

# 一种基于地理位置的传感器网络拓扑控制方案<sup>①</sup>

陆明洲<sup>1,2</sup> 沈明霞<sup>2</sup> 何菊<sup>3</sup>

(1.南京航空航天大学 信息科学与技术学院 江苏 南京 210016; 2.南京农业大学 工学院 江苏 南京 210032; 3.南京中医药大学 信息技术学院 江苏 南京 210029)

**摘要:** 网络节点掌握自身位置信息前提下如何设计成簇算法是传感器网络拓扑控制的一个重要研究方向。以提高网络的能耗有效性为目的进行深入研究,提出了一种新的基于精确地理位置的拓扑控制方案。仿真结果表明,该拓扑控制方案能耗低,适用于大规模网络,能够有效延长传感器网络的生存期。

**关键词:** 传感器网络; 拓扑控制; 分簇; 精确地理位置

## A Location - Based Topology Control Scheme in WSNs

LU Ming-Zhou<sup>1,2</sup>, SHENG Ming-Xia<sup>2</sup>, HE Ju<sup>3</sup>

(1.Information Science and Technology College, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China; 2.Engineering College, Nanjing Agriculture University, Nanjing 210032, China; 3.Information Technology Academy, Nanjing University of Traditional Chinese Medicine, Nanjing 210029, China)

**Abstract:** One of the important research areas in WSNs topology control is how to design the clustering algorithms on condition that nodes know their precision position. This paper presents an in-depth study on topology control in order to improve energy efficiency. It proposes a novel topology control scheme, and simulates the scheme in MATLAB. The results show that the energy consumption of the scheme is low, and the proposed scheme is fitful for the large scale networks and can effectively stretch the lifetime of WSNs.

**Keywords:** wireless sensor network; topology control; clustering; specific location

拓扑控制是传感器网络中许多其它研究课题的基础<sup>[1]</sup>。文献[2]证明了层次型拓扑控制是降低网络能耗更为有效的方法。LEACH<sup>[2]</sup>是传感器网络中最早被提出的层次拓扑控制算法,该算法选举簇头时并未考虑节点剩余能量且簇头分布不均匀。即便如此,LEACH的成簇思想对后来的层次型协议如 HEED<sup>[3]</sup>、GAF<sup>[4]</sup>等具有很大的影响。本文借鉴 GAF 按虚拟单元格划分簇的思想提出一种新的基于精确地理位置的层次型拓扑控制算法——LHTC(Location based Hierarchical Topology Control),该算法综合考虑节点位置及剩

余能量选举簇头,形成网络拓扑的能耗低,在该拓扑基础上的路由方案简单可靠。

本文第1节介绍网络模型,第2节详细描述 LHTC 算法,第3节仿真算法性能,最后给出本文小结。

## 1 网络模型

网络节点以基站为中心均匀部署,部署密度记为 $\lambda$ 。节点能够根据通信半径控制自己的发射功率,节点不具备全局唯一标识符。

将网络以基站为圆心划分为若干同心圆环且环内径长 $r$ ,环号从0计起且由内层到外层环号递增(如图

<sup>①</sup> 基金项目:国家高技术研究发展计划(863)(2008AA10Z226);江苏省农机局科研启动基金(gxz08007)

收稿时间:2009-10-13;收到修改稿时间:2010-03-13

1)。在各环内划分区域, 环  $i$  的区域  $j$  及区域  $j+1(i \geq 1, j=2 \times k, k=0, 1, 2 \dots)$  对应的扇区中心角都等于相邻内层环  $i-1$  的区域  $j \div 2$  ( $\div$  表示整除) 对应扇区中心角的一半。同一区域的各节点在网络运行期间作为一个簇。

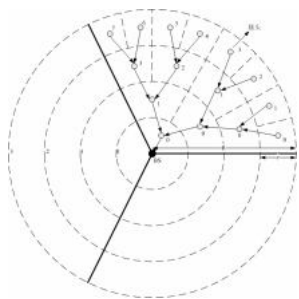


图 1 协议网络模型

网络模型假设:

- 1) 节点部署后位置不发生移动;
- 2) 网络中部分节点装载 GPS, 未装载 GPS 的节点根据定位算法<sup>[5-7]</sup>并参照装载 GPS 节点的位置确定自己在网络中的坐标;
- 3) 同区域节点间保持局部时间同步;
- 4) 网络部署在二维平面地区。

## 2 LHTC详细描述

### 2.1 节点属性确定

网络部署后, 任意节点  $N_i$  都可利用式(1)计算其与基站的距离:

$$d(N_i, BS) = \sqrt{(N_i.x - BS.x)^2 + (N_i.y - BS.y)^2} \quad (1)$$

其中  $(N_i.x, N_i.y)$  及  $(BS.x, BS.y)$  分别是该节点和基站的坐标。节点  $N_i$  的环号  $H_i$  可用式(2)表示如下:

$$H_i = \text{floor}(d(N_i, BS)) / r \quad (2)$$

$\text{floor}(x)$  为求不大于  $x$  的最大整数的函数。

为了计算节点  $N_i$  的区域号  $C_i$ ,  $N_i$  首先计算其所处环拥有的区域数  $K_i = 3 \times 2H_i$ , 则  $N_i$  所在区域对应的扇区中心角  $\theta = 2\pi / K_i$ 。  $N_i$  以基站为极点的极角  $\psi_i = \text{atan2}((N_i.y - BS.y), (N_i.x - BS.x))$ ,  $\text{atan2}(y, x)$  为可以返回四象限反正切值得函数, 则节点  $N_i$  的区域号  $C_i$  可用式(3)表示:

$$C_i = \text{floor}(\psi_i / \theta) \quad (3)$$

为了计算节点  $N_i$  与其所在区域中心点的距离, 首先计算以基站为极点的情况下中心点的极径  $\rho$

$= (H_i + 1/2) \cdot r$  以及极角  $\phi = (C_i + 1/2) \cdot \theta$ 。则中心点直角坐标为:

$$N_{ic}.x = BS.x + \rho \cdot \cos(\phi);$$

$$N_{ic}.y = BS.y + \rho \cdot \sin(\phi).$$

则  $N_i$  与中心点的距离可表示为:

$$d_{N_i, N_{ic}} = \sqrt{(N_i.x - N_{ic}.x)^2 + (N_i.y - N_{ic}.y)^2} \quad (4)$$

则  $N_i$  可以计算其当选簇头的权值  $w_i$ :

$$w_i = w \cdot \frac{E_{current}}{E_{ini}} + (1-w) \cdot \frac{1}{d_{N_i, N_{ic}} + 1} \quad (5)$$

其中  $E_{current}$  表示节点当前剩余能量,  $E_{ini}$  表示节点初始能量;  $w$  为可调节参数 ( $0 \leq w \leq 1$ ),  $w$  越大, 考虑能量的因素越大;  $w$  越小考虑节点位置因素越大。

### 2.2 簇内 ID 分配

基站在全网范围内广播基站坐标、 $R$  及  $r$ 。网络节点收到收到这些信息后运用式(2)、(3)、(4)分别计算出自己所处的环号、区域号及其与所处环区域中心点的距离。在各个环区域中选出一个标识符分配节点 (Identifier Distribute Node, IDN) 为网络节点分配簇内 ID。

为竞争 IDN 身份, 任意节点  $N_i$  设定一个与  $d_{Ni}$ ,  $N_{ic}$  成正比的 HELLO 消息等待时间  $t_{HELLO}$ 。节点等待  $t_{HELLO}$  时间段后醒来并监听其他节点的 HELLO 消息, 若在一定时间内没有任何 HELLO 消息则自主决定成为 IDN 并向邻居节点广播 HELLO 消息(如图 2 所示)。其中,  $H_{IDN}$  和  $C_{IDN}$  为 IDN 所处的环号及区域号,  $IDN\ NID$  为各 IDN 的簇内标识, LHTC 规定各个 IDN 节点的簇内标识符为 1。Message Type 指示消息类型及功能。



图 2 HELLO 消息格式



图 3 JOIN\_REQUEST 消息格式

所有收到 HELLO 消息的节点比较该消息中携带的环号和区域号, 若这两个属性跟自己的环号、区域号相同则向该 IDN 发送加入请求消息 JOIN\_REQUEST, 加入请求消息格式如图 3 所示, 其中 Src DID 为发送该 JOIN\_REQUEST 消息的节点的 MAC 地

址的一部分。

IDN 收到 JOIN\_REQUEST 消息后为发送该加入请求消息的节点分配簇内标识 New NID 并通过 JOIN\_RESPONSE 消息(格式如图 4 所示)通知该请求加入的节点。其中 Dst DID 为加入请求消息中 Src DID 的拷贝, New NID 为 IDN 给该申请加入的节点分配的簇内 ID。

|                              |              |         |         |
|------------------------------|--------------|---------|---------|
| $H_{IDN}  C_{IDN}  IDN\ NID$ | Message Type | Dst DID | New NID |
|------------------------------|--------------|---------|---------|

图 4 JOIN\_RESPONSE 消息格式

节点收到 JOIN\_RESPONSE 消息后首先根据 Dst DID 判断该消息是否是发送给自己的,若是,则向 IDN 发送 New\_NID\_RESPONSE 消息,消息格式如图 5 所示。其中,Src NID 为该节点收到的 JOIN\_RESPONSE 消息中携带的 IDN 分配给它的簇内 ID, Additional info 为该节点的附加信息,这些信息包括该节点的当前剩余能量及其与区域中心的距离。IDN 每收到一条 New\_NID\_RESPONSE 消息都将消息中的 Src NID 及该节点的 Additional info 携带的信息记录到一张同区域节点列表(the Same Area Node List, SANL)中。

|                              |              |         |                 |
|------------------------------|--------------|---------|-----------------|
| $H_{IDN}  C_{IDN}  IDN\ NID$ | Message Type | Src NID | Additional info |
|------------------------------|--------------|---------|-----------------|

图 5 New\_NID\_RESPONSE 消息格式

LHTC 要求各区域 IDN 使用 SANL\_BROADCAST 消息将生成的 SANL 广播给簇内节点。SANL\_BROADCAST 消息格式如图 6 所示:

|                              |              |          |          |       |          |
|------------------------------|--------------|----------|----------|-------|----------|
| $H_{IDN}  C_{IDN}  IDN\ NID$ | Message Type | NID info | NID info | ..... | NID info |
|------------------------------|--------------|----------|----------|-------|----------|

图 6 SANL\_BROADCAST 消息格式

其中 info 是该区域内各节点的附加信息。收到该消息的节点首先确定自己是否属于  $H_{IDN}||C_{IDN}$  指示的区域,若是,则根据消息携带的信息重构 SANL。

### 2.3 LHTC 簇头选举策略

LHTC 以“轮”方式维护簇结构,如图 7 所示。



图 7 LHTC 簇内轮结构示意图

#### 2.3.1 首轮操作描述

图 7 中除首轮之外的各轮称为后续轮,首轮操作包括首轮选簇头、首轮稳定阶段以及指定下任簇头,本部分介绍 LHTC 首轮选簇头的操作。

网络部署后所有节点根据式(5)计算自己的权值  $w_i$ ,记  $N_i$  的 SANL 中权值最大值为  $w_{max}$ ,与该权值对应的节点标识符记为  $NID_{wmax}$ (若有多个节点的权值相等且都等于  $w_{max}$  则选择标识符最小的 NID)。若  $w_i - w_{max} > 0$ ,则  $N_i$  自主决定成为簇头并在簇内公告 NID 及其为同区域其它节点分配的 TDMA 时隙;若  $w_i - w_{max} < 0$ ,则等待其它节点的簇头公告消息。若  $w_i - w_{max} = 0$ ,则参考节点标识符:若  $NID_{N_i} < NID_{wmax}$ , $N_i$  优先担任簇头,否则等待其它节点的簇公告消息。首轮选簇头是一种分布式操作,符合传感器网络自组织的特性。

#### 2.3.2 簇头轮换

簇头轮换包含前任簇头指定下任簇头以及新簇头在簇内分配新的工作时隙两种操作。前任簇头可以应用式(5)求得各个成员节点的权值  $w_i$ ,并选择权值最大的节点成为下任簇头,若存在多个具有相同  $w_i$  的节点,选择标识符最小的指定为下任簇头。

为保证前任簇头利用式(5)计算 SANL 中各成员节点的  $w_i$  时使用的是成员节点的当前剩余能量,在簇稳定阶段引入“剩余能量捎带”机制,该机制要求成员节点在发送给簇头的数据消息中捎带其当前剩余能量。簇头节点依据各成员节点捎带的能量更新 SANL 中的对应的节点剩余能量。

#### 2.4.3 簇稳定阶段

簇稳定阶段有簇内数据通信及簇间通信两种操作。簇内通信主要是成员节点在其工作时隙内将监测数据发送给簇头。簇间通信主要是簇头聚合簇内数据并发送给距离基站更近的簇头。假定环号和区域号分别为  $H_i$ 、 $C_i$  的簇头  $CH_i$  需要发送聚合数据给基站, $CH_i$  首先判断  $H_i$  是否为 0,若为 0 则直接发送给基站,否则下一跳簇头所在环号  $H_j = H_i - 1$ ,区域号  $C_j = \text{floor}(C_i/2)$ 。收到该消息的簇头节点比较自己的环号区域号与消息中的  $H_j$ 、 $C_j$ ,若相等则将该消息转发给内层簇头,否则丢弃。如此循环消息将被正确路由到基站。

## 3 算法仿真分析

仿真程序设定节点初始能量为 0.5J。约定簇成员

节点发送的数据包长为 500B, 包头长 25B。文献[8]仿真 LEACH 协议时约定节点密度为 0.01, 为比较 LHTC 与 LEACH 性能, 设定  $\lambda = 0.01$ 。

### 3.1 环内径取值与协议能耗关系仿真分析

取  $R=150\text{m}$ ,  $w=0.6$ ,  $\lambda = 0.01$ ,  $r$  分别取 30m、40m 以及 50m, 让 LHTC 仿真程序分别运行 600 到 1000 轮, 各种情况下死亡节点数占网络节点总数的比率如图 8 所示。

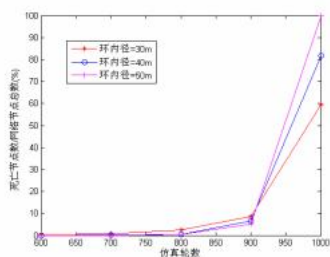


图 8 环内径取值与死亡节点比率对照

当网络部署半径一定时, 环内径越小, 簇头数越多, 而簇头能耗大于普通节点, 因此在网络死亡节点比较少少的情况下, 环内径越小, 网络能耗越大。随着网内节点平均能量的降低, 环内径越大, 簇头间的通信距离以及簇内节点与簇头的平均距离就越大, 网络死亡节点出现速率会明显加快, 这与仿真结果一致。

### 3.2 LHTC 与 LEACH 性能仿真对比

取 LHTC 协议参数  $w=0.7$ ,  $r=30\text{m}$ ,  $R=150\text{m}$ , 将 LEACH 和 LHTC 分别运行 600 到 1000 轮, 比较两种协议死亡节点率, 结果如图 9 所示。

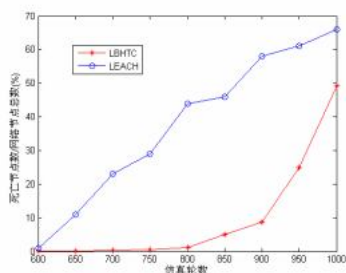
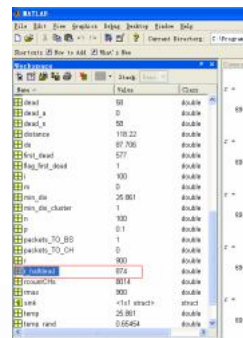


图 9 LEACH 与 LHTC 协议死亡节点情况对照

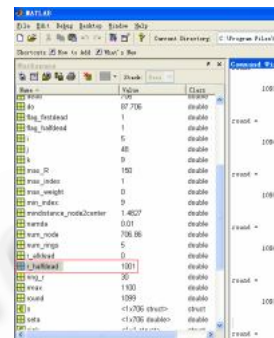
从图 9 可以看出, LHTC 运行 900 轮后仍保持很低的节点死亡率, 而 LEACH 算法从 650 轮开始就有超过 10% 的节点死亡。若让 LEACH 工作区域面积与 LHTC 工作区域面积相等, LHTC 相对 LEACH 的性

能优越性将更明显。因此, LHTC 是一种能够适应大规模部署的拓扑控制算法。

在仿真程序中设定参数求 LEACH 和 LHTC 协议节点死亡一半的准确轮数, 结果如图 10 所示。



(a) LEACH



(b) LHTC

图 10 LEACH 与 LHTC 一半节点死亡轮数

仿真结果表明, LEACH、LHTC 协议 50% 节点死亡的轮数分别为 874 轮和 1001 轮, 由此可见, LHTC 能耗较 LEACH 协议有所降低, 能够有效延长网络生存期。

## 4 结语

本文提出了一种基于地理位置的层次型拓扑控制算法, 算法将位于同一环区域的节点划归一个簇。在每个簇内综合考虑节点地理位置及节点剩余能量两个因素选举簇头。簇头聚合簇内数据并多跳发送给基站, 有效降低了簇头能耗。算法仅需局部时间同步, 该算法生成的网络拓扑基础上的路由方案简单可靠。仿真结果表明该算法能适应大规模部署的网络, 能有效延长网络生存期。

## 参考文献

- 1 张学, 陆桑璐, 陈贵海, 等. 无线传感器网络的拓扑控制. 软件学报, 2007, 18(4): 943-954.
- 2 Heinzelman WR, Chandrakasan AP, Balakrishnan H. Energy-Efficient communication protocol for wireless microsensor networks. Proc. of the Hawaiian Int'l Conf. on System Science (HICSS). Washington. 2000. 3005-3014.
- 3 Younis O, Fahmy S. HEED: A hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. IEEE Trans. on Mobile Computing, 2004,

(下转第 134 页)

- 3(4):660 – 669.
- 4 Xu Y, Heidemann J, Estrin D. Geography-Informed energy conservation for ad hoc routing. Rose C, ed. Proc. of the ACM Int'l Conf on Mobile Computing and Networking (MobiCom). New York: ACM Press, 2001. 70 – 84.
  - 5 Girod L, Estrin D. Robust range estimation using acoustic and multimodal sensing. Proc IEEE RSJ Int'l Conf Intelligent Robots and Systems (IROS'01), Vol 3 Maui Hawaii USA, 2001. 1312 – 1320.
  - 6 Savvides A, Han CC, Srivastava MB. Dynamic finger-grained localization in ad-hoc networks of sensors. In: Proc 7th Annual Int'l Conf on Mobile Computing and Networking (MobiCom). Rome, Italy. July 2001. 166 – 179.
  - 7 Niculescu D, Nath B. Ad hoc positioning system (APS) using AOA. Proc 22nd Annual Joint Conf of the IEEE Computer and Communications Societies (INFOCOM' 2003 ). IEEE, Vol. 3, 2003.
  - 8 Capkun S, Hubaux JP, Levente B. Mobility Helps Security in Ad Hoc Networks. Proc. of MobiHOC, 2003. 46 – 56.