

无线传感器网络的能量均衡路由算法

张海洋, 曾凡仔, 罗娟, 李仁发

(湖南大学计算机与通信学院, 长沙 410012)

摘要: 针对大部分无线传感器网络路由协议只能实现局部能量均衡的问题, 提出一种改进的能量均衡路由算法, 把传感器网络构建成分层网络, 利用节点直接传输和逐跳转发相结合的混合传输策略, 在多跳传输时, 使用改进的基于多路径路由的最大能量路径算法。仿真结果表明, 该算法能有效延长网络的生存时间。

关键词: 无线传感器网络; 能量均衡; 网络寿命; 多路径路由

Energy Balance Routing Algorithm for Wireless Sensor Networks

ZHANG Hai-yang, ZENG Fan-zi, LUO Juan, LI Ren-fa

(School of Computer and Communication, Hunan University, Changsha 410012)

【Abstract】 Most current protocols just achieve the local energy balance of Wireless Sensor Networks(WSNs). To solve this problem, a modified energy balance routing algorithm is proposed. It constructs a layered network by minimum hop count, and uses a mixed transmission mode resulting from hop-by-hop forwarding combined with direct forwarding. In hop-by-hop transmission, it uses the modified multi-path routing scheme based on maximum energy path. Simulation results show the lifetime of the networks is prolonged efficiently.

【Key words】 Wireless Sensor Networks(WSNs); energy balance; longevity of networks; multi-path routing

1 概述

在无线传感器网络中, 节点能量有限且一般没有能量补充。保持每个节点的能耗均衡对避免网络失效具有重要意义。多路径路由在路由过程中可以选择不同的路径, 能够平衡节点能量的消耗, 成为无线传感器网络路由研究的热点。

文献[1]提出高弹性能量有效多路径路由(HREEMR), 构造并维护少数几条备份路径, 主路径失效后, 可以从备份路径中选择一条替代, 但该协议在发现和维护多条路径方面开销很大。文献[2]提出负载均衡多路径路由(LBMPR), 依赖传感器节点的地理位置生成从源节点到汇聚节点(sink)的多条路径, 数据通信均匀分布于不同路径, 以达到节点间的负载均衡, 该算法的性能依赖于2个参数的选取, 但较难合适地选取。文献[3]提出能量感知路由(EAR), 通过一定的概率选择和维持多条路径来均衡网络能量的消耗, 但该机制没有考虑对距离(跳数)的优化, 且需周期性地实施洪泛查询来进行路由维护, 增加协议开销。文献[4]提出一个基于多路径路由的最大容量路由机制(MCP), 选择能量最大的路径进行路由, 能较好地实现局部区域的能量均衡问题, 但未解决 sink 附近区域能量消耗热点问题, 且该算法在路由维护时每次选择的不是最大能量路径。

本文提出一个基于文献[4]的改进算法, 改进了 MCP 的路由维护算法, 使其每次都能选取最大能量路径。

2 算法网络模型

本文研究的网络由一个 sink 和多个随机均匀分布的节点构成。当网络工作时, 节点既可以以一定的概率将数据转发给下一跳节点, 又可以直接把数据传送给 sink, 多跳传输时节点的通信半径相同。在网络工作前, 通过初始化把网络构造成分层网络。

基于最小跳数的分层网络实现方法如下: 令 h_v 为节点 v

的到 sink 的最小跳数。初始时, sink 设置为 $h_s = 0$, 其他节点 u 的 h_u 为无穷大。sink 向网络广播带 h_s 的路由创建消息。节点 u 从 v 接收到一个路由创建消息后, 取出 h_v 进行以下比较:

(1) 如果 $h_v > h_u - 1$, u 什么都不做。

(2) 如果 $h_v = h_u - 1$, u 把 v 加入到它的路由表中。

(3) 如果 $h_v < h_u - 1$, u 清空路由表中已有内容, 并将 v 加入路由表中。然后, u 设置 $h_u = h_v + 1$, 重新广播带 h_u 的路由创建消息。

3 层内能量均衡多跳路由机制

在上述分层网络中, 每个节点到 sink 可能存在多条最小跳数路径。本文的算法在多条最小跳数路径中选择能量最大的进行路由。路径能量指该路径中所有节点的剩余能量的最小值。

定义节点 w 到 sink 的路径为所有最小跳数路径中能量最大的路径, 设该路径能量为 $p(w)$, 则 $p(w) = \min(e(w), p(u^*))$, 其中, $e(w)$ 为节点 w 的剩余能量; $p(u^*)$ 为下一跳节点到 sink 所有最小跳数路径的能量的最大值。多路径分层网络如图1所示。其中, h 代表节点跳数; e 代表节点能量。节点 e 到 sink 有2条最小跳数路径:

$$e \rightarrow b \rightarrow a \rightarrow s, e \rightarrow d \rightarrow c \rightarrow s$$

其中, 路径 $b \rightarrow a \rightarrow s$ 的能量为节点 a 的剩余能量值, $p(b) = 35$; 路径 $d \rightarrow c \rightarrow s$ 的能量为节点 c 的剩余能量值,

基金项目: 国家自然科学基金资助项目(60673061); 湖南省自然科学基金资助项目(06JJ50113)

作者简介: 张海洋(1983-), 男, 硕士研究生, 主研方向: 无线传感器网络; 曾凡仔、罗娟, 副教授; 李仁发, 教授、博士生导师

收稿日期: 2009-05-20 **E-mail:** jiuzihei@163.com

$p(d)=10$ 。因此，

$$p(u^*) = \max(p(b), p(d)), \quad p(e) = \min(e(e), p(u^*))$$

节点 e 没有选择剩余能量最大的节点 d 为下一跳节点，而选择路径 $e \rightarrow b \rightarrow a \rightarrow s$ 来传输数据，避免了节点 c 过早死亡导致网络的不连通。

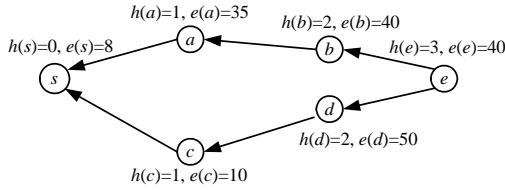


图1 多路径分层网络

3.1 层内能量均衡多跳路由的创建

每个节点 v 维护一个路由表，表中记录可能的下一跳路由由节点 (u_1, u_2, \dots, u_n) ，及其到 sink 路径的能量值 $(p(u_1), p(u_2), \dots, p(u_n))$ ，并记录 $p(u^*)$ ， $p(u^*) = \max(p(u_1), p(u_2), \dots, p(u_n))$ 。构造分层网络的同时进行路由的创建。sink 节点向网络广播带 $h_s=0, p(s)=\infty$ 的路由创建消息。节点 v 把 u 加入路由表时，记录 $p(u)$ 的值，同时与 $p(u^*)$ 进行比较，如果其值大于 $p(u^*)$ ，则令 $p(u^*) = p(u)$ 。

3.2 层内能量均衡多跳路由的维护

MCP 机制的路由维护通过数据包来更新节点的能量，不能保证每次选取的都是能量最大的路径。本文对此进行了改进。节点 u 发送数据时在数据包中加入它的更新的路径能量。利用无线通信的侦听机制，路由表中包含 u 的节点 v 更新 $p(u)$ ，同时更新 $p(u^*)$ ，并查看 $p(v)$ 是否有所变化，有则广播带 $p(v)$ 的消息。路由表中含有 v 且收到广播消息的节点做同样检测，依次类推。路由的维护示例如图 2 所示，其中， p 代表路径能量； e 代表节点的剩余能量。节点 b 发送数据给汇聚节点，消耗 8 个单位能量， b 在发送数据时把更新的路径能量 $p(b)=40-8$ 加入数据包中，节点 c 的路由表里有 b ，它能侦听到 $p(b)$ 的变化，更新路由表中的 $p(b)$ 值，同时更新 $p(u^*)$ ，此时 $p(c)=32$ ，其值发生了变化，因此， c 发送带 $p(c)$ 的广播消息， d 收到此消息后更新路由表里面的 $p(c)$ 值，同时更新其 $p(u^*)$ ， d 检测到原来的路径不再是最大能量路径后，选择 e 所在的路径传输数据。

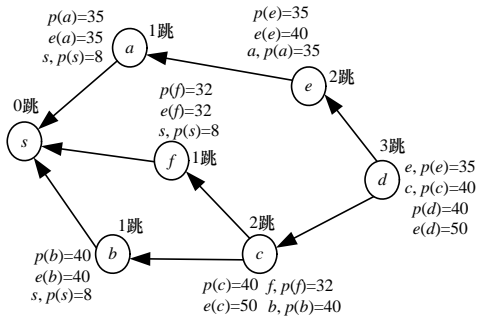


图2 路由维护示例

4 层间能量均衡的数据传输方法

上述方法构建的多跳路由只在同一层节点中(即局部区域)较好地实现了能量的均衡使用，而没在层间节点实现。例如在能量消耗热点区域(sink 附近区域)，没有实现与其他区域节点能量的均衡使用，这些节点会因能量消耗过多过早死亡。本文采用逐跳转发和直接转发相结合的混合传输模式解

决上述问题。基本思想为：每个节点以概率选择直接传输或逐跳转发数据，最外围区域的节点，由于无须转发或转发的数据较少，因此可以较大概率选择直接传输，从而减轻中间区域节点负载；中间区域的节点基本以逐跳转发数据；靠近 sink 区域的节点与 sink 距离较小，直接与 sink 通信消耗的能量有限，也可以较低概率直接与 sink 通信。计算节点直接与 sink 通信的概率 p_i 是实现层间能量均衡使用的关键。

计算 p_i 的算法如下^[5]：令 \mathfrak{S}_i 为第 i 层消耗的总能量，为实现各层之间能量的均衡，每层节点消耗的能量满足：

$$\frac{E[\mathfrak{S}_i]}{S_i} = \frac{E[\mathfrak{S}_j]}{S_j}$$

其中， S_i 为各层的区域面积，即：

$$E\left[\frac{\sum_{k=1}^{h_i} \partial_{ik}}{S_i}\right] = E\left[\frac{\sum_{k=1}^{h_j} \partial_{jk}}{S_j}\right] \quad (1)$$

其中， h_i 为 i 层要处理的负载数； ∂_{ik} 为处理每个负载消耗的能量。

定义： $E[g_i] = N \cdot D_i$ ，其中， N 为节点总数； $E[g_i]$ 为第 i 层节点发送自己产生的负载消耗的平均能量。

定义：

$$E[f_i] = -\sum_{k=1}^{n-i} \frac{\prod_{j=k}^{n-i} a_{n-j}}{\prod_{j=k}^{n-i} d_{n-j}} \times (\sum_{j=1}^{n-k} (a_j E[g_j] - a_{j+1} E[g_{j+1}]) + a_1 \times E[f_1])$$

其中， $a_i = \frac{i^2}{2i-1}$ ， $d_i = \frac{(i+1)^2-1}{2i+1}$ ； $E[f_i]$ 为第 i 层节点转发其他层节点消耗的平均能量，可由式(1)推导求得。

定义：

$$E[h_i] = E[g_i] + E[f_i] \quad (2)$$

$$E[f_{i-1}] = p_i E[h_i] \quad (3)$$

由式(2)、式(3)可得：

$$p_i = \frac{E[f_{i-1}]}{E[g_i] + E[f_i]}$$

5 性能分析

5.1 仿真环境

使用 omnet 进行算法仿真，将本文的算法和文献[4]的 MCP 算法进行比较。网络寿命用来评价路由算法的性能，定义为网络中第 1 个节点消耗完能量时网络延续的时间。用轮表示网络的寿命。一轮定义为网络中所有节点完成一次对 sink 节点的数据发送。两轮之间的时间间隔足够大，使得最后一个节点能向 sink 发送完数据。

在 $6 \text{ km} \times 6 \text{ km}$ 的长方形区域中，分别随机均匀部署 100 个和 200 个节点。仿真时拓扑变化 100 次。节点多跳传输半径为 700 m，且都能和 sink 通信。节点部署后位置不再发生变化。前一轮数据发送完成后，后一轮才开始发送。节点的初始化能量均匀分布在 $[2000, 3000]$ 中。假设数据包的长度是固定的。节点多跳传输时单跳消耗能量 3 个单元，与 sink 通信消耗的能量与距离平方成正比，节点接收数据时消耗能量 0.1 个单元。假设信道是无噪声的，数据丢失率为 0，节点传输覆盖区域为理想的圆，使用对称通信信道。

5.2 数据结果

本文算法和 MCP 算法的比较结果如图 3、图 4 所示。本文算法既考虑了层内区域能量的均衡使用，又考虑了层间区域能量的均衡使用，因此，在 95% 以上的拓扑中，本文算法 (下转第 138 页)