

高效节能的传感器网络数据收集和聚合协议*

刘明[†], 龚海刚, 毛莺池, 陈力军, 谢立

(计算机软件与新技术国家重点实验室(南京大学), 江苏 南京 210093)

A Distributed Energy-Efficient Data Gathering and Aggregation Protocol for Wireless Sensor Networks

LIU Ming[†], GONG Hai-Gang, MAO Ying-Chi, CHEN Li-Jun, XIE Li

(State Key Laboratory for Novel Software Technology (Nanjing University), Nanjing 210093, China)

+ Corresponding author: E-mail: wing.lm@gmail.com, http://www.nju.edu.cn

Received 2004-09-15; Accepted 2004-12-08

Liu M, Gong HG, Mao YC, Chen LJ, Xie L. A distributed energy-efficient data gathering and aggregation protocol for wireless sensor networks. *Journal of Software*, 2005,16(12):2106–2116. DOI: 10.1360/jos162106

Abstract: This paper proposes a distributed energy-efficient data gathering and aggregation protocol, in which a node, according to its residual energy and the strength of the signal received from its neighboring nodes, independently makes its decision to compete for becoming a cluster head. In addition, assume that the inter-cluster communication data in a multi-hop manner is sent to the designated node, it then sends the gathered data by the whole network to the base station. DEEG also proposes a simple approach to solve the cluster coverage problem. With the increase in node density, this approach produces a linear sensor network lifetime in the number of nodes. Experimental results have shown that compared with another two data gathering and aggregation protocols--- leach and PEGASIS, the DEEG algorithm, in the best case, can lead to the increase of sensor network lifetime by 1800% and 300% respectively. Moreover, since all the nodes in the sensor network die in the last 40 rounds (the last node dies) in DEEG protocol, the reliability of the sensing information in DEEG is higher than that in LEACH and PEGASIS.

Key words: wireless sensor networks; cluster; spanning tree; energy-efficient; cluster coverage

摘要: 提出了一种分布式的高效节能的传感器网络数据收集和聚合协议 DEEG. 此协议中节点自主地根据其剩余能量以及邻居节点的信号强度来竞争簇头, 同时为了减小簇头节点的能量开销, 簇头之间以多跳方式将收集到的数据发送到指定的簇头节点, 然后通过该节点将整个网络收集的数据发送到基站. 此外, 该协议还提出了一种简单的簇覆盖方法, 使得当节点密度提高时, 传感器网络寿命相应于节点数量呈线性增长. 实验证明, 在没有使用簇覆盖方

* Supported by the National High-Tech Research and Development Plan of China under Grant No.2001AA113050 (国家高技术研究发展计划(863)); the National Grand Fundamental Research 973 Program of China under Grant No.2002CB312002 (国家重点基础研究发展规划(973))

作者简介: 刘明(1972 -), 男, 四川成都人, 博士生, 主要研究领域为移动计算, 传感器网络技术; 龚海刚(1975 -), 男, 博士生, 主要研究领域为分布式计算, 并行处理; 毛莺池(1976 -), 女, 博士生, 主要研究领域为分布式计算, P2P 计算; 陈力军(1961 -), 男, 副教授, 主要研究领域为移动网络 Qos 技术; 谢立(1942 -), 男, 教授, 博士生导师, CCF 高级会员, 主要研究领域为分布式计算, 计算机网络.

法的情况下,DEEG 协议与其他两种数据收集和聚合协议(LEACH,PEGASIS)相比,在最好情况下,其网络寿命分别提高达 1800%和 300%,并且由于 DEEG 协议使得所有节点集中于最后 40 轮内全部死亡(网络寿命定义为最后一个节点死亡),因此,使用 DEEG 协议的传感器网络其监测结果具有很高的可靠性。

关键词: 无线传感器网络;簇头;汇集树;节能;簇覆盖

中图分类号: TP393 文献标识码: A

随着传感器技术、嵌入式技术以及低功耗无线通信技术的发展,生产具备感应、无线通信以及信息处理能力的微型无线传感器已成为可能。这些廉价的、低功耗的传感器节点共同组织成无线传感器网络,通过节点间的相互协作,将其监测和感应的多种环境信息(如温度、湿度等)传送到基站进行处理。无线传感器网络(wireless sensor networks)具有广泛的应用场景,可以应用在国防军事、救灾、环境监测等各个领域。由于其巨大的应用价值,WSNs 已经引起了各国军事部门、工业界和学术机构的极大关注,并纷纷展开该领域的研究工作,目前由微型传感器节点组成的无线传感器网络已经发展成一个重要的计算平台^[1,2]。

与无线移动自组网(MANETs)不同,无线传感器网络一般具有较大的节点密度以及较弱的节点移动性(一般部署后传感器节点不再移动)。同时,由于受到成本以及体积等原因的限制,无线传感器节点的处理能力、无线带宽以及电池容量等资源更为匮乏。特别是在很多应用中,由于传感器节点被部署在敌后或者环境恶劣的地区,节点的能量无法得到补充,这使得如何延长传感器网络的工作寿命成为设计上需要考虑的关键因素之一。目前已有大量的研究工作从不同角度来力求延长传感器网络的寿命^[3-7],本文则从通信协议的角度出发,提出了一种有效利用节点能量的数据收集和聚合协议 DEEG。

1 相关工作

传感器网络的主要任务是将网络中传感器节点收集的数据传送给基站。一种实现该任务的最简单方法是直接传送^[8],即网络中的每个节点把收集的数据直接传送给基站。然而,对于远离基站的传感器节点,节点传送数据消耗的能量代价太高将使节点很快死亡。为了解决这个问题,一些以节约能量为目的的算法相继被提出^[9-13]。

在文献[9]中,Wendi 等人提出了 LEACH 算法。它是一种分布式自组织的协议,其核心思想是减少与基站直接通信的节点数量来达到节能的目的。LEACH 协议按轮(round)运行,每轮分为设置(setup)和稳定(steady)两个阶段。在设置阶段,首先以自组织的方式随机选出部分传感器节点作为簇头,接着选出的簇头进行广播,普通节点根据接受信号的强弱来选择最近的簇头加入,并同该簇头一起形成簇。在稳定阶段中,簇中的成员节点把收集的数据传送给簇头,簇头将簇内所有成员节点收集的数据与本节点收集的数据进行聚合,然后发给基站。由于 LEACH 协议在每一轮中簇头都要重新随机选取,这种方式保证了能量损耗被均匀地分布到所有节点上。通过减少与基站直接通信的节点数量以及使用数据聚合(data fusion)技术,LEACH 协议与 DIRECT 协议相比,可以使传感器网络寿命(第 1 个节点死亡)得到 8 倍左右的提高。然而,LEACH 算法对以下几点没有考虑:1) 簇头数目的优化,LEACH 每轮中任意节点成为簇头的概率为 p ,因此簇头的数目与节点的数量规模成正比;2) 所有簇头直接与基站通信,对于远离基站的簇头其能量损耗很快;3) 由于簇头是随机选取的,因此算法不能保证簇头在网络中的分布,而簇头的分布将决定该轮中传感器网络的能量损耗状况。

在文献[10]中,Lindsey 和 Raghavendra 注意到在一个距离范围以内,节点收发电路所消耗的能量大于该节点放大电路所消耗的能量。为了减少传感器节点的能量损耗,他们提出了 PEGASIS 协议。该协议的具体方法是,把系统中所有传感器节点用贪婪算法构成一个边长之和接近最小的链,链上的每条边只接收(发送)数据一次。即在每一轮中,任意节点从位于链上的一相邻接点接收数据并把接收到的数据与本节点的数据聚合,接着把聚合后的数据发送到链上的另一相邻接点。这种发送和聚合是从链上的端节点开始,沿着链路直到指定节点,然后由指定节点把最后聚合的数据发送到基站。与 LEACH 协议相比,PEGASIS 协议与基站直接通信的节点更少而且数据聚合能力更强,从而显著减少了每一轮的能量损耗,特别是在基站远离监测区域的情况下,这种效果更加明显。根据模拟实验,PEGASIS 相对于 LEACH 其传感器网络寿命提高了 100%~300%。然而,PEGASIS 算法对传

感器网络寿命的提高建立在所有节点知道网络全局信息的基础上,因此我们认为 PEGASIS 协议存在以下几个问题:1) 由于传感器网络的节点规模庞大而传感器节点的处理能力和内存有限,因此保存网络的全局信息对于单个节点非常困难;2) 传感器网络大量应用在敌后或环境恶劣的地区,排除节点因为电能耗尽死亡以外还存在着其他令节点失效的原因,因此维护全局信息需要很大的代价;3) PEGASIS 算法将网络中的所有节点构造成一条链,如果链上的某一节点死亡则使得从链端到该节点的所有数据丢失,因此 PEGASIS 的容错性不佳(如图 1 所示,节点 4 死亡);4) 由于链的长度与节点数量有关,而传感器网络的节点数量规模很大,因此在收集数据时会产生很大的延迟.

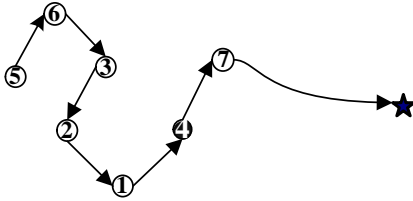


Fig.1 Illustration of PEGASIS protocol

图 1 PEGASIS 协议示意图

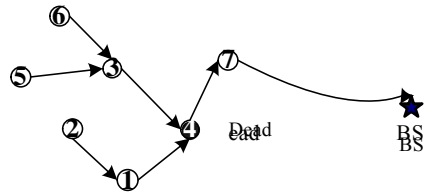


Fig.2 Illustration of PEDAP protocol

图 2 PEDAP 协议示意图

文献[11]中提出的 PEDAP 协议进一步发展了 PEGASIS 协议,其核心思想是把传感器网络的所有节点构造成一棵最小汇集树(minimum spanning tree).PEDAP 假定节点同构并且基站知道网络中所有节点的位置信息.除了节点的位置信息以外,根据无线通信模型节点在每一轮中的能量消耗是可以确定的,因此基站可以预知网络中任意节点的剩余能量.每过一定轮数后,PEDAP 协议需要基站重新计算路由信息用来排除死亡节点.计算完成以后,基站向每个节点发送其所需的消息(消息内容包括:在汇集树上的父亲节点、每一轮中节点发送数据到父亲节点的时隙等).通过这种方式,PEDAP 在设置阶段中,所有节点只需接收由基站发送的消息即可.因此,PEDAP 在设置阶段消耗的能量小于 LEACH 和 PEGASIS.模拟实验证明了 PEDAP 算法优于 PEGASIS 算法.然而对于 PEDAP 这种集中式的算法,如果节点因为其他(非能量耗尽)原因死亡,则基站无法得知该节点已死亡,将仍然按照原来的信息计算最小汇集树(如图 2 所示,节点 4 死亡),即当网络中有节点意外死亡时,PEDAP 难以及时排除死亡节点.

由上面的分析可以看到,基于全局或集中式的算法,如果要排除意外死亡节点,每个节点需要周期性地广播自己的状态,因此伸缩性差,并且会产生很大的能量开销.而基于簇的路由算法由于其分布式自组织的特点,具有较好的可扩展性和容错性.本文提出的 DEEG 协议正是基于簇的数据收集和聚合协议.

2 系统模型和问题描述

与文献[9-12]等所提出的算法一致,本文提出的基于簇的 DEEG 协议同样按轮运行.每轮分为簇生成、汇集树创建和数据收集 3 个阶段,分别用 $T_{clusters}$, T_{tree} 和 T_{data} 表示.其中 T_{data} 是网络进行环境监测的阶段,在此阶段中节点收集的数据被传送到基站进行处理.为了保证网络的有效工作时间,算法需要具有 $T_{data} \gg T_{cluster} + T_{tree}$ 的性质.

2.1 网络模型

本文中 N 个传感器节点随机均匀分布在一个正方形区域 A 内,并且该传感器网络具有如下性质:

- 传感器节点部署后不再移动.
- 基站部署在区域 A 以外的一个固定位置,并且基站是唯一的.
- 部署后网络不需要人为维护.
- 所有节点具有相似的能力(处理/通信),并且地位平等.
- 节点没有装备 GPS,也不能通过测量的方法知道其具体位置.
- 无线发射功率可控,即节点可以根据距离来调整发射功率的大小.

- 每轮中节点的能量消耗不统一。

网络的前 4 项属性是一般传感器网络的典型设置.第 5 项属性表明,本文提出的 DEEG 算法不需要利用节点的位置信息.我们认为,基于通过消息交换来确定节点位置的方法,当节点规模增大时,交换的消息数量将快速增长,从而导致网络效率下降并且直接影响到系统的可扩展性,同时,因为消息交换所产生的能量开销将对传感器网络的寿命造成影响.第 6 项属性主要是从节能的角度出发,为簇内通信和簇间通信定义不同的能量等级.发送消息可以显著减少节点的能量损耗,从而延长传感器网络的寿命.第 7 项属性指明了网络中节点的能耗不一致,所以算法必须保证能量消耗被所有节点均匀分担.

2.2 无线通信模型

近年来,在低能量无线通信方面进行了大量的研究工作.本文与文献[14]使用了相同的无线通信模型.该无线通信模型给出了一个阈值 d_0 (d_0 是常数,数值取决于使用环境),当发送节点与接收节点的距离小于 d_0 时,发送方发送数据的能量损耗与距离的平方成正比,否则与距离的四次方成正比.上述的两种能量衰减模型分别称为自由空间模型(free space)和多路衰减模型(multipath fading).因此,根据发送节点和接收节点之间的距离,发送节点可以使用不同的能耗模型计算发送数据所需要的能量.例如,节点 a 向距离 d 外的另一节点 b 发送 k 字节的数据,可以使用下面的公式计算其能量消耗:

$$E_{tr}(k, d) = E_{elec}(k) + E_{amp}(k, d) = \begin{cases} kE_{elec} + k\epsilon_{fs} d^2 \\ kE_{elec} + k\epsilon_{mp} d^4 \end{cases} \quad (1)$$

而 b 接收 a 发送的消息,其无线接收装置产生的能耗为:

$$E_{Rx}(k) = kE_{elec} \quad (2)$$

在上面的公式中, E_{elec} 表示无线收发电路所消耗的能量, E_{amp} 表示放大器消耗的能量,其大小取决于发送节点与接收节点间的距离以及可接受的位错误率.此外,大部分协议和算法都采用了数据聚合技术来减少发送和接收的数据量,从而达到节省能量的目的.DEEG 算法同样采用数据聚合技术来减少能量损耗.与 PEGASIS 协议一致,DEEG 算法假定数据聚合的能力为 $NK=K$,其中 N 表示传感器网络中所有节点的数量, K 表示数据包长度.数据聚合所消耗的能量用 E_{fusion} 表示.此外,本文假设无线信道是对称的,即从节点 a 传送消息 m 到节点 b 消耗的能量等于从节点 b 传送消息 m 到节点 a 消耗的能量.

2.3 问题描述

本质上按轮运行延长传感器网络寿命的方法都是在最小化系统每轮消耗能量的同时,把能量损耗均匀地分布到每个节点上.对于基于簇的算法,簇的构造将决定每轮系统能量的损耗状况.假设 N 个节点随机分布在区域 A ,我们认为,为了减小系统每轮中的能量损耗以及确保能耗均匀分布在每个节点,簇算法需要满足以下条件:

- 1) 算法应该是完全分布式自组织的,节点只根据本地信息独立地决定自身状态.每个节点在 $T_{cluster}$ 结束前必须决定是否作为簇头或是某个簇中的一个成员.
- 2) 节点之间的通信其能量损耗应该尽可能地满足自由空间模型,即簇内通信以及簇间通信都需要满足自由空间模型.由于 LEACH 算法不能保证簇头在网络中分布良好,当簇头分布较差时,簇内通信不满足自由空间模型,从而导致较大的能量开销,如图 3 所示.

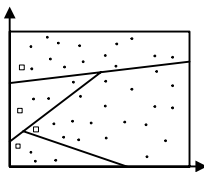


Fig.3 Illustration of the distribution of cluster in LEACH protocol
图 3 LEACH 簇分布示意图

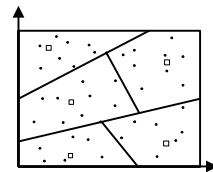


Fig.4 Illustration of the distribution of cluster in DEEG protocol
图 4 DEEG 簇分布示意图

3) 簇内和簇间分别使用不同的能量等级通信.在节点密度足够的情况下,簇间通信所使用的能量等级必须保证其通信的覆盖半径至少是簇半径的两倍以上,由簇头集合构成的覆盖网络是连通的.

4) 优化簇头的分布.簇头之间的通信在满足自由空间模型的情况下,应尽可能把保证任意簇头间的距离大于簇半径 r_c .本文中采用簇半径 r_c 来表示一个簇的大小,即只允许与簇头距离不超过 r_c 的节点加入该簇头所属簇.由于无线信道的干涉现象,当两个簇头间的距离小于 r_c 时,会加剧彼此簇内通信的干扰,造成不必要的消息重传,从而导致额外的能量开销.

5) 簇头的选择应该考虑到节点剩余能量的状况.与 LEACH 协议不同,本文提出的算法不假定簇头在每轮中的能量开销一致,因此需要算法保证能量开销被均匀地分布到每个节点上,避免个别节点过早死亡.

3 DEEG 协议描述

DEEG 协议按轮运行,每轮分为 $T_{cluster}$, T_{tree} 和 T_{data} 三个阶段.在 $T_{cluster}$ 中,簇头选出后,簇头接收所有即将加入该簇的成员节点的消息.根据簇内成员节点的数目,簇头创建一个 TDMA 时隙调度,并把该 TDMA 调度广播给簇内所有成员节点.为了进一步节约能量,在簇生成以后的 T_{tree} 阶段,从簇头集中选一簇头作为唯一与基站通信的节点 CH_b .以 CH_b 作为树根,其他簇头节点作为 CH_b 的子孙创建近似最小汇集树.在 T_{data} 中,簇内成员节点按所分配的时隙将数据发送到簇头,簇头收到簇内数据后继续等待,直到其汇集树上的所有子孙节点的数据到达后,该簇头再将所有收到的数据进行聚合,然后发送到其父亲节点.重复上述步骤直至根节点,最后,根节点将聚合后的数据发送到基站.

3.1 簇生成算法

DEEG 算法中簇的大小固定,其半径 $r_c < d_0/2$.为了保证簇头之间的连通性,我们定义簇头的通信半径 $R_{ch} > 2r_c$,两个相邻簇头间的距离 d 满足关系式 $r_c < d \leq R_{ch}$,通过这样的关系式可以让簇头在传感器网络中分布良好,如图 4 所示.由于簇的大小固定且簇间的关系需要满足上诉约束,所以在一个大小的确定的监测区域 A 中,覆盖整个区域 A 所需的簇的数量与 A 的面积以及簇半径 r_c 直接相关.文献[15]中给出了覆盖区域 A 所需最小节点数的公式,其中 n 表示所需的最小节点数目,如下所示:

$$n\pi r^2/A_{area} = 2\pi\sqrt{27} \quad (3)$$

根据式(3),对于每个节点 $S_i(1 < i < N)$,我们定义其成为簇头的初始概率为

$$P_{init} = 2A_{area} / \sqrt{27}Nr_c^2 \quad (4)$$

初始概率是为了限制初始簇头的数目,显然最终的簇头数目 $N_c \geq Np_{init}$.当节点运行 DEEG 协议以后,为了使能量损耗被均匀地分布到所有节点上,其成为簇头的概率 p_{ch} 与节点的能量有关:

$$P_{ch} = P_{init} \times \frac{E_{current}}{E_{max}} \quad (5)$$

其中 $E_{current}$ 表示节点现存能量, E_{max} 表示节点的初始能量.当节点的能量小于阈值 E_{min} 时,该节点不再参与簇头竞争.由于消息的发送、接收以及聚合所消耗的能量可以预知,所以阈值 E_{min} 可以根据下面的公式计算.

$$E_{min} = cycle \times (C_{degree} \times (E_{elec} + E_{DA}) + l \times E_{DA} + l \times \epsilon_{fs} \times d_0^2) \quad (6)$$

上式中, $cycle$ 表示每轮数据收集次数, C_{degree} 表示簇内成员的平均数目, l 表示数据包的长度.

对于任意节点 S_i ,在每轮的 $T_{cluster}$ 阶段需要执行簇生成算法,算法的主体是一段循环代码,循环的次数与节点的现存能量 $E_{current}$ 有关.每次循环需要时间 t , t 在长度上应该保证节点 S_i 可以收到以 S_i 为圆心半径 r_c 中任意节点发送的消息.同时我们注意到,为了延长传感器网络的寿命,节点在竞争簇头时能量不是唯一需要考虑的因素.循环开始前, S_i 在半径 r_c 内广播,并且接收所有半径 r_c 内邻居节点发送的广播消息.根据接收的无线信号强度,

令 $PRI = \sum_{n=1}^m RadioStr / m$,其中 m 是节点 S_i 半径 r_c 内邻居节点的数量. PRI 越高,以 S_i 为簇头的簇其能耗就越小,

故 PRI 将作为节点竞争簇头的优先级.静态传感器网络中节点的死亡(能量耗尽)是可以预测的,如果节点的故障率较小,则每个节点的邻居集合变化很小.此外,由于 DEEG 协议把网络能耗均匀地分担到每个节点上,延长了网

络中所有节点的寿命,进一步增加了节点邻居集合的稳定性.见算法 1 的第 3~6 行,DEEG 协议中所有节点每过 count 轮进行广播来计算 PRI,count 的取值与网络中节点的故障率有关.

算法 1. 簇生成算法伪码.

```

1.  If round=1
2.    Bs_str=Receive_Str(msg from BS)           // radio strength from bs
3.  If round%count=0
4.    Broadcast( $S_i \rightarrow ID$ )
5.    Receive_Str(msg from neighbor in  $s_i$ 's cluster range)
6.    PRI=Avr_Str(all neighbor in the cluster range of  $s_i$ )
7.     $p = \max(p_{ch}, p_{min})$                        //  $p_{min} = p_{init} * (E_{min}/E_{max})$ 
8.    is_CH=false

9.  while( $p \leq 1$ )
10.  If New_CH!=empty
11.    Add(CH_Set, New_CH)
12.  If CH_Set=empty and  $p \neq 1$ 
13.    If random(0,1) $\leq p$ 
14.      Broadcast( $S > ID$ , temp, PRI)
15.    Else If CH_Set=empty and  $p = 1$ 
16.      Broadcast( $S > ID$ , CH, PRI)
17.      is_CH=true
18.    Else If CH_Set!=empty
19.      cluster_head=Max_PRI(all nodes in CH_Set)
20.      If  $S_i \in CH\_Set$  and  $S_i$  has highest PRI in CH_Set
21.        If  $p = 1$ 
22.          Broadcast( $S_i \rightarrow ID$ , CH, PRI)
23.          is_CH=true
24.        Else
25.          Broadcast( $S_i \rightarrow ID$ , temp, PRI)
26.         $p = \min(2 * p, 1)$ 

27.  If is_CH=true
28.    Broadcast( $S_i \rightarrow ID$ , CH, PRI)
29.  Else
30.    If CH_Set=empty
31.      Broadcast( $S_i \rightarrow ID$ , CH, PRI)
32.  Else
33.    join_cluster(cluster_head,  $S_i \rightarrow ID$ )
    
```

每次循环开始时, S_i 首先判断有没有新的簇头广播消息,如果有,则把新的簇头加入候选的临时簇头集合 CH_Set.如果 CH_Set 不为空并且 CH_Set 中不包含 S_i 节点本身,则 S_i 失去选举机会,但是如果循环结束后,CH_Set 中没有任何候选簇头宣告自己正式成为簇头,则节点 S_i 认为自己没有被任何簇覆盖,于是 S_i 广播自己正式成为簇头.如果 S_i 是簇头的候选节点($S_i \in CH_Set$),则每次循环都需要判断 S_i 的优先级 PRI 是否最高,当其优先级最高且循环结束则 S_i 广播自己正式成为簇头,否则循环结束后如果有候选节点宣告自己成为正式簇头,则 S_i 加入该簇头.如果循环结束后 CH_Set 依然为空,即 S_i 没有被任何簇覆盖,则 S_i 广播自己正式成为簇头.需要注意的是,在竞选簇头的过程中,节点的广播半径为 r_c ,见算法 1.

3.2 近似最小汇集树生成

从算法 1 的第 1~2 行语句可以看到,在网络正式工作之前,基站将对整个网络进行广播,每个节点利用各自收到的广播信号强度 BS_str 作为以后竞选 CH_b 节点的参数.簇生成以后,各簇头竞争成为 CH_b 节点.我们定义每个簇头成为候选 CH_b 的概率为

$$p_b = (BS_str \times E_{current}) / (BS_str_{max} \times E_{max}) \quad (7)$$

其中,BS_str 是指簇头接收的基站广播信号的强度,BS_str_{max} 表示区域 A 中基站广播信号的最大强度.由于基站

和区域 A 的最短距离已知,故 $BS_{str_{max}}$ 的值可以通过公式(1)计算.显然,如果簇头距离基站越近且能量越大,则该簇头成为候选 CH_b 的可能性越大.簇头从 $(0,1)$ 之间取随机值 p ,如果有 $p \leq p_b$,则该簇头成为候选 CH_b ,并且向半径 R_{ch} 中的邻居簇头进行广播,广播消息由两部分组成:节点 ID 和 p_b ,邻居簇头收到广播消息后进行转发,直至所有簇头收到候选 CH_b 的广播消息,其中具有最大 p_b 值的候选节点被选为 CH_b 节点(如果存在两个以上的候选 CH_b 簇头的 p_b 相等,则 ID 值大的节点成为 CH_b).如果簇头没有成为候选 CH_b 节点,且在规定的时间内没有收到其他簇头节点成为候选 CH_b 的消息,则令 $p = p / 2$,直到 $p \leq p_b$,宣告其成为 CH_b 候选簇头.在网络生命后期,节点大量死亡,区域 A 中的节点密度不能保证簇头的连通性,所以当簇头在限定时间内没有收到候选 CH_b 的广播消息,则该簇头自动成为 CH_b ,这样可以保证仍然生存的节点可以将其监测数据传到基站.

CH_b 选出以后,以 CH_b 作为树根构造路由树.首先 CH_b 发送包含自身 ID 的广播消息给半径 R_{ch} 里的所有邻居簇头,邻居簇头收到消息后在消息中加上 CH_b 到本簇头的路径信息,然后继续广播,即用类似于 flooding 的方法直到所有簇头得到其通向 CH_b 的路由信息.为了减轻消息交换的复杂度以及节省节点有限的存储资源,如果在簇头收到的路径消息中,路径的跳数大于该簇头收到的其他路径消息中的路由跳数,则丢弃该广播消息.当广播过程结束后,每个簇头保留了所有 CH_b 到本节点的最短路径消息,然后从中选择最近的上游节点作为父亲(根据接收信号强度),并向其发送 CHILD 消息.例如,图 5(a)是簇头之间的邻居关系图,其中 A 是选出的 CH_b 节点,由 A 发起广播,当节点 B 和 C 收到来自 A 的消息后,向半径 R_{ch} 内的邻居簇头广播路径消息 $\{A,B\}$ 和 $\{A,C\}$.由于路径消息 $\{A,B\}$ 的长度大于 $\{A\}$,所以该消息被 C 丢弃,同理, B 丢弃路径消息 $\{A,C\}$.而对于节点 D ,从 B 收到的路径消息 $\{A,B\}$ 和从 C 收到的路径消息 $\{A,C\}$ 等长,所以 $\{A,B\}$ 和 $\{A,C\}$ 都被保存.广播结束后,各个节点保存的路径如图 5(b)所示. B 和 C 都是 D 的上游节点,假设 D 收到的来自 C 的信号强度大于 B ,则 D 向 C 发送 child 消息,最后得到的汇集树如图 5(c)所示.

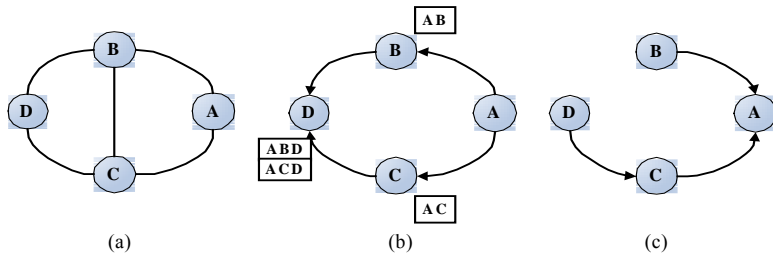


Fig.5 Illustration of the algorithm for generating approximation minimum spanning tree

图 5 近似最小汇集树生成算法示意图

3.3 簇覆盖

文献[16]对随机部署、规则部署和计划部署 3 种节点部署策略进行了讨论.随机部署策略中,节点在监测区域中的分布符合均匀分布.规则部署策略中,传感器节点按照规则的几何拓扑进行放置,比如按栅格(grid)置放.计划部署策略中,监测区域中重点地段的节点密度高于一般地段.规则分布由于其节点位置事先确定,因此不属于我们的研究范围.计划部署虽然在整个监测区域中节点不符合随机均匀分布,然而在一个小的范围内,节点的放置则近似于随机,所以下面对于随机部署的分析结果同样适用于计划部署.

对于随机部署而言,要完全保证部署的节点覆盖整个区域是不可能的.因此,我们期望解决的问题如下:在半径为 R 的监测区域 C ,需要随机部署多少传感半径为 r_s 的节点,才能保证整个区域被覆盖部分达到期望值.为了方便描述,我们令区域 C 的圆心坐标为 $(0,0)$,并作如下定义:

定义 1. 邻域.对于任意一点 $(x,y) \in C$,其邻域定义如下:

$$N(x,y) = \{ \forall (x'-x)^2 + (y'-y)^2 \leq r_s^2 \} | (x',y') \in C \} \tag{8}$$

定义 2. 区域 C' .对于区域 C' ,有 $C' \subset C$ 并且对任意一点 $(x,y) \in C'$,满足如下定义:

$$x^2 + y^2 \leq (R - r_s)^2 \tag{9}$$

对于任意一点 $(x,y) \in C$,如果在其邻域内至少存在一个节点,则该点被覆盖.由于区域 C 中节点的部署符合均

匀分布,所以节点落入 (x,y) 邻域内的概率为 $p=\mathcal{N}(x,y)_{area}/C_{area}$.假设区域 C 中部署了 m 个节点,则点 (x,y) 被覆盖的概率为

$$P_{coverage} = C_m^1 p(1-p)^{m-1} + C_m^2 p^2(1-p)^{m-2} + \dots + C_m^m p^m$$

$$= \sum_{n=1}^m C_m^n p^n (1-p)^{m-n} \quad (10)$$

对 $\forall(x,y) \in C$,其邻域面积 $\mathcal{N}(x,y)_{area} = \pi r_s^2$,故单个节点落入某点邻域内的概率为 $p = \mathcal{N}(x,y)_{area}/C_{area} = \pi r_s^2 / \pi R^2$.根据式(9),如果 C 中随机部署了 m 个节点,则对 $\forall(x,y) \in C$ 其被节点覆盖的概率为

$$P_{cover} = \sum_{n=1}^m C_m^n \left(\frac{r_s}{R}\right)^{2n} \left(1 - \frac{r_s^2}{R^2}\right)^{m-n} \quad (11)$$

在节点密度足够的情况下,每轮 $T_{cluster}$ 阶段形成的簇覆盖了整个监测区域 A .虽然一个簇的理论面积为 πr_c^2 ,然而为了覆盖区域 A ,簇与其邻居簇之间存在重叠,即簇的实际面积小于 πr_c^2 .因此,可以根据具体应用指定的 p_{cover} ,利用式(11)近似计算出达到该覆盖要求所需的节点数量 m .为了节省能量,簇头随机地从簇内成员节点中选出 m 个节点作为本轮的数据收集节点,而让剩余成员节点在本轮处于休眠状态.簇头对簇内成员节点的调度,通过簇内广播 TDMA 调度来实现.

4 性能分析

4.1 参数设置

我们使用 GlomoSim 作为模拟实验平台,模拟实验中各项参数见表 1.在模拟实验中,网络寿命包含 3 种不同的定义,分别为:第 1 个节点死亡、一半节点死亡和最后一个节点死亡.此外,在模拟 DEEG 协议时,设各节点的感知半径 $r_s=12\text{m}$,且当节点能量小于 $E_{min}(0.002\text{J})$ 时,认为该节点死亡.

Table 1 Simulation parameter

表 1 模拟中使用的参数列表

Parameter	Value
Network size	(100×100),(150×150),(200×200),(250×250),(300×300),(350×350),
Sink position	(50,175),(50,200),(20,215),(50,230),(50,245), (50,260),(50,275),(50,300),(50,400)
Node number	100,150,200,250,300,350,400,450,500,1000
Threshold distance(d0)	75m
Cluster radius(rc)	30m
Coverage radius(rs)	12m
Eelec	50nJ/bit
e _{fs}	13pJ/bit/m ²
e _{mp}	0.0013pJ/bit/m ⁴
E _{fusion}	5nJ/bit
Data packet size	500 bytes
Broad packet size	25 bytes
Packet header size	25 bytes
round	5 TDMA frames
Initial energy	2J

为了对 DIRECT,LEACH,PEGASIS 以及 DEEG 这 4 种协议进行全面的比较,我们在实验中分别模拟了监测区域大小变化、基站的位置变化和节点数量变化对协议性能的影响,结果及分析见第 4.3 节.其中 DEEG 协议分为 DEEG-1 和 DEEG-2(使用了簇覆盖算法)两种协议.

4.2 簇头数量及分布

为了验证 DEEG 协议使得生成的簇分布良好,我们分别在 100m×100m 和 200m×200m 两个场景中随机地对簇头的数量和分布进行了 200 次抽样.令覆盖整个区域 A 所需的最少簇头数量为 N_{opti} ,根据式(3)有 $N_{opti} = \lceil 2A_{area} / \sqrt{27}r_c^2 \rceil$.实验证明,每轮运行 DEEG 协议后,生成的簇头数量不超过 $\lceil 2.5 \times N_{opti} \rceil$ 的情况占全部抽样的 98%,其中最坏情况下生成的簇头数量为 $N_c = \lceil 3 \times N_{opti} \rceil$.实验结果说明,运行 DEEG 协议后,每轮生成的簇头数量与监测区域 A 的面积和簇半径 r_c 有关,并且生成的簇头区域 A 中分布良好.图 6 是场景为 100m×100m,节点数

量为 300 的节点部署图,图 7 是图 6 所指网络在第 420 轮时的簇头分布示意图,可以看到,任意簇头之间的距离大于簇半径 r_c ,并且生成的簇头完全覆盖了整个网络中的所有节点.



Fig.6 The distribution of nodes
图 6 节点分布图



Fig.7 The distribution of cluster head in 420 rounds
图 7 第 420 轮中簇头分布示意图

4.3 实验结果及分析

图 8 是基站位置为(50,245),监测区域 A 大小为 100m×100m,节点数量为 100 时节点死亡数量与网络工作时间(轮数)之间的关系图.可以看到,图 8 中表示 DEEG 协议的曲线几乎是一条平行于 X 轴的直线.由于 DEEG 协议使得网络能耗被均匀地分担到每个节点上,因此第 1 个节点和最后一个节点的死亡时间非常接近,一般在 40 轮以内.从图 8 中可以看到,与 LEACH 和 PEGASIS 相比,DEEG 协议使得网络寿命分别提高了 310%和 55%(第 1 个节点死亡).需要注意的是,由于区域 A 中节点密度较小,因此使用了簇覆盖方法的 DEEG-2 协议不能显著提高网络寿命,其性能略优于 DEEG-1.

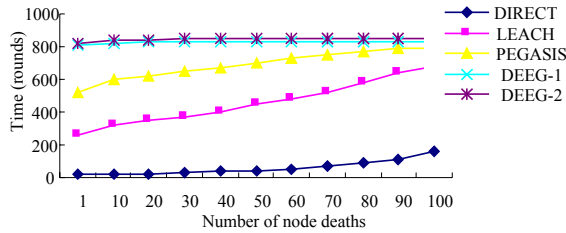


Fig.8 Timing of node deaths in a network of size 100m×100m
图 8 节点死亡时间图

图 9~图 11 分别描述了上述 3 种定义的网络寿命与基站位置的关系.由图 9 可以看到,当网络寿命定义为第 1 个节点死亡时,DEEG 协议随着监测区域 A 与基站距离的增加其网络寿命减小得最慢.实验表明,当基站位置从(50,175)~(50,300)时,网络寿命从 892 轮减小到 732 轮,减小幅度不到 18%,远优于 PEGASIS 和 LEACH 协议.同时还可以看到,DEEG 协议在 3 幅图中的曲线变化最小,这说明了运行 DEEG 协议的网络,其 3 种定义的网络寿命在时间上相差不大.

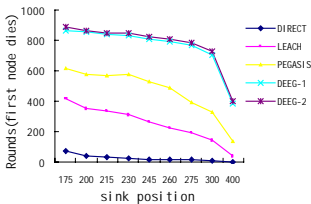


Fig.9 Network lifetime as sink travels farther (FND)
图 9 基站位置与网络寿命 (第 1 个节点死亡)

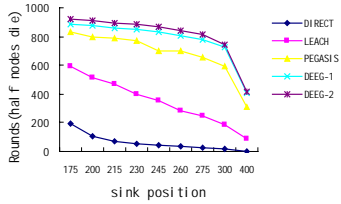


Fig.10 Network lifetime as sink travels farther (HND)
图 10 基站位置与网络寿命 (一半节点死亡)

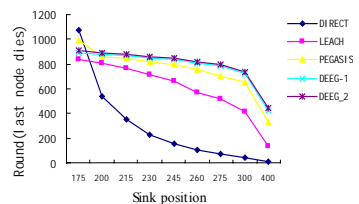


Fig.11 Network lifetime as sink travels farther (LND)
图 11 基站位置与网络寿命 (最后一个节点死亡)

在模拟时我们考虑了监测区域 A 大小变化对各协议性能的影响.从图 12~图 14 可以看到,随着监测区域 A 的变大,DEEG 协议仍然保持很好的性能,这是因为 DEEG 协议中簇分布合理,并且簇头之间通过多跳的方式将

数据传送到基站,减小了簇头的能量损耗,从而使得在相同条件下,DEEG 协议的性能优于 LEACH 和 PEGASIS 协议.

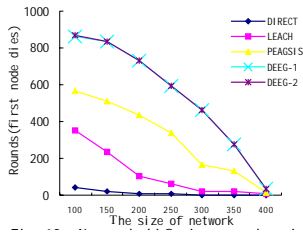


Fig.12 Network lifetime as the size of network increased (FND) 图 12 监测区域面积与网络寿命 (第 1 个节点死亡)

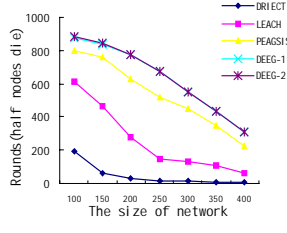


Fig.13 Network lifetime as the size of network increased (HND) 图 13 监测区域面积与网络寿命 (一半节点死亡)

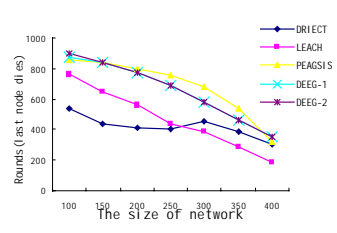


Fig.14 Network lifetime as the size of network increased (LND) 图 14 监测区域面积与网络寿命 (最后一个节点死亡)

图 15 反映了节点数量与网络寿命之间的关系.一个值得注意的现象是,当节点数量增加时,前 4 种协议都不能使网络寿命明显提高(DEEG-2 除外),这是因为这些算法没有考虑到覆盖问题.实际上,当节点密度足够大时,只需部分节点就可以覆盖整个区域 A,因此多余的节点应该处于休眠状态,从而减小节点的能量损耗达到延长网络寿命的目的.而前 4 种协议没有考虑覆盖问题,所有节点在每轮中都处于活动状态,从而使得网络寿命不随节点数量的增加而延长.此外,由于 DEEG 算法生成的簇头只与监测区域 A 的面积和节点的分布情况有关,因此当节点数量增加时,DEEG-1 协议中的簇成员数量增加,从而导致簇头的能量损耗增加,使得第 1 个节点的死亡时间提前.从图中我们可以看到,当节点数量为 100 时,其第 1 个节点的死亡时间为 890 轮.而当节点数量增加到 500 时,第 1 个节点的死亡时间提前到第 831 轮.DEEG-2 协议使用了簇覆盖方法,当簇中的成员节点超过应用需求时,簇头通知冗余成员节点在本轮处于休眠状态,实验证明,DEEG-2 协议随着节点数量的增加网络寿命成线性增长.

对于按轮运行的协议,每轮中所有活动节点收集环境数据的次数是一致的(在模拟中,我们设节点每轮数据收集次数为 5 次),所以可以通过网络中所有节点收集环境数据的总次数来衡量协议的性能.图 16 是我们根据基站与监测区域 A 之间的距离变化,分别计算了 4 种协议在网络失效(第 1 个节点死亡)前所有节点数据收集次数之和,可以看到,DEEG-1 协议的数据收集次数,远高于前 3 种协议,而且随着区域 A 与基站距离的增加,DEEG-1 协议的数据收集次数的减小幅度在 4 种协议中最慢.这是因为 DEEG 协议中生成的簇在网络中分布良好,各个簇所收集的数据在簇头节点间以多跳的方式传到基站,减小了簇头节点的能量损耗,使得各节点的能耗更均匀,前面的模拟实验证明,DEEG 协议中的第 1 个节点死亡时间一般在最后的 40 轮以内(最后一个节点死亡),因此,其数据收集次数远比其他协议要高,并且 DEEG 协议的监测质量高于其他 3 种协议.

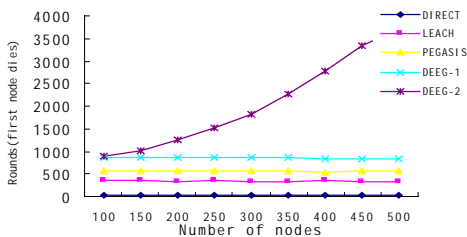


Fig.15 Network lifetime as the number of nodes Increased (FND)

图 15 节点数量和网络寿命关系图

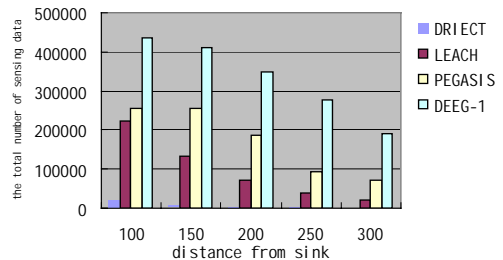


Fig.16 The total number of sensing data gathered by all nodes in the network (FND)

图 16 网络传感数据收集数量图

从上述的实验结果中可以看到,除了图 16 以外,在所有图中,DEEG-1 和 DEEG-2 协议具有相近的性能,这是因为在上述的实验中节点密度较小,只有很少的冗余节点,因此使用了簇覆盖方法的 DEEG-2 协议不能显著延

长网络寿命.而当节点冗余度增加以后,从图 16 可以看到,DEEG-2 协议的网络寿命远优于其他 4 种协议,但是由于 DEEG 协议规定所有节点在 $T_{cluster}$ 阶段都处于活动状态并参与簇的形成过程,故需要消耗节点的能量,因此当冗余度达到一定限度时,网络寿命将不再随着节点数量的增加而延长.此外,在实验中我们发现,DEEG 协议在 $T_{cluster}$ 和 T_{tree} 两个阶段所消耗的能量占全部能量的 0.4%~0.7%,该结果与 PEGASIS 协议相当而大于 LEACH 协议(LEACH 协议在簇生成阶段所消耗的能量约为总能量的 0.05%~0.07%).

5 结论及以后的工作

本文提出了一种分布式的高效节能的传感器网络数据收集和聚合协议,协议中节点根据其剩余能量的大小以及邻居节点的信号强度来竞争簇头,簇头选出以后普通节点选择最近的簇头加入形成簇.为了进一步减小能量开销,远离基站的簇头通过多跳的方式经过其他簇头将收集的数据传送到基站.此外,DEEG 协议还通过一种简单的簇覆盖方法使得当节点密度提高时,网络寿命也相应延长.实验证明,DEEG 协议提高了网络寿命,并且由于 DEEG 协议使得所有节点的死亡集中于最后的 40 轮内(最后一个节点死亡),因此 DEEG 协议的监测结果更为准确.

传感器网络往往部署在人难以维护的地区,由于受到恶劣环境因素的影响,传感器节点一般具有较高的故障率.因此单簇头的簇算法当簇头失效时,整个簇中成员收集的数据将会丢失.为了使网络具有更好的容错性,在下一步工作中我们将考虑在 DEEG 协议的基础上提高数据收集与传输的可靠性.此外,我们已订购了 CrossBow 公司开发的 motes 传感器节点用于构建传感器网络平台,并准备在其上实现 DEEG 协议.

References:

- [1] Estrin D, Girod L, Pottie G, Srivastava M. Instrumenting the world with wireless sensor networks. In: Proc. of the Int'l Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2001). 2001.
- [2] Pottie GJ, Kaiser WJ. Wireless integrated network sensors. Communications of the ACM, 2000,43(5):51-58.
- [3] Ye W, Heidenmann J, Estrin D. An energy-efficient MAC protocol for wireless sensor networks. In: Proc. of the IEEE INFOCOM. 2002. http://www.isi.edu/div7/publication_files/Ye02a.pdf
- [4] Sohrabi K, Gao J, Ailawadhi V, Pottie GJ. Protocols for self-organization of a wireless sensor network. IEEE Personal Comm. Mag. 2000,7(5):16-27.
- [5] Kulik J, Heinzelman WR, Balakrishnan H. Negotiation-Based protocols for disseminating information in wireless sensor networks. ACM Wireless Networks, 2002,8(2):169-185.
- [6] Cerpa A, Estrin D. ASCENT: Adaptive self-configuring sensor networks topologies. In: Proc. of the IEEE INFOCOM. 2002.
- [7] Kawadia V, Kumar PR. Power control and clustering in Ad hoc networks. In: Proc. of the IEEE INFOCOM. 2003. 459-469.
- [8] Estrin D. Next century challenges: Scalable coordination in sensor networks. In: Proc. of the MobiCOM'99. 1999. 263-270.
- [9] Heinzelman WR, Kulik J, Balakrishnan H. Adaptive protocols for information dissemination in wireless sensor networks. In: Proc. of the 5th Ann. Int'l Conf. on Mobile Computing and Networking. 2001. 174-185.
- [10] Lindsey S, Raghavendra CS. Pegasus: Power-Efficient gathering in sensor information systems. In: Proc. of the IEEE Aerospace Conf. 2002. 1-6.
- [11] Tan HO. Power efficient data gathering and aggregation in wireless sensor networks. SIGMOD Record, 2003.
- [12] Bandyopadhyay S, Coyle E. An energy-efficient hierarchical clustering algorithm for wireless sensor networks. In: Proc. of the IEEE INFOCOM. 2003.
- [13] Younis O, Fahmy S. Distributed clustering in Ad-hoc sensor networks: A hybrid, energy-efficient approach. In: Proc. of the IEEE INFOCOM. 2004.
- [14] Heinzelman WR. An application-specific protocol architecture for wireless microsensor networks. IEEE Trans. on Wireless Communications, 2002,1(4):660-670.
- [15] Slijepcevic S, Potkonjak M. Power efficient organization of wireless sensor networks. In: IEEE Int'l Conf. on Communications (ICC). 2001.
- [16] Tilak S, Abu-Ghazaleh N, Heinzelman W. Infrastructure tradeoffs for sensor networks. In: Proc. of 1st Int'l Workshop on Wireless Sensor Networks and Applications (WSNA 2002). 2002. 49-57.