

文章编号:1001-9081(2006)12-2787-02

无线传感器网络簇头优化分簇算法及其性能仿真

陈 静, 张晓敏

(山东大学 信息科学与工程学院, 山东 济南 250100)

(chenj@mail.sdu.edu.cn)

摘 要:基于对 LEACH 等算法的研究,提出一种传感器网络分簇算法——簇头优化分簇算法。它将节点周期性划分为数个在地理位置上分布均匀的“临时簇”,然后分别在每个临时簇内选择簇头;簇头选择时,遵循保护最低能量节点的原则,即要求所选簇头尽量靠近剩余能量最低的节点。仿真结果表明,与 LEACH 相比较,该算法能保证簇头较均匀分布在网络中,推迟第一个死亡节点出现的时间,同时也提高了基站接收的数据量。

关键词:无线传感器网络;低功耗自适应聚类路由算法;簇头;优化

中图分类号: TP393.03 **文献标识码:** A

Cluster heads optimized clustering algorithm for wireless sensor networks and its performance simulations

CHEN Jing, ZHANG Xiao-min

(School of Information Science and Engineering, Shandong University, Jinan Shandong 250100, China)

Abstract: Based on the study of LEACH, a clustering hierarchy algorithm named cluster heads optimized algorithm was proposed. The algorithm periodically put all nodes into several temporary clusters distributed evenly in geographical position, and then selected a cluster head in every temporary cluster. While selecting the cluster head, the principle of protecting the lowest energy node should be followed, the selected cluster head should be as close as the lowest energy node. Simulation results show that, compared with LEACH, the algorithm can enable the cluster heads to distribute more evenly in the network, postpone the emergency of the first dead node, and increase the number of data received at the base station.

Key words: Wireless Sensor Network (WSN); Low Energy Adaptive Clustering Hierarchy (LEACH) algorithm; cluster heads; optimization

0 引言

无线传感器网络(Wireless Sensor Network, WSN)由于工作环境和自身构造所限,网络传感器节点的计算、通信能力及能量都十分有限,对于节点的更换和充电也较难实现。因此,在对该种网络协议及传输机制的研究中,尽量减少节点耗能、延长网络生存时间成为一个主要目标。分簇算法就是一个重要的研究方面。一个好的分簇算法能够形成优良的网络拓扑结构,提高路由协议及 MAC 协议的效率,为数据融合、目标定位等提供基础。LEACH (Low Energy Adaptive Clustering Hierarchy)算法^[1,2]是一种可以随机选择簇头,动态成簇的能耗低,载荷均衡且易于实现的算法。它作为一个专门针对 WSN 的算法受到关注^[3~5]。

本文基于对 LEACH, PEGASIS^[5], DEED^[6]等算法的研究,提出一种新的分簇算法——簇头优化算法(Cluster Heads Optimized, CHO)。仿真表明,与 LEACH 相比,该算法延长了网络生存时间,均衡了节点的能耗,提高了数据量。

1 LEACH 算法

LEACH 算法是一种低功耗自适应分簇算法。该算法中使用了“轮”的概念,每一轮由初始化和稳定工作两个阶段组成:在初始化阶段,算法随机地选取节点作为簇头,簇头向所

有节点广播此消息,其他节点根据接收信号的强弱加入就近的簇;在稳定阶段,簇头节点将接收的数据融合并发送给基站。

LEACH 算法中各节点等概率担任簇头,节点能量消耗相对均衡。但是,由于没有考虑节点的具体地理位置,不能保证所选簇头在网络内均匀分布,因而可能产生簇头过于集中或分布在网络边缘的情况^[7]。簇头选取时,文献[1]也有对节点剩余能量的考虑,但它需要获取所有节点的总能量。针对以上两点,本文在保持“轮”的工作机制和稳定阶段数据传输机制不变的前提下,提出了簇头优化分簇算法。

2 簇头优化算法

该算法基本思想是周期性地将网络节点划分为“临时簇”,使网络节点在地理位置上较均匀地分为几个“临时簇”。每轮开始,分别在每个临时簇内选取簇头,以保证簇头的均匀分布。为实现节点能量的均衡消耗,在簇头选取时,将节点的剩余能量作为选择依据,并提出了保护能量最低点的原则。其后的成簇和数据传输过程都采用与 LEACH 相同的机制。

2.1 临时簇的划分

首先,各节点产生一个随机概率 p , 其中 $p > a$ 的节点向其覆盖半径 r 内的节点发出竞争簇头的 CANDIDATES 消息(r 可以根据具体的网络规模及所要划分的簇个数而定),消

收稿日期:2006-06-26;修订日期:2006-08-26

作者简介:陈静(1982-),女,山东德州人,硕士研究生,主要研究方向:无线传感器网络拓扑控制技术; 张晓敏(1955-),女,山东济南人,副教授,主要研究方向:Ad hoc 网络及无线传感器网络技术。

息中包括本节点的 ID 和产生的随机概率 p 。与此同时,这些节点也接收其他的 CANDIDATES 信息,将这些信息及自身的 CANDIDATES 信息保存至本节点的集合 C 中。在等待时间 T 后(T 的大小能保证接收到其覆盖范围内其他节点所发送的消息),节点在 C 中选出概率最大者(概率相等时比较 ID)。如果恰是本节点,就向其覆盖范围内的所有节点发送 HEAD 消息;假如不是本节点,则等待概率最高点的 HEAD 消息。如果一定时间 T 内没有收到 HEAD 消息,说明该概率最高点已失去竞争能力(即它的覆盖半径内存在比它概率高的节点,从而使它失去了成为簇头的资格),则将其从 C 中去除。然后在 C 中重新选择概率最大点,重复以上过程,直到每个节点都收到 HEAD 消息。收到多个 HEAD 消息的节点根据收到的信号强弱加入就近的簇。然后,临时簇头节点广播本簇内的节点信息,使每个节点知道所在簇的所有节点,以保证临时簇内的簇头选取。

由于临时簇的划分过程中节点广播消息仅含有节点 ID 及概率值,所以不会造成过多的额外耗能。另外,由于节点概率随机产生,为了优化临时分簇作用,应周期性进行临时分簇过程。

r 取值适当的情况下, α 的取值越小越可以保证划分出理想个数的簇,但是为了节能,可以适当减少参与竞争簇头的节点个数,即增大 α 的取值。图 1 为 100 个节点均匀分布在 $(100 \times 100) \text{m}^2$ 范围内, $r = 40 \text{m}^{[6]}$ 的情况下 α 的取值与生成临时簇个数关系(结果为 500 次随机测试的平均值)。

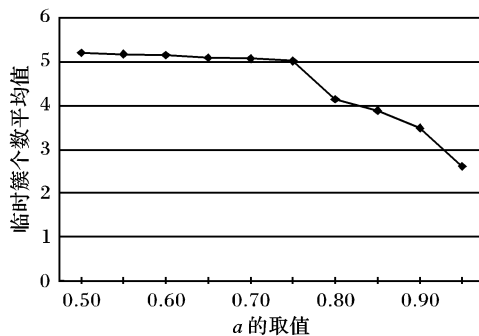


图1 α 的取值与临时簇个数的关系曲线

2.2 数据传输簇的划分

与 LEACH 不同,本算法中簇头的选取在各个临时簇内进行,并且在簇头选取时使用保护最低能量节点的原则,即要求所选簇头尽量靠近剩余能量最低的节点。

算法中,每个节点赋以下属性: $N, E(t), GE, Esum$ 和 $Eave$ 。其中 N 表示该临时簇内节点的个数, $E(t)$ 对应节点的剩余能量水平, GE 标记为比本节点能量低的节点的个数, $Esum$ 标记为本临时簇所有节点剩余能量的总值, $Eave$ 标记为本临时簇所有节点剩余能量的平均值,则 $Eave = Esum/N$ 。

初始时,所有节点的 $GE, N, Eave$ 都设为 0, $Esum$ 等于 $E(t)$ 。然后节点向该临时簇内所有其他节点通告自己的能量和 ID 号,同时,接收其他节点的能量值和 ID 号。节点每接收一个能量值则 N 加 1,并将能量值累加到 $Esum$ 中,同时与自己的能量进行比较,若比自身的能量值低则 GE 位加 1。这个阶段完成后, GE 位为 0 的节点为该临时簇内剩余能量水平最低者,有权进行簇头的选择。它根据接收到的其他节点的信号强弱发出信号,让 $E(t) > Eave$ 且距离自己最近的节点成为簇头,簇头选择过程完成。节点当选为簇头以后,发布通告消息告知网内其他节点自己是新簇头,非簇头节点根据收到

的簇头广播信号的强弱就近加入簇,并告知该簇头。成簇和数据传输的过程都采用与 LEACH 相同的机制。

算法中节点需的参数均可以通过 LEACH 原有的广播模式获得,而且节点仅需获知临时簇内节点的能量,因此不会造成算法复杂度的提高和额外的能耗。

3 算法仿真分析

在无线传感器网络中,第一个死亡节点(First Node Dead, FND)出现的时间是衡量网络寿命的一个重要参数。因此,为了提高网络性能,应该尽量推迟第一个死亡节点出现的时间。

仿真以 NS-2 为平台,为与 LEACH 作比较,参照文献[1],仿真情景如下:100 个节点均匀分布在 $[(0, 100) \times (0, 100)] \text{m}^2$ 的区域内,基站位于 $(50, 175)$ 处,每帧数据为 500 字节,并假定各节点总是有数据向基站发送。节点初始能量为 15J,每轮持续时间为 20s,采用与文献[1]相同的能耗模型。另外,根据节点传输数据与临时分簇广播消息大小关系,临时分簇的周期定为 400s, $\alpha = 0.75$ 。随机选取几轮,比较两种算法的簇头分布如图 2,图 3 所示。

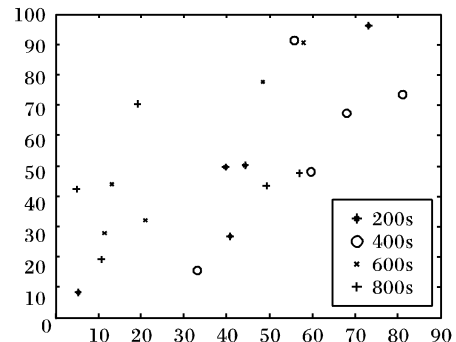


图2 LEACH 算法簇头分布

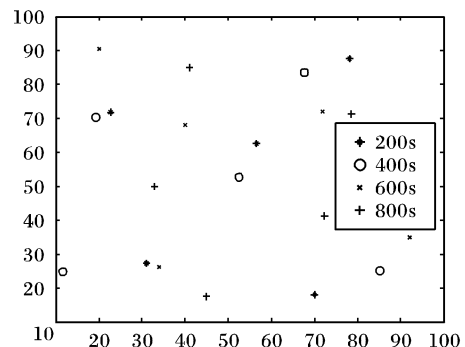


图3 CHO 算法簇头分布

由图可见,CHO 算法比 LEACH 算法所选出的簇头要更均匀地分布在网络中。

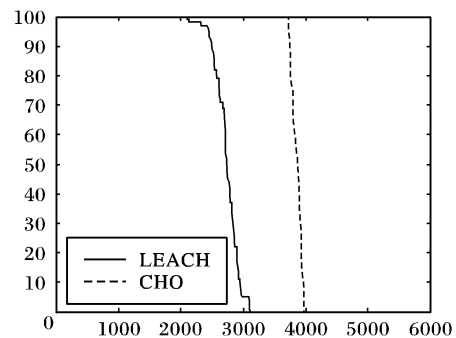


图4 运行时间与生存节点个数

图4为生存节点个数与仿真时间的关系。采用 LEACH

(下转第 2792 页)

的匹配阈值下返回的匹配结果数目,证明了新算法的查全率得到了提高。

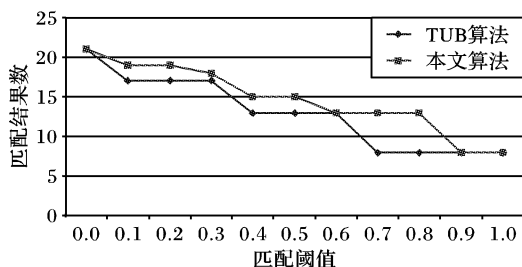


图3 查全率比较

从图3中可以看到,使用基于过程分析的服务发现算法,查全率有一定的提高,而且随着匹配阈值的降低,返回的服务数目增长曲线比较平滑,这说明用户改变匹配阈值时,可以更好地控制返回服务的数目。

对于查准率,由于目前的服务发现测试数据集中没有包含转换本体等更丰富的语义信息,仅是一些 Input 和 Output 信息,所以没有在实验部分表现出算法查准率的变化,但上面通过实例验证了算法的正确性,从理论上分析了如何保证查准率。

5 结语

在分布式电子商务环境中,如何更有效地发现服务供应者与服务需求者,也就是说如何具有更高查全率与查准率的 Web 服务发现是研究的中心问题。虽然 TUB, CMU 等机构都在这方面进行了研究工作,但仍存在着一些不足。基于过程分析的服务发现,在确保查准率的前提下,利用部分 Output 匹配情况下对过程进行分析的方法,提高了查全率。本文还

对新的服务发现算法与现有其他算法进行了比较实验,对实验结果的分析证明了新算法的有效性。

参考文献:

- [1] HAAS H. Web Services activity statement[R]. Technical report, W3C, <http://www.w3.org/2002/ws/Activity>, 2001.
- [2] MILLER E. Semantic Web activity statement[R]. Technical report, W3C, <http://www.w3.org/2001/sw/Activity>, 2003.
- [3] ANTONIOU G. A Semantic Web Primer (Cooperative Information Systems) [M]. MIT Press, 2003.
- [4] MARTIN D. OWL-S: Semantic Markup for Web Services[R]. Technical report, W3C, <http://www.w3.org/Submission/OWL-S/>, 2004.
- [4] BELLWOOD T. UDDI Version 2.04 API Specification[R]. Technical report, Oasis, <http://uddi.org/pubs/ProgrammersAPI-V2.04-Published-20020719.htm>, 2002.
- [5] PAOLUCCI M, KAWAMURA T, PAYNE TR. Semantic matching of Web services capabilities[A]. Proceedings of the 1st International Semantic Web Conference (ISWC) [C]. Sardinia, Italia, 2002. 333 - 348.
- [6] SRINIVASAN N, PAOLUCCI M, SYCARA K. Adding OWL S to UDDI: implementation and throughput [A]. First International Workshop on Semantic Web Services and Web Process Composition [C]. San Diego, USA, 2004.
- [7] JAEGER MC, TANG S. Ranked Matching for Service Descriptions using DAML-S [A]. CAISE Workshops (3) [C]. 2004. 217 - 228.
- [8] KLEIN M, BERNSTEIN A. Searching for services on the semantic Web using process ontologies [A]. The Emerging Semantic Web - Selected papers from the first Semantic Web Working Symposium [C]. Amsterdam, 2002. 159 - 172.

(上接第 2788 页)

算法,节点从第 2020s 开始死亡,1100 多秒后,所有节点死亡。采用本算法,第一个死亡节点出现的时间推迟到了第 3750s,到所有节点死亡仅用了 240s。本算法将网络 FND 向后推迟的同时,所有节点接近同时死亡。这是因为簇头的均匀分布可以避免各节点与簇头之间距离差异而引起的耗能的差距。选取簇头时,依据节点的剩余能量水平,避免了簇头节点由于耗能较大而能量消耗过快,可能过早死亡的情况,同时对于每轮中剩余能量最小的节点,由于会距离簇头节点较近,传输耗能受到了“保护”,从而也利于节点能耗均衡。

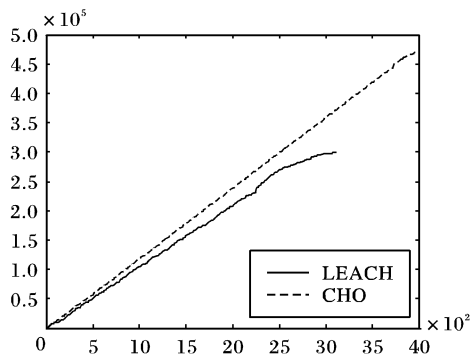


图5 运行时间与接收数据量

仿真时间与基站收到的总数据量之间的关系如图5所示。从图中可以看出,本算法基站接收的数据量要比 LEACH 多。虽然在每一轮基站不会收到更多的数据,但是因为本算法延长了网络生存时间,所以基站就可以有更多的时间去接收数据。从而提高了整个网络传输的数据量和网络的效率。

本算法在分簇时避免了簇头过于集中,同时在选取簇头时“保护”了能量最低点,从而保证了节点能耗更为均衡,延长了网络生存寿命,同时也提高了数据传输量。

参考文献:

- [1] HEINZELMAN WB, CHANDRAKASAN AP, BALAKRISHNAN H. An application - specific protocol architecture for wireless microsensor networks Wireless Communications [J]. IEEE Transactions, 2002, 1(4): 660 - 670.
- [2] HEINZELMAN WR, CHANDRAKASAN A, BALAKRISHNAN H. Energy-Efficient Communication Protocol for Wireless Microsensor [A]. Proceeding of the 33rd Hawaii International Conference on System Sciences [C]. 2000.
- [3] MHATRE V, ROSENBERG C. Homogeneous vs heterogeneous clustered sensor networks: a comparative study [A]. 2004 IEEE International Conference on Communications [C]. 2004, Vol 6: 3646 - 3651.
- [4] YONIS O. HEED: A Hybrid, energy-efficient, distributed clustering approach for ad - hoc sensor network [J]. IEEE Trans on Mobile Computing, 2004, 3(4): 366 - 379.
- [5] LINDSEY S, RAGHAVENDRA CS. Pegasus: Power efficient gathering in sensor information systems [A]. Proceedings of IEEE Aerospace Conference [C]. Montana, USA: IEEE Computer Society, 2002. 23 - 29.
- [6] 龚海刚, 刘明, 陈力军, 等. DEED: 一种无线传感器网络中高效节能的数据通信协议 [J]. 电子学报, 2005, 8(33): 1391 - 1396.
- [7] 孙利民. 无线传感器网络 [M]. 北京, 清华大学出版社, 2005.