14,15]:

1) LEACH协议最大的问题就是分簇不均匀。由于LEACH协议采用随机的机制选举簇首, 没有当选过簇首的任何位置的节点都可以成为簇首。分簇不均匀的具体表现为: 首先导致簇首个数不确定, 可能出现簇首个数远小于期望簇首数 K 的情况; 簇首可能分布很不均匀, 比如集中某一区域, 这样普通节点可能距离最近的簇首仍然非常远; 簇内节点个数可能很不均衡, 有些簇内节点个数过多的簇首可能迅速死亡。

2) LEACH协议无论簇首位于何处, 都单跳将数据传输给基站, 而无线传感器网络节点的传输能量消耗和节点间距离成正比, 因此如果簇首位置离基站很远, 那簇首势必在传输数据时消耗大量能量, 加速簇首死亡。

3) LEACH协议有很多对能量消耗缺乏考虑的表现。首先簇首阈值公式1-1中没有考虑能量因素, 簇首不但可能位置不合理, 如果选举低能量节点当选簇首, 将导致簇首迅速死

亡；另外LEACH协议簇首选举过程中的大量广播信息也消耗大量能量，而且LEACH协议隔一段时间就要进行所有簇首节点的重选，这样的能量消耗是不值得的。

2. PEGASIS协议(Power-Efficient Gathering in Sensor Information Systems)^[11]

PEGASIS协议全称是传感器信息系统高效能收集协议，它是在LEACH协议的基础上提出的一种“链”式结构的无线传感器网络分层路由协议。PEGASIS协议认为LEACH协议按轮循环的簇结构建立阶段开销大，而且网络中会有若干个簇首和较远的基站通信消耗的能量比较大，另外簇内节点和簇首之间仍然存在通信距离过大的情况。PEGASIS协议提出将网络中所有传感器节点串成一条链(也可以看成是所有节点分成一个簇)，所有的节点只和链结构中的上下跳节点通信，并且在所有节点只选出一个链头节点直接和基站通信，这样的结构相比LEACH协议大大减少了节点间通信的距离，也减少了簇建立和维护的开销。

PEGASIS协议的链式结构如图1-6所示，PEGASIS协议也是每轮产生一条链路结构，并且产生一个链头节点直接和基站通信。其机制是由基站选择一个没有当选过链头的节点成为新的链头，然后按照贪婪算法的思想从该链头节点开始形成一条链，然后数据就从各个源节点按照链的方向将数据传输到链头节点，然后链头节点再把数据传输给基站，传输过程中每个节点都可以进行数据融合。PEGASIS协议数据传输采用Token机制，如图1-6所示，要传输数据时，节点 C_2 将Token沿着链传给 C_0 ， C_0 将数据传给 C_1 ， C_1 将自己采集的数据和 C_0 的数据融合后传给链头 C_2 ， C_2 以同样方式收集 C_4 和 C_3 的数据，最后 C_2 处理融合所有数据后，将数据传输给基站。

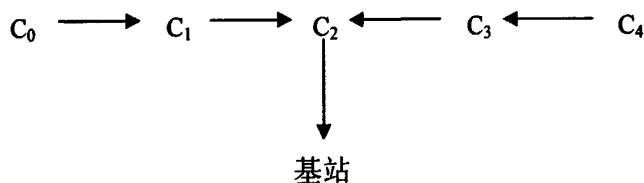


图1-6 PEGASIS协议的链式结构

PEGASIS协议相比LEACH协议有如下优点：由于每个节点在传输数据时都是和自己距离最近的节点通信，因此能量消耗是最小的；所有节点只形成一条链，相比LEACH协议全局选举多个簇首要节约不少开销；每个节点都可以进行数据融合，可以进一步减小数据传输量；PEGASIS协议轮换没有当选过链头的节点当选新的链头，这样可以将能量消耗均匀地分摊于每个传感器节点。就能量消耗的角度来讲，PEGASIS协议比LEACH协议节省能量1-2倍左右。

然而PEGASIS协议的能量高效却是以牺牲另外两个网络重要指标为代价的。首先PEGASIS协议将所有节点串成一条链，其传输时延是相当大的，并且每个节点都可能进行数据融合，将更加大时延。而LEACH协议中每个簇内节点将数据传输到基站只需要两跳即可。第二个因素就是健壮性，由于所有节点都串成一条链，有些关键位置的节点若被损坏比如链头节点，将导致整个网络或者很大部分网络失效。另外PEGASIS协议需要节点获取位置信息。

3. TEEN协议(Threshold Sensitive Energy Efficient Sensor Network Protocol)^[12]

TEEN协议的全称是阈值敏感的高能效的传感器网络协议，它也是在LEACH协议基础上发展而来的一种基于分簇的路由协议。TEEN协议与LEACH协议的实现机制很相似，只不过TEEN协议是响应型协议，而LEACH协议是主动型协议。主动型协议持续监测感知对象并以恒定的速率发送监测数据，而响应型协议只在被监测对象发生突变时才传送数据。TEEN协议的特点是定义了两个概念，硬门限和软门限。硬门限定义了一个门限值，当传感器节点接收到数据高于这个门限值时节点必须打开射频向簇首发送数据。软门限定义了一个门限值，当节点感应到数据的变化值大于这个门限值时节点才打开射频向簇首发送数据。在不发送数据时，传感器节点关闭射频以节约能量。

TEEN协议采用和LEACH协议相同的成簇机制，在每次簇首重建后，簇首还要广播硬门限和软门限的值。TEEN协议的工作过程如下：簇划分好后首先由基站通过簇首向所有传感器节点报告硬门限和软门限的值，然后各个簇内节点开始持续监测感知对象的数据。当某个节点采集到的数据首次大于硬门限值时，该节点打开射频模块向它的簇首发送数据，并且把发送的数据值缓存为一个状态变量SV(State Viable)。然后在后续的监测过程中，只有当节点采集的数据大于硬门限值且这个值和SV的差值大于等于软门限的时候，该节点才将该数据发送给簇首，并且把当前数据值更新为新的SV。

TEEN协议通过设定硬门限值限定了用户感兴趣的数据范围，只有当监测值达到用户感兴趣的数据范围，节点才会向用户发送数据，这可以减少不必要的数据发送量。通过设定软门限值限定了当数据变化到达什么范围时才需要向用户发送监测数据，也可以减少数据发送量。TEEN协议这样的设计是针对无线传感器网络一种类似于报警监测的应用，通过软硬门限值的设定可以大大节省不必要的能量消耗。另外软门限值的设定是可以根据用户的对数据的要求随时改变的。较小的软门限值使数据更加精确但数据传输会变得很频繁导致能量消耗过大，而较大的软门限值可能对监测的灵敏度不够，具体操作时可以根据应用的特点设定合理的软门限值。

TEEN协议通过软硬门限值的设定大大减少了数据传输次数,节省了能量。TEEN协议适合于实时响应性的应用,比如入侵警报,火灾预警等,但它不适合于需要周期性获取监测数据的应用。而且软硬门限值如果设定不合理,可能导致节点虽然采集了大量有用的数据却不能发送的情况。另外由于TEEN协议分簇机制和LEACH协议相同,因此它仍然存在分簇不均匀等严重问题。

4. HEED协议(Hybrid Energy Efficient Distributed Clustering)^[13]

HEED协议的全称是混合高能效的分布式分簇协议,它也是一种基于分簇结构的路由协议。HEED协议认为生命周期,可扩展性和负载平衡是无线传感器网络中最重要的三个需求,并且它的目标是尽量在这个三个需求中达到平衡。

HEED协议的簇首选择机制考虑节点剩余能量这个主因素和簇内通信代价这个次要因素。具有剩余能量多的节点有较大的概率成为候补簇首,而最终节点能否成为簇首取决于该节点的剩余能量是否比周围的节点多。簇内通信代价用于确定落在多个簇范围内的节点最终属于哪个簇,以及平衡簇首之间的负载。考虑到分簇后簇内的通信开销,HEED以AMRP(簇内平均可达能量)作为衡量簇内通信代价的标准。

HEED的簇头选择算法具有以下特点:完全分布式的簇头产生方式;簇头产生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,并簇头的选择过程考虑多个约束条件。另外HEED在簇头选择标准和簇头竞争机制上都与LEACH也不同。实验结果表明,HEED协议不但分簇速度快,而且簇首分布比较均匀,整个网络拓扑结构合理。

1.2.3 LEACH协议的改进

LEACH协议是无线传感器网络中提出来的第一个经典的分簇路由协议,对无线传感器网络分层路由协议的发展具有指导意义。但是正如2.3节所描述的那样,LEACH协议具有很多严重的缺陷,比如分簇不均匀,簇首与基站单跳通信消耗大量能量等等,这些都有许多值得研究和改进的地方。LEACH协议的分簇结构是无线传感器网络中一种经典的分层结构,再加上LEACH协议具有的一些缺陷,于是自LEACH协议提出以来国内外就涌现出很多基于LEACH的分簇结构并且针对LEACH缺陷进行改进的研究成果。由于本文也是基于无线传感器网络分层的网络结构并且针对LEACH的缺陷提出一种新的分簇路由协议,因此本文在此首先对近年来国内外针对LEACH的研究成果作一个总结。

1. 簇首阈值公式改进

在对LEACH协议的改进方法中，有一类研究专门针对LEACH协议的簇首阈值公式进行改进。

M.J.Handy等人^[16]对LEACH协议的簇首阈值公式进行改进，它将节点的剩余能量因素考虑进去，在选择簇首时剩余能量少的节点当选簇首的概率降低。M.J.Handy等人对阈值公式改造如下：

$$T(n) = \frac{P}{1 - p \left(r \bmod \frac{1}{P} \right)} \left[\frac{E_{cur}}{E_{max}} + \left(r_s \operatorname{div} \frac{1}{P} \right) \left(1 - \frac{E_{cur}}{E_{max}} \right) \right] \quad (1-2)$$

其中 P 是簇首所占所有节点百分比， E_{cur} 是现在剩余能量， E_{max} 是初始能量， r_s 是该节点到目前为止还没当选簇首的轮数。 r_s 从1开始到 $1/p$ 为止，然后从1开始变化，当 r_s 的值变为 $1/p$ 时，LEACH协议的簇首阈值公式又变为公式1-1，即如果连续 $1/p$ 轮该节点都没有当选过簇首，则认为它的能量由于没有当选过簇首则剩余能量充足，让它以一个比较高的概率当选簇首。M.J.Handy等人提出的这个方法在一定程度上避免了低能量节点当选簇首，但由于其概率性的特点仍具有随机性，因此仍然有可能选取低剩余能量节点当选簇首。而且该方法只考虑了剩余能量因素，其余LEACH协议的缺点都没有改进。

胡钢等人^[17]也提出对LEACH协议的簇首阈值公式进行改进，他们不但考虑了传感器节点的剩余能量，还考虑了传感器节点周围的邻居节点个数。他们将公式改进如下：

$$T(n) = \frac{P}{1 - p \left(r \bmod \frac{1}{P} \right)} \left[\frac{E_{cur}}{E_{max}} + \left(1 - \frac{E_{cur}}{E_{max}} \right) \frac{neighborNum}{P(chTimes + 1)} \right] \quad (1-3)$$

其中与公式(1-2)中同名的变量具有相同的意义， $neighborNum$ 是当前传感器节点周围的邻近节点数目， $chTimes$ 表示当前节点在以前轮次中充当簇首节点的次数。从公式(1-3)中可以看到，节点剩余能量多的有较高的概率当选簇首；而当前节点若邻近节点较多，则以该节点为簇首的簇内节点能更紧密围绕在簇首周围更加节省能量，因此也有较高概率当选簇首；另外在前几次轮次中当选簇首次数较少的节点也有更高概率成为簇首，这也是将能量消耗均衡分布在整个网络中思想的体现。胡钢等人的改进方法不但考虑了节点的剩余能量，还考虑了让在节点密集的区域选出簇首从而使簇中的节点能比较集中。但是该改进方法仍然不能改变LEACH协议的簇首选取随机性的缺陷，在簇首选举过程中仍然可能出现簇首个数过少，以及簇首集中某一区域等簇首位置不合理的情况。

2. 簇首与基站间的单跳传输改进

由于基站多数情况下离无线传感器网络监测区域较远,因此LEACH协议中簇首与基站间单跳传输会消耗大量的能量,加上LEACH协议的簇首位置不合理的现象将更加加重这一缺点,因此对LEACH协议的改进中有一部分是专门针对簇首与基站间的单跳传输机制进行改进的。

王国芳等人^[18]提出了一种以节点剩余能量为参数限制中间节点选择的簇首多跳算法。该算法在生成包括基站在内的所有簇首节点的最小生成树的同时,考虑作为中间节点的簇首的剩余能量,使簇首之间形成一个多跳的路径与基站通信。

该算法的簇形成过程和LEACH协议一样,在簇首确定之后,将各个簇首间的拓扑关系看成是一个无向连通图 $G(V,E)$,其中 V 是所有簇首与基站的节点集合, E 是 V 中任意两个节点所组成的边的集合, E 中将任意两点间的距离作为其边的权值。然后对无向连通图 G 运用Prim最小生成树算法计算出以基站为根节点包含所有簇首在内的最小生成树,形成各个簇首到基站的多跳路径。

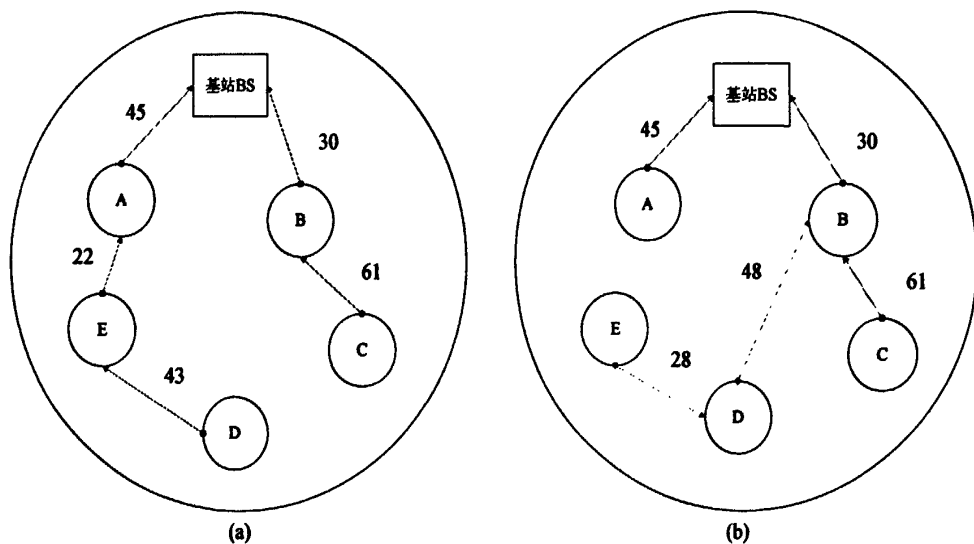


图1-7 簇首与基站的最小生成树算法

如图1-7中(a)所示,簇首B与基站BS距离最小,则节点B就成为基站的候补子节点,还要继续判断B节点的剩余能量是否大于原始能量的25%,若是则簇首B的下一跳就是基站,否则应该选择除B外的簇首作为基站的子节点。依次类推,直到形成整棵最小生成树。该算法还随节点剩余能量调整簇间多跳路径。如图1-7中(b)所示,当簇首A的剩余能量小于一定值时,它不能继续作为中间节点承担转发数据的任务,算法调整路径,可以看到虽然E簇首离簇首A更近,可是簇首E仍然选择簇首为它的下一跳节点,然后D也相应地调整路径。

王国芳等人提出的算法大大减少了簇首为了传输数据给基站所消耗的能量，而且该算法使用剩余能量感应来动态调节簇首间的多跳路径，这确实是改进了LEACH协议簇首与基站间单跳传输消耗大量能量的缺点。然而最小生成树目的是生成连通所有簇首且距离总和最小，而改进目标应该是某一个簇首将数据传输到基站所耗费能量最小，因此该改进算法并不很符合改进目标。而且它对LEACH的其他缺点没有进行改进。

3. LEACH-C协议与LEACH-F协议

Heinzelam.W在他的博士论文中提出了LEACH-C协议和LEACH-F协议^[19]，和标准LEACH协议不同，这两种协议都采用集中式的方法选举簇首，也就是基站通过某种算法计算出合理的簇首，而不是像LEACH协议那种由节点自己决定是否成为簇首的分布式算法。

LEACH协议是通过一种随机的方式选举簇首，也就是完全没有任何依据地选择簇首，因此会出现分簇不均匀的现象。LEACH-C协议根据全局节点信息来选举簇首，可以有效地避免LEACH协议的缺点。LEACH-C协议中每个传感器节点把自己的地理位置和当前剩余能量报告给基站。基站首先根据所有传感器节点报告的剩余能量计算出一个节点平均剩余能量，节点剩余能量低于这个平均值的时候不能当选簇首。然后在符合条件的节点中挑选出合适数量和最优地理位置的节点当选为簇首，这是一个NP难问题。LEACH-C协议中采用簇内所有成员节点到簇首距离总和最小的原则，采用模拟退火算法^[20]计算出全局最优的簇首，最后由基站把簇首的选举结果广播出去。

LEACH-F协议是在LEACH-C协议的基础上又做了一些改进。LEACH-F协议在簇形成阶段和LEACH-C协议是一样的，也是由基站获取所有传感器节点的剩余能量和节点位置信息后由基站运行模拟退火算法根据全局节点位置信息计算簇首的位置。在簇结构确立以后，LEACH-F协议为每个簇生成一个簇首“值班”列表，该列表中列出所有簇内节点和簇首并指出该簇中节点轮流当选簇首的顺序。一旦簇形成后，簇的分布就不再改变，只是每个簇由一个簇内节点组成的“值班”顺序轮流当选簇首。

LEACH-C协议与LEACH-F协议采用一种集中式的算法根据全局信息计算簇首，相比LEACH协议随机性的选举簇首机制，集中式的算法能够产生比较合理的簇首和簇斤结构，从而延长网络生命周期。LEACH-F协议相比LEACH-C协议的优点是一旦簇结构决定就不再变化，这样可以减少每轮循环建立簇所带来的开销。但是LEACH-F协议并不适合真实的网络环境，因为它不能动态处理传感器节点加入失效的情况，另外还增加了簇间的干扰。

LEACH-C协议与LEACH-F协议这种集中式选举簇首的算法还是很有借鉴意义，它不但能根据全局信息选出合理的簇首，而且大部分工作由能量无限的基站来操作，也大大减

少了LEACH协议建簇过程带来的能量消耗。本文就是采用这种集中式算法的思想,结合群智能算法研究领域,提出了一种基于集中式选举簇首的智能无线传感器网络分簇路由协议。

4. 基于群智能算法改进LEACH协议

现在国内外的研究成果中也有不少利用群智能算法改进LEACH协议缺点的成果。

梁英等人^[21]提出基于粒子群算法来优化分簇结果,该方法主要思想是首先用LEACH算法进行初步分簇,然后依据节点自身能量和周围节点能量分布利用粒子群算法在已经用LEACH协议分好的簇内优化分簇的效果。该方法将无线传感器网络分簇分为四个阶段:

1) 簇的初步形成阶段:首先采用与LEACH协议一样的选举簇首方法选出候补簇首,而形成的簇结构将不会改变,而最终簇首的确定将根据粒子群算法在形成的簇内节点和候补簇首中选择。

2) 候补簇首收集簇内节点信息阶段:在初步形成的簇里的簇内节点将各自个位置和剩余能量信息发送给候补簇首。这时候补簇首存储了簇内节点位置信息 $P\{p_1, p_2, \dots, p_n\}$ 和剩余能量信息 $E\{e_1, e_2, \dots, e_n\}$ 。 p_i 和 e_i 代表第 i 个节点的位置和剩余能量, n 表示有 n 个节点。

3) 利用粒子群算法确定簇首阶段:该阶段确定粒子群算法的适应度函数,并且利用候补簇首收集的簇内节点的位置信息和剩余能量信息,由候补簇首计算出当前簇内最优的簇首位置作为该簇内最终的簇首。该算法是认为当选簇首的节点不但自身要有充足的剩余能量,且离簇首远的节点剩余能量应该较大,离簇首近的可以具有较小剩余能量。在该思想下,算法提出粒子群算法的适应度函数如下:

$$f(k) = x e_k + \frac{y}{n-1} \sum_{i=1, i \neq k}^n \frac{e_i r_i}{r_i + 1} \quad (1-4)$$

其中 k 为在当前粒子群算法迭代中选中作为簇首的节点, e_k 为当前节点的剩余能量, r_i 为本簇内第 i 个节点与当前簇首节点 k 的距离, e_i 为节点 i 的剩余能量。 $\frac{e_i r_i}{r_i + 1}$ 为第 i 个节点的等效能量,满足若距离簇首 k 较远并且节点 i 的剩余能量较大则等效能量较大;若距离簇首 k 较近节点 i 的剩余能量较小则等效能量较大。 n 为该簇的簇内节点个数, x 和 y 为权值参数,满足 $x+y=1$ 。以上述适应度函数进行粒子群算法迭代,迭代中将较大的适应度值作为较优值,最终迭代结束选出该簇内适应度函数最大的节点作为该簇的簇首。

4) 优化簇的形成阶段:由候补簇首计算得出本簇内最优簇首后将该信息发布给簇内节点和新簇首节点(如果不是候补簇首本身),最终形成优化簇首的簇。

梁英等人提出的基于粒子群算法对用LEACH协议分好的簇内来优化簇首,粒子群算法

中以能量分布作为考虑因素，该方法在一定程度上能选取出能量分布合理的簇结构。但是对解决LEACH协议分簇不均匀这个最大缺点这个方法并没有什么作用，因为它是先用LEACH协议分完簇后再进行簇首位置的调整，而且它的着眼点只在于能量的分布合理性，并没有涉及簇首位置的合理性。另外由簇首进行粒子群算法的迭代运算是相当耗费能量的。

杭海存等人^[22]通过将LEACH算法与蚁群算法相结合提出一种新的路由算法增加了网络中数据包的发送量，分散节点的能量消耗，均衡网络的能量消耗，延长了网络的生命周期。和LEACH协议一样，该算法也分为簇形成阶段和稳定运行阶段。在簇的形成阶段，首先节点产生一个随机数，该随机数若小于阈值，节点就成为簇首，并发送簇首公告信息，若随机数大于阈值，节点就监测簇首公告信息并根据所信号的强弱来决定所加入的簇，并向相应的簇首发送加入信息。在簇的稳定阶段，簇首在接收数据的同时，以该簇首节点为起点，基站作为终点，运行蚁群算法，找到由簇首到基站的最优路径，通过中间节点的转发，更好地实现多跳性能。

杭海存等人提出的算法利用蚁群算法全局寻优的能力，在用LEACH协议分完簇后，使用蚁群算法找到从各个簇首节点为源节点，基站为目的节点，并且通过各个簇内节点或者簇首节点的最优路径。该算法可以找到从簇首节点到基站的能量消耗最小的路径，但是却一定程度上增加了时延，而且LEACH协议的本质缺陷也没有解决。

1.3 本文的研究内容和组织结构

本文以设计节能高效简单的无线传感器网络路由协议为目标，在充分研究和分析现有无线传感器网络路由协议的基础上，结合群智能算法，以无线传感器网络经典分簇路由协议 LEACH 协议为基本模型，提出了一种基于群智能算法的无线传感器网络智能分簇路由协议 SI-CRP 协议(Swarm Intelligence—Clustering Routing Protocol)。本文的主要研究内容和创新点如下：

- 1) 分析现有无线传感器网络经典分簇路由协议的特点，并对它们进行分析。特别是针对LEACH协议存在的使能量消耗过大的几个问题进行详细地分析。
- 2) 介绍和分析两种群智能算法粒子群算法PSO(Particle Swarm Optimization)和蚁群算法ACO(Ant Colony Optimization)的研究成果。
- 3) 针对LEACH协议分簇不均匀的缺陷，改造粒子群算法，本文提出用改造后的离散粒子群算法直接计算全局最优的簇首，使分簇均匀合理。

- 4) 针对LEACH协议簇首与基站单跳通信耗费大量能量的缺陷, 本文提出用改造的蚁群算法构造簇首到基站间的多跳传输路径。
- 5) 针对LEACH协议按轮全局重新选举簇首带来的能耗过大的问题, 本文提出基于能量自感应机制的局部簇首选举机制。
- 6) 详细阐述基于群智能算法的智能分簇路由协议SI-CRP的主要技术和实现细节, 并通过仿真实验从各方面比较本文所提出SI-CRP协议和LEACH协议。

本文一共分为五章, 具体组织结构如下:

第一章: 绪论。首先简单介绍了无线传感网络的概念、结构、关键技术等, 然后介绍和分析无线传感器网络分层路由协议特别是详细分析了 LEACH 协议和国内外针对 LEACH 协议的改进研究成果。

第二章: 群智能算法。详细介绍粒子群算法和蚁群算法, 并对它们的相关研究成果进行介绍和分析。

第三章: 基于群智能算法的无线传感器网络分簇路由协议。详细介绍本文提出的 SI-CRP 协议的主要创新点和技术内容。

第四章: SI-CRP 协议的实现与仿真。首先介绍了 SI-CRP 协议的实现流程, 然后通过基于 MATLAB 的仿真实验从分簇效果和协议整体运行能耗与节点生命周期这几个方面比较了 SI-CRP 协议和 LEACH 协议的性能。

第五章: 总结和展望。首先总结了本文所做的工作和研究意义, 然后在展望中提出了 SI-CRP 协议存在的限制性因素和值得改进的地方作为以后工作的努力方向。

第2章 群智能算法

群智能算法(Swarm Intelligence Algorithm)作为一种新型的仿生类进化算法已经成为国内外学者研究的热点领域。一些生物学家通过观察和研究发现,诸如鸟群鱼群和蚁群等生物虽然个体智能不高,行为简单,但由一个个简单个体组成的群体却能拥有复杂而有序的群体智能行为。群智能算法就是模拟自然界生物群体的这种行为而构造的随机搜索方法。它强调个体行为的简单性,群体的涌现特性以及自上而下的研究策略,并将其演化成为一种很好的优化方法。与大多数优化算法不同,群智能算法是基于概率搜索的算法。群智能算法实现简单,具有较强的鲁棒性,在组合优化、网络路由、函数优化、数据挖掘、机器人路径规划、无线传感器网络性能优化领域等领域得到了广泛的应用,并且取得了较好的效果。已有研究表明群智能算法为解决许多应用问题提供了新的途径和方法。

目前,群智能算法研究领域有两种主要算法:粒子群优化算法 PSO(Particle Swarm Optimization)和蚁群优化算法 ACO(Ant Colony Optimization)。本文也正是将这粒子群优化算法和蚁群优化算法经过改造后应用于本文提出的无线传感器网络分簇路由中的寻优问题。

2.1 粒子群优化算法

粒子群优化PSO算法起源于对鸟群群体行为运动方式的观察和研究。研究者发现鸟群在飞行过程中经常会突然的改变方向,时而散开时而聚集,其行为不可预测,但其整体总能保持一致性,个体与个体间也保持着最适宜的距离。研究者通过研究发现原来群体生物中存在一种社会信息共享机制,它为群体的进化提供了一种优势,这就是后来粒子群优化算法形成的基础。

基本粒子群优化算法由Kennedy和Eberhart博士在1995年首次提出^[23],其引入了近邻速度匹配、多维搜索及距离加速概念,形成了粒子群算法的最初版本。Shi和Eberhart在1998年引入了惯性权重的概念^[24],从而实现了全局搜索与收敛速度的平衡,形成了现在粒子群优化算法的标准版本。粒子群算法一经提出就引起广泛关注,短短几年时间里,许多学者在基本粒子群算法的基础上,就参数设置、群体多样性、算法离散化等多个方面进行改进,使粒子群算法的研究迅速发展,成为目前智能计算领域的研究热点之一。

粒子群优化算法已经被证明是一种解决工程领域NP难问题的有效的解决方法^[25-27]。近年来将粒子群算法应用于无线传感网络领域的例子也不少，粒子群算法相比于其他算法收敛速度更快，全局搜索效率更高^[21,28,29]。

2.1.1 标准粒子群优化算法

粒子群优化算法中每个优化问题中的可能解都被抽象成d维搜索解空间上一个粒子(Particle)，每个粒子都有一个由目标适应度函数决定的适应度值。和其他进化算法类似，粒子群优化算法也是利用该适应度值的大小为依据进化。每个粒子都以一定的速度在解空间飞行搜索，该速度将决定粒子在解空间的飞行的方向和位置。粒子的速度是根据该粒子自身的最佳飞行经验和其它粒子的最佳飞行经验来动态地调整。粒子自身的最佳飞行经验称为个体最优解，其它粒子的最佳飞行经验称为全局最优解。整个粒子群体在解空间中不断搜索最优解，在搜索过程中不断调整自己的飞行速度，直到找到最优解算法结束。

假设一个优化问题解空间的维数为d维，则任一个粒子i在某次迭代中也就是粒子i的位置可以表示为 $x_i = (x_{i1}, x_{i2}, \dots, x_{id})$ 。所谓解空间的维数，就是为达到问题的目标相关参数的个数，而粒子i的位置也就是到达问题目标时d个相关参数的一个解。根据优化问题的目标，粒子群算法首先提出一个适应度目标函数 $fitness=f(x_i)$ ， x_i 即是粒子i的位置 $x_i = (x_{i1}, x_{i2}, \dots, x_{id})$ 。一般优化问题的目标是使适应度目标函数fitness达到最小值或者最大值，粒子群算法就是要找到这个极值，并且找到当目标函数fitness达到这个极值时，d个相关参数的解。标准粒子群算法不但定义了d维的粒子位置，还定义了d维的粒子速度 $v_i = (v_{i1}, v_{i2}, \dots, v_{id})$ 。每个粒子在算法迭代中就是不断根据个体最优解和全局最优解以及自己历史的位置来更新自己下一轮迭代中的飞行速度，并由这个更新速度得出新的粒子位置。粒子群算法在每轮的迭代中根据以下两个公式来更新粒子的速度和位置的：

$$v_{id}(t+1) = wv_{id}(t) + c_1r_1(P_{id}(t) - x_{id}(t)) + c_2r_2(P_{gd}(t) - x_{id}(t)) \quad (2-1)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \quad (2-2)$$

公式(2-1)和(2-2)的 x_{id} 和 v_{id} 即代表d维的粒子位置和粒子速度。 t 代表当前迭代轮数， $t+1$ 就是下一代迭代轮数。 $P_{id} = (p_{i1}, \dots, p_{id})$ 代表的是d维的粒子历史的个体最佳位置， $P_g = (p_{g1}, \dots, p_{gd})$ 代表的是d维的整个粒子群历史的全局最佳位置。 w 称为惯性权重，它控制当前轮迭代的速度对下一轮迭代速度的影响程度。 C_1 称为个体学习因子，它控制粒子当前的飞行位置受个体历史最优位置的影响程度。 C_2 称为社会学习因子，它控制粒子当前的飞行位置受粒子群全局最优位置的影响程度。 r_1 和 r_2 是两个随机数。

从公式(2-1)中可以看到,粒子的速度更新受三部分因素影响。第一部分是 wv_{id} ,它表示下轮速度对当前速度的继承,可以平衡算法快速收敛和全局收敛。第二部分是 $C_1r_1(P_{id}(t)-X_{id}(t))$,该部分是对自身飞行经验的学习,向自身历史最优位置靠拢,可以增强全局搜索能力,避免局部最小。第三部分是 $C_2r_2(P_{gd}(t)-X_{id}(t))$,该部分是社会学习部分,是对群体飞行经验的共享。三部分共同作用使粒子群迅速向最优解的位置靠近。

粒子群优化算法流程如下^[30]:

- 1) 初始化一个种群规模为 N 的粒子群(即所有粒子的个数为 N),随机初始化每个粒子的速度与位置。设定每个粒子个体历史最优位置和全局最优位置为随机初始化的粒子位置。然后根据优化问题目标得出粒子群算法的适应度目标函数。
- 2) 开始粒子群算法迭代过程,根据粒子位置计算每个粒子的适应度函数值。
- 3) 将每个粒子的适应度目标函数值和个体历史最优适应度目标函数值比较,如果该轮的适应度函数值好于个体历史最优适应度函数值,更新个体历史最优位置和个体历史最优适应度函数值。
- 4) 比较所有粒子的适应度函数值,得出本轮全局最优适应度函数值和全局最优位置,如果本轮全局最优适应度函数值好于全局历史适应度函数值,更新全局最优位置和全局最优适应度函数值。
- 5) 根据公式(2-1)和(2-2)更新每个粒子下一轮迭代的速度和位置。
- 6) 判断是否达到最大迭代轮数,是则算法结束,否则更新迭代轮数并转第2)步。

图2-1显示了标准粒子群算法的流程图。

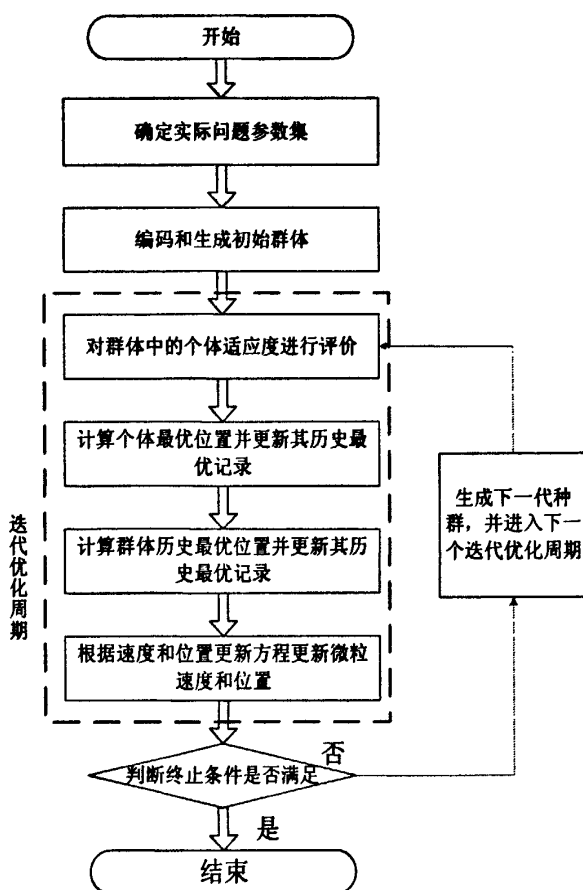


图 2-1 标准粒子群算法流程图

2.1.2 粒子群算法惯性权重

粒子群优化算法速度更新公式(2-1)中的惯性权重 w 是一个重要的参数, 是Shi和Eberhart在1998年为了改进基本粒子群算法而引入的。惯性权重的引入主要是为了粒子群算法早熟和收敛停滞问题, 使其有扩展搜索区域的能力, 这样可以平衡算法的全局搜索能力和局部搜索能力。惯性权重 w , 个体学习因子 C_1 和社会学习因子 C_2 的取值大小对粒子群算法的运行效率和结果有较大的影响。当惯性权重 w 取值较大时, 粒子群算法的全局搜索能力较强; 而当惯性权重 w 取值较小时, 粒子群算法的局部收敛性较好^[31]。

目前对惯性权重取值大小的研究主要可以分为两类: 固定惯性权重和动态惯性权重。固定惯性权重就是指在粒子群算法迭代过程中惯性权重 w 取值为一个固定常数, 始终不变。动态惯性权重选定惯性权重 w 取值的变化范围, 在粒子群算法迭代过程中根据某一规律动态调整惯性权重 w 的取值。固定惯性权重使粒子群算法在迭代过程中始终具有相同的搜索能力, 而动态惯性权重可以根据具体情况改变搜索的策略。

关于固定权重研究领域, 主要目标就是找到粒子群算法迭代时, 找到粒子群算法迭代过程中最适合的惯性权重 w 的取值。Eberhart和Shi经过实验证明得出结论当惯性权重

$w=0.729$, $c_1=c_2=1.494$ 时, 粒子群算法性能最优^[32]; Trelea得出的结论是当 $w=0.6$, $c_1=c_2=1.7$ 时粒子群算法性能最优^[33]; 张丽平应用遗传算法对标准PSO算法的三个参数进行了优化, 并就10位的Rosenbrock函数和Griewank函数进行了仿真, 得出结论, 对于Rosenbrock函数, 当 $w=0.659$, $c_1=2.796$, $c_2=1.336$ 时, 算法性能最优, 而对于Griewank函数, 当 $w=0.613$, $c_1=2.456$, $c_2=2.260$ 时算法性能最优^[31]。

动态惯性权重相比固定惯性权重更加灵活, 能在迭代过程中动态地调整惯性权重的大小, 使粒子群算法表现的更优异, 因此动态惯性权重研究的成果更多一些。

Shi和Eberhart提出了一种基于递减惯性权重的粒子群优化算法^[34]。他们令速度更新公式(3-1)中的 $c_1r_1=\Phi_1$, $c_2r_2=\Phi_2$, $\Phi_1 \in [0, 2]$, $\Phi_2 \in [0, 2]$ 。另外他们对标准粒子群算法最大的改进是定义惯性权重 w 是一个随迭代轮数线性递减的函数, 该函数表达式如下所示:

$$w = w_{\max} - (w_{\max} - w_{\min}) * \frac{iter}{\max iter} \quad (2-3)$$

其中 w_{\max} 是开始迭代的 w 值, w_{\min} 是迭代结束的 w 值, $iter$ 是当前迭代的轮数, $\max iter$ 是最大的迭代轮数。在文献[34]中, Shi和Eberhart取 $w_{\max}=0.9$, $w_{\min}=0.4$, 也就是随着迭代轮数的增加, 惯性权重 w 的值从0.9递减到0.4。

文献[35]中Zheng等人又提出了惯性权重随迭代轮数递增的粒子群算法。他们取 $c_1r_1=\Phi_1$, $c_2r_2=\Phi_2$, $\Phi_1 \in [0.5, 2]$, $\Phi_2 \in [0.5, 2]$, 惯性权重 w 的函数定义如下:

$$w = w_{\min} + (w_{\max} - w_{\min}) * \frac{iter}{\max iter} \quad (2-4)$$

Zheng等人也取 $w_{\max}=0.9$, $w_{\min}=0.4$, 即在迭代过程中惯性权重 w 的值从0.4递增到0.9。

Shi和Eberhart提出的随迭代轮数的惯性权重调整方法在粒子群算法迭代前期具有较好的全局搜索能力, 迭代后期收敛性也较好, 但缺点是收敛速度慢。而Zheng等人提出递增方法虽然前期收敛速度很快, 但是容易陷入局部最优, 而且迭代后期收敛性也不好。为改进以上两个缺点, 文献[36]中崔红梅等人提出具有折线先增后减的动态惯性权重调整方法, 这样粒子群算法前期具有较快的收敛速度, 后期收敛性也较好。他们定义函数使惯性权重在前一半迭代轮数从0.4递增到0.9, 后一半迭代轮数再从0.9递减到0.4。该方法改进了递增惯性权重后期收敛性差的特点, 然而前期惯性权重小容易使粒子群陷入局部最优。

文献[34-36]中都是基于线性调整惯性权重的方法, 也有不少研究成果是基于非线性调整惯性权重的, 这种方法相比于线性方法更加灵活。

王丽等人^[37]引入了递减指数和迭代阈值的概念对线性递减惯性权重的策略进行了改

进, 在粒子群算法迭代过程中, 惯性权重随当前迭代轮数, 递减指数和迭代阈值非线性变化, 惯性权重的动态调整公式如下:

$$w = \left(\frac{r-1}{r_T-1}\right)^\lambda (w_f - w_i) + w_i \quad (2-5)$$

其中 λ 为递减指数, r 为当前迭代轮数, r_T 为迭代阈值, w_i 为初始惯性权重值, w_f 为达到迭代阈值时的惯性权重值。当迭代轮数达到迭代阈值 r_T 时令 $w=w_f$, 并保持到迭代结束。由于递减指数 λ 的引入, w 随着迭代轮数的增加非线性递减, 有利于避免局部最优。迭代前期惯性权重 w 较大, 有利于粒子群加大搜索范围至整个解空间。随着迭代的进行, 惯性权重 w 非线性减小, 粒子群渐渐向全局最优解靠近。到迭代末期迭代轮数到达迭代阈值时惯性权重被限定为 w_f , 粒子群以较小的惯性权重 w 迅速收敛到全局最优解。

刘建华等人^[38]提出了一种根据不同粒子距离全局最优粒子的距离动态调整惯性权重的方法, 该文献中惯性权重 w 按公式(2-6)调整。公式(2-6)中, L_{ig} 表示第 i 个粒子与全局最优粒子之间的距离, 参数 L_{max} 与 L_{min} 最大距离和最小距离, 它们随着问题不同取值不同。 w_{max} 表示预设的最大的惯性权重 w 值, w_{min} 表示预设的最小的惯性权重 w 值, r_{max} 是最大的迭代轮数, r 表示当前的迭代轮数。

$$w = \begin{cases} w_{max} & L_{ig} > L_{max} \\ w_{max} - \frac{(L_{ig} - L_{min})}{(L_{max} - L_{min})} \frac{r}{r_{max}} (w_{max} - w_{min}) & L_{min} < L_{ig} < L_{max} \\ w_{min} & L_{ig} < L_{min} \end{cases} \quad (2-6)$$

韩江洪等人^[39]根据群体早熟收敛程度和粒子个体适应度函数值自适应地调整粒子的惯性权重, 使粒子群在进化过程中保持惯性权重的多变性。惯性权重 w 按照以下公式动态调整:

$$w(r) = \begin{cases} w(r) - (w_{max} - w_{min}) \left| \frac{f_i - f_{avg}}{f_m - f_{avg}} \right| & f_i > f_{avg} \\ w(r) & f_{avg} < f_i < f_{avg} \\ 1.5 - \frac{1}{1 + k_1 \exp(-k_2 \Delta)} & f_i < f_{avg} \end{cases} \quad (2-7)$$

公式(2-7)中 f_i 为第 r 轮迭代中第 i 个粒子的适应度函数值, f_{avg} 为所有粒子适应度函数值的平均值, f_{avg}^* 为适应度函数值优于 f_{avg} 的粒子的适应度函数值的平均值, $\Delta = |f_m - f_{avg}|$,

f_m 为最优粒子的适应度函数值, Δ 用来评价粒子群的早熟收敛程度, Δ 越小越说明粒子群趋于早熟收敛。 k_1 和 k_2 是两个参数, 该参数的数值选择对算法性能影响较大, 文献[39]中取 $k_1=1.5$, $k_2=0.3$ 。韩江洪等人提出的方法在粒子群算法的全局收敛性和收敛速度之间取得了个折中。

2.1.3 离散粒子群算法

粒子群算法最初被用于求解连续优化问题, 但在实际工程应用领域中许多优化问题的模型都是基于离散模型的, 这使得标准粒子群算法无法运用于这些基于离散模型的优化问题。因此不少国内外学者对标准粒子群算法进行离散改造, 提出了一些离散粒子群 DPSO 算法(Discrete PSO)以满足解决基于离散模型优化问题的需要。

在现有的研究成果中, 对粒子群算法的离散改造基本可以分为两类^[40]: 一种称为基于连续空间的 DPSO, 它是针对特定问题将离散空间映射到连续空间, 在计算中仍保留粒子群算法中速度位置更新的公式和运算规则; 另一种称为基于离散空间的 DPSO, 是以粒子群算法思想和速度位置更新公式为基础, 针对不同问题重新定义不同的粒子离散结构和速度位置更新公式的运算规则。前者是将离散问题映射到粒子连续运动空间后, 在连续空间中计算, 而后者是将粒子群算法映射到离散空间, 在离散空间中计算。

1. 基于连续空间的 DPSO 算法

粒子群算法创始人 Kennedy 和 Eberhart 在基本粒子群算法基础上提出了针对 0-1 规划问题的 BPSO 算法(Binary PSO), 并建立了不同基本粒子群算法的新计算模式^[41]。在 BPSO 算法中粒子位置的每一维 x_{id} 限制为 1 或者 0, 对速度 v_{id} 则不作这种限制。使用粒子速度更新粒子位置时 v_{id} 越大粒子位置 x_{id} 越可能为 1, v_{id} 越小则 x_{id} 越可能为 0。BPSO 算法的状态更新公式如下:

$$x_{id}^{k+1} = \begin{cases} 1, & p_{id} < sig(v_{id}^{k+1}) \\ 0, & \text{其他} \end{cases} \quad (2-8)$$

$$sig(v_{id}^{k+1}) = \frac{1}{1 + \exp(v_{id}^k)} \quad (2-9)$$

其中 sig 函数可以保证粒子位置 x_{id} 的每个分量都在 [0,1] 之间, p_{id} 为预定阈值一般是个随机数。BPSO 算法其他部分和标准粒子群算法相同。文献[42]的实验中, 对于大多数测试函数, BPSO 算法都比遗传算法快, 尤其在问题维数增加时。BPSO 算法并不直接优化二进制变量本身, 而是通过优化连续变化的二进制变量为 1 的概率, 达到间接优化离散变量

的目的。

由于BPSO算法不能适用于许多离散优化组合问题中存在序结构表达和约束条件等问题,一些学者又提出了对粒子位置的近似取整策略,即对迭代产生的连续解进行取整运算并评价解的质量,常采用向上取整和舍尾取整^[43-45]。这类算法的核心就是对粒子位置进行取整计算,取整策略是粒子群算法一种直接离散化方式,算法其余部分和标准粒子群算法相同。

由于基于连续空间的DPSO算法生成的连续解与整数规划问题的目标函数值之间存在多对一的映射,因此该目标函数不能完全反映连续解的质量。另外整数规划问题的连续化导致大量冗余解空间和冗余搜索,从而影响算法的收敛速度。

2. 基于离散空间的DPSO算法

基于离散空间的DPSO算法往往需要根据优化问题的特点构建相应的粒子结构,并对标准粒子群算法中的粒子速度位置更新公式进行重定义来求解。

Clerc针对旅行商问题TSP(Traveling Salesman Problem)提出了基于离散空间的TSP-DPSO算法^[46]。在该算法中,粒子的位置被表示成所有城市的一个排列,所有的排列就构成了问题的搜索空间。该算法中还定义了交换子和交换序的概念, $S=swap(j,k)$ 就称为一个交换子,它的含义是交换粒子位置中编号为j与编号为k的元素,由一组特定顺序的交换子组成的集合称为一个交换序如 $SS=\{swap(j,k), swap(i,y), \dots swap(a,b)\}$ 。而该算法就将粒子的速度定义为达到目标状态所需要的当前状态执行的交换序。基于这些概念,该算法重新定义了标准粒子群算法中粒子速度位置更新公式中的加减操作,以及速度与随机数相乘的操作,实现了粒子群算法向离散空间的映射。算法中还引入了基本交换序的概念,即在产生相同新解的等价交换序中,交换子最少的交换序列称为基本交换序。在粒子群算法运行过程中每次粒子速度被更新后就要重新计算速度V的基本交换序,作为粒子新的历史速度,以避免由于算法的迭代造成速度记录列表中速度数据太多。本文提出的基于群智能算法的无线传感网络路由协议中,基于离散粒子群算法直接计算最优簇首就是借鉴Clerc的思想,针对分簇问题对标准粒子群算法进行离散改造,本文在第四章会详细介绍这一过程。

文献[47]中作者提出了针对0-1规划问题提出的离散二进制PSO,记为B-PSO,以与基于连续空间的BPSO相区别。B-PSO算法中,定义粒子的位置和速度为由0和1组成的同维度的矢量,矢量间的加减法定义为对二进制位的“异或”操作,矢量间的乘法定义为二进制位的“与”操作,从而构造了一种基于离散空间的新型二进制PSO算法。该论文中还借鉴

了免疫算法的思想以避免粒子群算法陷入局部最优。实验表明B-PSO算法比基于连续空间的BPSO算法和遗传算法的效率都高。与基于连续空间的BPSO算法不同，B-PSO算法中粒子速度是与位置同维的二进制矢量，粒子的更新计算在离散空间中进行。

基于离散空间的DPSO算法不存在冗余搜索问题，且对离散问题表达自然，易与其他演化算法结合，发展前景好。但基于离散空间的DPSO算法需要针对不同问题需要定义不同的模型，缺少一个通用的标准模型。

2.2 蚁群优化算法

蚁群优化算法是受到自然界中真实蚁群的觅食行为而得到启发产生的。仿生学家通过大量的观察和研究发现，蚂蚁总是能够从蚁穴出发，选择最短的路径找到食物源。

蚁群优化算法首先是由 M.Dorigo 根据蚁群觅食行为的原理首先提出的^[49]，最早是用来解决著名的 TSP 旅行商问题。随着这一思想的不断改进与发展，现在已经被成功应用于解决很多组合优化问题。

2.2.1 蚁群优化算法原理

蚁群优化算法来源于对蚁群觅食行为的研究，大量研究发现，蚂蚁之所以能在觅食时找到最短路径是因为蚂蚁在觅食时会释放一种叫做“信息素”的化学物质。而且后来的蚂蚁能检测到信息素，并且倾向于向信息素浓度高的地方移动，而在移动过程中蚂蚁又会继续释放信息素增加信息素的浓度，于是经过蚂蚁越多的路径信息素越强，而后来的蚂蚁选择该路径的可能性也越大。由于在相同时间段内越短的路径会被越多的蚂蚁释放信息素，所以后来的蚂蚁选择较短路径的可能性也越大，最后所有的蚂蚁都走最短的那条路径。大量蚂蚁的这种基于信息素的选择路径方法其实就是一种正反馈的机制。

基于基本 TSP 问题的蚁群算法如下：假设蚂蚁种群数量为 m ，有 n 个城市， $d_{ij}(i,j=0,1,\dots,n-1)$ 是城市 i 和城市 j 之间的距离， $tabu_k$ 禁忌表表示某只蚂蚁 k 已经访问过的城市，即以后不能再访问禁忌表 $tabu$ 中的城市。算法初始时将 m 只蚂蚁随意分布到 n 个城市中去，每只蚂蚁初始的 $tabu$ 为它所在的城市编号。然后每只蚂蚁根据概率公式选择下一步的城市，在 t 时刻某只蚂蚁 k 由城市 i 到城市 j 的概率选择公式为^[49]：

$$p_{ij}^k(t) = \begin{cases} \frac{\tau_{ij}^\alpha(t)\eta_{ij}^\beta(t)}{\sum_{j \in allowed_k} \tau_{ij}^\alpha(t)\eta_{ij}^\beta(t)} & j \in allowed_k \\ 0 & otherwise \end{cases} \quad (2-10)$$

其中 $\zeta_{ij}(t)$ 表示 t 时刻的城市 i 和城市 j 之间的信息素浓度, α 表示信息素浓度因素对概率选择公式的影响因子, $\eta_{ij}(t)$ 表示城市 i 到城市 j 的期望函数, 通常是 $\eta_{ij}=1/d_{ij}$, β 是该期望函数的影响因子, $allowed_k$ 表示蚂蚁 k 在当前城市 i 中还可以选择的城市。

初始时每只蚂蚁在城市 i 的 $\zeta_{ij}(j=0,1\dots i-1,i+1\dots n-1)$ 为一个常数, 因此选择每个城市的概率只与期望函数 η_{ij} 有关系。此后蚂蚁 k 每走一步就往禁忌表 $tabu$ 中加入刚选择的城市。经过 n 个时刻, 蚂蚁 k 走完所有的城市完成一个循环, 此时需要对各条路径上的信息素浓度进行更新。信息素浓度不但因为经过蚂蚁的释放而增强, 也由于时间的推移而挥发。当某蚂蚁 k 完成一次循环后按照以下公式对信息素浓度进行更新:

$$\tau_{ij}(t+n) = (1-\rho) * \tau_{ij}(t) + \Delta\tau_{ij} \quad (2-11)$$

其中 ρ 表示城市 i 和城市 j 之间信息素挥发系数, 因此 $1-\rho$ 表示信息素的保留系数, $\Delta\zeta_{ij}$ 表示本次循环路径 ij 之间的信息素浓度增量, 其计算方法根据计算模型而定。

蚁群优化算法的步骤如下:

- 1) 将 m 个蚂蚁随机地放到 n 个城市中去, 设置每个蚂蚁的禁忌表 $tabu$ 中第一个已访问城市为当前所在的城市, 并设置每个城市之间的 $\zeta_{ij}(0)$ 为一个常数, 并且 $\Delta\zeta_{ij}=0$ 。
- 2) 对于每只蚂蚁 $k(k=1,2,\dots,m)$ 按概率公式(2-10)中计算禁忌表 $tabu$ 之外城市中概率最大的城市 j 为下一步的城市, 并把城市 j 放入禁忌表 $tabu$ 中。
- 3) 经过 n 个城市蚂蚁 k 走完所有城市, 完成一次循环。计算每只蚂蚁走过的总路径长度 L_k , 并看是否比当前保存的全局最短路径更短, 若是则更新当前路径为全局最短路径。
- 4) 根据公式(2-11)更新每个城市路径间的信息素浓度 $\zeta_{ij}(t+n)$ 。
- 5) 将每个城市路径间的 $\Delta\zeta$ 置 0, 并将迭代轮数 $iter$ 加 1。
- 6) 判断当前迭代轮数 $iter$ 是否达到最大的迭代轮数 $maxiter$, 若是则结束迭代, 否则转第 2 步。

蚁群优化算法是一种正反馈算法, 某一段路径上信息素浓度越大就会吸引更多的蚂蚁释放更多的信息素。由于这种正反馈存在, 加快了算法收敛速度。蚁群算法也是一种并行算法, 每个蚂蚁搜索过程独立, 可以减少计算时间。另外蚁群算法模型也直接针对离散问题, 可以直接应用于无线传感网络路由等离散问题。和粒子群算法一样, 蚁群算法也容易陷入局部最优解。

2.2.2 信息素浓度增量计算模型

信息素浓度增量 $\Delta\zeta_{ij}$ 根据不同的策略有不同的计算模型, M.Dorigo 共提出三种信息素浓

度增量计算模型。

1) Ant Cycle System

$$\Delta\tau_{ij}^k = \begin{cases} \frac{Q}{L_k} & \text{蚂蚁k在t到t+1时刻经过路径ij} \\ 0 & \text{否则} \end{cases} \quad (2-12)$$

其中 Q 是一个常数, L_k 为蚂蚁 k 在本次循环中走过总的路径长度。

2) Ant Quantity System

$$\Delta\tau_{ij}^k = \begin{cases} \frac{Q}{d_{ij}} & \text{蚂蚁k在t到t+1时刻经过路径ij} \\ 0 & \text{否则} \end{cases} \quad (2-13)$$

其中 d_{ij} 为城市 i 和城市 j 之间的距离。

3) Ant Density System

$$\Delta\tau_{ij}^k = \begin{cases} Q & \text{蚂蚁k在t到t+1时刻经过路径ij} \\ 0 & \text{否则} \end{cases} \quad (2-14)$$

Ant Quantity System和Ant Density System与Ant Cycle System的主要区别在于在Ant Quantity System和Ant Density System中每个蚂蚁每走一步也就是经过一个时刻都要更新某两个城市间的信息素浓度, 而Ant Cycle System则是等某蚂蚁完成对 n 个城市的访问之后进行全局更新。Ant Quantity System和Ant Density System是利用局部信息, 而Ant Cycle System是利用全局信息。在采用全局信息更新信息素浓度时, 如果某两个城市路径没有被选中, 该路径上的信息素浓度就会随着挥发系数的作用而逐渐降低, 因此算法会逐渐忽略概率选择公式中概率低的路径。但是利用局部信息更新信息素浓度会大大加快算法收敛速度, 但也很容易陷入局部最优解。因此在运用蚁群优化算法解决问题时, 应该在算法迭代中Ant Quantity System或者Ant Density System进行局部更新信息素浓度, 而利用Ant Cycle System进行信息素浓度的全局调整。

2.2.3 蚁群算法参数设置

蚁群算法的参数研究主要是针对蚂蚁下一跳概率选择公式(2-10)中信息素浓度影响因子 α 和期望函数影响因子 β 以及信息素浓度更新公式(2-11)中信息素挥发系数 ρ 这三个参数的研究^[50]。

当信息素浓度影响因子 α 取值较小时, 则表示蚂蚁在运行过程中对信息素的关注度比较小, 也就是蚂蚁在迭代过程中对某条路径上选择下一跳节点时对历史信息素浓度考虑较

小,这会使得蚁群较难找到全局最优路径,会延长算法收敛时间;而当 α 取值较大时,算法可以很快收敛,但由于搜索的随机性减弱,因此容易陷入局部最优解。

期望函数影响因子 β 类似于信息素浓度影响因子 α ,当 β 取值过小时,算法搜索的随机性比较大,收敛时间长;当 β 取值过大时,容易陷入局部最优解。

信息素挥发系数 ρ 也是个很重要的参数,当 ρ 取值较小时,信息素挥发比较慢,也就是算法注重考虑历史迭代中的信息素浓度数值,而对新的迭代轮次中的信息素浓度变化考虑较少,这样算法不容易找到全局最优解,算法收敛速度慢;而当 ρ 取值较大时,信息素挥发比较快,算法更多考虑当前信息素浓度的变化,可以加快收敛速度,但容易陷入局部最优,而且当 ρ 数值增大到一定程度时,导致蚂蚁在迭代过程中不能很好地利用以前蚂蚁释放的信息素,使算法很难找到全局最优解。

就目前的研究成果来看,这三个参数一般设置范围如下: $0 < \alpha < 5$, $0 < \beta < 5$, $0.1 < \rho < 0.9$ 。

2.3 本章小结

本章介绍了两种典型的群智能算法—粒子群优化算法 PSO 和蚁群优化算法 ACO 的基本原理和研究成果。特别是对粒子群算法以及其相关的研究成果进行了深入的讨论,首先介绍了标准粒子群算法的原理和流程,并对粒子群算法研究领域比较重要的惯性权重参数的研究和离散粒子群改造的研究成果进行了比较详细的介绍。由于本论文最大的创新点在于采用改造的离散粒子群算法来计算无线传感网络中全局最优的簇结构,而且文中根据分簇问题的特点都有涉及对粒子群算法中惯性权重参数和离散改造的研究,因此对粒子群算法的这两块内容做了比较深入的介绍。本章第 2 小节也详细介绍了蚁群优化算法的基本原理和流程以及对信息素浓度更新机制的研究。本文针对无线传感网络分簇路由问题中簇首多跳通信的问题,采用改进的蚁群算法来计算簇首到基站的多跳路径计算。

通过本章对粒子群算法和蚁群算法的介绍,对下一章介绍如何利用这两种群智能算法来设计无线传感网络智能分簇路由协议起到了很好的铺垫作用。

第 3 章 基于群智能算法无线传感网络分簇路由协议

本章将详细阐述基于群智能算法的无线传感网络智能分簇路由协议 SI-CRP 协议的主要技术内容和创新点，并且介绍 SI-CRP 协议的整体实现流程。

在详细阐述本文提出路由算法的主要内容之前，先介绍下无线传感器网络能量损耗模型，它也是本文提出的整个路由协议研究出发点。

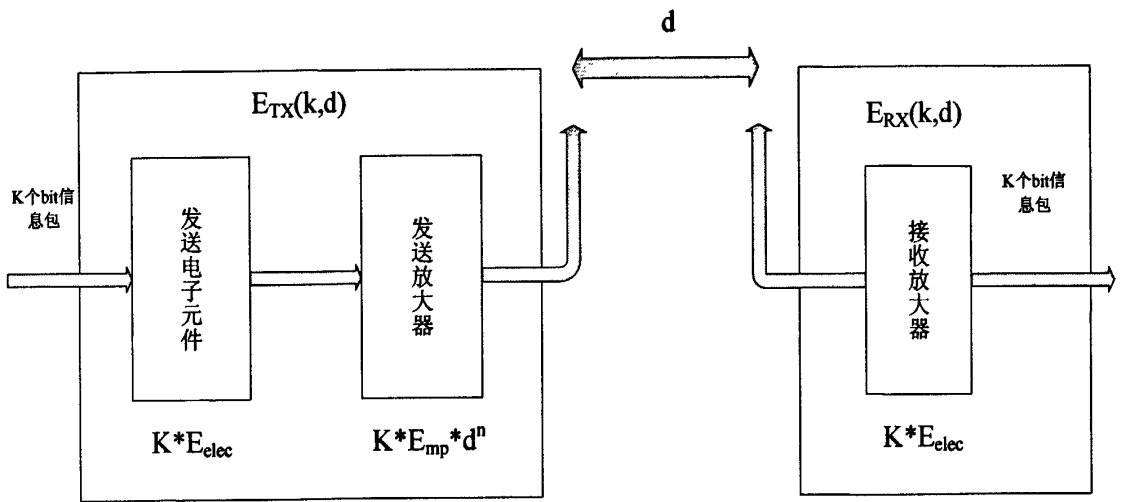


图 3-1 无线传感器网络节点信号传输模型

上图就是无线传感器网络节点信号传输模型，表示相距距离为 d 的两个传感器节点发送和接收数据的信号传输情况。下面用公式进一步说明^[48]：

$$E_{TX}(k,d) = \begin{cases} kE_{elec} + kE_{fs}d^2 & d \leq d_0 \\ kE_{elec} + kE_{mp}d^4 & d > d_0 \end{cases} \quad (3-1)$$

$$E_{RX}(k,d) = kE_{elec} \quad (3-2)$$

$E_{TX}(k,d)$ 表示向距离自身 d 处的节点发送 k 个 bit 数据所消耗的能量， $E_{RX}(k,d)$ 表示从距离自身 d 处的节点接收 k 个 bit 数据所消耗的能量。在发送数据时能量消耗与距离 d^n 成正比，短距离传输即 $d \leq d_0$ 时， $n=2$ ，这种能量衰减模型称为自由空间模型，在长距离传输即 $d > d_0$ 时， $n=4$ ，这种模型称为多径衰减模型^[48]。 E_{elec} 表示发送或接受 1bit 数据电路消耗的能量， E_{fs} 是自由空间模型中无线通信放大电路消耗的能量， E_{mp} 是多径衰减模型中无线通信放大电路消耗的能量， d_0 门限值等于 $\sqrt{E_{fs} / E_{mp}}$ 。

从公式(3-1)和(3-2)中可以看到,两个传感器节点通信能耗的主要因素还是两个传感器节点的距离。因此本文提出的路由协议就是在 LEACH 协议分簇的结构上尽量使整个网络中传感器节点通信距离尽可能短,并且设计能够节省能量的路由机制。

本文提出的智能无线传感器网络分簇路由协议 SI-CRP 是以无线传感器网络经典分簇路由协议 LEACH 协议的网络结构和路由机制为基础,针对 LEACH 协议中的缺点,结合对群智能算法领域的研究提出几种改进机制,设计出一种基于群智能算法的智能无线传感器网络分簇路由协议。

针对 LEACH 协议中分簇不均匀的缺点,本文根据分簇优化的目标提出相适应的离散粒子群算法来直接计算所有传感器节点中的全局最优簇首节点集合,从而使无线传感器网络的分簇均匀,簇首节点位置合理;针对 LEACH 协议中簇首节点与基站单跳通信消耗大量能量的缺点,本文提出以无线传感网络能量损耗为权值采用蚁群算法计算从簇首节点到基站簇首间的多跳路径来节省簇首发送数据给基站的能量消耗;针对 LEACH 协议按轮进行全局簇首重选机制带来能耗过大的问题,本文提出了基于能量自感应机制的局部簇首更新机制来进一步节省能量消耗。

3.1 基于离散粒子群算法的分簇策略

在本文第一章已经阐述了经典分簇路由协议 LEACH 协议最大的缺点就是其随机选择簇首节点的机制带来的分簇不均匀的问题,本文针对分簇问题的特点,提出了适合该问题的离散粒子群算法,并用该算法直接计算全局位置最优的簇首节点,从而形成合理的簇结构。

本文提出的分簇算法假设传感器节点能量有限且能感知剩余能量,并且可以根据发送距离长短调整发射的功率。另外本文假设传感器节点拥有 GPS 定位功能能获取自己相对基站的位置信息。基站能量无限,计算最佳全局最优位置簇首节点的工作由基站来进行,并将计算出的全局最优簇首信息广播给整个网络,从而大大减少了分布式选举簇首节点带来的能量消耗。

3.1.1 适应度目标函数

要利用粒子群算法选举全局最优位置的簇首节点,首先要根据分簇问题的目标,提出粒子群算法的适应度目标函数。优化分簇的目标是从整个监测区域的无线传感器网络所有节点中选出一组处于最佳位置的节点担当簇首的角色使分成的簇均匀。当整个无线传感器

网络中每个簇的簇内节点能紧密地围绕在簇首节点周围时可以被认为合理的簇结构, 对一个簇来说, 当簇内所有节点到簇首的距离总和最小且它们到簇首距离都差不多大小时(也就是簇内所有节点到簇首的距离方差最小时)可以刻画出紧密围绕的特征。因此从整个网络来看, 当所有簇的簇内节点到各自簇首距离总和的平均值与所有簇的簇内节点到各自簇首的距离方差的平均值同时达到最小时, 所有的簇的结构应该是均匀合理的。适应度函数F定义如下:

$$F = \text{mean} \left(\sum_{j=1}^n \left(\sum_{i=1}^m D_{ji} - jch \right) \right) * \text{mean} \left(\sum_{j=1}^n \left(\text{var} \left(\sum_{i=1}^m D_{ji} - jch \right) \right) \right) \quad (3-3)$$

其中 D_{ji} - jch 代表第 j 个簇中第 i 个簇内节点到簇首的距离, i 变量从簇内第一个节点到第 m 个节点, j 变量从第一个簇到第 n 个簇。 mean 是求平均值的函数, var 是求方差的函数。适应度函数F前半部分乘积就是网络中各个簇的簇内节点到簇首节点距离总和的平均值, 而F函数的后半部分乘积是各个簇的簇内节点到簇首的距离方差的平均值, 当这两个部分的乘积达到最小值时整个网络的簇结构是均匀合理的。而粒子群算法就是要在不断的迭代中找寻使这个适应度目标函数F尽可能小的粒子位置。

这里还有一点要说明, 根据文献[51], 当簇首节点个数占网络中所有传感器节点个数百分比为5%时, 整个网络的能量最节省。本文中运用了这个研究成果, 将簇首节点个数就设定为所有传感器节点个数的5%, 每个簇的簇内节点个数都相等(每个簇的簇内节点其实应该就是20个, 设整个网络中有 N 个传感器节点, 则整个网络中有 $5\% * N$ 个簇首节点, 也就是 $5\% * N$ 个簇, 因此每个簇的簇内节点为 $N / 5\% * N = 20$, 其实公式(3-3)中 m 变量就应该为20)。

3.1.2 分簇问题的离散粒子群改造

要利用粒子群算法来计算无线传感器网络中全局最优位置的簇首节点, 必须根据分簇优化问题的特点将标准粒子群算法改造成离散粒子群算法。由于分簇优化的目标是要找到所有传感器节点的一个簇划分, 本文将粒子结构也就是粒子的位置里的维数数据存放整个网络中所有传感器节点的编号, 也就是无线传感器网络节点个数就是粒子的维数。然后用一个数据结构存放节点编号和该节点的位置坐标信息, 从而将每个节点的编号和位置坐标信息关联起来, 这样就可以根据粒子位置中的节点编号互换来获得不同的簇划分, 从而通过关联的传感器节点坐标信息计算公式(3-3)中适应度函数值确定较好的簇划分, 最终通过粒子群算法的迭代找到全局最优的簇划分。

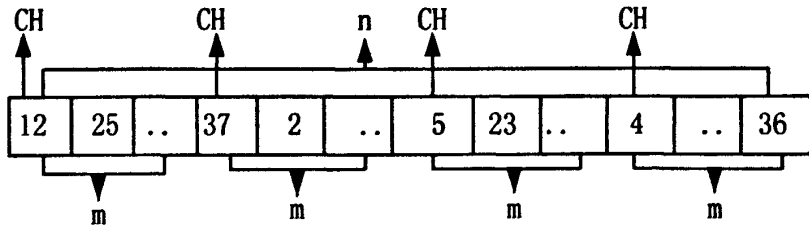


图 3-2 本文粒子群算法粒子结构

如图 3-2 所示，粒子的维数 n 代表有 n 个传感器节点，其中每一维里的数字代表每个传感器节点的编号。 m 代表每个簇内都有相等的 m 个节点，这里 m 应为 $20(1/5\%=20)$ 。CH 代表簇首(Cluster Head)，假设 i 是 n 维粒子的坐标，如果 i 对 20 取余结果为 1，则 i 代表的是簇首的坐标，否则为簇内节点坐标。

在定义了针对分簇优化问题的粒子结构后，就要将标准粒子群算法的速度和位置更新公式(2-1)和(2-2)的运算规则进行改造，使它成为针对特定问题的离散粒子群算法，能够在粒子群算法迭代中不断向全局最优解靠近。本文定义了如图 3-2 的粒子结构后，为了在迭代算法中不断根据适应度函数值改变簇划分，算法需要对图 3-2 中的粒子编号进行交换。借鉴 clerc 在文献[46]中 TSP-DPSO 的离散粒子群改造方法，本文也将粒子速度定义为一系列粒子坐标中节点编号的交换函数 $v_i=swap(j,k)$ 组成的交换序，即代表将粒子 i 位置坐标 j 的节点编号与坐标 k 的节点编号进行交换，一个速度可能是若干个这样的交换序列的集合。同时也更改了公式(2-1)和(2-2)中的“+”，“-”和“*”的定义，具体如下：

(粒子位置+粒子速度)：旧的粒子位置与粒子速度“+”的结果为新位置。让速度中的所有交换序列依次改变旧位置的节点编号。假设旧位置为(1,4,5,3,2)，速度为((1,2),(3,5))，则先交换坐标为 1 和 2 的数据，然后交换坐标为 3 和 5 的数据，然后则新位置为(4,1,2,3,5)。

(粒子位置-粒子位置)：两个粒子位置的“-”的结果为一个速度。假设 p_1 为第一个位置， p_2 为第二个位置， p_1-p_2 产生的 v 应满足 $p_2+v=p_1$ 。如上 p_1 为(4,1,2,3,5)， p_2 为(1,4,5,3,2)，则“-”的结果应为((1,2),(3,5))。

(粒子速度+粒子速度)：两个粒子速度的“+”的结果为新速度。两个粒子速度的“+”操作简单将两个速度的交换序列合并。设一个粒子速度为((1,2),(3,5))，另一个为((3,4))，新速度为((1,2),(3,5),(3,4))。

(实数因子*粒子速度)：一个实数因子与粒子速度的“*”结果仍然为一个粒子速度，定义为速度中的每一个交换序列以因子的值为概率保留(公式(3-1)和(3-2)中的因子 w ， c_1 ， c_2 都为一个小于 1 的实数)。本文做法是对粒子速度中每一维的数据生成一个小于 1 的随机

数,若该随机数小于实数因子,则保留该维数据,否则将该维速度数据删除。设一个粒子速度为 $((1,2),(3,5))$,实数因子为0.4,假设为粒子速度第一维数据 $(1,2)$ 生成随机数为0.5,为第一维数据 $(3,5)$ 生成随机数为0.3,则最后 $0.4*((1,2),(3,5))$ 的运算结果为 $((3,5))$ 。

通过以上对标准粒子群算法中粒子结构和粒子速度位置更新公式的改造,已经可以使粒子群算法运用于无线传感网络的分簇优化问题。

3.1.3 混合惯性权重调整和粒子变异策略

本文提出的SI—CRP协议的分簇方法在粒子群算法中粒子速度位置更新公式中所用惯性权重调整策略是一种混合惯性权重调整策略。线性递减的惯性权重调整策略是一种不错的惯性权重调整策略,在粒子群算法迭代前期惯性权重值较大,有利于粒子群算法能找到全局最优解,而在迭代后期惯性权重值较小,有利于加速局部收敛。然而线性递减的惯性权重调整策略的缺点就是收敛速度比较慢,本文通过采用线性递减的惯性权重调整策略并融合提出的一种基于粒子多样性的惯性权重调整方法来避免陷入局部最优并加快粒子群收敛速度。这种基于粒子多样性的惯性权重调整方法根据每个粒子位置与全局最优粒子的相似度来设定每个粒子的惯性权重。这种方法的思想是当某个粒子位置与全局最优粒子位置相似度越大则其惯性权重应该越大,使其能够保持原有的速度避免快速向全局最优粒子位置靠近;当某个粒子位置与全局最优粒子位置相似度较小时则惯性权重应该越小,使其能够加快收敛速度。也就是说采用基于粒子多样性的惯性权重调整方法在收敛速度和避免全部最优取得平衡。本文首先定义了粒子多样性函数:

$$f(p_i) = \frac{\text{count}(p_i, p_g)}{N} \quad (3-4)$$

其中 $f(p_i)$ 代表某个粒子 i 的多样性函数, N 是粒子群的粒子位置的维数, $\text{count}(p_i, p_g)$ 代表粒子 i 的位置和全局最优粒子 g 的位置维数相异的个数。假设粒子 i 的位置为 $p_i=(1,2,3,4,5)$,全局最优粒子位置为 $p_g=(1,2,4,5,3)$,则 $\text{count}(p_i, p_g)$ 的值为3,则 $f(p_i)$ 的值为 $3/5=0.6$ 。整个基于粒子多样性函数的惯性权重的调整公式为:

$$w(p_i) = w_{\max} - (w_{\max} - w_{\min}) * f(p_i) \quad (3-5)$$

其中 $f(p_i)$ 代表某个粒子 i 的多样性函数, $w(p_i)$ 代表粒子 i 的惯性权重, w_{\max} 代表线性惯性权重调整策略中迭代开始时的惯性权重值, w_{\min} 代表迭代结束时的惯性权重值。

本文提出的混合惯性权重调整策略是在迭代轮数的前20%轮和迭代轮数的20%轮数中采用线性递减惯性权重调整,而在中间60%的迭代轮次中采用公式(3-5)中基于粒子多样性

的惯性权重调整,并在迭代中保持粒子速度位置更新公式(2-1)和(2-2)中 $w+c_1+c_2=1$,且 $c_1=c_2$ 。

本文还提出使用变异的策略来避免陷入局部最优即在粒子群算法迭代过程中判断如果每个粒子隔一定轮数个体最优解没有发生变化,并且该粒子位置和全局最优粒子位置的相似度在一定程度以上就使该粒子位置的某些维(即粒子编号)进行交换。本文中具体的做法是如果经过总迭代次数 $maxiter$ 的10%轮后,如果某个粒子隔一定轮数个体最优解没有发生变化,并且该粒子位置和全局最优粒子位置相似度在80%以上即多样性函数 $f(p_i)$ 的函数值小于0.2时,则在粒子位置中的每个簇中随机选择一维与当前的簇首所在的维交换即选择本簇内其他节点当簇首。当然本来就是全局最优粒子不需要变异。这种变异策略可以使粒子在迭代过程中产生一定的概率找到比当前全局最优解更好的解,避免陷入局部最优。

3.1.4 启发式算法

为了提高粒子群算法计算全局最优簇结构的效率,加速粒子群算法迭代的收敛速度,本文针对分簇问题的特点提出两种有效的启发式算法,使粒子群算法能更快找到全局最优的簇首节点。

(1) 交换簇之间簇内节点的位置

如果在粒子群算法迭代过程中发现某粒子位置中形成的簇结构中某个簇内节点比自己当前的簇首更靠近另外某个簇首,那这样的簇内节点应该被交换到合适的簇中去。

该启发式算法使用一个阈值来决定是否需要某个簇内节点做交换操作。假设传感器节点均匀布撒的区域面积为 S , N 为总的节点个数,那每个簇的理想覆盖面积应该是 $S/(N*5\%)$ 。把每个簇的覆盖面看成是以簇首为中心的圆面,每个簇的簇半径应为

$\sqrt{S/(N*5\%*\pi)}$ 。为了提高灵活性和减少误判,算法设定阈值为 $1.5*\sqrt{S/(N*5\%*\pi)}$ 。

算法在粒子群算法每轮迭代中检查粒子位置形成的簇结构中每个簇的簇内节点,若发现某个簇内节点与当前簇首的距离大于 $1.5*\sqrt{S/(N*5\%*\pi)}$ 则启动交换过程。假设发现超过阈值的是A簇的簇内节点为A1,当前簇首为A-H。首先寻找最接近A1的簇首,如果仍为当前簇首A-H,则终止交换过程。若找到的为其他簇首B,则在B簇中寻找最接近A-H的簇内节点B1,然后判断B1到B簇簇首B-H的距离是否超过阈值,若也超过阈值则交换簇内节点A1和B1,否则不交换。在本算法中可以根据粒子位置中传感器节点编号与节点位置的关联信息判断每个簇内节点是否需要交换,如需要交换只要交换粒子位置中节点的编号即可。

(2) 调整簇首节点位置

在(1)中的启发式算法中处于不合适簇的簇内节点将被调换到合适的簇当中去,但是即使每个簇内节点都在合适的簇当中,当前簇首节点的位置仍然可能不是最合适的。比如以一个圆形的簇为例,理想的簇首节点应该位于圆形区域的中心,但如果当前簇首节点位于圆形区域的边界区域位置,则簇的结构就比较差,也就是簇内各成员节点到当前簇首节点距离的差距过大。

设某个簇中簇首节点与离它最远的簇内节点之间的距离为 $Dis-far$,与离它最近的簇内节点距离为 $Dis-near$ 。若 $Dis-far$ 与 $Dis-near$ 之间的差值过大则认为该簇不均匀即各簇内节点到簇首的距离方差较大,此时需要在当前簇的节点中找出位置更适合的节点当选簇首。本文设定这个差值阈值为 $1.3 * \sqrt{S / (N * 5\% * \pi)}$,当发现某个簇中的 $Dis-far$ 与 $Dis-near$ 之间的差值大于这个差值阈值时,算法将所有簇内节点到当前簇首的距离排序,然后将距离当前簇首中等的那个簇内节点当选为新的候补簇首,继续上述检查过程,直到找到合适的簇首节点成为该簇最终的簇首节点。

使用两种启发式算法的策略是每经过总迭代次数 $maxiter$ 的5%轮数后全局最优解,则对所有粒子依次使用启发式算法(1)和(2)对粒子位置进行指导调整,从而加速找到全局最优簇首节点的速度。

3.1.5 选择全局最优簇首流程

上面主要介绍了本文智能分簇路由协议SI-CRP协议中主要内容创新点—利用改造的粒子群算法计算全局最优位置簇首节点的主要技术,在这一小节中,将介绍该分簇方法的完整算法流程。SI-CRP协议中基于改造粒子群算法的分簇方法流程如下:

1) 首先每个传感器节点通过自身的GPS定位系统获取自身相对于基站的坐标位置信息,然后每个传感器节点将自身的位置信息和节点编号广播给基站。基站将节点编号和对应的位置信息存储起来。

2) 基站开始运行改进的粒子群算法。算法首先随机初始化种群规模为 N 的粒子群中每个粒子的位置和速度。如3.1.2节所述,粒子的位置就是网络中的传感器节点编号的一个排列,开始随机初始化粒子位置为一个不重复的节点编号的排列。而粒子的速度则随机初始化为两个为一组的传感器节点编号交换序列,如有 D 个传感器节点,则一个的粒子速度为 $D/2$ 个节点编号的交换序列。

3) 由初始粒子位置根据适应度目标函数公式(3-3)得出所有粒子中适应度函数值最小的粒子位置,并将相应的粒子位置设定为全局最优粒子位置,每个粒子的个体最优粒子位

置设定为初始化的粒子位置。

4) 接下来开始粒子群算法的迭代过程,更新迭代轮数,将迭代轮数加1。根据公式(2-1)和(2-2)计算这一轮的每个粒子的速度和位置。

5) 根据公式(3-3)计算每个粒子的适应度函数值,得出所有粒子中最小的适应度函数值,并记录该全局最小的适应度函数值及相应的粒子。

6) 判断每个粒子的当前适应度函数值是否比个体历史最优函数值更小,则更新当前适应度函数值为当前粒子的个体最优适应度函数值,并更新当前粒子位置为该粒子的个体历史最优位置。判断该轮全局最小的适应度函数值是否比历史全局最优适应度函数值更小,若是则将该轮全局最小的适应度函数值更新为全局最优适应度函数值,并将对应的粒子位置更新为全局最优粒子位置。

7) 判断迭代轮数,每隔最大迭代轮数 $maxiter$ 的5%则随机对所有粒子运用两种启发式算法,然后转第(8)步。否则直接转第(8)步。

8) 判断每个粒子个体最优位置是否连续最大迭代轮数 $maxiter$ 的10%轮未发生变化,若是则按照3.1.3节中所述对粒子位置进行变异,然后转第(9)步,否则直接转第(9)步。

9) 判断是否达到迭代最大轮数,若是则结束迭代,否则按照3.1.3节所述混合惯性权重调整策略更新惯性权值然后转第(4)步继续迭代。

10) 根据粒子群算法迭代完成时选出的最终全局最优粒子位置,按照3.1.2节所述的簇首节点坐标和簇内节点坐标的定义将整个粒子位置分成若干个簇,并确定每个簇中的簇首节点和簇内节点的编号,从而划分出全局最优的簇结构。

图3-3显示了基于离散粒子群算法的分簇算法的详细流程图。

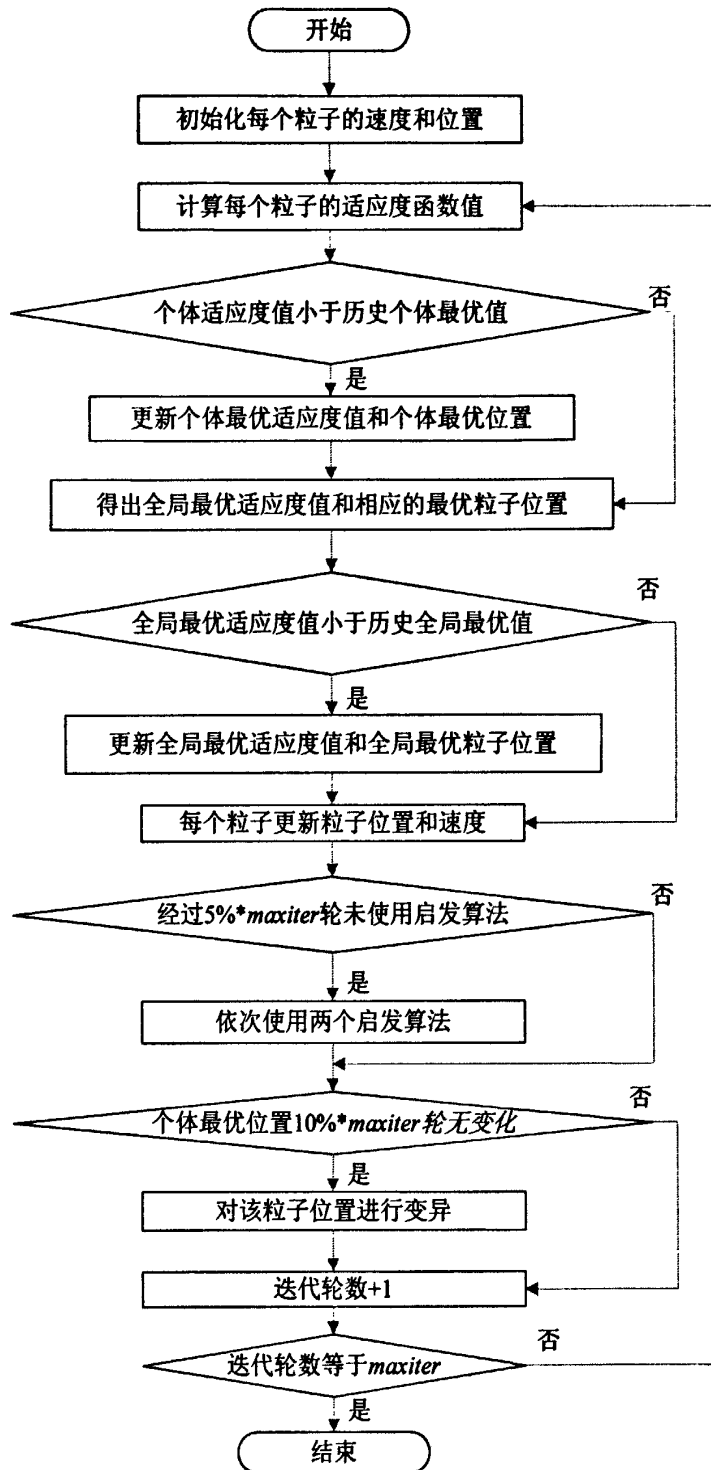


图 3-3 基于离散粒子群算法的分簇算法流程

3.2 基于改进蚁群算法的簇首间多跳策略

LEACH协议第二个比较严重的缺陷是簇首与基站间的单跳通信,从公式(3-1)中可以看出当两个通信的传感器节点之间通信距离大于某一个阈值,通信能耗将和通信距离的

4次方成正比,簇首将由于如此大的能耗迅速死亡。因此簇首间应该采用多跳通信,降低通信能耗。本文提出的SI-CRP智能分簇路由协议采用改进的蚁群优化算法计算簇首间的最优多跳路径,从而节省了簇首通信的能耗。

3.2.1 概率选择公式期望函数改进

标准蚁群算法的概率选择公式(2-10)中的期望函数 η_{ij} 为城市 i 和城市 j 之间距离 d_{ij} 的倒数,为的是求得TSP问题中的路径距离最短。而将其应用于无线传感器网络的簇首间多跳路径计算时目标应是使簇首间的多跳通信能耗最小,因此就要对标准蚁群算法的概率选择公式中的期望函数 η_{ij} 进行改进。

根据无线传感器节点通信能耗公式(3-1)和(3-2),相对于TSP问题中两个城市间的距离这个概念,在采用蚁群算法计算簇首间多跳路径时的期望函数时,应该采用一对通信的传感器节点间的通信总能耗作为期望函数。因此在这里将标准蚁群算法的概率选择公式中的期望函数 η_{ij} 改为如下公式:

$$\eta_{ij}(d) = \frac{1}{E_{ij}} = \frac{1}{E_{TX}(d) + E_{RX}(d)} = \begin{cases} \frac{1}{2E_{elec} + E_{fs}d^2} & d \leq d_0 \\ \frac{1}{2E_{elec} + E_{mp}d^4} & d > d_0 \end{cases} \quad (3-6)$$

其中 E_{TX} , E_{RX} , E_{elec} , E_{fs} , E_{mp} 和 d_0 等变量的含义与公式(3-1)和(3-2)相同,改进后的期望函数 η_{ij} 表示相距距离为 d 的两个传感器节点 i 和 j 之间发送1bit数据所消耗的通信总能耗的倒数,也就是某个簇首节点到另一个簇首节点的通信能耗越低,蚂蚁选择该簇首节点为自己下一跳节点的概率越大。将公式(3-6)作为蚁群算法概率选择公式中的期望函数能够在蚁群算法的迭代中找到簇首到基站的最小能耗多跳路径。

3.2.2 信息素浓度增量改进

信息素浓度增量 $\Delta \zeta_{ij}$ 的三种计算模型中都是以距离作为参数,因此也需要进行修改。本文的改进蚁群算法是采用Ant Cycle System的全局更新信息素浓度的机制,因此对Ant Cycle System的信息素浓度增量的计算模型进行改进。

$$\Delta \tau_{ij}^k = \begin{cases} Q^* \eta_k = Q^* \frac{1}{\sum E_{path(i-b)}} & \text{蚂蚁} k \text{ 经过路径 } i-j \\ 0 & \text{否则} \end{cases} \quad (3-7)$$

其中 η_k 为蚂蚁 k 在本次循环中从源簇首节点到基站所通过簇首节点形成的路径上总能耗的倒数,其中每对簇首节点之间的通信能量消耗都如公式(3-6)之中所述。任何两个簇首

节点之间信息素浓度就从对应的簇首节点开始计算到基站的总能耗。

3.2.3 基于无线传感网络路由的蚁群算法机制改进

标准蚁群算法原本是用来解决旅行商问题的，算法在迭代过程中需要遍历所有的城市节点，从而找到遍历所有城市节点且经过一次的最短路径。而将蚁群算法运用于本文簇首间到基站的多跳路径问题时，其算法机制就需要有所改变。首先和旅行商问题不同，簇首间多跳路径问题中源节点和目的传感器节点是确定的，而不像旅行商问题最后的解源节点和目的节点可以是任何城市节点。再者从某一个源簇首节点出发通常只经过所有的簇首中某几个簇首节点到达基站，而不需要像旅行商问题遍历所有的城市节点。

具体的改进措施如下：当利用改造的蚁群算法求某一个源簇首节点到基站的多跳路径时，假设有 n 个簇首节点和1个基站节点，初始时将 m 只蚂蚁都放在该源簇首节点，每只蚂蚁仍然保留一个禁忌表 $tabu$ ，初始时存放的是源簇首节点的编号。然后每只蚂蚁开始独立地搜索基站节点，每只蚂蚁按照公式(2-10)所示的概率选择下一跳的簇首节点，其中的期望函数 η_{ij} 是如公式(3-6)所示的传感器两节点的能耗模型。初始时簇首节点 i 和簇首节点 j 之间的信息素浓度 ζ_{ij} 为一个常数，因此概率选择是只依赖于两个簇首节点之间的通信能耗。另外和标准蚁群算法机制一个比较大的不同是，一旦某只蚂蚁根据概率公式选择下一跳簇首节点的编号为基站节点编号时，这只蚂蚁此轮迭代就结束。当所有蚂蚁都找到基站时，按照公式(3-7)所示的全局信息素浓度增量更新公式更新需要增加的信息素浓度。

3.2.4 基于蚁群算法簇首间多跳路径计算流程

前面介绍了利用改造的蚁群算法计算簇首到基站多跳路径的具体的技术内容，下面将介绍整个计算流程，使对本算法有更加清晰的认识。基于改造的蚁群算法计算簇首到基站多跳路径的具体计算流程如下：

- 1) 首先各个簇首节点将自己的节点编号和位置信息报告给基站，基站记录各个簇首的信息。
- 2) 基站为每个簇首节点运行蚁群算法找到从自身到基站的总能耗最小的多跳路径。以某一个簇首节点为例，初始化时将 m 只蚂蚁都放在该簇首节点上作为源节点，禁忌表 $tabu$ 中存放自身节点编号，每只蚂蚁初始时选择下一跳概率公式(2-10)中信息素浓度 ζ 为一常数，期望函数按照公式(3-6)所示的能耗模型计算。并将迭代变量置为1。
- 3) 开始迭代过程，每只蚂蚁 $k(k=1,2,\dots,m)$ 按概率公式(2-10)中计算禁忌表 $tabu$ 之外的簇首节点中概率最大的城市为下一跳节点，并把节点依次放入禁忌表 $tabu$ 中作为本轮迭代的

路径,如果遇到的下一跳节点为基站,则将基站放入禁忌表中后该蚂蚁此轮路径探索结束。

4) 计算每只蚂蚁得出路径的总能耗,并且得出全局最少的总能耗和对应的最优路径。然后比较该轮全局最优总能耗和历史全局最优总能耗,如果当前全局最优总能耗更小,则更新历史全局最优总能耗为当前的全局最优总能耗,并保存相应的最优路径。

5) 每只蚂蚁根据公式(2-11)更新它本轮迭代产生多跳路径上每对簇首节点间的信息素浓度 ζ_{ij} 并对所有蚂蚁的更新结果进行叠加。

6) 将每个城市路径间的 $\Delta\zeta$ 置0,并将迭代轮数 $iter$ 加1。

7) 判断当前迭代轮数 $iter$ 是否达到最大的迭代轮数 $maxiter$,若是则结束迭代,否则转第3步。

8) 输出最终得到从该源簇首节点出发的全局最优路径,并由基站通知每个路径上的簇首节点他们的上下跳节点编号,各簇首节点收到该信息后保存自己的上跳和下跳节点编号。

图3-4显示以确定某一个簇首为源起点基站为终点计算基于蚁群算法的最小能耗路径的详细流程。

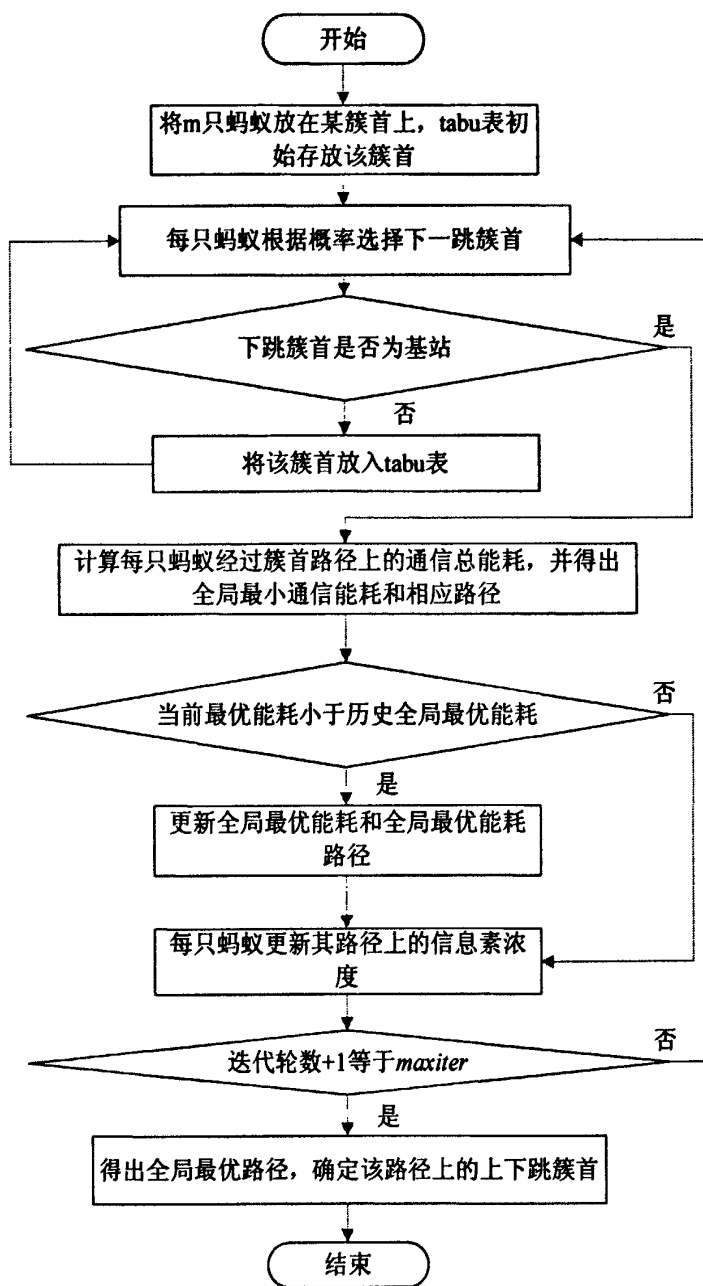


图 3-4 基于蚁群算法最小能耗路径算法流程

这里还有点需要说明, 在计算出某个簇首节点为源节点基站为终点的路径上的所有簇首节点, 他们其实都获得了以自身为源节点到基站的多跳最小能耗路径, 因此算法不需要再为这些路径上的簇首节点再次运行蚁群算法, 只需要为还没有上下跳信息的簇首节点运行蚁群算法计算相应的多跳最小能耗路径即可。

3.3 基于能量自感应的局部簇首更新机制

LEACH协议中还有一个比较严重的问题是在给整个网络中分簇时所耗费的大量能量,

以及其周期性重新分簇的机制使浪费的能量进一步增加,无线传感器网络中的传感器节点能量本来就有限,频繁的重新分簇会很快使传感器节点死亡。

在LEACH协议中,簇首的选举和分簇工作都需要每个传感器节点的参与,其中还包括传感器节点之间大量的信息交互,比如传感器节点根据簇头概率公式确定自己当选为簇首后,需要向全网节点广播这一个消息,簇内节点还要根据信号强弱确定出离自己最近的簇首节点,并向它发送加入簇的消息,簇首节点收到所有簇内节点的加入消息后,还要向它们发送确认消息。可以想象,这些广播信息对能量有限的传感器节点来说是非常大的消耗。而本文提出的无线传感器网络智能分簇路由协议SI-CRP协议设计的首要目标就是节能,本章开始提出的基于改造粒子群的分簇算法和基于改造蚁群的簇首间的多跳算法都很好地体现节能这一目标,基于改造粒子群的分簇算法只需要开始时每个传感器节点将它们各自的节点编号和位置信息发送给基站,然后整个分簇计算过程都在基站由能量充足的高性能的计算机计算,计算完成后由基站将分簇结果发送给传感器节点即可。基于改造蚁群的簇首间的多跳算法也是如此,由基站计算出簇首间的多跳路径后将簇首相应的上下跳节点发送给簇首即可。因此本文SI-CRP协议中的两种算法都是采用将能量消耗转嫁到基站的策略来减少网络中传感器节点的能量消耗。

至于LEACH协议中周期性重新分簇带来的能量消耗,本文提出了一种基于能量自感应的局部簇首更新机制。在这种机制中,SI-CRP协议只在首次将网络分簇时需要传感器节点发送其节点编号和位置信息给基站,而后当节点获得了基站计算得到的分簇结构和簇首间多跳路径后,在以后整个网络工作过程中不再进行全局的重新分簇,而是只根据节点剩余能量情况动态地局部小范围重新选举簇首。

基于能量自感应的局部簇首更新机制具体如下:网络中的簇内节点在将监测数据发送给簇首节点的同时,会捎带一个很小数据量的剩余能量信息,而簇首节点需要分离出簇内节点的剩余能量信息,并将它所有的簇内节点当前剩余能量情况存储起来。而簇首节点在工作的同时则不断监测自身的剩余能量情况,当发现自身的剩余能量不足初始能量的10%时,则该簇首认为自己不再适合做簇首,然后从它的簇内节点剩余能量信息中查询出本簇内剩余能量最多的节点当选为新的簇首(如果所有节点的剩余能量都不足初始能量的10%时则新当选的簇首节点将百分比阈值下降一半,即变为5%,依次类推)。旧的簇首将该簇首当选的信息在本簇范围内进行广播,并将旧簇首的上下跳节点编号发送给新的簇首,由新的簇首和上下跳节点进行确认。可以看到,这种局部簇首更新机制的重新选举簇首的信息交互只在有限几个传感器节点之间进行,相比于LEACH协议全网范围内的重新选举簇

首带来的广播交互能耗要小得多。而且通过对能量的监测，可以有效避免簇首节点迅速死亡，能量小于初始能量10%的节点成为簇内节点，可以使该节点通信能耗迅速下降，延长生命周期。

图3-5显示了以某一个簇工作情况为例基于能量自感应的局部簇首更新机制的工作流程。

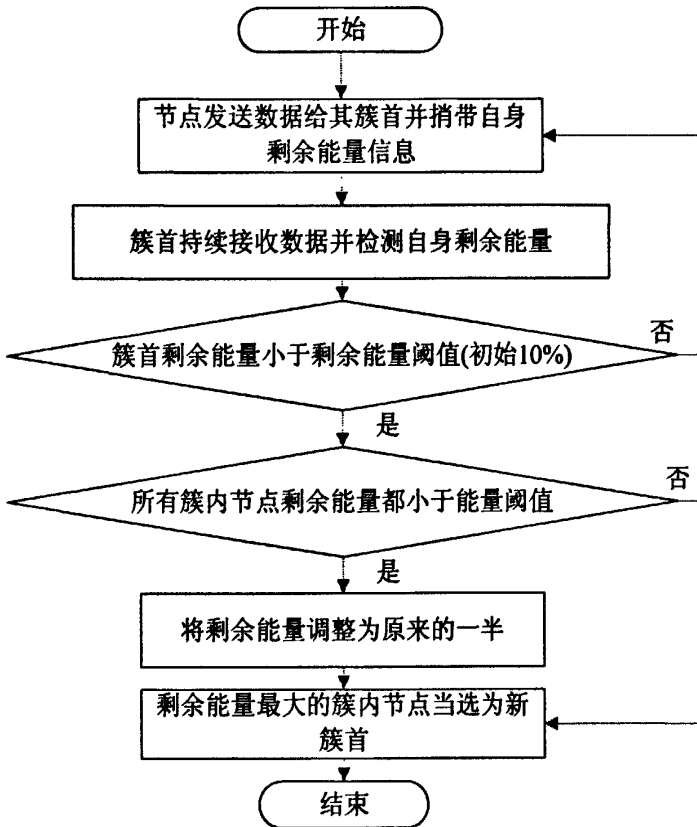


图 3-5 基于能量自感应的局部簇首更新机制流程图

3.4 本章小结

本章是本论文的核心章节，在该章中详细地介绍了本文提出的无线传感网络智能分簇路由协议SI-CRP协议的主要的创新点和技术内容以及SI-CRP协议的详细流程。

首先在3.1节中详细介绍了基于改造粒子群算法计算全局最优簇首的原理和流程，这个过程是整个SI-CRP协议分簇阶段的最重要的一个步骤，也是本文最重要的创新点。基于该方法计算得到的簇首节点都是在所有传感器节点中位置最适合做簇首的节点集合。在3.2节中详细介绍了基于改造蚁群算法计算簇首到基站的多跳路径，这是属于SI-CRP协议分簇阶段的第二个重要步骤，也是本文主要内容。基于改造蚁群算法计算得到的簇首多跳路径

得到了从簇首到基站的能耗最小的多跳路径。SI-CRP协议就是将改造的粒子群算法和改造的蚁群算法运用于无线传感网络路由协议中为主要技术特征的。在3.3节中介绍了SI-CRP协议在稳定数据传输阶段的一个重要创新内容就是在簇首基于能量自感应的局部簇首更新机制。有了局部簇首更新机制，不必在协议运行过程中周期性重新分簇，从而节省能量。

第 4 章 SI-CRP 协议的实现与仿真

4.1 SI-CRP 协议的实现

本节将详细阐述基于群智能算法的无线传感网络分簇路由协议—SI-CRP 协议的实现细节和流程。

SI-CRP 协议和 LEACH 协议一样,将整个路由协议分为两个大阶段:网络簇结构生成阶段和稳定数据传输阶段。这里首先需要说明一点的是,与 LEACH 协议周期性循环进行网络簇结构生成阶段和稳定数据传输阶段不同,SI-CRP 协议只在协议首次运行时生成最终的分簇网络结构,而后就一直处在稳定的数据传输阶段。SI-CRP 协议不像 LEACH 协议一样循环地进行全局簇首选举,而只是在检测到某簇首节点剩余能量低于阈值时进行局部的簇首选举,从而降低网络工作的能耗。

4.1.1 网络簇结构生成阶段流程

本小节首先介绍 SI-CRP 协议的网络簇结构生成阶段,该阶段包括 3.1 节介绍的基于离散粒子群算法计算全局最优簇首和 3.2 节介绍的基于改造蚁群算法计算簇首间多跳路径。

当网络中传感器节点被布置好后,SI-CRP 协议就开始生成簇结构,首先所有传感器节点会将自己的节点编号和位置信息发送给基站,基站会建立节点编号和位置信息的关联,并将所有传感器节点的信息存储起来。然后按照 3.1 节中所表述分簇方法计算出最优簇首并获得整个网络的簇结构。在获得全局最优的簇首后,将簇首节点编号和其位置信息作为参数根据 3.2 节所表述方法计算簇首间的多跳路径,得出每个簇首节点做为源节点的下跳节点以及作为中间转发节点的上跳节点。在这两个过程结束后,基站已经生成了整个网络的结构图,下面需要将这些信息发送给每个传感器节点。

基站首先报告簇首节点信息,基站用簇首节点编号作为标识符,并发送簇首节点相关信息。该信息包括以它为簇首的所有簇内节点的编号,并且用簇首节点编号关联该簇首节点的上下跳簇首节点信息(可能有簇首节点没有上跳簇首节点),上下跳簇首节点信息包括上下跳簇首节点的编号以及簇首节点发送给下跳簇首节点所需要的发送功率(发送功率根据两个簇首节点之间的距离得出)。所有传感器节点保持监测基站信息,每个节点根据自己的节点编号和基站发送来的簇首节点编号做对比,如果节点编号与基站发送的节点编号不

同则不接收，当发现发来的簇首节点编号与自己编号相同则接收信息并存储起来，然后将自己的身份存储为簇首节点，这样每个簇首节点最后得到了相关的信息。类似地，基站将给所有簇内节点报告它们的簇首节点编号以及它们发送给相应的簇首节点所需要的发送功率。该信息结构也是以每个簇内节点的编号作为标识符，信息内容包括该簇内节点的对应簇首节点编号以及发送给该簇首节点所需要的发送功率。每个簇内节点也是按照比对节点编号的方法确定是否是发送给自己的信息，簇内节点接收发送给自己的信息后就存储起来。至此，整个网络的所有传感器节点都得到了各自的身份信息，因此整个网络的簇结构就建立起来了。每个传感器节点也得到了根据该分簇结构工作的必要信息，比如簇内节点知道它的簇首节点编号以及将数据发送给簇首节点所需要的功率，而簇首节点知道它所有的簇内节点编号，以及将数据传送给基站的多跳路径的上下跳节点编号和相应发送功率。和 LEACH 协议一样，簇首节点按照 TDMA 方式给本簇内所有簇内节点安排了一个发送数据的周期表，并通知簇内节点它们各自可以发送监测数据的周期时序，这样簇内节点可以无冲突地给簇首节点发送监测数据。图 4-1 是 SI-CRP 协议的网络簇结构生成阶段的流程图。

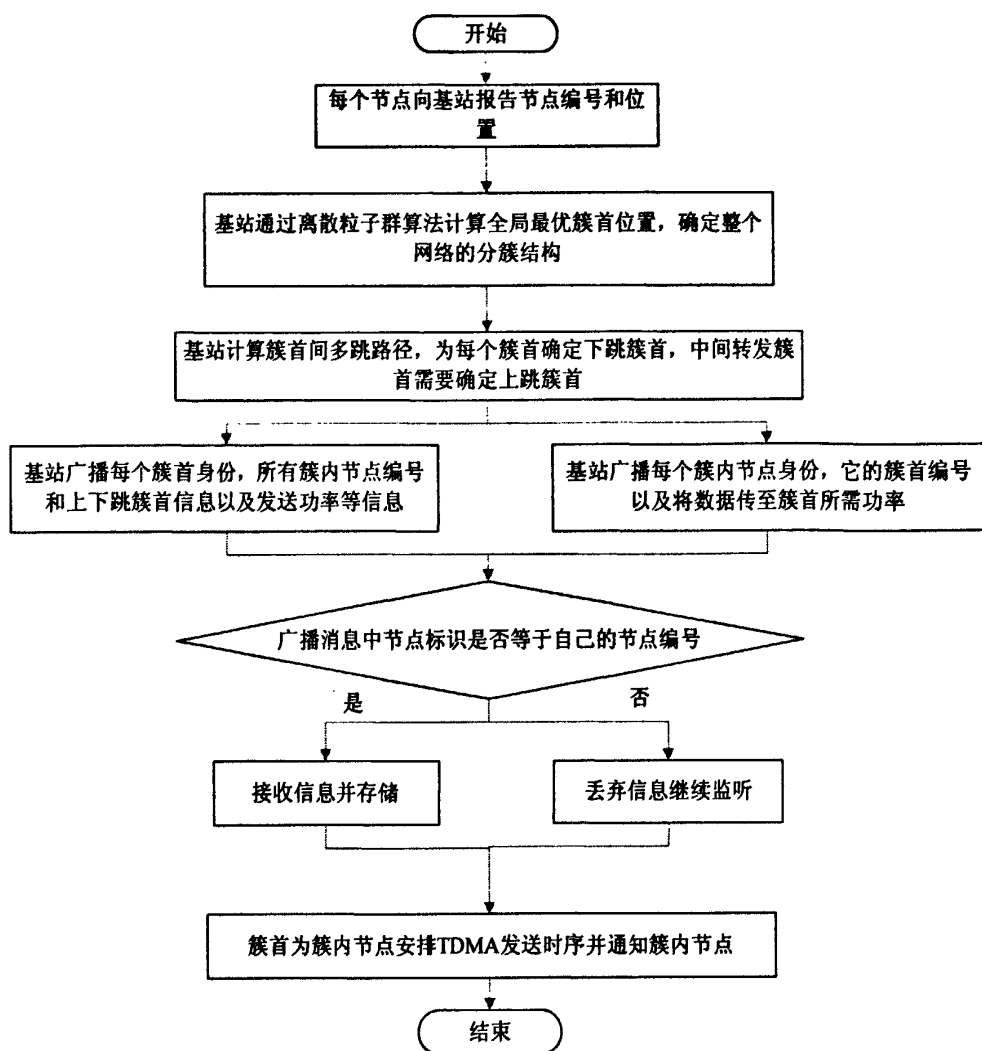


图 4-1 SI-CRP协议网络簇结构生成阶段流程图

4.1.2 稳定数据传输阶段流程

在 4.1.1 节网络分簇结构生成阶段之后就进入稳定的数据传输阶段。在网络分簇结构生成阶段结束之后，网络最终的簇结构已经生成，并且基站已经将簇首节点的信息和簇内节点的信息发送给各个传感器节点。也就是每个传感器节点都了解了它自己工作时需要的信息，比如簇首节点知道了它的簇内成员节点信息和它的上下跳簇首信息以及发送给它的下跳簇首节点所需要的发送功率；而簇内节点则知道了它所属的簇首节点以及发送给簇首节点所需要的功率以及何时发送等信息。具备了以上信息，各个节点就可以按照协议的规定进行工作了。

网络开始工作后簇内节点持续地监测数据，并按照簇首节点给自己安排的时序将监测的数据按照特定的发送功率发送给自己的簇首节点。簇内节点通过查找存储的簇首节点编号和发送给簇首节点所需要的发送功率获得发送监测数据参数，然后以簇内节点编号为标

标识符，将采集的数据按照查得的功率发送给簇首节点。簇首节点查看发来的数据信息的标识符是否为本簇的簇内节点，若是则接收，不是则舍弃。到整个簇的簇内节点都将数据发送给簇首节点后，簇首节点对数据进行一定的融合处理，然后查找存储的多跳路径的下跳簇首节点编号和发送功率，然后以自己的簇首节点编号为标识符，将融合后的数据按照查得的功率发送给下跳簇首节点，簇首节点发现是其他簇首节点发来的融合数据信息就将发来信息的簇首节点标识符和存储的上跳簇首节点编号对比，若相同则接收，若不是则舍弃。然后重复这一个过程，继续将该数据传送给下跳簇首节点，直到将数据传到基站。以上工作在各个簇内过程不断重复，至于簇首节点出现低剩余能量情况时按照 3.3 节的局部簇首更新机制处理。簇内节点出现剩余能量接近为 0 时会发送一个消息给簇首节点，簇首节点会将该簇内节点从接收周期表中删除。整个网络一直工作，直到所有传感器节点能量耗尽而死亡。

图 4-2 显示了某个传感器节点从采集数据到将发送数据到基站的整个流程，而整个网络的工作过程就是所有按照图 4-2 的流程图进行工作的簇内节点的集合。

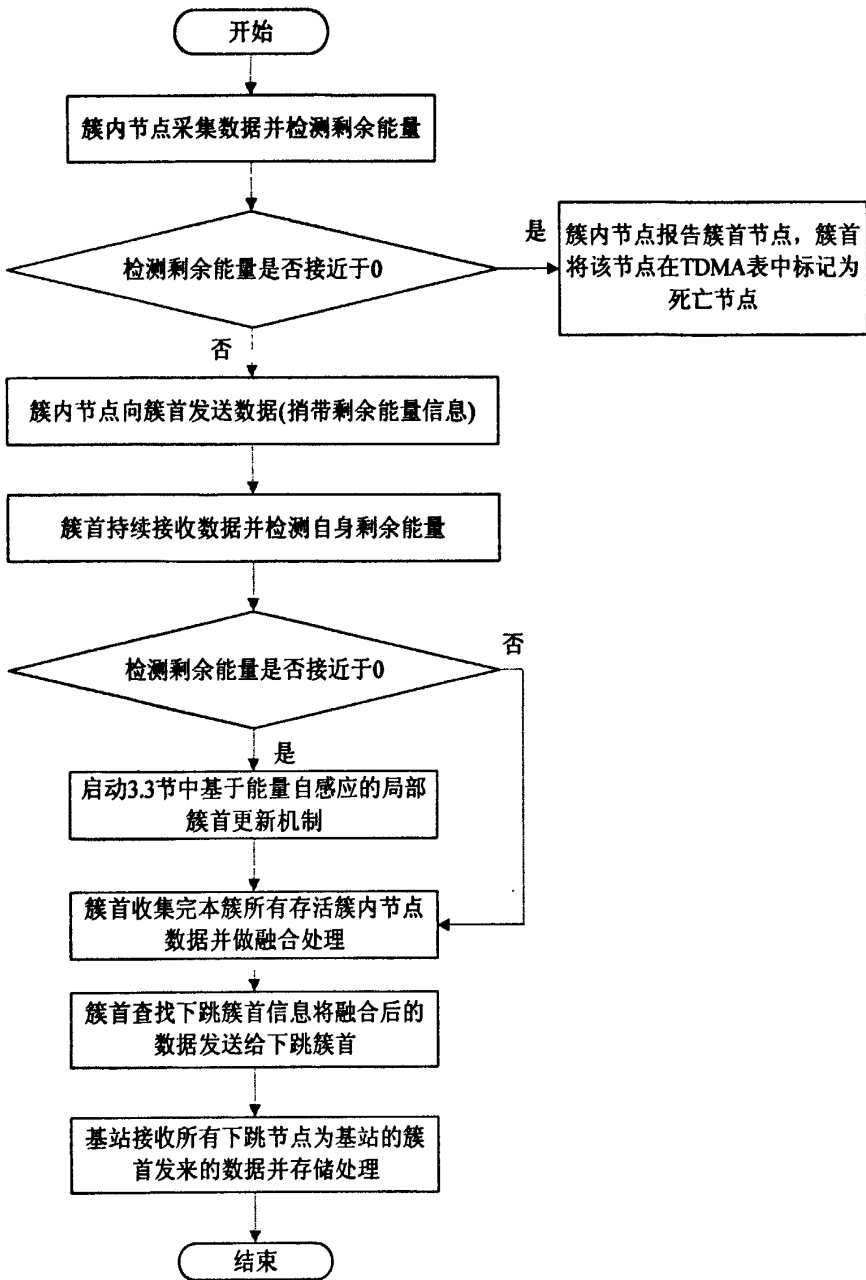


图 4-2 SI-CRP协议网络稳定工作阶段流程图

4.2 分簇实验仿真分析

本节以MATLAB为仿真工具，通过实验比较了SI-CRP协议分簇方法和无线传感器网络经典分簇路由协议LEACH协议的分簇效果。首先介绍实验场景：100个传感器节点随机地被布撒在一个200米为边长的正方形区域。在分簇实验中，假设基站处于该正方形区域的正中心。实验中用相同图形代表同一个簇的簇内节点，黑色图形代表簇内节点，红色图形代表簇首节点。另外为了更明显地对比SI-CRP协议和LEACH协议分簇效果的对比，这里的分簇实验中的传感器节点位置分布都是相同的。

首先来看看LEACH协议的分簇效果，前面已经论述了LEACH协议的随机选举簇首节点的机制会导致分簇不均匀的现象，比如导致簇首个数不确定，簇首可能分布很不均匀，簇内节点个数不均衡的情况。下面用MATLAB仿真LEACH协议的分簇效果，假设LEACH协议簇首节点选择阈值公式中期望簇首选择公式 K 为 $100 \times 5\% = 5$ 个。

图4-3显示的是一种较极端的情况，虽然期望簇首个数为5个，但是实际LEACH协议只分得2个簇首，而且每个簇负责太多簇内节点，簇内节点到簇首节点距离差距也很大。

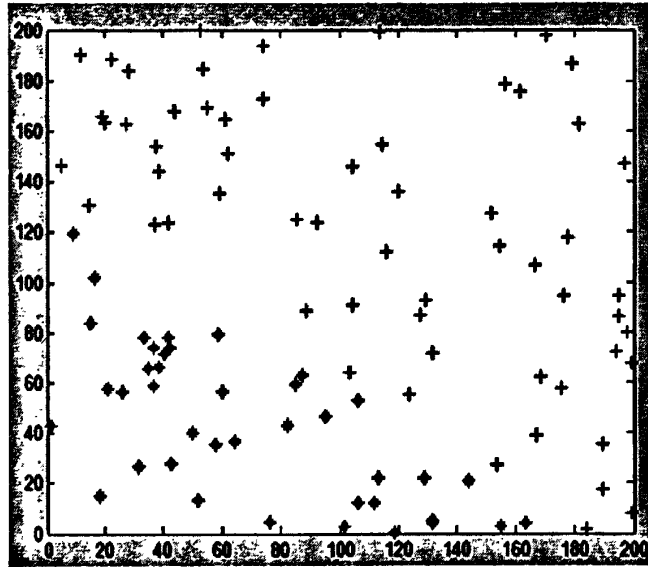


图 4-3 LEACH协议分簇效果1

图4-4显示的是相同传感器节点分布的LEACH协议的分簇效果。可以看到虽然分得了5个簇首，可是簇首的分布却很差，比如其中几个簇首靠的很近，还有个簇首位于网络边界，簇内也存在节点个数不均衡的问题。

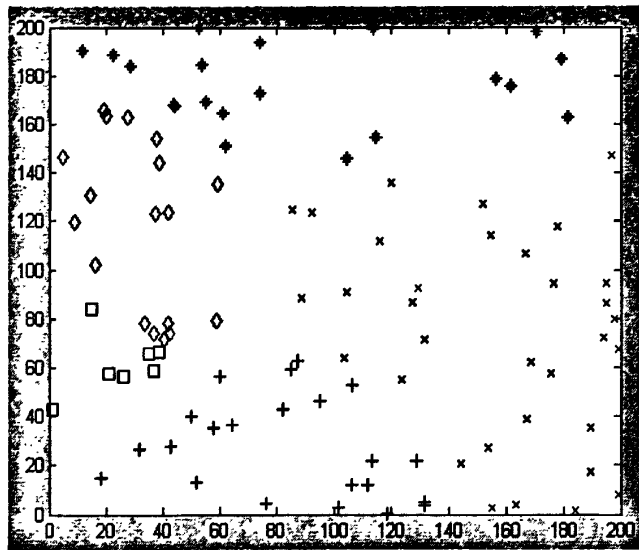


图 4-4 LEACH协议分簇效果2

从图4-5中还可以看到，虽然这个分簇效果每个簇负担的簇内节点个数差不多，但是仍然存在簇首位置不合理的缺点。另外由于LEACH协议选举的簇首位置不合理，虽然普通节点选择离自己最近的簇首加入，可是仍然存在簇内节点到簇首节点距离过大的缺点。

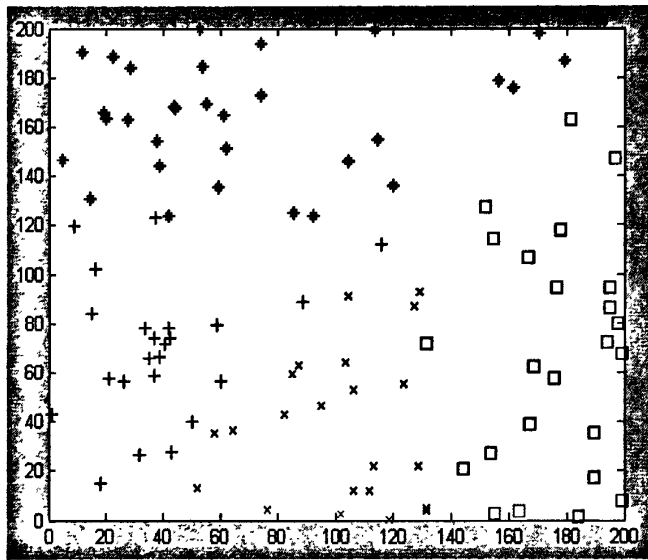


图 4-5 LEACH协议分簇效果3

上面几个图显示了标准LEACH协议在相同传感器节点位置分布时的三个分簇效果，从这几个图可以直观地看到LEACH协议随机选举簇首机制带来分簇不均匀的缺陷。

下面我们来看看本文提出的SI-CRP协议基于改进离散粒子群算法的分簇方法的效果。

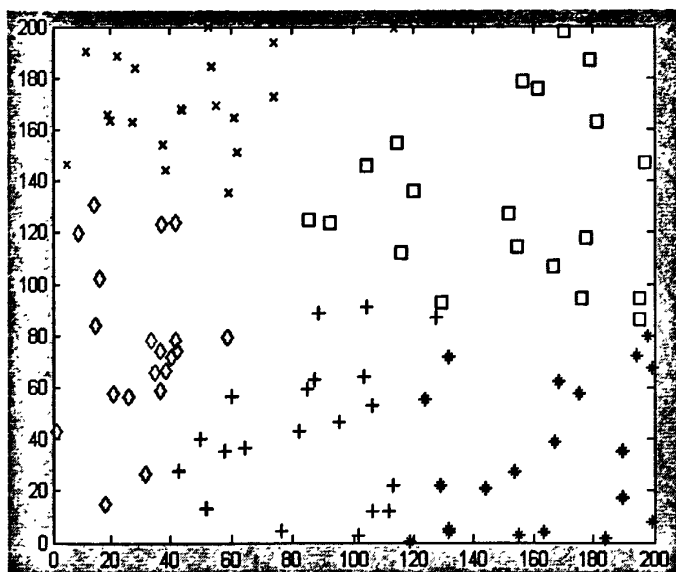


图 4-6 SI-CRP协议分簇效果1

图4-6和图4-7显示了与LEACH协议分簇实验相同传感器节点分布的两次运行SI-CRP协议分簇方法的效果图。从图中可以看到，SI-CRP协议分簇方法使簇首节点均匀地分布在整个网络监测区域，每个簇中的簇内节点都能比较紧密地围绕在簇首节点周围，另外由于SI-CRP协议分簇方法规定每个簇中的簇内节点个数都为19个，因此也不存在LEACH协议簇内节点个数不均衡的问题。从LEACH协议和SI-CRP协议的分簇效果图可以看出，SI-CRP协议的分簇方法相比LEACH协议簇首位置合理，分簇均匀。

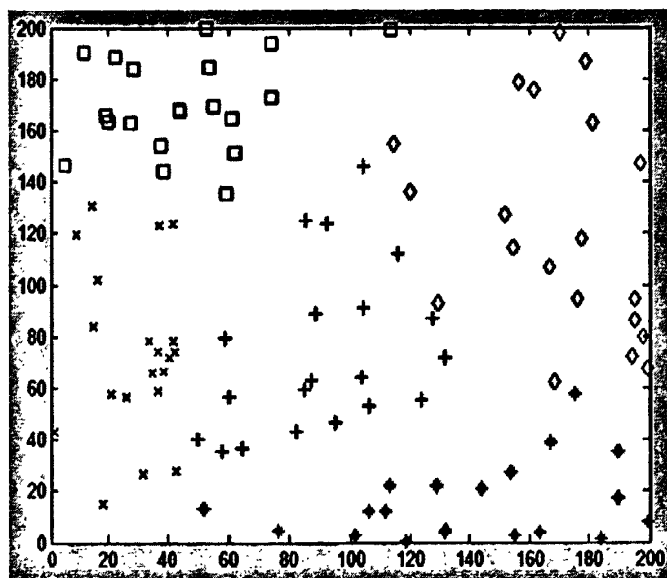


图 4-7 SI-CRP协议分簇效果2

为了检验3.1.1节中提出的适应度函数的合理性，这里比较LEACH协议和SI-CRP协议分

簇方法的适应度函数(公式3-3)的值。公式3-3分成 $f1$ 和 $f2$ ，其中 $f1 = \text{mean} \left(\sum_{j=1}^n \left(\sum_{i=1}^m D_{ji} - jch \right) \right)$ ，

$$f2 = \text{mean} \left(\sum_{j=1}^n \left(\text{var} \left(\sum_{i=1}^m D_{ji} - jch \right) \right) \right)。$$

表 4-1 LEACH协议和SI-CRP协议分簇方法适应度函数值对比

	f1	f2	F=f1*f2
LEACH	2543.1	1291.5	3284400
SI-CRP	715.6	227.5	162810

上表数据是相同分布的传感器节点连续各运行3次LEACH协议和SI-CRP协议适应度函数的平均值。从表中可以看出，无论是 $f1$ 的值还是 $f2$ 的值，LEACH协议的值都比SI-CRP协议大好几倍，总适应度函数值LEACH协议是SI-CRP的20倍左右。这说明LEACH协议中簇首和簇内节点不但距离总和过大而且距离大小非常不均匀。

4.3 协议运行仿真分析

在本小节里，将用 MATLAB 对本文提出的无线传感网络智能分簇路由协议 SI-CRP 协议的运行过程进行仿真。为了说明 SI-CRP 协议在节能方面的有效性，本文将以相同的仿真条件仿真 LEACH 协议的运行过程，并从传感器节点的生命周期以及网络平均能耗等方面比较 SI-CRP 协议和 LEACH 协议的能量有效性。

下面首先描述仿真 SI-CRP 协议和 LEACH 协议的实验条件：本文假定 100 个传感器节点随机地布撒于一个 200*200 的正方形区域，假设该区域四个顶点坐标为(0,0)，(0,200)，(200,0)，(200,200)，则基站位于坐标为(300,300)的位置。为了准确地比较 SI-CRP 协议和 LEACH 协议的能量有效性，仿真实验采用和 4.2 节相同的方法，即 SI-CRP 协议和 LEACH 协议在运行时都采用完全相同的节点位置分布。两种协议运行时传感器能量情况如下：每个传感器节点的初始能量都为 0.5J，公式(3-1)和(3-2)中的能量消耗参数为 $E_{elec}=50$ nJ/bit， $E_{fs}=10$ pJ/(bit*m2)， $E_{mp}=0.0013$ pJ/(bit*m4)， $d_0=87$ M。另外 SI-CRP 协议和 LEACH 协议都是分簇协议，且有相同的网络模型，为了突出 SI-CRP 协议能量有效性，仿真实验规定协议在将网络簇结构确定完以后都按照轮的概念来进行，在每轮中 SI-CRP 协议和 LEACH 协议都由簇内节点产生一个数据包，然后将该数据包在自己规定的周期内发送给自己的簇首

节点, 簇首收集所有簇内节点的数据包, 并做融合处理(融合率为 0.6), 然后按照各自协议的流程将数据发送给基站。除了两种协议稳定运行阶段的数据包, 用于簇结构建立的消息包的能量消耗也被计算在内, 规定消息包大小为 100 bit, 数据包大小为 4000 bit。另外规定 LEACH 协议每稳定运行 100 轮进行全局的簇结构重构, 这能够符合 LEACH 协议稳定运行阶段时间远大于簇建立阶段的要求。

SI-CRP 协议在运行过程中, 基于改造离散粒子群算法的分簇方法各参数如下: 改造粒子群的最大迭代次数 $maxiter$ 为 1000 轮, 初始惯性权重 $w_{max}=0.7$, 迭代结束的惯性权重 $w_{min}=0.1$, 个体学习因子 $C_1=0.2$, 群体学习因子 $C_2=0.2$, 如 3.1.3 小节中所述在总迭代前 20% 轮和后 20% 轮使用线性递减惯性权重调整方法, 中间 60% 轮使用基于粒子多样性的惯性权重调整方法, 且始终保持 $C_1=C_2$ 并且 $C_1+C_2+w=1$ 。而 LEACH 协议中期望簇首概率 P (即 K/N)=0.05。其余计算全局最优簇首的细节和过程均如 3.1 节中所述。

而在基于改造蚁群算法运行过程中的各参数如下: 信息素浓度影响因子 $\alpha=3$, 期望函数影响因子 $\beta=4$, 信息素挥发系数 $\rho=0.7$ 。其余计算簇首间多跳路径的细节和过程均如 3.2 节中所述。

图4-8比较了LEACH协议与SI-CRP协议的节点生命周期, 其中纵坐标代表了整个网络中传感器节点剩余能量大于0的也就是存活的传感器节点个数, 而横轴是协议运行的轮数。由于该仿真实验规定了LEACH协议与SI-CRP协议每轮所发的数据包个数相同, 其他传感器节点工作参数也都相同, 因此传感器节点的生命周期长短和网络能耗就只与网络工作的路由协议的好坏有关系, 因此该仿真实验能明确比较出LEACH协议与SI-CRP协议这两种分簇路由协议的能量有效性。

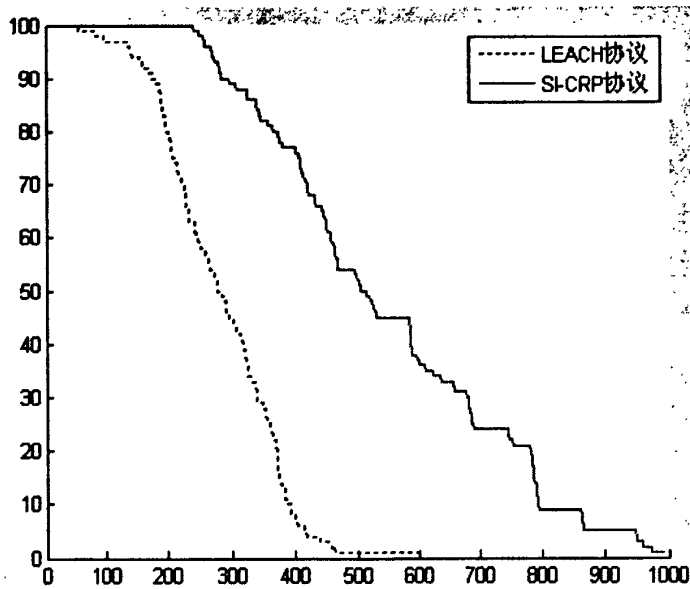


图 4-8 LEACH 与 SI-CRP 节点生命周期比较

从图中可以看到，相比于LEACH协议，SI-CRP协议的平均节点生命周期延长了近1倍。为了更清楚地比较两种分簇路由协议在运行过程中的情况，表4-2对协议运行的数据进行了统计。

表 4-2 LEACH协议和SI-CRP协议仿真运行数据

	第一个节点死亡轮数	最后一个节点死亡轮数	前 95 个节点死亡轮数	网络每轮平均能耗 (以 95 个节点死亡轮数为准)
LEACH 协议	75	600	412	$12.13 \times 10^{-2} \text{J}$
SI-CRP 协议	241	996	862	$5.8 \times 10^{-2} \text{J}$

从表中可以看出，LEACH协议第一个节点死亡的轮数是75轮，而本文提出的SI-CRP协议由于采用了基于能量自感应的局部簇首更新机制，第一个节点死亡的轮数是241轮，比LEACH协议延长了2倍还多。LEACH协议最后一个节点死亡的轮数是600轮，而SI-CRP协议是996轮。这里有必要提的是前95个节点死亡轮数，由于仿真实验假定LEACH协议和SI-CRP协议的期望簇首是5个，因此仿真实验规定在整个网络的传感器节点在死亡到只剩5个时，不再进行分簇处理，而是直接将数据单跳发送给基站。因此从能耗考察的角度看，我们比较关注的是前95个传感器节点工作时的网络工作能耗。从表中可以看出，LEACH协议前95个节点死亡轮数是412轮，而SI-CRP协议是862轮，相比于LEACH协议延长了近1倍。而从前95个节点工作的网络每轮平均能耗来看，LEACH协议每轮通信的平均能耗是

$12.13 \times 10^{-2} \text{J}$, SI-CRP协议在每轮和LEACH协议发送相同数据包的情况下每轮通信的平均能耗是 $5.8 \times 10^{-2} \text{J}$, 能量比LEACH协议节省了一半还多。

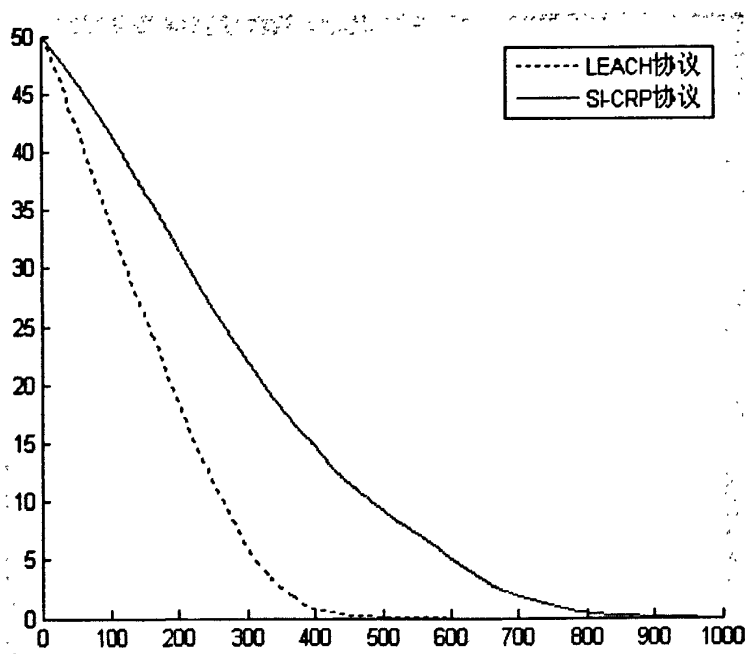


图 4-9 LEACH与SI-CRP能耗示意图

图4-9比较了仿真实验中LEACH协议和SI-CRP协议总的网络能耗情况。横坐标是协议运行的轮数, 纵坐标是整个网络总的剩余能量(初始网络中100个传感器节点的总剩余能量为 $0.5\text{J} \times 100 = 50\text{J}$)。从该图中可以清楚地看到LEACH协议网络中总的剩余能量随仿真迭代的进行下降地很快, 而SI-CRP协议网络剩余能量的下降曲线的下降趋势要比LEACH协议平缓地多。

4.4 本章小结

本章主要介绍了SI-CRP协议的实现流程和并从分簇效果和协议整体运行能耗与节点生命周期这几个方面仿真对比了SI-CRP协议和LEACH协议。

4.1节中详细介绍了SI-CRP协议工作的网络簇结构生成阶段和网络稳定的数据传输阶段的实现细节和流程, 并给出每个阶段的流程图。4.2节中用仿真实验比较了SI-CRP协议和LEACH协议在相同传感器节点分布的条件下的分簇效果。仿真实验采用位置完全相同的传感器节点进行对比, 使分簇效果的对比更加具有说服力。仿真实验对相同的传感器节点分布运行两次SI-CRP协议和LEACH协议, 得到的分簇结果显示SI-CRP协议的分簇方法相比

LEACH协议簇首位置合理，分簇均匀。4.3节中仿真SI-CRP协议工作的完整流程，并以传感器节点生命周期和网络能耗为考察指标，比较了SI-CRP协议和LEACH协议的能量有效性。仿真实验中SI-CRP协议和LEACH协议也同样作用于位置完全相同的传感器节点分布。最后的仿真结果表明，SI-CRP协议相比于LEACH协议，节点生命周期延长1倍多，网络能耗也节省一半以上。

第5章 总结与展望

5.1 总结

无线传感器网络是一种应用前景非常广阔的新技术，它能广泛地应用于军事、环境检测和预报、健康护理、智能家居、建筑物状态监控等领域，因此近年来吸引了国内外诸多学者对无线传感器网络技术的研究。

在无线传感器网络技术研究领域，路由协议一直是一个最热门的研究方向。由于无线传感网络中的传感器节点能量非常有限，因此无线传感网络技术研究领域大多都是以节能为首要研究目标的。而网络路由协议的设计将直接决定整个网络的拓扑结构以及传感器节点数据传输机制，这两个因素是影响传感器节点能量耗费的最直接的因素。因此设计一种能量有效的无线传感网络路由协议成为无线传感器网络技术研究中最重要最热门的研究方向。

无线传感网络分簇路由协议是无线传感网络路由协议中的一类，这种路由协议不需要太多路由控制信息，扩展性好，适用于大型网络应用。但是以 LEACH 协议为代表的无线传感网络分簇路由协议存在着分簇不均匀，簇首与基站通信能耗过大等缺点。本文以 LEACH 协议分簇网络结构为基础，结合群智能算法研究领域，提出了一种基于群智能算法的无线传感网络智能分簇路由协议—SI-CRP 协议。本文主要工作和创新点如下：

1) 介绍和分析了无线传感网络中经典的几种分层路由协议，特别是详细分析了经典分簇路由协议 LEACH 的优点和缺点。

2) 介绍了粒子群优化算法以及相关的研究成果，在对 LEACH 协议分簇不均匀问题和粒子群优化算法深入分析的基础上，提出了 SI-CRP 协议中最重要的创新点：根据无线传感网络分簇问题的特点，提出粒子群优化算法的目标适应度函数，并改造基本粒子群算法为适用于解决分簇问题的离散粒子群算法，用它计算整个网络中位置最优的节点为簇首节点。在改造粒子算法群过程中包括创新地提出混合惯性调整策略及两种加速求得最优解的启发式算法。

3) 介绍了蚁群优化算法以及相关的研究成果，在对 LEACH 协议簇首到基站通信能耗过大及蚁群优化算法深入分析的基础上，提出了 SI-CRP 协议中第二个创新点：改造标准蚁群优化算法，以传感器节点通信能耗为权值，计算无线传感网络中簇首到基站的多跳路

径,从而找到从某簇首节点出发,经由其他簇首节点到达基站的最小能耗的多跳路径。

4) 研究 LEACH 协议中周期性重新分簇机制的弊端,创新地提出 SI-CRP 协议中第三个创新点:基于能量自感应的局部簇首重选机制,通过该机制,大大减小了重新选举簇首所消耗的能量,并且只需在协议初始时对网络分簇,协议以后运行过程中网络分簇结构不变,不再进行全局性周期性的重新分簇。

5) 提出整个 SI-CRP 协议,进行编码,并仿真比较了 LEACH 协议和 SI-CRP 协议的分簇效果以及协议整个工作过程中能耗情况和节点生命周期情况。

仿真结果表明,SI-CRP 协议通过基于改造粒子群计算得到的簇结构比 LEACH 协议随机机制得到的簇结构要均匀得多,簇首位置也要合理得多;对 SI-CRP 协议和 LEACH 协议整个工作过程的仿真结果表明,在相同的仿真条件和工作模式下,SI-CRP 协议比 LEACH 协议节点生命周期延长 1 倍多,网络能耗也小一半多。

LEACH 协议虽然提出了一种适合大型传感网络的分簇结构模型,可是它本身存在很多的缺陷,这些缺陷造成整个传感器网络在运行过程中有很多不必要的能量浪费,这不符合无线传感网络路由协议节能的首要目标。本文提出的 SI-CRP 协议只需要基站在路由协议开始运行前计算并生成优化的网络分簇结构,那传感器节点在工作过程中大大节省能量,使网络生命周期比 LEACH 协议延长一倍左右。这就是本文的研究意义。

本文提出的 SI-CRP 协议可以应用于大型的网络监测,比如环境监测或军事应用领域,这类应用场合的传感器节点经常是大量传感器节点一次布撒就不能或者难以再为传感器节点更换电源。SI-CRP 协议高效节能的特点可以使类似网络监测应用的生命周期大大延长。

5.2 展望

本文提出的 SI-CRP 协议通过对 LEACH 协议中缺陷机制的改进,在能耗方面有大幅度的改进,然而 SI-CRP 协议并不是完美无缺的,有进一步研究的必要。

首先 SI-CRP 协议中要求传感器节点具有定位功能的 GPS 设备,用来获取自己的位置坐标,而现有的传感器节点都比较廉价,增加 GPS 设备将会增加网络组建的成本。另外一个使传感器节点获得位置信息的方法是无线传感器节点定位算法的研究,这也是一个无线传感器网络研究领域一个热门研究方向。本人下一步的研究目标就是研究低能耗的传感器节点定位算法,使本文提出的 SI-CRP 协议不再依赖硬件设备,并且使节点定位算法定位传感器节点能耗尽可能少,不使 SI-CRP 协议失去节能的优点。

另外SI-CRP协议在工作过程中没有考虑传感器节点由于各种原因突然失效的情况，比如各种环境破坏或人为破坏。当簇内节点突然失效时，SI-CRP协议还不至于无法工作，即在某个簇内节点应该发送的时序周期内一直收不到该簇内节点数据。而若是簇首节点突然失效，情况就比较严重了，SI-CRP协议没有判断簇首节点是否失效也没有做出任何对簇首节点失效的补救措施，因此一旦簇首节点失效，簇内节点会在每个周期内向无法工作的簇首发送数据，造成不必要的能量浪费。因此以后需要对节点失效的情况进行研究，首先需要有一种机制判断节点是否失效以及确定某个簇内节点或者簇首节点失效时协议的补救措施。

如果能针对以上两点对SI-CRP协议进行改进，SI-CRP协议将变得更加实用并且具有更好的健壮性。

参 考 文 献

- [1] 孙利民,李建中,陈渝等. 无线传感器网络[M].北京:清华大学出版社,2005.
- [2] Ten emerging technologies that will change the world. *Technology Review*. Feb.2004,Vol.106,NO.1:22-49P.
- [3] 王永斌,屈晓旭.无线传感网络:体系结构与协议.北京:电子工业出版社,2007,10-19.
- [4] 彭静,刘光祐,谢世欢. 无线传感器网络路由协议研究现状与趋势[J].计算机应用研究,2007,2:4-9.
- [5] Haas ZJ, Halpern JY, Li L. Gossip-Based ad hoc routing[A]. Proc. of the IEEE INFOCOM[C]. New YORK. IEEE Communications Society,2002,1707-1716.
- [6] Intanagonwiwat C, Govindan R, Estrin D, Heidemann J. Directed diffusion for wireless sensor networking[A]. *IEEE/ACM Trans. on Networking*[C],2003,11(1): 2-16.
- [7] Sohrabi K, Gao J, Ailawadhi V, Pottie GJ. Protocols for self-organization of a wireless sensor network[A]. *IEEE Personal Communication*[C],2007,7(5):16-27.
- [8] Heinzelman WR, Kulik J, Balakrishnan H. Adaptive protocols for information dissemination in wireless sensor networks[A]. Proc. of the ACM MobiCom' 99[C]. Seattle: ACM Press, 1999, 174-185.
- [9] Braginsky D, Estrin D. Rumor routing algorithm for sensor networks[A]. Proc. of the 1st Workshop on Sensor Networks and Applications[C]. Atlanta: ACM Press, 2002, 22-31.
- [10] Heinzelman W, Chandrakasan A, Balakrishnan H. Energy-Efficient Communication Protocol for Wireless Microsensor Networks[A]. Proc. of the 33rd Hawaii International Conference on System Sciences[C]. IEEE Computer Society, 2000, 3005-3014.
- [11] Lindsey S, Raghavendra CS. PEGASIS: Power-Efficient Gathering in Sensor Information Systems[A]. Proc. of the IEEE Aerospace Conf[C]. Montana: IEEE Aerospace and Electronic Systems Society, 2002, 1125-1130.
- [12] Manjeshwar A, Grawal DP. TEEN: A protocol for enhanced efficiency in wireless sensor networks[A]. Proc. of the 15th Parallel and Distributed Processing Symp[C]. San Francisco: IEEE Computer Society, 2001, 2009-2015.
- [13] Younis O, Fahmy S. HEED: A hybrid, energy-efficient, distributed clustering approach for ad-hoc sensor networks[A]. *IEEE Trans. on Mobile Computing*[C], 2004, 3(4): 660-669.
- [14] 张伟华, 李腊元, 张留敏, 王选政. 无线传感器网络LEACH协议能耗均衡改进[J]. 传感技术学报, 2008, 21(11): 1918-1922.
- [15] Wairagu G. Richard, "EXTENDING LEACH ROUTING ALGORITHM FOR WIRELESS SENSOR NETWORKS", in Msc. Thesis, Makerere University, March, 2009.
- [16] M.J.Handy, M.Haase, D.Timmermann. Low Energy Adaptive Clustering Hierarchy with Deterministic Cluster-Head Selection[A]. *IEEE Trans. On MWCN*[C], 2002.
- [17] 胡钢, 谢冬梅, 吴元忠. 无线传感器网络路由协议LEACH的研究与改进[J]. 传感技术学报. 2007, 20(6): 1391-1396.
- [18] 王国芳, 李腊元, 李春林, 刘会静. 无线传感器网络中基于能量约束的簇首多跳算法[J]. 传感技术学报, 2009, 22(7): 997-1001.
- [19] Heinzelman W. Application - Specific protocol architectures for wireless networks [D] .

- Boston : Massachusetts Institute of Technology,2000.
- [20] Murata T,Ishibuchi H. Performance evaluation of genetic algorithms for flowshop scheduling problems[A]. Proc. of the 1st IEEE Conf. on Evolutionary Computation[C], Orlando: IEEE Press, 1994.812-817.
- [21] 梁英,于海斌,曾鹏. 应用PSO优化基于分簇的无线传感器网络路由协议[J].控制与决策,2006,21(4):453-457.
- [22] 杭海存,郭爱煌,舒文杰. 基于LEACH与蚁群算法的WSN路由机制及性能分析[J].传感技术学报,2008,21(10):1735-1737.
- [23] Kennedy J, Eberhart R. Particle Swarm Optimiaztion[A]. Proceeding of The IEEE International Conference on Neural Networks.Piscataway[C], NJ:IEEE Service Center,pp 1942-1948,1995.
- [24] Shi Y, Eberhart R C. A modified particle swarm optimizer [A]. Proc. IEEE International Conference on Evolutionary Computation. Piscataway[C], NJ : IEEE Press, 1998: 69-73.
- [25] Coello, C.A.C, Pulido,G.T, Lechuga MS, M.S. Handling multiple objectives with particle swarm optimization[A], IEEE Transactions on Evolutionary Computation[C], Vol. 8, No. 3.(2004) 256-279.
- [26] Wachowiak, M. P., Smolikova,R., Zheng, Y. F., Zurada, J. M.,Elmaghraby A. S:An approach to multimodal biomedical image registration utilizing particle swarm optimization[A], IEEE Trans. Evol[C]. Comput., Vol.8, No.3. (2004) 289-301.
- [27] Fukuyama, Y., Yoshida, H.: A Particle Swarm Optimization for Reactive Power and Voltage Control in Electric Power Systems[A]. Proc.Congress on Evolutionary Computation[C], Seoul, Korea. Piscataway, NJ: IEEE Service Center. (2001) 87-93.
- [28] 王雪,王冕,马俊杰.无线传感网络移动节点位置并行微粒群优化策略[J].计算机学报.2007.30(4).563-568.
- [29] Arabshahi,P. Gray,A. Kassabalidis, I. et al. Adaptive routing in wireless communication networks using swarm intelligence[A]. AIAA 19th Annual Satellite Communications System Conference Toulouse[C], rance. 2001.
- [30] 寇晓丽. 群智能算法及其应用研究[D]. 西安:西安电子科技大学,2009.
- [31] 张丽平. 微粒群算法的理论与实践[D]. 杭州:浙江大学,2005.
- [32] Eberhart R,Shi Y. Comparing inertia weights and constriction factors in particle swarm optimization[A]. IEEE Congress on Evolutionary Computation. Piscataway[C]: IEEE Service Center, 2000: 84- 88.
- [33] Trelea I.The particle swarm optimization algorithm: convergence analysis and parameter selection [J].Information Processing Letters, 2003, 85(6) : 317- 325.
- [34] Shi Y, Eberhart R. Empirical study of particle swarm optimization[A]. Proceedings of the 1999 Congress on Evolutionary Computation[C],1999:1945-1950.
- [35] Zheng Yong- ling, Ma Long- hua, Zhang Li- yan, et al. On the convergence analysis and parameter selection in particle swarm optimization[A]. Proceedings of the Second International Conference on Machine Learning and Cybernetics[C]. Xi' an,2003:1802-1807.
- [36] 崔红梅,朱庆保. 微粒群算法的参数选择及收敛性分析[J]. 计算机工程与应用,2007,43(23):89-91.
- [37] 王丽,王晓凯. 一种非线性改变惯性权重的粒子群算法[J]. 计算机工程与应用,2007,43(4):47-48.
- [38] 刘建华,樊晓平,瞿志华. 一种惯性权重动态调整的新型粒子群算法[J]. 计算机工程与应用,

2007,43(7):68-70.

- [39] 韩江洪,李正荣,魏振春.一种自适应粒子群优化算法及仿真研究[J].系统仿真学报, 2006,18(10):2969-2971.
- [40] 钟一文. 智能优化方法及其应用研究[D]. 杭州:浙江大学, 2005.
- [41] Kennedy J, Eberhart R C. A discrete binary version of the particle swarm algorithm[A]. Proc. the World Multi conference on Systemics[C],Cybernetics and Informatics,IEEE Press, 1997:4104-4109.
- [42] Kenny J , Spears W M. Matching algorithm to problems: an experimental test of the particle swarm and some genetic algorithm on the multimodal problem generator [A]. Proc. Int. Conf . on Evolutionary Computation. Piscataway[C], NJ: IEEE Press, 1999 : 78-83.
- [43] 李宁,邹彤,孙德宝. 带时间窗车辆路径问题的粒子群算法[J]. 系统工程理论与实践,2004,24 (4) : 131-135.
- [44] Salman A, Ahmad I, Al-Madani S. Particle swarm optimization for task assignment problem [J]. Microprocessors and Microsystems,2002,26(8):363-371.
- [45] 霍霄华,陈岩,朱华勇等. 多UCAV协同控制中的任务分配模型及算法研究[J]. 国防科技大学学报, 2006, 28 (3) :83-88.
- [46] M.Clerc.Discrete Particle Swarm Optimization, illustrated by the Traveling Salesman Problem New Optimization Techniques in Engineering, Springer, 2004, pp.219-239.
- [47] Afshinmanesh F, Marandi A, Rahimi-Kian A. A Novel Binary Particle Swarm Optimization Method Using Artificial Immune System[C] EUROCON, 2005.
- [48] 过文亮.无线传感器网络节能路由协议研究[D].上海:上海大学硕士论文,2009年1月.
- [49] Marco Dorigo, Vittorio Maniezzo,Alberto Colomi. The Ant system: Optimization by a Colony of Cooperating Agents[J]. IEEE Transactions on Systems , Man , and Cybernetics, 1996, 26 (1):1-13.
- [50] 冯春时. 群智能优化算法及其应用[D]. 安徽:中国科学技术大学, 2009.
- [51] Wendi B Heinzelman, Anantha P Chandrakasan, Hari Balakrishnan. An Application-Specific Protocol Architecture for Wireless Microsensor Networks[A]. IEEE Transactions on Wireless Communications[C],2002,1 (4) :6602670.

致 谢

在浙江工业大学计算机科学与技术学院攻读硕士学位的两年半时间里，我得到了众多老师和同学的关怀与帮助，在此对他们表示真诚的感谢。

感谢导师王万良教授在我硕士学习阶段的辛勤教导和培育，其高深的学术造诣，严谨的治学态度，及和蔼可亲、平易近人的宽宏气度感召和激励着每一个弟子。身为其学生，倍感骄傲和荣幸。感谢导师对我学习乃至生活上的关心和指导，对每一次选择的理解和支持。在此谨向王老师致以最诚挚的谢意！

本篇硕士论文是在王万良教授和范兴刚副教授两位导师的精心指导下完成。从论文的选题、开题、文章结构的构筑，到最后的定稿，都得到恩师的细心指导和建议。十分感谢蒋一波老师在小论文、毕业设计中的悉心指导和宝贵建议。从对论文构架的迷感到最后的完稿，每一次修改，每一点进步无不倾注了导师的心血。每一次耐心的指导和鼓励，都让我铭记在心。其严谨的学术态度、敏捷的思维、充满创造力的才华和极具亲和力的人格魅力也给我留下了深刻的印象。在此向两位导师致以诚挚的谢意！

也感谢同窗，在这里对他们给予的支持和帮助致以由衷的感谢。感谢郑泽萍，王翊，李川，陈青丽，濮约庆，陆洋，岑跃峰等同学对我完成论文的帮助，同窗好友的存在使得我的求学之路变得充实而又欢愉。

最后祝愿母校有更美好的明天，早日跻身于名校行列。

二〇一〇年十一月于浙江工业大学

攻读学位期间参加的科研项目和成果

录用和发表的论文

- [1] 侯佳斌,范兴刚,王万良等. Clustering Strategy of Wireless Sensor Networks based on Improved Discrete Particle Swarm Optimization. 2010年第6届 ICNC 国际会议论文集.
- [2] 范兴刚,侯佳斌,王万良等. 基于 DPSO 的智能 WSN 分簇路由算法[J]. 传感技术学报 (已录用)

参加的科研项目

- [1] 国家自然科学基金: 移动自组织网络中实时流媒体传输的控制与调度协同方法 (61070043)。
- [2] 浙江省自然科学基金: 无线传感器网络的分簇安全研究 (Y1080374)。
- [3] 浙江省教育厅项目: 无线传感器网络的分簇安全研究 (20070184)。

科研成果

- [1] 发明专利: 一种基于 DPSO 的智能无线传感网络分簇控制方法(申请号: 201010270654.8). 已经被国家专利局受理. 第二作者.