

物联网感知层局域按需簇维护模型与算法^{*}

胡向东¹, 徐慧芬², 张力¹

¹(重庆邮电大学 自动化学院, 重庆 400065)

²(重庆邮电大学 通信与信息工程学院, 重庆 400065)

通讯作者: 胡向东, E-mail: huxd@cqupt.edu.cn, http://www.cqupt.edu.cn

摘要: 基于无线传感网的物联网感知层传统的“全网”、“周期性”重新成簇的簇维护模式因超范围过度维护, 存在维护成本高、能量浪费严重、服务全面中断、响应不及时等缺点. 局域按需簇维护方法(local and on-demand maintenance of clusters, 简称 LDMC)将簇维护操作控制在簇受损的时间和空间范围内, 通过设置触发源、预处理和维护动作分别解决簇维护启动、簇维护方式和簇维护范围问题, 不仅能够克服簇更新周期确定的困难, 而且可在节点失效和新节点加入时对网络拓扑和路由变化及时进行响应, 减小突发事件对网络功能的影响, 改善网络的稳定性并降低其维护开销. 基于 NS2 仿真平台, 分别从能量消耗、数据传输、负载平衡和突发事件响应等角度对该方法进行了测试对比, 仿真结果表明, 该方法能够明显减少簇维护的能量消耗、延长网络生存时间, 并增加传输数据包的总量.

关键词: 物联网感知层; 簇维护; 事件驱动; 局域; 按需

中图法分类号: TP393

中文引用格式: 胡向东, 徐慧芬, 张力. 物联网感知层局域按需簇维护模型与算法. 软件学报, 2015, 26(8): 2020–2040. <http://www.jos.org.cn/1000-9825/4675.htm>

英文引用格式: Hu XD, Xu HF, Zhang L. Model and algorithm of local and on-demand maintenance of clusters in sensing layer of the Internet of things. Ruan Jian Xue Bao/Journal of Software, 2015, 26(8): 2020–2040 (in Chinese). <http://www.jos.org.cn/1000-9825/4675.htm>

Model and Algorithm of Local and On-Demand Maintenance of Clusters in Sensing Layer of the Internet of Things

HU Xiang-Dong¹, XU Hui-Fen², ZHANG Li¹

¹(School of Automation, Chongqing University of Posts and Telecommunications, Chongqing 400065, China)

²(School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China)

Abstract: The sensing layer of the Internet of things based on wireless sensor network requires intensive maintenance over the involved nodes and beyond by the conventional whole-network and periodic model of cluster maintenance. It therefore results in some shortcomings such as high cost of maintenance, heavy waste of energy, full service interruption, and delayed response to incident. This paper proposes a new method, namely, local and on-demand maintenance of clusters (LDMC), to carry out the operations of maintenance only in a restricted range of time and space which is decided by the damaged clusters. LDMC resolves issues such as when to start, which type or who to be maintained by the triggers, pre-processing or different operations of maintenance. The presented scheme is helpful not only to make critical decision on the timing of cluster update, but also to provide in-time response to the change of topology or route of network with the disabled or new nodes. As a result, it allows to reduce the impact of incidents on the function of network, to improve the stability of network, and to cut down the cost of maintenance. Comparison tests are performed on consumption of energy, transmission of

^{*} 基金项目: 国家自然科学基金(61170219); 重庆市自然科学基金(CSTC2013jcyjA40002); 重庆市高等学校优秀人才支持计划([2011]65)

收稿时间: 2013-11-25; 定稿时间: 2014-07-01

data, balance of load and response to incident based on NS2 simulation platform, and the results suggest that the proposed method is able to significantly reduce energy consumption in maintenance of clusters, to prolong the lifetime of network, and to increase the total amount of the transferred data packets.

Key words: sensing layer of the Internet of things; maintenance of cluster; incident-driven; local; on-demand

确保高效率运行和能量节省,是具有实用意义的基于无线传感网的物联网感知层的基本需求.但网络中传感器节点无人值守的分布方式和能量、存储、通信能力等资源严格受限的突出特点,决定了它具有高度动态性和脆弱性,使得物联网感知层的维护成为一项较为棘手的工作^[1],特别是有众多传感器节点分布的大型网络,面临着如何及时、安全、高效地进行簇维护的困难,以及如何确保簇维护模式具有良好的适应性、可扩展性和对突发事件的快速响应^[2].然而,目前还没有真正意义上的簇维护算法,以 LEACH^[3]为典型代表、以预设的时间周期为单位的传统周期性全网簇头轮换与拓扑重建的重新成簇方法基于时间驱动或能量驱动,从效果上能够实现簇维护,但存在超范围过度维护、轮换周期难以优化确定、维护成本高、能量浪费严重、服务全面中断、灵活性差、应急响应不及时等突出问题.

本文以灵活的簇维护方式、节省节点能量消耗和延长网络生命周期^[4]、提高快速响应突发事件的能力为基本目标,提出了一种基于事件驱动的物联网感知层局域按需簇维护方法(local and on-demand maintenance of clusters,简称 LDMC),从何时开始簇维护、采用何种簇维护方式和如何实现簇维护的角度来突破传统的“全网”、“周期性”重新成簇的簇维护方法存在的诸多弊端,将簇维护操作限定在网络内特定簇受损的时间和空间域内,以明显降低网络维护开销、确保网络高效而稳定地运行.

1 相关研究综述

作为最早被提出来的无线传感网分簇路由协议,LEACH^[3]采用典型的周期性簇头轮换簇维护模式,以预设的时间为周期,按轮循环随机地在全网内选择簇头节点. TEEN^[5]采用与 LEACH 相同的周期性簇头轮换的簇维护模式,不同之处在于:每轮簇头轮换时,簇头节点除了通过 TDMA 方法实现数据的调度,还向簇内成员广播有关数据的硬阈值和软阈值两个参数,用于节点判定并传送最新的监测数据. TEEN 实现了在数据传输方面对被监测突发事件的快速响应和维护,但 TEEN 的维护仅是对簇内通信的维护,使被监测突发事件的数据能够更快地被传输,对于影响到网络通信的突发事件,TEEN 仍采用周期性簇头轮换方式实现对簇结构的维护. HEED^[6]规定簇头的选择主要依据主、次两个参数,其中,主参数依赖于剩余能量,具有更高剩余能量的节点将有较高的成为临时簇头的概率,其算法的收敛速度较快,次参数包括节点邻近度或者节点密度,处于相同簇覆盖范围的多个簇头节点通过次参数(平均可达能级 AMRP)来竞争出最终的簇头; HEED 使用临时簇头对 AMRP 值过高的临时簇实现维护,得到更高效的簇结构,但 HEED 只在簇形成阶段进行这种维护,在簇形成之后仍使用周期性簇头轮换的簇维护模式. 在 LEACH-F^[7]中,簇是由基站采用模拟退火算法生成的,且基站为每个簇生成一个簇头列表,指示簇内节点轮流当选簇头的顺序;一旦簇形成,簇的结构就不再改变,簇内节点根据簇头列表依次成为簇头; LEACH-F 最大的优点就是每次对簇结构进行维护时无须再生成簇,减少了生成簇的开销,但 LEACH-F 并不适合真实的、动态性较强的网络. 文献[8]对 LEACH 协议进行改进,提出了一种提高无线传感网能效的双轮成簇协议 EEDRCP,在簇头选取算法中引入剩余能量参数,结合平面拓扑、层次拓扑和位置拓扑的优点形成混合型拓扑,同时,改单轮成簇为双轮成簇. 仿真结果表明,与 LEACH 协议相比, EEDRCP 协议可提高网络能效 33%~82%. 文献[9]提出了一种分布式能量有效的传感器监测网络成簇协议 EECT,节点根据邻居节点的分布以及自己的剩余能量竞争簇头. 为了降低簇头的能量消耗,簇头间形成一个以基站为根的最小生成树,将监测到的数据通过多跳方式直接发送到生成树的上游节点. 此协议还提出了一种簇内调度方法 EECTS-1,可监测到网络中的大部分区域,并在此基础上提出了改进方法 EECTS-2. 仿真结果表明,在监测性能相同的情况下,运行 EECTS-1 协议的网络寿命与 HEED 协议相同,比 DEEG 协议的网络寿命延长约 35%, EECTS-2 协议比 EECTS-1 和 HEED 协议的网络寿命延长约 70%~80%.

近年来出现了一些非周期的簇维护方法^[10],如基于最小 ID 的 CBRP 算法、基于能量阈值的 EDAC 与 EDCR

算法和基于最大连通度的簇维护算法.CBRP^[11]的目标是实现一个分布式、高效的可扩展路由协议,采用最小 ID 分簇算法降低簇维护开销,并且采用本地修复机制来增加分组投递率、减少路由发现时延及其开销,同时,使用路由缩短机制基于节点的两跳拓扑信息来优化路由.CBRP 特别适合大部分业务流量是由很少一部分发送节点产生的网络,并且其应用应能容忍路由发现的时延.EDAC^[12]和 EDCR^[13]基于动态计算的能量阈值来触发簇头轮换.文献[12]分析了 EDAC,EDCR 等算法采用的能量阈值计算方式,并给出了特定条件下的最优解.文献[13]以单簇为对象,分析证明了能量驱动的簇头轮换策略优于时间驱动的簇头轮换策略,并给出了单簇模型下的次优解求解方法.随着节点剩余能量的降低,这种方法将会导致簇头轮换越来越频繁,在网络平均能量较低时,节点的剩余能量将被大量消耗于频繁的簇头轮换而非数据传输,降低了节点能量的利用率.基于最大连通度^[14]驱动簇头生成算法的目的在于减少簇的数目,节点之间通过交互控制消息了解其邻居节点数,和其他节点具有最大连通度的节点被选为簇头;当连通度相同时,则选择 ID 最小节点作为簇头,簇头的一跳邻节点则成为该簇的成员节点.重复进行上述过程,直到所有节点都加入某个簇.其优点在于簇的数目较少,能够减少分组的投递延时,但建立的簇重叠度较高,同时未考虑负载均衡问题.

前述研究主要集中在如何成簇、成簇的拓扑控制方面,没有对网络基于簇结构运行一段时间之后的能量高效簇维护需求提出系统的解决方案,而网络中传感器节点的高度动态性地决定了网络生命周期内多次的簇维护需求是必然的,传统的全网周期性重新成簇会引起大量不必要的能量消耗,特别是当网络中只有个别或少量传感器节点失效、受到攻击或被破坏时.如何对具有高度动态性的物联网感知层建立起具有自适应特征的能量高效簇维护机制并实现可操作性强的簇维护算法,是本文关注的主要内容.

2 局域按需簇维护模型的建立

2.1 局域按需簇维护机制的提出

传统的周期性簇头轮换方法以预设的时间周期为单位进行全网簇头轮换与拓扑重建,如图 1 所示.图中 T_3 时刻是新一轮的簇头轮换时刻,在 T_3 之前的 T_2 时刻停止所有节点的通信;在 T_5 时刻簇头轮换结束,网络恢复通信.该方法存在几个明显的缺点:

- 1) 轮换在全网内进行,导致整个网络功能的周期性中断;
- 2) 会造成本身数据处理量不大或监控任务不重的区域因重新成簇产生大量不必要的能量浪费;
- 3) 网内不同区域的能量消耗不均匀导致轮换周期难以优化确定;
- 4) 网络发生故障时,受影响的节点要在下一次轮换簇头后才能回归网络,对故障的响应恢复不及时.

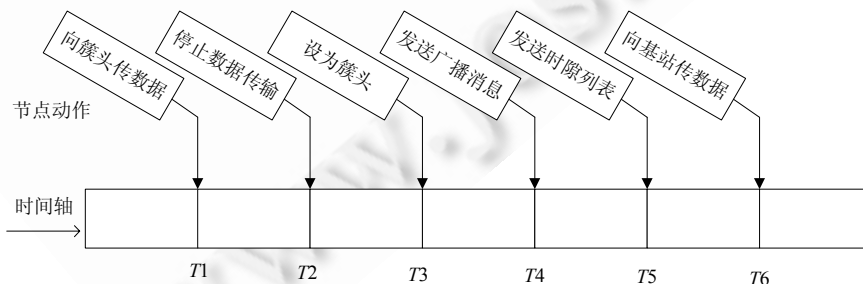


Fig.1 Periodic turn of cluster-head

图 1 周期性簇头轮换过程

为了应对上述问题,本文提出局域按需簇维护(LDMC)方法,主要思路是基于事件驱动且只在受影响的局域范围内进行,即,在完成首次成簇之后,依据网络对事件的响应需求,如簇头节点的剩余能量不足、新节点加入或恶意节点隔离等,只在事件所影响的有限范围内通过注销簇、新建簇或更新簇的方式对网络的簇结构进行快速、及时的维护.LDMC 方法重点回答 3 个问题:

- 何时开始维护?
- 选择何种维护方式?
- 如何实现维护?

分别通过设置触发源、预处理和设置维护动作来解决.

2.2 事件驱动机制

事件驱动机制包括事件侦听、事件预处理、维护动作这 3 个过程,如图 2 所示.事件侦听的作用是监测需维护的事件,并在事件发生后收集事件信息存入事件表;事件侦听得到的事件信息主要来自超时触发源、能量过低触发源、恶意行为触发源和统计异常触发源.事件预处理包含优先级比较和维护动作检索两个过程:优先级比较是比较事件表中事件的优先级,优先处理最高级事件;维护动作检索是根据待处理事件信息去匹配某一种簇维护动作.维护动作就是受事件影响节点对事件的响应,按照这些维护动作所实现的效果,可以将维护动作分成 4 类:新建簇、注销簇、更新簇和安全维护.

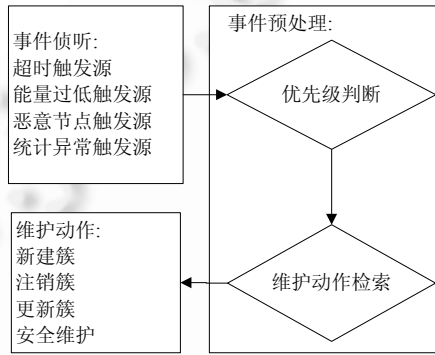


Fig.2 Incident-Driven mechanism

图 2 事件驱动机制

2.3 触发源

在事件侦听模块,传感器节点通过触发源监测需要触发维护的事件,一个触发源对应着一种事件.当网络中发生一个需要启动维护的事件时,受到该事件影响的传感器节点上对应触发源的触发条件将得到满足,从而触发该节点将触发信息添加到事件表中.

传感器节点发生的不同事件对应着不同的触发源,见表 1.不同工作状态下的传感器节点需要监测的事件是不同的,如簇头节点需要监测自身的剩余能量能否胜任簇头的事件,有与之对应的触发源,而成员节点不需要监测这个事件,就不需要这个触发源.

Table 1 Sources of trigger

表 1 触发源

触发源类别	触发源名称	所属对象	事件描述
超时触发源	TO_CH_NR	簇头节点	成员节点无应答
	TO_MN_NR	成员节点	簇头节点无应答
	TO_CH_NW	簇头节点	成员节点未工作
	TO_MN_NW	成员节点	簇头节点未工作
能量过低触发源	EN_CH_L	簇头节点	剩余能量低于阈值
恶意行为触发源	SE_CH_MB	簇头节点	簇头发现成员节点有恶意行为
	SE_MN_MB	成员节点	成员节点发现簇头有恶意行为
统计异常触发源	ST_CH_NL	簇头节点	成员节点个数过少
	ST_MN_NN	成员节点	当前簇头非最近

2.3.1 超时触发源

当物联网感知层出现传感器节点连接故障时,会引发连接超时或者长时间无应答等现象.因此,超时触发源主要用来监测网络的连接.物联网感知层发生网络连接故障的情况主要有以下3种:

- (1) 簇头节点故障或死亡,此时会引起簇头不再响应成员节点的请求或成员节点不再能收到簇头定期的广播消息.因此,成员节点从发送请求时开始计时,如果发生超时,则说明该事件发生;同理,成员节点设置一个计时器计时,收到簇头的定期广播时对其清零,计时器如果发生溢出,则也说明该事件发生.
- (2) 成员节点故障或死亡,此时会引起该成员节点不再响应簇头的请求或簇头在该成员节点的时隙中不能收到数据消息.因此,簇头节点从发送请求时开始计时,如果发生超时,则说明该事件发生;同理,簇头在每个成员节点的时隙开始时计时,收到成员节点发送的数据时对其清零,如果计时器的时间超出时隙的时长,则也说明该事件发生.
- (3) 簇头节点与成员节点间通信被阻碍.当物联网感知层所处的环境发生较大变化(如地震、泥石流等)时,可能发生簇头与成员节点间通信受阻,引起成员节点以为簇头故障或死亡,或簇头以为成员节点故障或死亡.因此,该事件会被当作以上的两种事件被监测出来,使传感器节点做出应对.

2.3.2 能量过低触发源

为了保证网络中能量消耗的均衡并延长网络整体的生命周期,LDMC方法设置一个能量过低触发源,当无异常事件发生时,簇头轮换则由能量过低触发源驱动;能量过低触发源在簇头节点的剩余能量低于一个动态的能量阈值时产生触发信号.该方法可以较好地解决周期性轮换算法中轮换周期难以优化确定的问题.动态能量阈值 E_{th} 的计算方法为

$$E_{th} = (\mu + \varepsilon \times p_{now}) E_{rec} \quad (1)$$

其中, μ, ε 分别为预设值,满足 $(\mu + \varepsilon \times p_{now}) \in [0, 1]$; E_{rec} 为簇头上一次维护结束时的剩余能量; p_{now} 为当前能耗速率, p_{now} 越大, E_{th} 越大,节点充当簇头的时间就越短.

2.3.3 恶意行为触发源

这是一类与网络安全有关的触发源.在网络运行过程中,簇头或成员节点可以基于恶意节点判定规则,根据对方的行为特征来判断其是否存在恶意行为;当判定为恶意节点时,将发出安全维护触发信号.LDMC方法设置了两个恶意行为触发源.

2.3.4 统计异常触发源

传感器节点对收集的信息或数据进行统计分析,并把依据分析结果的异常而建立的触发源称为统计异常触发源.LDMC方法设置了2个统计异常触发源.网络中簇头根据统计信息发现簇内成员节点个数小于允许的最小值时,将发出一个统计异常触发信号 ST_CH_NL . ST_CH_NL 触发源是对当前簇内成员节点个数与最小允许成员个数 $member_min$ 的比较, $member_min$ 的计算方法为

$$member_min = p \times alive_node \quad (2)$$

其中, $p \in [0, 1]$, p 为预设值; $alive_node$ 为当前存活节点个数.

为了使成员节点尽量与离自己最近的簇头通信,成员节点通过分析存储的可用簇头列表.当发现存在比当前所在簇的簇头离自己更近的簇头时,该成员节点将发出一个统计异常触发信号 ST_MN_NN .

2.4 触发预处理

LDMC的维护动作在特定的时隙开始执行,所以当节点获得一个触发信息时,可能无法立即执行对应的维护动作.在这期间,如果有其他事件发生,则传感器节点在事件的处理上将会产生冲突.为了避免这类冲突的发生,在事件驱动机制中引入了优先级.同时,为了对事件的响应更灵活和更合理,LDMC方法采用事件定制模式,即将触发源与响应动作分离,二者通过映射关系建立联系.通过修改映射关系,可以方便地改变需要响应的事件和响应动作之间的关联,从而灵活地实现不同触发源触发相同的维护动作,或相同的触发源在不同节点状态下触发不同的维护动作.

2.4.1 优先级设置

触发源的优先级分为两个级别:不同类别的触发源有不同的优先级,这属于第 1 级;同一类别的不同触发源也有不同的优先级,这属于第 2 级.综合考虑网络性能需求,安全性和连通性是需要优先保障的;其次,要尽可能地延长网络的生存时间;最后,尽可能优化网络的结构.因此,将一级优先级按恶意行为触发源、超时触发源、能量过低触发源和统计异常触发源顺序确定.其内部二级优先级先后顺序设置为:

- 恶意行为触发源类:SE_CH_MB,SE_MN_MB;
- 超时触发源类:TO_CH_NR,TO_MN_NR,TO_CH_NW,TO_MN_NW;
- 统计异常触发源类:ST_CH_NL,ST_MN_NN.

2.4.2 维护动作检索

维护动作检索在每次维护时隙开始时执行,根据当前优先级最高的触发源信息在源与动作映射中查找对应的维护动作函数.若未找到则从事件表中删去当前事件,若找到则给动作设定执行时间并从事件表中删去当前事件;然后再重新检索事件表,直到事件表为空.

2.5 维护动作

维护动作是指完成对一个事件响应的一系列节点行为.传感器节点根据发送或接收特定类型的消息来改变自身的工作方式和网络的簇结构,实现对网络的维护.

2.5.1 新建簇

现有分簇协议有分布式和集中式:分布式成簇算法包括由节点自身根据阈值决定是否当选簇头(如 LEACH)和通过节点间交互信息动态产生簇头(如 HEED);集中式成簇算法是由基站基于整个网络的状态信息选择合适的簇头(如 LEACH-C).这两种新建簇方法各有优势与不足:分布式成簇速度快、耗能少,但不能控制簇个数,簇间路由由恢复难度大,一般用于节点分布密集区域;集中式成簇能够控制簇个数,簇间路由由恢复容易,但比分布式成簇速度慢且耗能多,一般用于节点分布稀疏区域.LDMC 方法同时引入这两种新建簇方式,不同的触发事件将择优选用其中一种.

LDMC 方法的分布式成簇使用基于剩余能量竞争的模式,其原理为:维护时隙开始时,每个待命节点根据自身的剩余能量产生一个延时,剩余能量越大,延时就越短.当延时时间到若未收到来自其他节点的簇头广播消息,则对外发送簇头广播.这里,延时时间 T_d 的计算方法为

$$T_d = 50 \cdot \frac{E_{init} - E_{now}}{E_{init}} + R \quad (3)$$

其中, E_{init} 是节点的初始能量; E_{now} 是节点当前的剩余能量; R 是 0~1 之间的一个随机数; T_d 的单位是 ms, 范围在 [0,51]ms. 由于簇头广播消息是很简短的一类消息,消息的发送和传播过程所需的时间非常短,因此,50ms 的时间可以完成簇头竞争;引入随机值 R 的原因在于:避免存在相同剩余能量的节点同时竞争簇头,引起信道冲突.分布式成簇的过程如图 3 所示,其中涉及的消息类型见表 2.

Table 2 Type of messages of distributed clustering

表 2 分布式成簇的消息类型

消息类型	发送半径	接收方	描述
NEW_ADV_CH_SO	额定广播距离	所有节点	广播消息.由新簇头发给一定范围内的待命节点,收到消息的待命节点可以申请成为该簇的成员节点
NEW_JOIN_REQ_SO	到簇头的通信距离	簇头节点	成员节点发给最邻近簇头请求加入其簇的消息
NEW_ADV_SCH_SO	到最远成员节点的通信距离	成员节点	广播消息.由簇头发给簇内的成员节点,成员节点根据该消息计算自身可用的时隙

LDMC 方法通常不涉及全网的重新成簇,待成簇节点的个数相对较少、分布范围较小,因此,必要时可采用局域集中式成簇方法,即,可以通过其中一个节点根据局域范围内的节点状态信息选择簇头,不通过基站来确定簇头.局域集中式成簇过程如图 4 所示,其中涉及的消息类型见表 3.

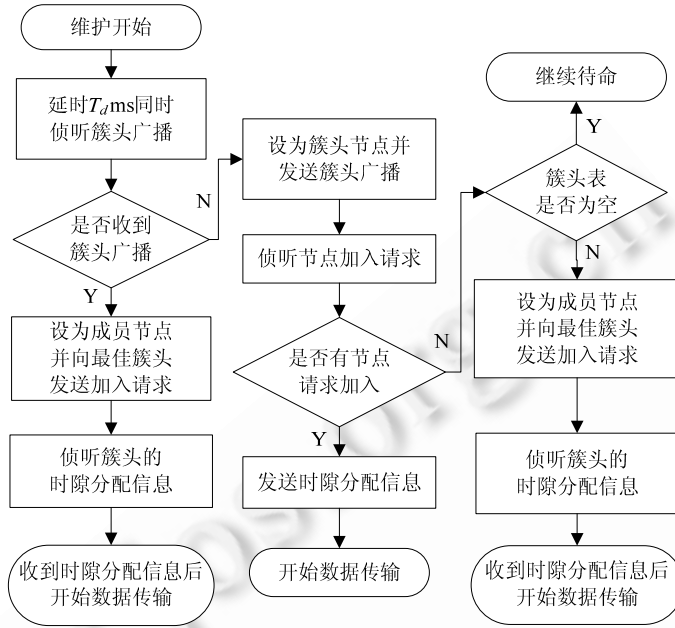


Fig.3 Process of distributed clustering

图 3 分布式成簇过程

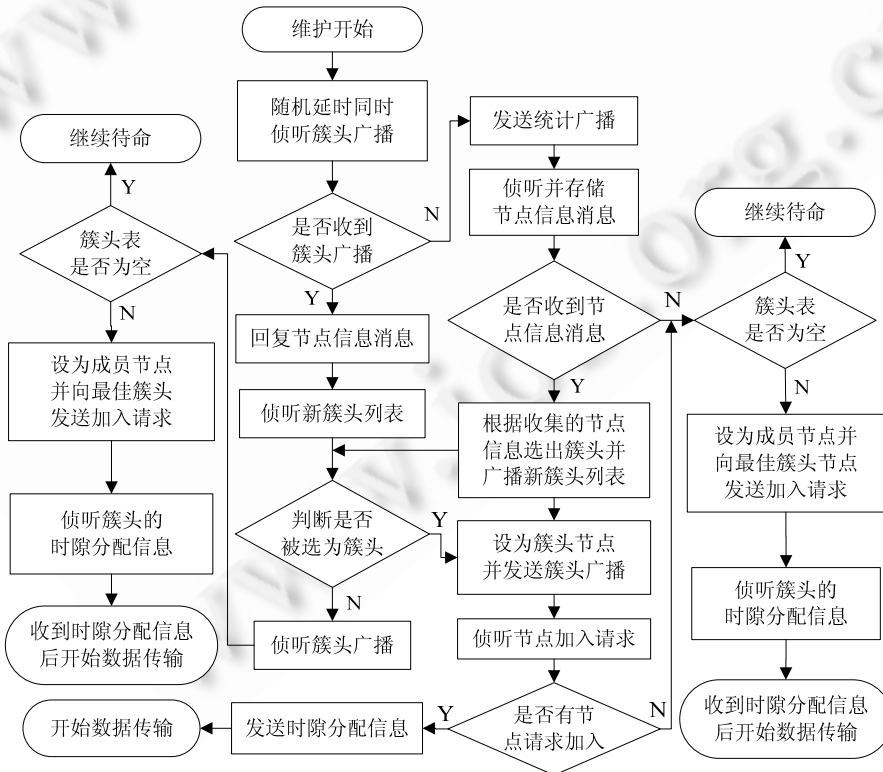


Fig.4 Process of local and integrated clustering

图 4 局域集中式成簇过程

Table 3 Type of messages of local and integrated clustering

表 3 局域集中式成簇消息类型

消息类型	发送半径	接收方	描述
NEW_ADV_ST_CE	额定广播距离	待命节点	广播消息,由统计节点发出,收到该消息的待命节点回复自身信息
NEW_INFO_ST_CE	到广播节点的通信距离	待命节点	包含节点信息的信息,发送给统计节点
NEW_ADV_ARR_CE	额定广播距离	待命节点	广播消息,统计节点广播簇头列表
NEW_ADV_CH_CE	额定广播距离	所有节点	广播消息,由新簇头发给一定范围内的所有节点,收到消息的节点更新簇头信息
NEW_JOIN_REQ_CE	到簇头的通信距离	簇头节点	成员节点发给最邻近簇头请求加入其簇的消息
NEW_ADV_SCH_CE	到最远成员节点的通信距离	成员节点	广播消息,由簇头发给簇内的成员节点,成员节点根据该消息计算自身可用的时隙

局域集中式成簇时,完成消息汇总统计功能的节点根据收到的 NEW_INFO_ST_CE 消息统计出当前簇内成员节点的个数 Num_MN,再根据 Num_MN 计算出当前簇是否需要分裂生成簇的个数 Num_CH.当 Num_MN 小于 member_min 的 λ 倍时,Num_CH 设为 1,此时,当前簇不需要分裂;否则,当前簇就要分裂并生成 Num_CH 个新簇.Num_CH 的计算方法如公式(4):

$$Num_CH = \lfloor \frac{Num_MN}{\lambda \times member_min} \rfloor \tag{4}$$

其中,符号 $\lfloor a \rfloor$ 为小于 a 的最大整数.

2.5.2 注销簇

注销簇的目的在于更新节点的簇头列表和更新簇头的簇间路由表.触发注销簇的事件包括:簇头节点剩余能量过低需要轮换;簇头死亡或受到攻击无法继续工作.对应地,有两种注销簇动作:有簇头注销和无簇头注销.涉及的消息见表 4.

- 有簇头注销是由簇头广播簇头辞职消息,收到消息的簇内成员节点从簇头列表中删除当前簇头信息,并将自身设为待命节点,等到下一个维护时隙新建簇;收到消息的簇外成员节点从簇头列表中删除辞职簇头的信息;收到消息的簇头节点从簇间路由表中删除辞职簇头的信息.
- 无簇头注销由簇内成员节点对外发布广播消息,为了能使所有保存有将注销簇的簇头信息的节点都能收到广播以更新自身所存储的簇信息,发送簇注销广播的成员节点往往需要加大发送功率,以实现对该簇影响范围的全部覆盖.簇注销消息的广播距离是由成员节点与簇头的通信距离加上簇头广播距离得到,如图 5 所示.

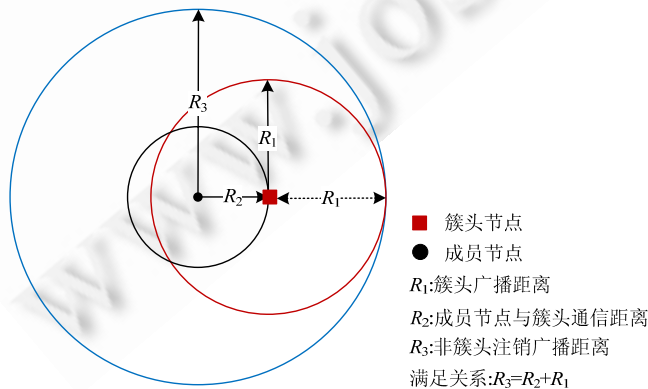


Fig.5 Count of radio distance of logged-off cluster

图 5 簇注销消息广播距离计算

Table 4 Type of messages of logged-off clusters

表 4 注销簇消息类型

注销类型	消息类型	发送半径	接收方	描述
有簇头注销	END_ADV_RES_CH	额定广播距离	所有节点	广播消息,由需要进行轮换的簇头发布
无簇头注销	END_ADV_RES_MN	到簇头的通信距离加额定广播距离	所有节点	广播消息,由将注销的簇的成员节点发布

2.5.3 更新簇

更新簇的目的在于实现成员节点更换簇头节点或待命节点加入正在运行的簇.当成员节点更换簇头时,需要离开原簇并加入新簇,因此,原簇要回收离开节点的时隙,新簇要给加入节点分配时隙.当待命节点加入正在运行的簇时,只需要该簇给新加入节点分配时隙.

在 LDMC 方法中,原簇头在失去节点的连接后会得到超时的触发信息,因此,成员节点不需要发送消息通知原簇头.更新簇动作包括两类消息,见表 5.

Table 5 Type of messages of renewed clusters

表 5 更新簇消息类型

消息类型	发送半径	接收方	描述
UPDATE_JOIN_REQ	到簇头的通信距离	簇头节点	发给簇头节点的请求加入簇的消息
UPDATE_ADV_SCH	到最远成员节点的通信距离	成员节点	广播消息,由簇头广播的时隙分配表

更新簇的流程如图 6 所示.

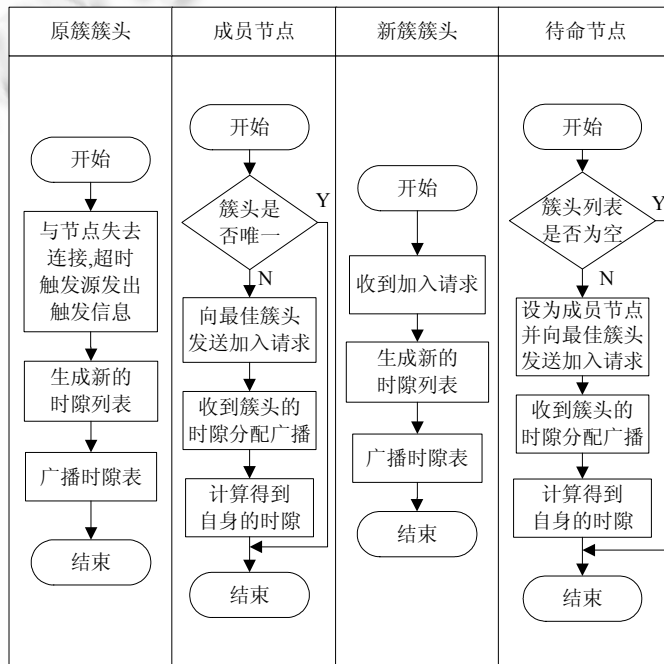


Fig.6 Process of renewed clusters

图 6 更新簇的流程

2.5.4 安全维护

簇的安全维护主要是为了公布发现的恶意节点以减少其对网络的危害.安全维护的流程如图 7 所示.安全维护动作包括 3 类消息,见表 6.

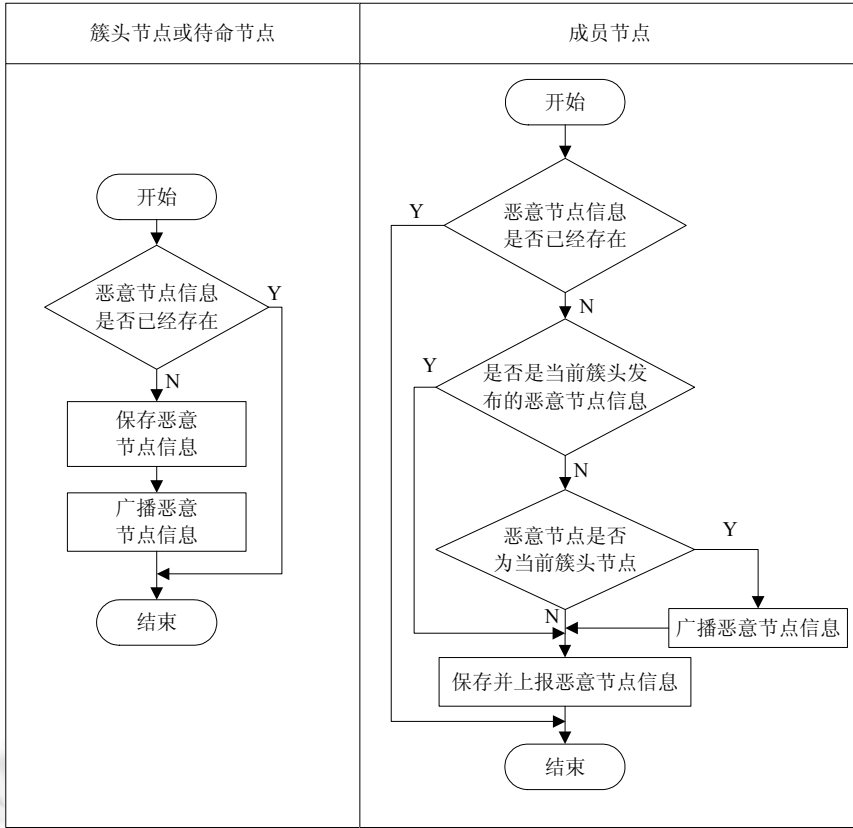


Fig.7 Process of secure maintenance of clusters

图 7 安全维护的流程

Table 6 Type of messages of secure maintenance of clusters

表 6 安全维护消息类型

消息类型	发送半径	接收方	描述
SECURITY_ADV_CH	额定广播距离	所有节点	广播消息,广播包含恶意节点 ID 的消息
SECURITY_ADV_MN	到簇头的通信距离加额定广播距离	所有节点	广播消息,广播包含恶意簇头 ID 的消息
SECURITY_UP_MN	到簇头的通信距离	簇头节点	发送包含恶意节点 ID 的消息给簇头

2.6 簇运行模式

传统周期性轮换算法在进行下一轮簇头选举前,网络拓扑结构是固定的,故可用静态 TDMA 调度方式;而 LDMC 的拓扑变化是非周期的,应采用动态 TDMA 调度机制.本文方法的 TDMA 帧分 4 部分:预留时隙、维护时隙、簇内通信时隙和簇间通信时隙(如图 8 所示).预留时隙用于扩展其他功能;维护时隙用于维护消息的收发,如果当前维护时隙无法完成维护动作,则在下一个维护时隙继续;簇内通信时隙用于成员节点向簇头传输数据,它被细分为多个节点时隙,一个节点时隙只允许一个成员节点使用,当簇内成员节点发生变动时,簇内通信时隙会重新调度;簇间通信时隙用于簇间的数据传输,根据簇头间的路由表将数据传输到基站.

LDMC 方法的时隙调度具有以下特点:

- (1) 所有簇的帧都等长且对齐.若不同簇的帧不等长或不对齐,则两个相邻簇的帧将会有先有后,当其中一个簇进行维护时,另一个簇的睡眠节点无法感知,这将影响簇维护的效果.为使相邻簇能参与到簇维护中以提高簇维护效果,就需要各个簇同时进入和结束维护时隙,故每个簇的帧对齐且帧长相同.

- (2) 节点时隙的分配由成员节点与簇头间的通信距离决定,通信距离越短,时隙就越靠前.成员节点在请求加入消息中附上收到簇头广播消息信号强度的信息,簇头根据信号强度排序,然后广播时隙消息.
- (3) 簇内通信时隙的时长是固定的,节点时隙时长可以改变.为保证在一个节点时隙内至少能完成一个数据包的传输,节点时隙有一个最小时长,最小时长由单个包的大小和带宽等确定.将簇内通信时隙按成员节点个数均分,均分后如果时长大于节点时隙的最小时长,节点时隙的时长就等于均分的时长;均分后如果时长小于节点时隙的最小时长,节点时隙的时长就等于节点时隙的最小时长,并从后向前复用时隙.复用时隙的方式是节点按帧轮流使用时隙,若有两个节点复用时隙,则每一个节点将每两帧使用一次时隙,单位时间内发送的数据包数量减半.由于节点使用的节点时隙越靠后,节点的通信距离则越远,所以通信距离远的节点先进行时隙复用.

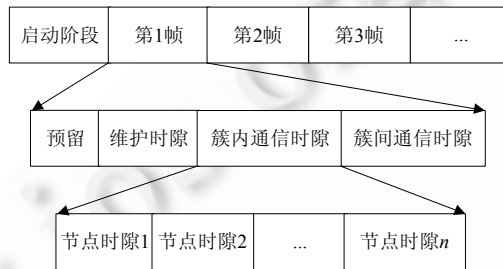


Fig.8 TDMA structure of LDMC

图 8 LDMC 方法的 TDMA 结构

3 局域按需簇维护模型的算法仿真实现

3.1 仿真平台搭建

本文在 NS2 软件平台下仿真局域按需簇维护方法,以模拟庞大且复杂的网络运行.

3.1.1 节点配置

为满足局域按需簇维护方法对传感器节点底层通信协议的需求,仿真所用节点参数配置见表 7.

Table 7 Setting of node's parameters

表 7 节点参数设置

名称	参数	值	描述
路由协议	-adhocRouting	dsdv	主动路由
链路层	-llType	LL	逻辑链路
MAC 层	-macType	802_11	802.11MAC 子层协议
队列类型	-ifqType	PriQueue	优先级队列
队列长度	-ifqLen	50	优先级队列长度为 50
天线类型	-antType	OmniAntenna	全向天线
传输模型	-propType	TwoRayGround	双径传播
物理层	-phyType	WirelessPhy	无线通信
信道类型	-channelType	WirelessChannel	无线信道

3.1.2 网络设置

网络设基站一个,位置可变;节点个数可调,并随机分布.节点的初始能量为 2J.成员节点将采集的数据打包发送给簇头节点,数据包大小为 500Bytes,包头大小为 25Bytes,节点在每个其对应的时隙向簇头发送数据包,网络带宽设为 1Mbps.网络数据输出周期为 0.1s.

3.1.3 通信过程设置

为了在 NS2 中仿真物联网感知层的传感器节点通信,需要在 NS2 中增加移动节点(mobile node)、MAC 协

议和信道传播模型 Channel.Applications 类的头文件用 Tcl 语言写成,节点的其他功能用 C++语言写成.通信过程包括数据包的发送和接收.

数据包的发送过程:Applications 创建数据包,然后发送给 Agent.Agent 执行协议栈运输层和网络层功能,将数据包发送给 CMUTrace.CMUTrace 将数据包的统计数据写入 trace 文件,然后将数据包发至 Connector.Connector 将数据包传送给链路层(LL);经过一小段时间的延迟后,数据包由 LL 发送给 Queue 缓冲队列.如果是还没有传送过的数据包,Queue 将以队列进行存储.然后,Queue 将数据包移出队列,发送到 MAC 层,开始运行 MAC 协议.最终,数据包被发送到网络接口层,网络接口层在数据包中加上传输能量信息,然后将其发送到 Channel. Channel 拷贝数据包,并发往连接信道的每一个节点.该通信过程如图 9 所示.

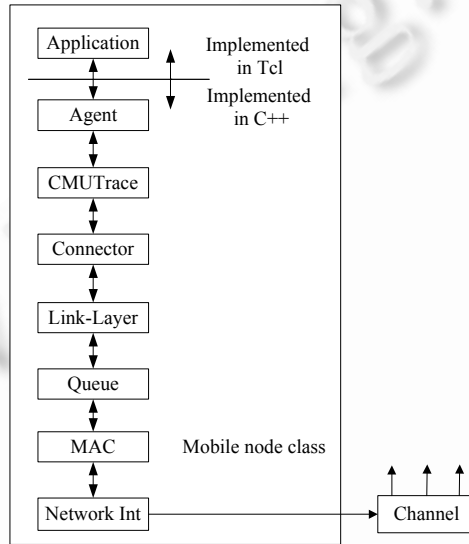


Fig.9 Process of node's communication

图 9 节点通信过程

数据包的接收过程与发送过程传输的路径相反,即,数据包被节点的网络接口接收后,依次向上传送至 MAC 层、链路层、Connector、CMUTrace 和 Agent.Agent 对收到的数据包进行判定,并通知 Application 数据包到达的信息.

3.2 系统功能的实现

3.2.1 系统结构

LDMC 方法在 Application 层实现对节点的控制,包括控制节点发送数据、处理接收的数据、节点睡眠与唤醒、节点检测需维护的事件等.这些控制功能都用 Tcl 语言编写,仿真代码按实现功能被写成多个 tcl 文件,通过一个仿真入口文件对它们进行调用,其结构如图 10 所示.

wireless.tcl 文件是仿真系统的入口,其主要功能是准备和开始仿真,包括设置节点分布拓扑、配置和创建节点、设置存储仿真数据的文件和启动仿真.uamps.tcl 文件主要是设置仿真参数,包括距离计算、数据统计、基站功能、信道配置和跟踪以及自由空间和多径衰落两种情形的能量传输模型;clusters.tcl 文件提供首次成簇策略,并在成簇之后启动 LDMC 维护策略.

LDMC 方法的实现集中于 ns-locm.tcl,event-listener.tcl,actions.tcl 这 3 个文件:ns-locm.tcl 文件实现对触发事件的优先级判断和对维护动作中收发函数的调用;event-listener.tcl 文件包含事件触发机制中所有的触发源,每个触发源按照自身需求以不同的频率循环查询触发条件,当条件满足时,将触发信息添加到事件表中;actions.tcl 文件包含 LDMC 方法中所有类型消息的收发函数.

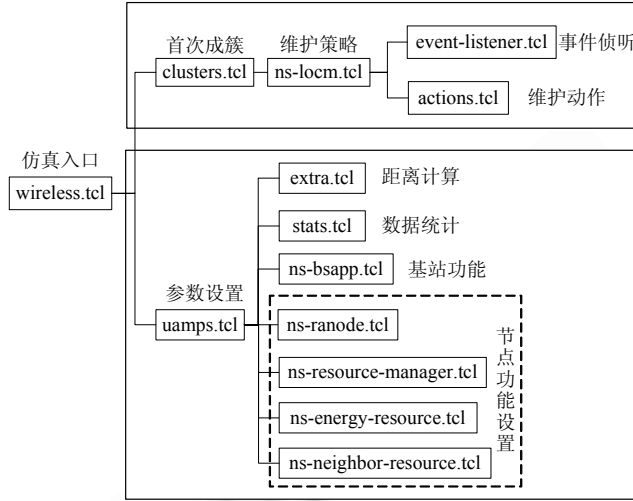


Fig.10 Simulation structure of LDMC

图 10 LDMC 方法仿真系统结构

3.2.2 事件侦听的实现

event-listener.tcl 文件实现事件侦听过程.该文件对 LDMC 方法的所有触发源进行定义和配置.触发源有触发条件、优先级、侦听周期和工作状态这 4 个主要属性.

触发条件在触发源的自检函数中进行设置和判断;每个触发源的优先级有两位:一位是触发源类别的优先级,另一位是自身的优先级;侦听周期是一个以 s 为单位的数值;工作状态标明触发源有效的节点工作状态.这里以 EN_CH_L 触发源为例,给出触发源的具体结构.

```

set Q_EN 2 #设置能量过低触发源优先级
set EN_CH_L 5 #设置 EN_CH_L
set Q_EN_CH_L(0) Q_EN #设置第 1 位优先级
set Q_EN_CH_L(1) 1 #设置第 2 位优先级
set T_EN_CH_L [expr 5*$frame_time_] #设置侦听周期
set S_EN_CH_L {1} #设置工作状态
Application/locm instproc checkEN_CH_L {} { #设置自检函数
    global ns_opt node_ #加载 global 变量
    $self instvar status_events_pe_Er_ #加载类内部变量
    $self instvar Q_EN_CH_L T_EN_CH_L S_EN_CH_L
    set ind [lsearch S_EN_CH_L $status_]
    if {$ind>0} { #判断是否处于有效的节点工作状态
        set E [[self getER] query] #获取当前能量
        set p [expr [expr $E-$pe_]/$T_EN_CH_L]
        set pe_$E
        set thresh [expr[expr $sopt(u)+$sopt(d)*$p]*$Er_] #阈值计算
        if {$E<$thresh} { #触发条件判断
            set events_[lappend events_ $EN_CH_L] #加入事件表
        }
    }
}
    
```

```
#下次侦听时间设置
$ns_at [expr [$ns now]+$ T_EN_CH_L] "$self checkEN_CH_L"
}
```

3.2.3 预处理的实现

ns-locm.tcl 文件实现预处理过程,该文件提供针对事件表的一系列函数,如获取事件表中最高优先级触发信息、删除事件表中的触发信息等.同时,设置触发源与响应动作的映射,实现根据触发信息检索维护动作的功能.预处理过程如图 11 所示;LDMC 方法中,触发源与响应动作的映射设置见表 8.

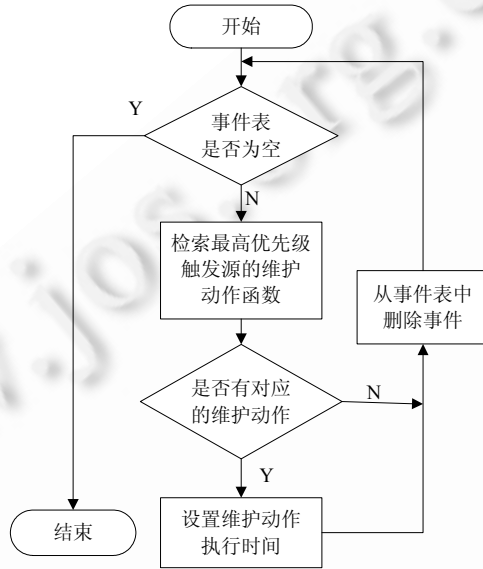


Fig. 11 Flow chart of pre-processing
图 11 预处理流程图

Table 8 Setting of sources of trigger and actions
表 8 触发源与动作映射设置

触发源名称	响应动作
TO_CH_NR	更新簇
TO_MN_NR	新建簇
TO_CH_NW	更新簇
TO_MN_NW	新建簇
EN_CH_L	新建簇、注销簇
SE_CH_MB	安全维护、更新簇
SE_MN_MB	安全维护、新建簇
ST_CH_NL	注销簇、更新簇
ST_MN_NN	更新簇

3.2.4 维护动作的实现

actions.tcl 文件实现维护动作.一个维护动作包括一系列收发函数、一个动作开始函数和一个动作进度函数.收发函数是实现动作的主体,一种类型的消息对应一个发送函数和一个接收函数.动作开始函数由预处理过程调用,同时为动作的执行提供准备,如延时时间的生成.动作进度函数在完成一个收发函数后执行,为动作的下一阶段做准备,并反馈当前的动作进度.

3.2.5 TDMA 调度机制的实现

LDMC 方法使用 TDMA 调度机制,其实现代码如下:

```

#单个包发送时间设置
set opt(slot_time) [expr[TxTime[expr $opt(sig_size)+$opt(hdr_size)]]]
#TxTime 函数计算以 1Mbps 的速度传输 n bit 数据所需要的时间
proc TXTime {numbytes} {
    global opt
    return [expr $numbytes*8/$opt(bw)]
}
#最小节点时隙计算
set opt(ss_slot_time) [expr $opt(slot_time)*$opt(spreading)]
#帧长的计算
set opt(frame_time) [expr $opt(ss_slot_time) * $opt(fl)]
//随机选取缓冲时间设置
l1stRndDelay_=Random::uniform(0,0.01);
//节点的数据发送时间设置
xmitTime_=config_.ssSlotTime_*(clusterNodes_.size(-))+l1stRndDelay_;
不同簇用不同的 CDMA 编码,相同簇采用同一个 CDMA 编码进行数据传输.实现 CDMA 编码分配的代码
如下:

```

```

#获取最大可用编码数
set numCodesAvail[expr 2*$opt(spreading)-1]
#设置 CDMA 编码
set ClusterCode[expr int(fmod($ind,$numCodesAvail))+1]

```

4 实验测试与性能分析

基于以上仿真系统的配置,下面主要从功能实现、能量消耗、数据传输、负载平衡和突发事件处理这 5 个方面对 LDMC 方法进行功能测试和性能评估。

4.1 功能实现测试

在没有突发事件时,LDMC 方法使用动态能量阈值驱动簇头轮换,其中涉及到两个参数: μ 和 ε 。不同的参数设置会影响网络的存活时间,如图 12 所示。

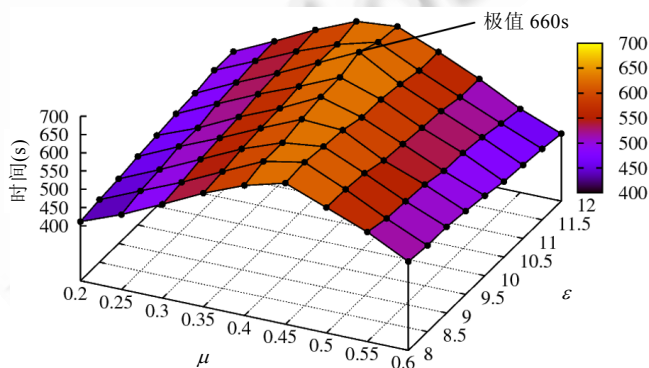


Fig.12 Effect of lifetime of network on μ and ε

图 12 μ 和 ε 对网络存活时间的影响

图 12 选用的数据是在多种网络结构下所得数据的均值,由图可见,当 $\mu=0.4$, $\varepsilon=11$ 时,网络存活时间较长,均

值最大.此时, $(\mu+\varepsilon \times p_{new})$ 的值在 0.58~0.66 之间波动,接近文献[15]中提出的最优能量阈值参数 $p_{opt}=0.644$.本文随后的仿真均采用这一组参数.

LDMC 方法的簇头轮换功能测试使用的网络参数是:

- 节点分布范围:200m×200m;
- 节点数量:400;
- 基站位置:(100,100);
- 簇内最少成员节点数/存活节点数:15/400;
- 首次成簇协议:LEACH.

在 NAM 下得到的簇头轮换过程如图 13 所示.图中方框标识节点为簇头节点,图 A 是首次成簇后的簇结构.图 B 是在 29s 时,1-1 号簇簇头剩余能量低于阈值,进行新建簇和注销簇的动作,此时,1-1 号簇转变为 1-2 号簇,实现了第 1 次簇头轮换.同时,由于该局部区域内节点更新了簇头列表,在 1-1 号簇与 2-1 号簇和 3-1 号簇边界附近的一些节点的 ST_MN_NN 触发源被触发,进行了更新簇的动作,使得 1-2 号簇与 2-1 号簇和 3-1 号簇的边界不同于 1-1 号簇与 2-1 号簇和 3-1 号簇的边界.图 C 是在 31s 时,2-1 号簇转变为 2-2 号簇,2-1 号簇与 4-1 号簇和 1-2 号簇的边界发生了变化,实现了第 2 次簇头轮换.图 D 是在 42s 时,3-1 号簇转变为 3-2 号簇,3-1 号簇与 5-1 号簇和 1-2 号簇的边界发生了变化,实现了第 3 次簇头轮换.从 NAM 展示的网络运行过程可以看出,LDMC 方法确保了簇维护在网络的局部范围进行,有效地解决了传统方法因全网维护导致的网络功能中断问题,并能避免未受影响区域因重新成簇导致的过度维护和不必要的能量浪费.

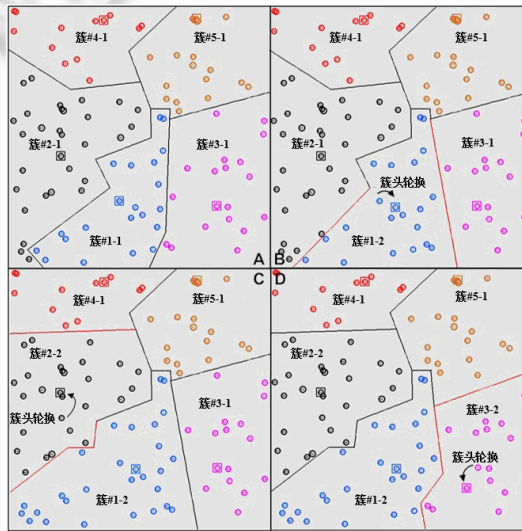


Fig.13 Local rotation of cluster head

图 13 局域范围内的簇头轮换

4.2 能量消耗测试

将 LDMC 方法与经典的 LEACH 算法进行对比,使用的网络环境参数是:

- 节点分布范围:100m×100m;
- 节点数量:100;
- 基站位置:(50,150);
- 簇内最少成员节点数/存活节点数:15/100;
- 首次成簇协议:LEACH.

分别从全网节点能量消耗和网络存活时间角度进行测试,得到的对比结果如图 14 和图 15 所示.

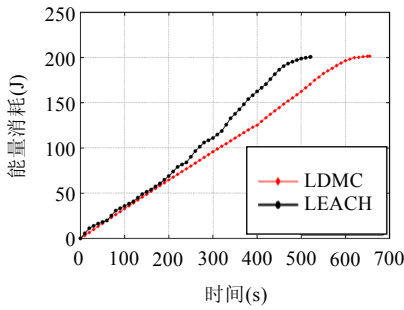


Fig.14 Comparison of expensed-energy in network

图 14 全网能量消耗对比

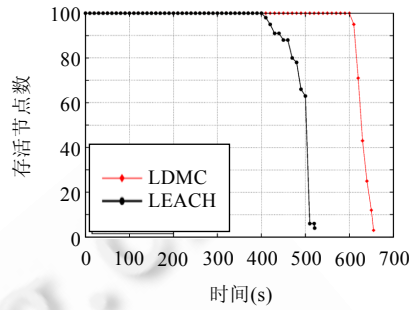


Fig.15 Comparison of lifetime of network

图 15 网络存活时间对比

由图 14 可知,LDMC 方法比 LEACH 算法能量消耗更稳定,且随着时间的增加,LDMC 方法的能量消耗速率并未像 LEACH 一样明显加快;在给定能量耗尽时,LDMC 方法所支持的网络存活时间持续更长.由图 15 可知,基于 LEACH 算法的网络在第 400s 时开始出现因能量耗尽的死亡节点,整个网络的存活时间为 525s;而基于 LDMC 方法的网络在第 600s 时才出现死亡节点,整个网络的存活时间达到 665s,延长了约 26.7%.可见,LDMC 方法明显降低了簇维护的能量消耗,能够有效延长网络的生命周期.

4.3 数据传输测试

数据传输测试时使用与能量消耗测试相同的网络参数,测试结果如图 16 所示.由图 16 可知,LDMC 方法中基站共收到数据包 65 928 个;而基于 LEACH 算法的基站收到的数据包总数为 52 374 个,增加了约 25.9%.同时还能看出,LDMC 方法的数据传输更快且更稳定,这表明基于 LDMC 方法的网络的数据收集能力更强.

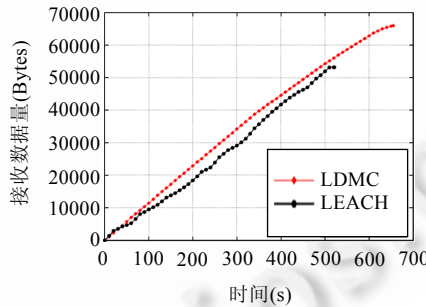


Fig.16 Number of received data packets of basestation

图 16 基站收到的数据包总数

4.4 负载均衡测试

LDMC 方法通过控制簇内成员节点数实现对网络簇个数的控制,以此调节负载均衡.控制节点数涉及 p 和 λ 两个参数.使用能量消耗测试相同的网络参数,得到不同参数设置下网络中的簇个数,见表 9.

Table 9 Number of clusters in network under different p and λ

表 9 不同 p 和 λ 下网络的簇个数

λ	p				
	0.05	0.1	0.15	0.2	0.25
1.1	9.56[4,16]	7.42[4,11]	5.80[2,9]	4.11[2,8]	2.92[1,6]
1.3	9.39[4,16]	7.12[4,10]	5.31[2,8]	3.60[2,6]	2.62[1,6]
1.5	9.20[4,16]	6.77[3,10]	4.88[2,7]	3.38[1,6]	2.41[1,6]
1.7	8.98[4,15]	6.40[3,9]	4.53[2,6]	3.13[1,6]	2.28[1,5]
1.9	8.81[4,14]	6.14[3,9]	4.26[2,6]	2.99[1,6]	2.19[1,5]

表中数据格式:簇个数均值[最小簇个数,最大簇个数].由表 9 可以看出,当 p 和 λ 都较小时,成员节点较少的簇能正常运行,成员节点较多的簇在进行簇维护时将被分成多个簇,使得网络中存在的簇个数较多且波动较大;当 p 和 λ 都较大时,成员节点较少的簇会被注销,其节点加入到其他簇,成员节点较多的簇在进行簇维护时不被分裂成多个簇,使得网络中的簇个数较少且波动较小.

网络的负载状况常用负载平衡因子(load balance factor,简称 LBF)来描述,由于簇头的负载主要取决于其簇内的成员节点个数,因此可将 LBF 定义为簇内成员节点数方差的倒数,即

$$LBF = n_c / \sum_{i=1}^{n_c} (x_i - \bar{x})^2 \tag{5}$$

其中, n_c 是网络中簇的个数, x_i 是第 i 个簇中成员节点的个数, \bar{x} 是各个簇中成员节点的平均数.LBF 值越大,则网络负载越平衡.

使用能量消耗测试相同的网络参数,同时设置 LEACH 协议的簇头个数为 5.在 LDMC 方法中 p 和 λ 分别为 0.15 和 1.5,由表 9 可知,其平均簇头个数接近于 5,与 LEACH 协议的簇头个数相当.测试结果如图 17 所示.由图 17 可以看出,在簇个数相同的条件下,LDMC 方法的负载平衡因子波动范围更小、均值更大,说明网络的负载更均衡.

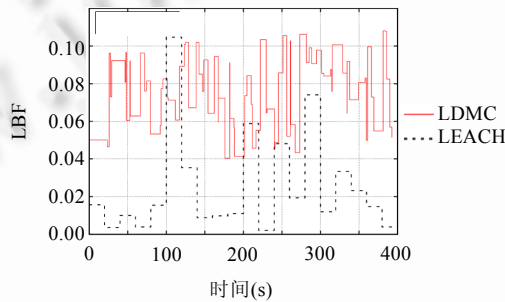


Fig.17 Comparison of LBF
图 17 负载平衡因子对比

4.5 突发事件处理测试

LDMC 方法使用事件驱动机制,能够就一些影响较大、优先级较高的突发事件进行及时响应并展开相应的簇维护,如簇头失效、新节点加入或恶意节点攻击等.测试结果分别如图 18~图 20 所示.

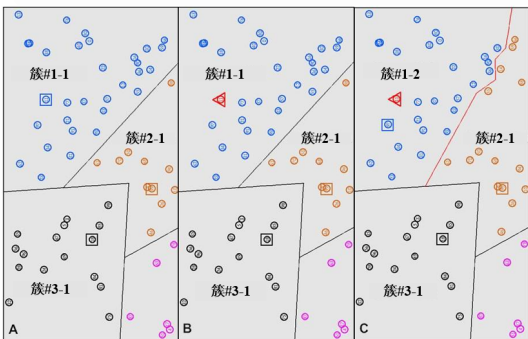


Fig.18 Response of LDMC on invalid cluster head
图 18 LDMC 方法对簇头失效的响应

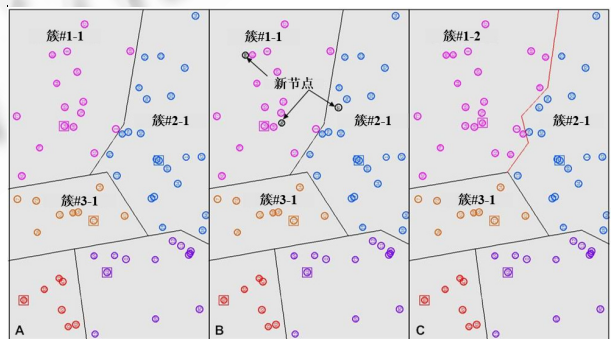


Fig.19 Response of LDMC on new node
图 19 LDMC 方法对新节点加入的响应

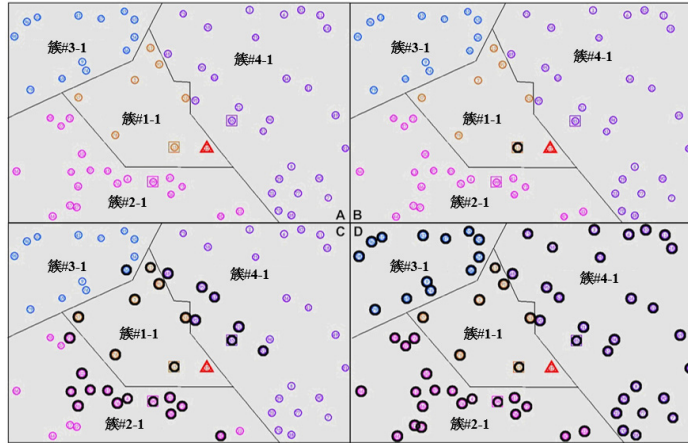


Fig.20 Response of LDMC on malicious nodes

图 20 LDMC 方法对恶意节点的响应

图 18 中,方框标识节点为簇头节点,三角形标识节点为失效簇头节点.图 A 是簇头未失效前的簇结构.图 B 是在第 20s 时的簇结构,此时,1-1 号簇的簇头由于突然死亡引起失效.图 C 是在第 21.1s 时,由于 1-1 号簇内的成员节点发现簇头失效后进行新建簇动作产生了新的簇头,1-1 号簇转变为 1-2 号簇,完成对簇头节点失效的维护,同时,在 1-1 号簇与 2-1 号簇边界附近部分节点的 ST_MN_NN 触发源被触发,进行了更新簇的动作,使得 1-2 号簇与 2-1 号簇的边界不同于 1-1 号簇与 2-1 号簇的边界;另外,在 1-1 号簇与 3-1 号簇边界附近的节点更新了簇头列表,但未满足 ST_MN_NN 触发源的触发条件,未进行维护动作,所以 1-2 号簇与 3-1 号簇的边界和 1-1 号簇与 3-1 号簇的边界相同.

图 19 中,图 A 是新节点未加入前的簇结构.图 B 是在第 20s 时,有 3 个新节点出现时的簇结构.图 C 是在第 20.8s 时,由于新加入节点进行新建簇的动作产生了一个新簇头,创建了 1-2 号簇,1-1 号簇内成员节点由于 ST_MN_NN 触发源被触发进行了更新簇动作,加入了 1-2 号簇,这导致 1-1 号簇中成员节点数变少,并使 1-1 号簇中簇头的 ST_CH_NL 触发源被触发,进行了注销簇和更新簇的动作,使得 1-1 号簇中剩余的所有节点也加入 1-2 号簇,从而实现对新节点加入的维护;同时,在 1-1 号簇与 2-1 号簇边界附近的部分节点的 ST_MN_NN 触发源被触发,进行了更新簇的动作,使得 1-2 号簇与 2-1 号簇的边界不同于 1-1 号簇与 2-1 号簇的边界,在 1-1 号簇与 3-1 号簇边界附近的节点更新了簇头列表,但未满足 ST_MN_NN 触发源的触发条件,未进行维护动作,所以 1-2 号簇与 3-1 号簇的边界和 1-1 号簇与 3-1 号簇的边界相同.

图 20 中,三角形标识节点为恶意节点,方框标识节点为簇头节点,黑圈代表已获取恶意节点 ID 的节点.图 A 是在第 20s 时出现恶意节点的情形.图 B 是在第 20.2s 时,1-1 号簇簇头发现恶意节点的情形.图 C 是在第 20.3s 时,1-1 号簇簇头广播恶意节点 ID,在其广播范围内的节点获取了恶意节点 ID.图 D 是在第 20.7s 时,由于已获取恶意节点 ID 的成员节点将恶意节点 ID 上报给簇头和已获取恶意节点 ID 的簇头节点转发包含恶意节点 ID 的广播,实现了全网对恶意节点的识别和剔除.

5 结 论

传统的全网周期性重新成簇的簇维护模式存在多种弊端,本文建立了基于事件驱动的物联网感知层局域按需簇维护(LDMC)模型与算法,使簇维护操作被局限于需要进行簇维护的时间和空间范围内,不仅能够明显地降低网络维护开销、解决周期性簇轮换的周期设置困难问题,而且能够对节点失效和通信受阻等网络故障情形快速响应,提高网络的稳定性、服务的连续性和网络生命周期的持久性.

基于 NS2 仿真平台对本文提出的 LDMC 方法和经典的 LEACH 算法进行全面的仿真测试对比,在仿真条

件下,基于 LDMC 方法的网络生存时间增加约 26.7%,基站收到的数据包总量增加约 25.9%。结果表明,LDMC 方法能够明显减少网络维护的能量消耗,增加数据传输量,负载更均衡,并能对节点失效、新节点加入和恶意节点攻击等突发事件实现及时维护。

References:

- [1] Ahmed A, Mohamed Y. A survey on clustering algorithms for wireless sensor networks. *Computer Communications*, 2007,30(14-15):2826–2841. [doi: 10.1016/j.comcom.2007.05.024]
- [2] Hu XD, Yu PQ, Wei QF. Securing sensor networks based on optimization of weighted confidence. *China Communications*, 2012, 9(8):122–128.
- [3] Heinzelman W, Chandrakasan A, Balakrishnan H. Energy-Efficient communication protocol for wireless microsensor networks. In: *Proc. of the 33rd Hawaii Int'l Conf. on System Sciences*. Hawaii: IEEE Computer Society, 2000. 3005–3014. [doi: 10.1109/HICSS.2000.926982]
- [4] Xu J, Hong YF, Wang C, Bai XZ. Upper bounds on lifetime of ordinary clustering ultra wide band sensor networks. *Ruan Jian Xue Bao/Journal of Software*, 2011,22(2):313–322 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/3748.htm> [doi: 10.3724/SP.J.1001.2011.03748]
- [5] Manjeshwar A, Agrawal DP. TEEN: A routing protocol for enhanced efficiency in wireless sensor networks. In: van de Geijn RA, ed. *Proc. of the 15th Parallel and Distributed Processing Symp.* San Francisco: IEEE Computer Society, 2001. 2009–2015. [doi: 10.1109/IPDPS.2001.925197]
- [6] Younis O, Fahmy S. HEED: A hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. *IEEE Trans. on Mobile Computing*, 2004,3(4):366–379. [doi: 10.1109/TMC.2004.41]
- [7] Heinzelman WB, Chandrakasan AP, Balakrishnan H. An application-specific protocol architecture for wireless microsensor networks. *IEEE Trans. on Wireless Communications*, 2002,1(4):660–670. [doi: 10.1109/TWC.2002.804190]
- [8] Chen QZ, Zhao XM, Chen XY. Design of double rounds clustering protocol for improving energy efficient in wireless sensor networks. *Ruan Jian Xue Bao/Journal of Software*, 2010,21(11):2933–2943 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/3668.htm> [doi: 10.3724/SP.J.1001.2010.03668]
- [9] Shen Y, Qi WD, Dai H. An energy efficient clustering protocol for surveillance sensor networks. *Ruan Jian Xue Bao/Journal of Software*, 2008,19(9):2432–2441 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/19/2432.htm> [doi: 10.3724/SP.J.1001.2008.02432]
- [10] Chan HW, Perrig A. ACE: An emergent algorithm for highly uniform cluster formation. In: Karl H, Willig A, Wolisz A, eds. *Proc. of the 1st European Workshop on Wireless Sensor Networks (EWSN 2004)*. Berlin, Heidelberg: Springer-Verlag, 2004. 154–171. [doi: 10.1007/978-3-540-24606-0_11]
- [11] Hou TC, Tsai TJ. An access-based clustering protocol for multihop wireless ad hoc networks. *IEEE Journal on Selected Areas in Communications*, 2001,19(7):1201–1210. [doi: 10.1109/49.932689]
- [12] Wang YC, Zhao QC, Zheng DZ. Energy-Driven adaptive clustering data collection protocol in wireless sensor networks. In: Huang DG, ed. *Proc. of the 2014 Int'l Conf. on Intelligent Mechatronics and Automation*. Chengdu: IEEE Press, 2004. 599–604. [doi: 10.1109/ICIMA.2004.1384266]
- [13] Gamwarige S, Kulasekera C. An algorithm for energy driven cluster head rotation in a distributed wireless sensor network. In: Meng MQH, ed. *Proc. of the Int'l Conf. on Information and Automation (ICIA 2005)*. IEEE Press, 2005. 354–359.
- [14] Yuan JY, Shi WR. WSN self-maintenance cluster algorithm based on connectivity. *Computer Engineering and Applications*, 2007, 43(19):138–140 (in Chinese with English abstract). [doi: 10.3321/j.issn:1002-8331.2007.19.040]
- [15] Gamwarige S, Kulasekera C. Optimization of cluster head rotation in energy constrained wireless sensor networks. In: Omidyar G, ed. *Proc. of the Int'l Conf. on Wireless and Optical Communications Networks*. Singapore: IEEE Press, 2007. 1–5. [doi: 10.1109/WOCN.2007.4284155]

附中文参考文献:

- [4] 徐娟,洪永发,王成,白星振.常规分簇的超宽带传感网生存期的上界. *软件学报*,2011,22(2):313–322. <http://www.jos.org.cn/1000-9825/3748.htm> [doi: 10.3724/SP.J.1001.2011.03748]

- [8] 陈庆章,赵小敏,陈晓莹.提高无线传感器网络能效的双轮成簇协议设计.软件学报,2010,21(11):2933-2943. <http://www.jos.org.cn/1000-9825/3668.htm> [doi: 10.3724/SP.J.1001.2010.03668]
- [9] 沈洋,齐望东,戴浩.一种能量有效的传感器监测网络成簇协议.软件学报,2008,19(9):2432-2441. <http://www.jos.org.cn/1000-9825/19/2432.htm> [doi: 10.3724/SP.J.1001.2008.02432]
- [14] 袁久银,石为人.基于连通度的无线传感器网络自维护分簇算法.计算机工程与应用,2007,43(19):138-140. [doi: 10.3321/j.issn:1002-8331.2007.19.040]



胡向东(1971—),男,四川广安人,博士,教授,CCF 高级会员,主要研究领域为网络化测控及其信息安全,物联网与智慧空间,复杂系统建模,仿真与优化.



张力(1988—),男,硕士生,主要研究领域为传感网,物联网.



徐慧芬(1990—),女,硕士生,主要研究领域为物联网理论与技术.