

能量预约法改善簇头轮内死亡问题

邓亚平, 蒋新春

DENG Ya-ping, JIANG Xin-chun

重庆邮电大学 计算机学院, 重庆 400065

College of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

E-mail: jxc-13s@163.com

DENG Ya-ping, JIANG Xin-chun. Using energy-reserved method to improve problem of cluster-heads' death in round. Computer Engineering and Applications, 2008, 44(23): 98-101.

Abstract: A problem of death of cluster-heads in a round due to a long round duration is an important issue in routing algorithms based on LEACH in which an operation is divided into rounds, because the problem effects the networks' usability. In order to deal with the problem, the authors introduce a variable round mechanism and an energy-reserved method in this paper. Parameters in the energy-reserved method must be satisfied for a number of constrained factors so that the energy-reserved method takes effect. The simulation shows that the optimal energy-reserved method enhances the usability of WSNs.

Key words: Wireless Sensor Network (WSN); Low Energy Adaptive Clustering Hierarchy (LEACH); round

摘要: 在基于低功耗自适应集簇分属协议将操作划分为轮的算法中, 因轮长过大而导致的多数簇头轮内死亡是影响网络有效使用的重要问题。为有效地控制簇头轮内死亡的发生, 提出了变长轮的思想, 引入了能量预约法。为使能量预约法产生作用, 其参数的确定必须满足多方面的制约因素。仿真结果表明一个优化的能量预约法增强了网络的可用性。

关键词: 无线传感器网络; 低功耗自适应集簇分属协议; 轮

DOI: 10.3778/j.issn.1002-8331.2008.23.030 **文章编号:** 1002-8331(2008)23-0098-04 **文献标识码:** A **中图分类号:** TP393

1 引言

无线传感器网络由于其独特的优势, 便于应用于不同的领域。由于网络中结点的能量、带宽的限制, 对无线传感器网络的设计提出了特有的挑战^[1-3]。从网络层角度看, 为无线传感器网络高效地感应信号, 尽可能地节约能量, 出现了多种数据路由协议^[2](SPIN^[4], Direct Diffusion^[5], LEACH^[6-7])。它们的核心思想是在某一时刻仅有少量的相关结点参与数据路由, 减少能量的消耗。现有路由协议算法按网络结构可以划分为层次路由和平坦路由。由于层次路由具有较好的扩展性, 可应用于大范围的网络中, 其中 LEACH 是层次路由的经典算法。

2 基于 LEACH 的路由算法简介

2.1 LEACH 算法

LEACH 是一种自适应聚类算法, 其基本思想是: 在整个无线传感器网络中随机选择结点, 分布数据能量于整个网络, 完成数据收集任务。为实现上述目标, LEACH 要完成的基本任务: 一是局部小区域内结点相互协作, 共同形成一个数据操作单元——簇; 二是随机选取小区域的代表点即簇头结点, 并构造分配簇成员; 三是在与基站通信之前, 在簇内实现局部数据聚合、压缩, 以减少数量。

LEACH 的基本操作单元是轮, 每一轮可以分成两个阶段: 建立阶段和稳定数据传送阶段。簇建立阶段又可以分为簇头选择、形成簇、建立 TDMA 时间表三个子阶段。

簇头选择子阶段: LEACH 采用一种能够自治、无需中心控制的分布式算法来实现簇头的选择。该算法要保证: 每一轮时间片内网络中的簇头个数为规定值, 必须将数据能量消耗分布于各个结点, 防止部分结点能量消耗过大而死亡。LEACH 所采用的分布式算法为: 在每一轮开始, 网络中所有结点自己产生一个 0~1 随机值来表示成为簇头的概率, 当概率值小于整个网络的一个门限函数 $T(n)$ 时, 该结点就可以成为簇头。

形成簇子阶段: 当簇头产生之后, 簇头结点必须确保网络中的所有结点都知道簇头的信息。为完成这项工作, 簇头结点使用 CSMA MAC 向网络广播 ADV (Advertise) 数据包, 该数据包含有簇头的 ID 号标识。每一个非簇头结点将根据本轮自身收到的 ADV 信号强度选择距离自己最近的簇头加入 (距离近, 信号强)。非簇头按照收到 ADV 的 ID 号返回一个 Join-REQ 数据包给相应簇头结点。当所有结点均加入各自相应的簇头域时, 网络将分成若干个小区域, 成为一个簇。

建立 TDMA 时间表子阶段: 当簇头获得所有簇成员结点的信息后, 簇头结点将为每个簇成员分配一个时隙, 用来接收

作者简介: 邓亚平 (1948-), 男, 教授, 主要研究领域为计算机网络与通信和信息安全; 蒋新春 (1981-), 男, 硕士研究生, 主要研究领域为计算机网络与通信、无线传感器网络。

收稿日期: 2008-02-29 **修回日期:** 2008-05-05

相应簇成员的感应信息。所有簇成员接收簇头发送的 SCH? ADV 数据包,从中找出自己的时隙位置数据。

数据传输阶段:当 TDMA 时间表建立之后,在簇内,簇头与成员之间采用 TDMA 方式工作。而在不同簇之间,为了减少信号的干扰,LEACH 采用 CDMA 方式工作,每一个簇用不同 CDMA 码传送数据。当簇头结点的每个数据帧收集到簇成员的数据之后,将相应数据进行聚合、压缩之后传给基站,此数据传输阶段一直持续至该轮结束。

LEACH 通过上述几个分布算法,实现了自适应式无线传感器网络数据收集任务,是基于簇的层次算法的代表之作。

2.2 基于 LEACH 的改进算法

LEACH-C 针对 LEACH 算法在产生簇头阶段未考虑簇头位置信息、未实现均匀分布的问题,提出了在簇头建立阶段,由基站完成簇头选择的方法。在每轮开始的簇头选择阶段,LEACH-C 所有结点向基站汇报相应的位置信息(可由 GPS 提供)和能量信息。基站统计所有结点的能量平均值,使那些低能的结点,在本轮中不被选为簇头。基站对选出的后备簇头,使用 annealing 算法来从中选择 k 个优化簇头。由基站选出簇头后,对相应结点发送本轮的“任命”命令。簇头结点实施 LEACH 相应的广播 ADV、接收 Join-REQ、分配时隙等工作。簇头分布均匀了,从全局来看 LEACH-C 的能量消耗比 LEACH 低,延长了使用寿命。

为了减少在 LEACH-C 中,每轮结点都向基站报告能量情况而带来的能量消耗问题,预测能量消耗法(predicted current remaining energy^[8])通过每轮的预测剩余能量来减少向其基站汇报能量的次数。

同时还有 EMCA^[9]、BCDCP^[10]等基于 LEACH 的改进算法。EMCA(Enhanced Multi-hop Clustering Algorithm)主要应用于大面积的网络,其簇建立过程可以分为簇头的形成、建立路由、创建 TDMA 时间表和同步等阶段。在 EMCA 中,提出了一种新的 TDMA 时间表机制,其基本思想是增强数据融合,保证在同一时隙有尽可能多的结点同时传送,为缩短数据融合的时间,实现节能的目的。

BCDCP(Base Station Controlled Dynamic Clustering Protocol)是一种类似于 LEACH-C 的路由算法,其充分利用基站的作用,由基站负责建立簇、寻找路由、实施管理的任务。BCDCP 的核心思想是:(1)由基站确定的所有簇尽可能包含均等个结点;(2)采用簇头间传递信息的方式,将数据逐跳传到基站。

正如上述各个算法一样,所有的基于 LEACH 的算法都将基本操作分成了轮,每轮由一个一个数据帧组成,每一个结点在簇内与该簇头通信对应一个时隙。

3 簇头轮内死亡问题与能量预约法

3.1 簇头轮内死亡问题

所有基于 LEACH 的算法,都有相似的数据帧格式,如图 1。

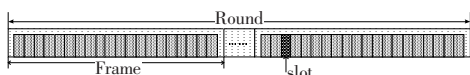


图1 LEACH 帧结构图

每一轮可根据簇成员个数划分为若干帧,每一帧的一个时隙对应一个簇成员。每帧中的最后时隙用作簇头结点来与基站通信。由于簇头结点在网络中的重要作用,既要接收簇成员的

数据,又要向基站汇报数据,因此需要簇头结点具有较高的可靠性。虽然 LEACH-C、EMCA、BCDCP 等基于 LEACH 的改进算法皆能保证簇头结点能量相对较高,但是当在网络算法运行的后期,所有结点能量均不高,而当一轮时间片过长时,簇头结点将无法完成数据聚合信息的任务。这种问题出现在所有以轮为操作单元的算法中,而对于 EMCA、BCDCP 等适用于大规模的网络、采用单个结点与基站通信的算法而言则更加突出。

在基于 LEACH 的算法中,某一簇头结点因一轮时间未到而过早死亡时,将导致簇内结点失去了汇聚点,无法向基站汇报感应信息,基站也将失去相应数据,这种情况给一个网络的正确使用带来很大影响。

该情况下,算法可以做以下三种选择:一是,所有簇内结点就此等待下一轮开始;二是,所有成员直接与基站通信;三是,该簇内的结点再重新选择一个新簇头来完成剩余轮时间片工作。

第一种情况,算法中实现简单,会失去相应区域的结点信息,因此不可取;第二种情况,将导致整个网络的能量消耗剧增,导致部分低能量结点很快死亡;第三种情况,对于一个多跳的簇结构网络来说,实现很困难:不仅要向本簇内广播,还要向相关联的簇头结点报告。

综上所述,簇头轮内死亡是无线传感器网络亟待解决的一个重要问题。要改善该问题可以采用变长轮的方法。

3.2 能量预约法——变长轮机制

所有基于 LEACH 的算法中,轮长由算法事先设定,且在整个网络运行过程中保持不变。由于轮长不仅与网络的结点数,还与结点能量有关,而结点数和能量将随着网络算法的运行而不断变化,因此对于轮长固定的定长轮而言,不可能满足变化的能量要求。如果将轮长设置太长,将会出现簇头结点由于能量不足而死亡,这样的结果将严重影响网络的使用。为防止簇头结点轮内死亡,可以采取预约能量的方法来解决。

能量预约法:首先能够保证簇头结点具有相对较高的能量,不会出现量能太低的情况,这种算法可以参照 LEACH-C。在一个有 n 个成员的簇内,初始能量均等,在 n 个结点中预约能量 E 供簇头使用,而剩余能量则由非簇头结点消耗。这样簇内结点能量总和被分成了两部分:簇头能量和非簇头能量。能量分配的要求是,在每个结点都成为簇头之前,尽可能少的结点死亡或者不死亡。

每帧非簇头结点消耗的能量为:

$$E'_{non-CH} = l \cdot E_{elec} + \epsilon_{fs} d_{toCH}^2 \quad (1)$$

在 n 个结点的簇内,每帧簇头结点消耗的能量为:

$$E'_{CH} = \begin{cases} (n-1) \cdot lE_{elec} + lE_{elec} + \epsilon_{fs} d_{toBS}^2 & (d_{toBS} < d_0) \\ (n-1) \cdot lE_{elec} + lE_{elec} + \epsilon_{mp} d_{toBS}^4 & (d_{toBS} > d_0) \end{cases} \quad (2)$$

在以时间长度为 T 、每一个时隙大小为 μ ,由 $n-1$ 个成员结点组成的簇中,每轮能量消耗为:

$$E_{non-CH} = \frac{T}{(n-1)\mu} \times E'_{non-CH} \quad (3)$$

$$E_{CH} = \frac{T}{(n-1)\mu} \times E'_{CH} \quad (4)$$

按能量预约法将得到变长轮时间片的大小。这样得到:

$$nE/E'_{CH} = (E_{total} - nE)/(n-1)\beta E'_{non-CH} \quad (5)$$

其中:(1- β)为本轮内死亡的结点百分数。

轮时间片长度为 $T = (n-1)\mu E/E'_{CH}$ 得到:

$$T = \frac{(n-1)\mu E_{total}}{n \cdot ((n-1)\alpha\beta + 1) \cdot E'_{CH}} \quad (6)$$

其中： $\alpha = \frac{E'_{non-CH}}{E'_{CH}}$ 非簇头与簇头能量消耗的比值。

3.3 能量预约法量化分析

由于基于 LEACH 的算法都有一个生成簇头、形成簇、构建 TDMA 时间表的过程，这一个过程的开销也是能量消耗的一部分，即 $E_{overhead}$ ，而 $E_{overhead}$ 的大小因不同情况而不等。为减小 $E_{overhead}$ 值，最佳的轮长为 T_{Net} ，整个算法过程中仅有产生一次 $E_{overhead}$ 。但由于传感器结点的能量有限，任何一个结点都不可能在 T_{Net} 内承担簇头工作。

为了不出现轮内簇头结点死亡问题，簇头结点的能量消耗公式(4)，必须满足： $E_{CH} < E_{initial}$ ($E_{initial}$ 为一轮开始时，簇头结点初始能量)

将得到 T 的上限：

$$T < \frac{E_{initial} \cdot (n-1)\mu}{E'_{CH}} \quad (7)$$

在整个网络中，结点能量逐渐被消耗， $E_{initial}$ 将随时间推移而不断减少，因此轮长 T 固定为某一个值，是不能满足整个网络运行要求的。加之在网络运行后期，簇头结点可能由于能量较低而出现轮内死亡的情况，针对定长轮不足，引入轮变长机制是很必要的。

按能量预约法，得到变长轮轮长值，如公式(6)。变长轮轮长与簇内总能量、簇内活动结点个数、非簇头结点与簇头结点能量比 α 、本轮内剩余的结点百分数 β 相关。轮长的决定因素： α 值主要取决于传感器结点的硬件、基站的位置等客观因素。 β 则取值于 0~1 之间， $\beta=0$ 时，将取到轮长 T 的上限，算法的前几轮将有簇头死亡； $\beta=1$ 时，要保证每一轮之后，结点个数保持不变，这样的结果是轮长因过多照顾非簇头结点而过于短小，而不适于网络的有效使用。从有效的利用无线传感器网络的角度出发， β 取 0.6~0.8。

4 仿真与分析

4.1 系统模型

仿真实验中，为了变长轮与定长轮性能相比较，在此采用文献[7]中与定长轮相同的参数设置进行 LEACH 仿真，主要参数见表 1。

参 数	值
网络规模	100×100
结点个数	100
基站位置	(50,200)
簇头个数	5
E_{elec}	50 nJ
ϵ_f	10 pJ
e_{mp}	0.001 3 pJ
μ	4.2 ms

4.2 变长轮与定长轮数据对比

为得到变长轮与定长轮的数据对比，仿真过程中，结点的初始能量为 2 J。首先按文献[7]中所规定轮长，即 $T=20$ s 作为轮时间片长度。仿真结果显示出在网络的后期，轮长为 20 s 时将导致大量的簇头结点轮内死亡，如图 2 所示。

如图 2 可知：在第 17~23 轮中，5 个簇头至少有 3 个在簇内死亡，且多数簇头在每轮的前半段死亡。

为解决由于轮时间过长而导致的轮内死亡的问题，将轮时

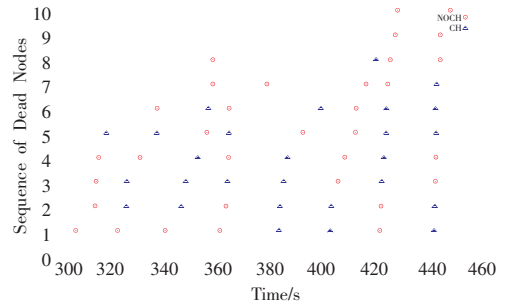


图 2 轮长为 20 s 时，簇头轮内死亡情况

间片长设为 10 s。簇头轮内死亡的情况有所改善，但依旧影响使用，如图 3 所示。而且将轮长固定于 10 s，不适应于能量较足的网络初期，簇头结点更迭频繁。

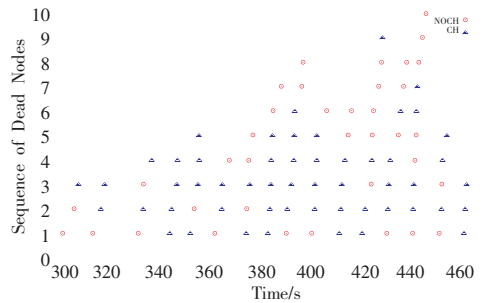


图 3 轮长为 10 s 时，簇头轮内死亡情况

从图 3 得知，轮长为 10 s 的网络后期，簇头轮内死亡问题依旧存在。如在 380 s、390 s 开始的第 38、39 轮中，5 个簇头都在轮内死亡。对于一个轮长为 10 s 的网络而言，前 30 轮内能量的使用效率不高，导致没有必要的更换簇头的情况发生。

不论轮长规定值的大小如何，没有一个固定的轮长既能满足能量效率要求，又能防止簇头轮内死亡。根据公式(6)，仿真变长轮的算法，仿真结果证明，对于网络初期，轮时间片因能量充足而较长，提高了能量使用效率，减少了不必要的簇头更迭；网络后期，由于能量不足，为减少簇头的轮内死亡情况的发生，轮时间片逐渐减小。如图 4 和图 5 所示。

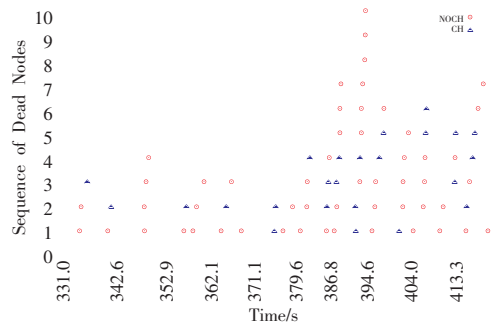


图 4 不定轮长簇头结点死亡情况

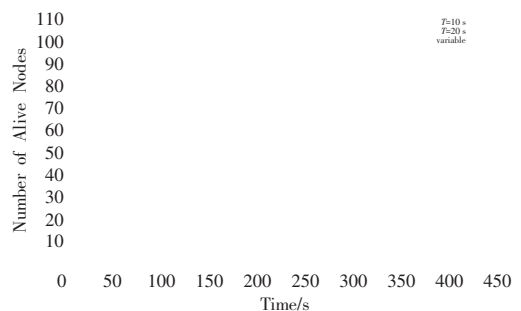


图 5 不同轮长时网络结点死亡情况对比图

如图 4 所示,变长轮算法有效地控制了簇头结点在轮内死亡的问题。对比定长轮的情况,变长轮结点死亡皆在轮时间片的后期,而采用定长轮时,簇头轮内死亡多数发生在每一轮的前半段。

如图 5 所示,变长轮算法中,结点死亡状况好于轮长为 20 s 的情况。

通过上述分析可以看出,对于基于 LEACH 的以轮为操作单位的算法,轮时间长度是影响网络性能的重要因素。如果轮持续过长,过早出现结点死亡的情况,而在网络后期将导致多数簇头轮内死亡;当轮长时间过小,会造成过多的能量消耗和频繁的簇头更迭。采用本文采用变长轮方法,即轮长随网络能量的变化而变化,可以改善上述问题,较传统的定长轮增强了网络的可用性。

4.3 β 值对能量预约法的影响

从公式(6)中,得知 β 值是决定能量预约法的重要因素,不同的 β 值将产生不同的效果。为得到效果图,该节中结点仿真初值为 1 J。

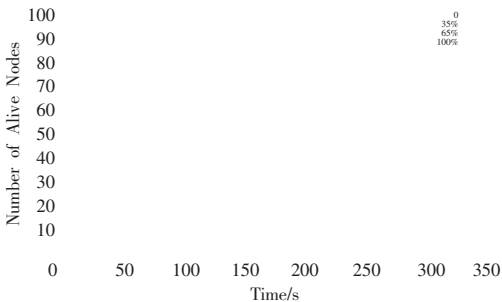


图 6 β 值对网络结点死亡的影响

图 6 和图 7 为 β 值对网络中结点的死亡情况的影响。从图 6 中可以看到:当 $\beta=0$ 时第一个结点死亡发生在网络第 25 s 左右;随着 β 值增加,前 10 个结点死亡所用的时间越来越长, $\beta=0$ 用时 70 s, $\beta=0.35、0.65、1$ 时用时分别为 125 s、180 s、240 s。

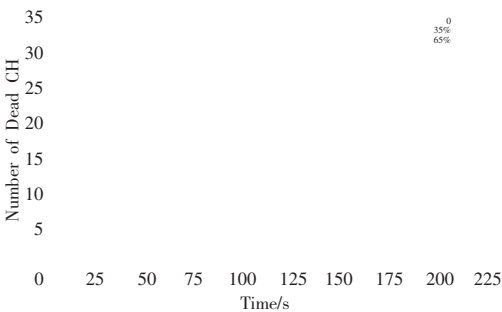


图 7 β 值对簇头结点死亡的影响

图 7 为前 50 个死亡结点中,以簇头身份死亡的结点数,可以看出: β 值为 0、0.35、0.65、1 时,其中死亡的簇头个数为 33、14、5、0。这说明 β 值越大对簇头死亡的改善情况效果越好。

从网络内部来看,产生图 6 和图 7 结果的原因是如果每一轮中簇头结点预约的能量多(β 值小),结点完成簇头任务需要的能量大,导致轮内死亡。相反,簇头结点预约的能量少(β 值大),保证了结点在担任簇头时间内发生死亡情况少。

图 8 和图 9 为 β 值对网络轮数的影响,可以得知:(1) β 值越小,轮数越少,建立阶段所消耗的能量也就越少。 β 值为 0、0.35、0.65、1 时网络生存期将被划分为 25、40、60、265 轮。(2) β 值越大,网络后期轮长越短。 β 值为 1 时,稳定传送时间小于 1 s

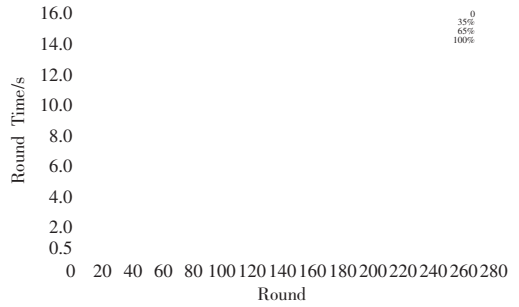


图 8 β 值对轮长的影响

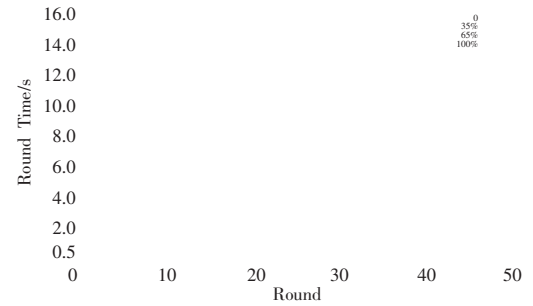


图 9 前 50 轮轮长情况

的轮共有 200 个之多。

图 10 为 β 值对基站收到的数据包个数的影响, β 值越小,基站收到的数据包越多。其原因是: β 值过大时,网络将消耗过多的时间用于网络建立。

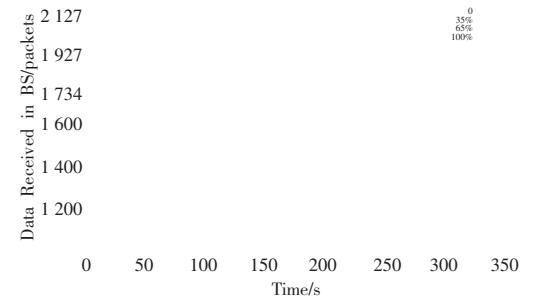


图 10 β 值对网络吞吐量的影响

通过本节的讨论可以看出,使用能量预约法时 β 值对网络的使用具有十分重要的影响。 β 值小时,轮时间片长,网络吞吐量大,有利于提高无线传感器网络的使用效率;但会过早发生结点死亡的情况,且簇头死亡情况严重,这将影响网络可靠性。 β 值大时,网络的使用寿命长,簇头轮内死亡情况少,可以保证每一个簇头在轮时间片内完成各自的任务,增强了网络簇头结点的可靠性;但由于轮长短,造成簇头更迭频繁,降低了网络使用效率。因此,为有效的使用能量预约法来改善簇头轮内死亡问题,选择 β 值时必须慎重,需要考虑多方面因素。经过仿真得到:对于要有效改善簇头轮死亡、提高网络使用效率的网络而言, β 一般取于 0.6~0.8。

5 结束语

由于 LEACH 等基于簇的路由算法,需将操作划分为轮,每一轮的长短对整个网络有很大影响。如果 T 过小,那么能量的使用效率低,导致没有必要的簇头更迭,而当 T 过大时将导致整个网络出现轮内死亡问题。本文采取能量预约的方法来改