

# 一种基于网格的兼顾拥塞避免与能耗均衡的WSN路由算法

牛玉刚, 陈文广

(华东理工大学 化工过程先进控制和优化技术教育部重点实验室, 上海 200237)

**摘要:** 提出一种兼顾拥塞避免和能耗均衡的路由算法. 首先, 利用一种特殊的网格模型为每个节点建立多条路径, 在数据转发时通过选择权重值最大的路径作为转发路径以避免拥塞和均衡能耗; 然后, 考虑节点均匀分布时, 通过建立虚拟网格确定网络模型, 并通过路径更新均衡能耗和避免拥塞. 仿真结果表明, 所提出的算法在保证数据传输的情况下, 能够减少丢包率和均衡网络能耗.

**关键词:** 无线传感器网络; 路由算法; 网格模型; 拥塞避免; 能耗均衡

**中图分类号:** TP393

**文献标志码:** A

## A grid-based energy-aware and congestion-aware routing algorithm in WSN

NIU Yu-gang, CHEN Wen-guang

(Key Laboratory of Advanced Control and Optimization for Chemical Process of Ministry of Education, East China University of Science & Technology, Shanghai 200237, China. Correspondent: NIU Yu-gang, E-mail: acniuyg@ecust.edu.cn)

**Abstract:** The grid-based energy-aware and congestion-aware routing(GECCR) algorithm can avoid the network congestion and balance the energy consumption. At first, for a special grid model, each node establishes multiple paths. To avoid congestion and balance energy consumption, the routing algorithm chooses the path with the maximum weight value as the forwarding path. On this basis, it is further considered that when the nodes are deployed in the uniform way, a virtual grid is used to establish the grid model. The network congestion can be released and the energy congestion is balanced by updating the path. The simulation results show that the routing algorithm not only can guarantee the data transmission, but also can reduce the loss rate and balance the energy consumption.

**Keywords:** WSN; routing algorithm; grid model; congestion avoidance; balanced energy consumption

## 0 引言

无线传感器网络(WSN)由大量传感器节点通过自组织的方式形成<sup>[1]</sup>. 传感器节点把采集到的数据传送给 sink 节点. 合理的路由算法可以提高数据的传输率和网络的服务性能. 另外, 传感器节点一般是由储能有限的电池供电, 并且通常无法更换电池. 因此, 有效地利用整个网络的能量是设计网络算法时需要考虑的因素<sup>[2-6]</sup>. 文献[3]通过使用下一跳节点的剩余能量、数据包的接收率以及到达 sink 节点的跳数寻找最优的路径并进行数据转发, 以达到高效地利用网络能量的目的; 文献[4]提出了一种考虑“最小能量消耗”和“能耗均衡”的分布式路由算法, 该算法使用每个节点的“能量密度”平衡网络能量的消耗. 由于上述路由

算法都使用某个特定的节点或路径进行数据转发, 这会造成网络能量的非均衡消耗; 而 WSN 中多对一的数据传输也会造成网络能量的非均衡消耗<sup>[7]</sup>. 能量非均衡消耗会使部分节点过早地死亡, 致使网络中出现“能量空洞”, 最终导致网络生命周期的减小. 因此, 在设计路由算法时, 必须考虑如何均衡网络能耗, 以延长网络生命周期.

在设计路由算法时, 还需考虑网络拥塞对网络性能的影响, 它不仅会增大数据传输的延迟或丢包, 而且会减小网络的吞吐量, 降低网络的性能<sup>[7]</sup>. 文献[8]提出了分层的路由算法, 该算法使用 Dijkstra 算法为每一个簇建立路由树, 同时在数据转发时使用邻居节点的平均队列长度进行拥塞预防; 为了缓解拥塞, 文

收稿日期: 2015-10-30; 修回日期: 2016-01-13.

基金项目: 国家自然科学基金项目(61273073, 61374107).

作者简介: 牛玉刚(1964—), 男, 教授, 博士生导师, 从事随机系统、网络拥塞控制、网络控制系统等研究; 陈文广(1991—), 男, 硕士生, 从事无线传感器网络拥塞控制的研究.

献[9]通过整合路由层以及MAC层的信息设计路由算法,并使用主动拥塞避免实时拥塞检测和缓解机制来达到避免网络拥塞的目的。

WSN的路由协议可分为单路径协议和多路径路由协议,其中多路径路由协议使用多条路径进行数据转发.文献[10]提出了多路径距离矢量路由协议(AOMDV),该协议在源节点采集到数据时进行路由请求,从而建立多条到达sink节点的路径.然而,已有的路由算法在建立路径时或者考虑了能量均衡消耗,或者考虑了拥塞避免问题.但是,它们都没有考虑如何在平衡能耗的同时避免网络拥塞.因此,为了在保证网络能量均衡消耗的情况下避免网络拥塞,本文利用网格技术提出一种兼顾拥塞避免和能耗均衡的多路径路由算法——GECR路由算法.该算法包括GECR\_Grid和GECR\_Balance两部分,且GECR\_Grid考虑了节点以网格形式布置的情况,包括路径建立和数据转发两个阶段.在路径建立阶段,GECR\_Grid为每个节点建立“相对位置坐标”,“相对位置坐标”可以使每个节点通过统一的方式确定数据的传播方向,即建立路径.以这种方式建立路径时,节点只需要了解自身和邻居节点的信息,从而大大减小路由建立的能量开销.在数据转发阶段,GECR\_Grid会根据下一跳节点的剩余能量、队列长度以及到达sink节点的跳数为每条路径设置权重值,节点则根据路径的权重值来选择转发路径,以达到避免拥塞和均衡能耗的目的。

在GECR\_Grid的基础上,本文针对节点随机均匀分布的情况,进一步提出GECR\_Balance路由算法.GECR\_Balance把网络分为虚拟的网格,节点通过“网格坐标”以及“相对位置坐标”确定自己所属的网格.GECR\_Balance为每个网格选择一个主节点,负责转发邻居网格的数据.位于同一网格的节点拥有相同的“相对位置坐标”.GECR\_Balance为拥有相同“相对位置坐标”的节点建立相同的路径,从而达到建立路径的同时减小能量开销的目的.网格中的主节点根据GECR\_Grid算法中的路径建立方式和路径选择方法进行路径建立和数据转发.同时,GECR\_Balance算法还采用路径更新策略,该策略基于能耗均衡和拥塞避免进行主节点的更替,即路径的更新.“相对位置坐标”确保位于同一网络中的节点可以通过统一的方式完成路径更新。

## 1 网络拓扑模型

### 1.1 网络模型假设

本文对网络的特性给出如下假设:

- 1) sink节点拥有充足的能量供应;
- 2) 每个传感器节点都有相同的能力;

3) 每个传感器节点都装备有GPS定位装置;

4) 传感器节点一旦布置完成以后就不再移动,并且在各个方向拥有相同的通讯能力。

### 1.2 网格模型的建立

本文首先考虑一种特殊的网络拓扑模型,即每个节点都被布置在事先规划好的网格顶点上,如图1所示.而为了保证节点与邻居节点之间正常地传输数据,每个网格的边长 $d$ 需满足

$$d^2 + d^2 \leq R^2, \tag{1}$$

其中 $R$ 为节点的传输距离。

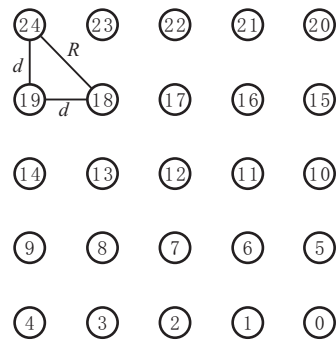


图 1 网格拓扑的网格划分

大部分WSN都是通过随机均匀的方式布置传感器节点的,因此本文为随机布置的WSN建立了一个虚拟网格.当节点布置完成以后,如果sink节点位于部署区域的顶点,则整个网络被统一划分为多个虚拟网格;如果sink节点位于部署区域内部,则以sink节点为原点把整个网络分为4个象限,并把每个象限统一划分为一个虚拟网格.拥有相同顶点或边的网格为邻居网格。

为每个网格设置了用于确定网格位置的网格坐标.sink节点所在网格的网格坐标为(0,0),而位于该网格上方和下方的网格坐标依次为(0,1),(0,2),...,位于该网格左方和右方的网格坐标则依次为(0,1),(0,2),...,其他网格则根据这些网格坐标确定自己的网络坐标.图2为sink节点位于顶点或第二象限的网格坐标示意图。

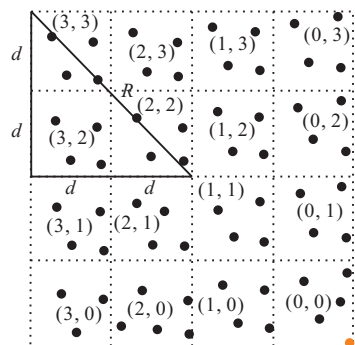


图 2 均匀拓扑的网格划分

为了保证每个节点都能够与其邻居网格中的所

有节点进行正常通讯, 边长  $d$  需要满足

$$(2d)^2 + (2d)^2 \leq R^2. \quad (2)$$

虚拟网格建立完成以后, 每个节点会确定其相对位置坐标以及所属的网格. 为了确定节点的相对位置坐标, sink 节点会向整个网络广播一个包含 sink 节点位置的 Locate\_notice 信息 (如表 1 所示). 节点  $i$  在接收到 Locate\_notice 以后会采用如下等式计算“相对位置坐标”:

$$\begin{cases} X_i = \left\lfloor \frac{x_i - x_{\text{sink}}}{d} \right\rfloor, \\ Y_j = \left\lfloor \frac{y_i - y_{\text{sink}}}{d} \right\rfloor. \end{cases} \quad (3)$$

其中:  $(x_i, y_i)$ 、 $(x_{\text{sink}}, y_{\text{sink}})$ 、 $(X_i, Y_i)$  分别是节点  $i$  和 sink 节点的绝对位置坐标以及节点  $i$  的相对位置坐标; 而符号  $\lfloor \text{value} \rfloor$  是取得 value 的整数值.

表 1 Locate\_notice 信息

sink 节点地址	X 的坐标	Y 的坐标
0	$x_0:760$	$y_0:0$

计算相对位置坐标以后, 与该节点相对位置坐标相等的网格坐标所确定的网格就是该节点所属的网格. 拥有相同相对位置坐标的节点位于同一网格内.

## 2 GEGR\_Grid 路由算法

本文首先针对图 1 所示的网格拓扑模型提出一种路由算法——GEGR\_Grid, 该算法包括路径建立和数据转发两部分:

1) 路径建立: 节点部署完成以后, sink 节点也会广播一个如表 1 所示的地理位置通知信息 Locate\_notice. 每个节点在接收到 Locate\_notice 信息以后会采用下式计算自己的“相对位置坐标”:

$$\begin{cases} X_i = \left\lfloor \frac{x_i - x_{\text{sink}}}{d} \right\rfloor, \\ Y_j = \left\lfloor \frac{y_i - y_{\text{sink}}}{d} \right\rfloor. \end{cases} \quad (4)$$

其中:  $(x_i, y_i)$  和  $(x_{\text{sink}}, y_{\text{sink}})$  分别是节点  $i$  和 sink 节点的绝对地理位置坐标, 而  $(X_i, Y_i)$  和  $d$  是节点  $i$  的相对位置坐标和网格宽度. sink 节点的相对位置坐标为  $(0, 0)$ . 图 1 中节点 7 和节点 18 的相对位置坐标分别为  $(2, 1)$  和  $(3, 3)$ .

表 2 Route\_discovery 信息

节点地址	绝对 $x$ 坐标	绝对 $y$ 坐标	相对 $X$ 坐标	相对 $Y$ 坐标
13	190	380	3	2

节点在计算相对位置坐标后会广播一个用于路由建立的路由发现信息 Route\_discovery (如表 2 所示). 节点  $i$  接收到邻居节点  $j$  的 Route\_discovery 信息以后, 首先会把节点  $j$  添加到自己的邻居列表中, 邻居列表如表 3 所示; 然后, 比较自己的“相对位置坐标”和 Route\_discovery 中  $j$  节点的“相对位置坐标”. 如果节点  $i$  的“相对位置坐标”和节点  $j$  的“相对位置坐标”满足下式中的一条:

$$\begin{cases} (X_i - 1) = X_j \text{ 且 } (Y_i - 1) = Y_j, \\ (X_i - 1) = X_j \text{ 且 } Y_i = Y_j, \\ X_i = X_j \text{ 且 } (Y_i - 1) = Y_j, \end{cases} \quad (5)$$

则节点  $i$  建立一条通过节点  $j$  到达 sink 节点的路径.

表 3 邻居列表

节点 ID 号	剩余能量/J	队列长度	绝对 $x$ 坐标	绝对 $y$ 坐标	相对 $X$ 坐标	相对 $Y$ 坐标
13	85	56	190	380	3	2

2) 数据转发: GEGR\_Grid 通过为路径设置权重的方式避免网络拥塞和均衡网络能耗. 每一条路径的权重值 ( $L_{i \rightarrow j}$ ) 都是下一跳节点的剩余能量 ( $RE_j$ ) 和队列长度 ( $QL_j$ ) 以及相对位置坐标  $(X_j, Y_j)$  的函数, 采用下式计算得到:

$$L_{i \rightarrow j} = \alpha \times \frac{RE_j}{IE} + \beta \times \frac{X_i + Y_i}{X_j + Y_j} + \gamma \times \frac{TQL - \left( \left( \left\lfloor \frac{QL_j}{5\% \times TQL} \right\rfloor + 1 \right) \times (5\% \times TQL) \right)}{TQL}. \quad (6)$$

其中: IE 和 TQL 分别是节点初始能量和存储空间的可能存储数据包的总队列长度,  $(X_i, Y_i)$ 、 $(X_j, Y_j)$  分别是节点  $i$  和节点  $j$  的相对位置坐标. 同时, 式 (6) 中的 3 个参数  $\alpha$ 、 $\beta$ 、 $\gamma$  满足下式:

$$\alpha + \beta + \gamma = 1. \quad (7)$$

不同的应用对网络的性能要求不同, 因此, GEGR\_Grid 可以通过调节参数  $\alpha$ 、 $\beta$ 、 $\gamma$  的值来适应不同的应用.

路径权重计算函数 (6) 包含节点的剩余能量和队列长度, 因此为了实时更新每条路径的权重值, 每个节点会定期广播一个 Hello 信息. 邻居节点接收到以后, 根据该信息更新其剩余能量和队列长度.

表 4 Hello 信息

节点地址	剩余能量/J	队列长度	绝对 $x$ 坐标	绝对 $y$ 坐标	相对 $X$ 坐标	相对 $Y$ 坐标
13	85	56	190	380	3	2

每个节点在转发数据之前都会比较所有路径的权重值大小, 然后选择权重值最大的路径进行数据转发. 由式 (6) 可知: 下一跳节点的队列长度较小和剩余能量较多时路径权重值较大, 因此通过选择权重值最大的路径可以达到避免网络拥塞和均衡网络能耗的目的.

## 3 GEGR\_Balance 路由算法

本文进一步在 GEGR\_Grid 的基础之上提出了适用于以随机均匀的方式部署的 WSN 的路由算法——GEGR\_Balance. 它以图 2 所示的网格模型为基础, 不

仅包含 GECR\_Grid 中的路径建立和数据转发阶段,还包括路径更新阶段。

1) 路径建立: 传感器节点部署完以后, GECR\_Balance 使用 1.2 节的方法建立一个虚拟网络. 所有节点确定自己所属的网格以后, GECR\_Balance 会在每个网格中选择一主节点负责数据的转发。

与 GECR\_Grid 类似, 广播一个如表 2 所示的 Route\_discovery 信息, 把其中的相对坐标改为节点的网格坐标. 节点  $i$  接收到邻居节点  $j$  的 Route\_discovery 信息以后, 会把节点  $j$  的信息添加到自己的邻居列表中(如表 3 所示)。

节点  $i$  会比较自己的网格坐标与 Route\_discovery 信息包含的邻居节点网格坐标. 如果满足式 (5) 中的条件, 则节点  $i$  会建立一条把节点  $j$  作为下一跳节点的路径, 即节点  $j$  就是  $j$  所属网格的主节点, 否则直接丢弃 Route\_discovery 信息. 而当节点  $i$  接收到与节点  $j$  属于同一网格中节点  $h$  发送的 Route\_discovery 信息时, 节点  $i$  会使用绝对地理坐标比较主节点  $j$  和节点  $h$  与 sink 节点之间的距离. 如果节点  $h$  比节点  $j$  更靠近 sink 节点, 则节点  $i$  会用节点  $h$  替换掉节点  $j$  作为新建路径的下一跳节点, 即选择节点  $h$  作为主节点。

2) 数据转发: 路径建立完成以后, GECR\_Balance 会使用式 (6) 设置每条路径的权重值, 并选择其中权重值最大的路径进行数据转发以达到拥塞避免和能耗均衡的目的. 同时, 每个节点也会定期广播 Hello 信息用于更新路径的权重值。

3) 路径更新: 在 GECR\_Balance 中还包含了路径更新阶段, 通过主节点的更替(即路径的更新)来进一步避免拥塞和平衡能耗. 具体的更新过程包括基于能耗均衡的路径更新和基于拥塞控制的路径更新这两种情况:

① 基于能耗均衡的路径更新: GECR\_Balance 首先将节点的剩余能量按照 0%~10%, 11%~20%, ... 等分为 10 个能量等级. 然后, 根据“网格能量等级”进行主节点的更换, “网格能量等级”是指网格中剩余能量最多的节点所在的能量等级. 在网络运行期间, 网格中的主节点通过“Hello”信息获得网格中其他节点的能量等级, 并通过比较网格中所有节点的能量等级确定“网格能量等级”。

主节点在转发数据之前都会比较自身能量等级与“网格能量等级”, 如果自身的“能量等级”低于“网格能量等级”, 则主节点会广播如表 5 所示的“Route\_update”信息用于寻找新的主节点. 与主节点属于同一网格的节点在接收到“Route\_update”信息以后, 首先检测自己的能量等级. 如果自身的能量等级等于“网

格能量等级”, 则该节点根据自身剩余能量通过下式计算得到一个“等待时间”:

$$WT_i = 0.05 \times \left( 1 - \frac{RE_i - \left[ \frac{RE_i}{10\% \times IE} \right] \times (10\% \times IE)}{10\% \times IE} \right). \quad (8)$$

其中:  $RE_i$  和  $IE$  分别是该节点的剩余能量和初始能量, 符号 [value] 表示对 value 取整数值. 该节点在“等待时间”结束后, 立即广播一个如表 6 所示的“Route\_replace”信息。

表 5 Route\_update 信息

节点地址	绝对 $x$ 坐标	绝对 $y$ 坐标	相对 $X$ 坐标	相对 $Y$ 坐标	网格能量等级 $I$
13	190	380	3	2	79.5

表 6 Route\_replace 信息

节点地址	绝对 $x$ 坐标	绝对 $y$ 坐标	相对 $X$ 坐标	相对 $Y$ 坐标
13	190	380	3	2

主节点邻居网格中的节点接收到“Route\_update”信息以后, 会检测自己的路由列表中是否有到达该主节点的路径. 如果检测到存在这样的路径, 则该节点会等待第一个“Route\_replace”信息, 并把发送“Route\_replace”的节点更替为该路径的下一跳节点, 即把该节点更新为主节点。

② 基于拥塞控制的路径更新: 主节点在转发数据之前也会检测自身的队列长度, 并在队列长度到达总缓存空间的 80% 时广播一个不包含“网格能量等级”的“Route\_update”信息. 与该主节点属于同一网格的节点在接收到“Route\_update”信息以后, 会根据下式计算得到一个“等待时间”:

$$WT_i = 0.05 \times \frac{QL_i}{TQL}. \quad (9)$$

其中:  $QL_i$  和  $TQL$  分别是节点的队列长度和缓存空间大小, 并且在“等待时间”结束以后, 立即广播一个“Route\_replace”信息. 邻居网格的节点收到“Route\_update”信息以后, 将检测自己的路由列表是否有到达发送“Route\_update”的主节点的路径. 如果检测到存在这样的路径, 则该节点会使用基于能耗均衡的路径更新中的方法进行主节点的更替, 即路径的更新. 通过使用队列长度较小的节点作为主节点的路径更新策略, 可以进一步避免网络中拥塞的产生。

## 4 仿真分析

### 4.1 参数设置

本文采用 NS-2.35 进行仿真实验, 通过对比分析本文提出的 GECR 路由算法和多路径路由算法 AOMDV<sup>[10]</sup> 的丢包率、平均时延和吞吐量等服务性能, 验证了 GECR 在避免拥塞方面和均衡能耗方面的

效果, 仿真参数如表 7 所示.

表 7 仿真参数设置

环境参数	网络拓扑配置值	均匀拓扑配置值	环境参数	网络拓扑和均匀拓扑的配置值
区域范围/m <sup>2</sup>	800×800	450×450	流量类型	CBR
网格边长/m	130	87	MAC 协议	802.11
通讯距离/m	250	250	缓存空间	100
节点数	49	65	分组大小/bytes	128
仿真时间/s	300	300	数据流数	10
带宽/M/s	2	2	接收功率/W	0.02
发送功率/W	2.5	2.5	空闲功率/W	0.01
休眠功率/m·W	0.6	0.6	初始能量/J	100

### 4.2 仿真结果

为了评估 GECR 在拥塞避免方面的性能, 本文统计了在不同发送时间间隔下源节点和 sink 节点发送和接收的数据包以及网络的丢包率. 由图 3 可知: GECR 与 AOMDV 发送的数据包相同, 同时, 在发送间隔较大即发送速率较小时, sink 节点接收数据包也相同. 但是, 随着发送速率的增加, sink 节点接收到的数据包开始低于源节点发送的数据包, 这是因为网络中产生了拥塞, 导致数据包丢失. 从图 3 中也可以看到: 当发送时间间隔低于 0.14 s 时, GECR 中 sink 节点接收到的数据包数量要高于 AOMDV 中 sink 节点接收到的数据包数量, 同时图 3 中的网络丢包率也验证了这一点. 由图 3 可知: 与 AOMDV 相比, GECR 降低了网络丢包率, 提高了网络的性能. 这是因为 GECR\_Grid 和 GECR\_Balance 通过权重值的比较选择负载较小的路径进行数据的转发, 这可以避免网络拥塞的产生. 而在网络将要发生拥塞时, GECR\_Balance 会进行主节点的更替, 这可以进一步避免网络中发生拥塞, 最终减小网络的丢包率.

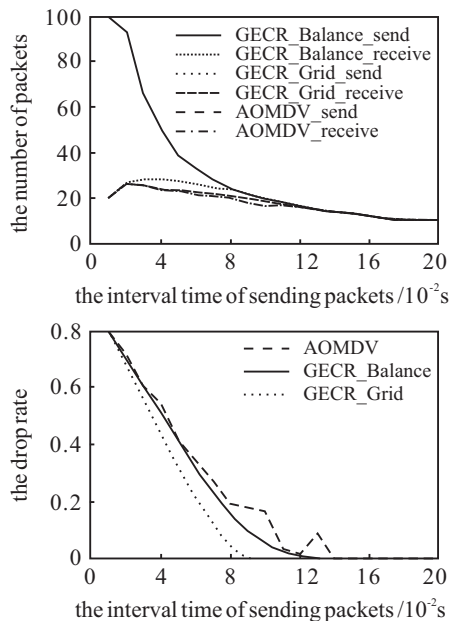


图 3 不同发送时间间隔下发送和接收的数据包和丢包率

本文也统计了 GECR 和 AOMDV 中的时延与吞吐量. 由图 4 可知: GECR 和 AOMDV 的时延大致相同, 而 GECR 则提高了整个网络的吞吐量. 这是因为 GECR 会在网络将要发生拥塞时, 选择其他路径进行数据转发. 同时, GECR\_Balance 在拥塞发生以后会进行路径的更新, 从而使网络可以使用多个节点的存储空间来存储数据包. 由图 4 可知: GECR\_Balance 的最大吞吐量为 115 kbit/s, 而 AOMDV 的最大吞吐量为 103 kbit/s, 即 GECR\_Balance 最大吞吐量比 AOMDV 提高了 11% 左右.

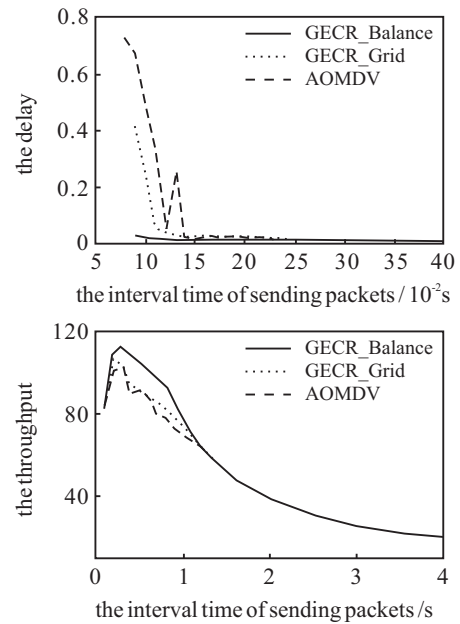


图 4 不同发送时间间隔下的网络时延和吞吐量

GECR 路由算法也考虑了如何均衡网络消耗. 文献 [11] 指出, 网络的“能量空洞”问题并不能完全避免, 因此, 为了延长网络的生命周期, 必须最大限度地均衡 sink 节点邻居节点的能量消耗. 本文统计了 sink 节点的邻居节点的平均剩余能量和方差. 由图 5 可知: AOMDV 与 GECR\_Grid 的平均剩余能量大致相同, 而 GECR\_Balance 的剩余能量则比 AOMDV 多, 这是因为 GECR\_Balance 拥有更多的 sink 节点的邻居节点负责数据的转发. 图 5 也统计了 sink 节点的邻居节点剩余能量的方差, 可以发现, GECR 的方差比 AOMDV 的方差要小, 这就说明了 GECR 能够平衡 sink 节点的邻居节点的能量消耗. 这是因为 GECR\_Grid 和 GECR\_Balance 在设置路径权重值时考虑了下一跳节点的剩余能量, 同时 GECR\_Balance 也会使用“网格能量等级”来平衡同一网格中节点的能耗.

为了进一步说明整个网络的能量消耗状况, 本文绘制了网络剩余能量三维图, 如图 6 所示. 所有节点均匀分布在 450×450/m<sup>2</sup> 的区域内, sink 节点位于 (450, 0). 图 6 说明 sink 节点的邻居节点的剩余能量大

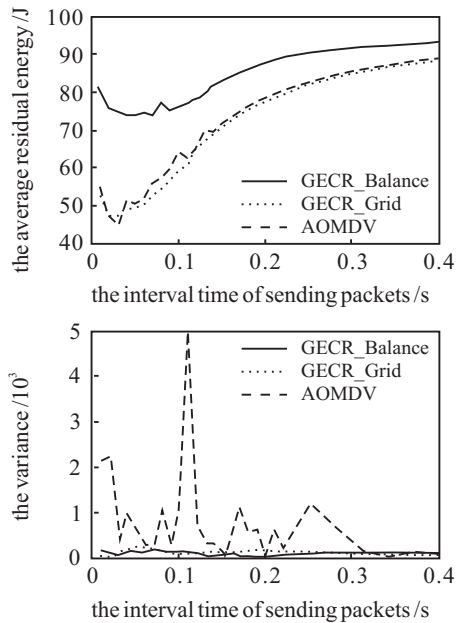


图5 不同发送时间间隔下 sink 节点  
邻居节点的平均剩余能量和方差

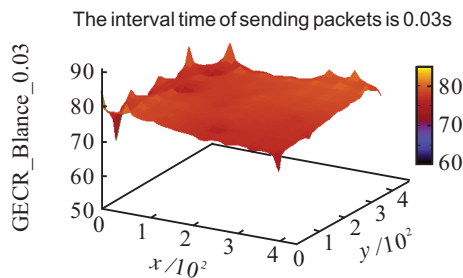


图6 发送时间间隔为 0.03 s 和 0.05 s 的剩余能量分布图

致相同,这进一步说明了 GECR 能够均衡 sink 节点的邻居节点的能量消耗。从图 6 中也会发现,边缘节点并不能够均匀地消耗能量。这是因为边缘节点不参与数据的转发,而 GECR.Balance 仅仅能够平衡转发节点的能量消耗。这就造成了边缘节点能量消耗不均。

## 5 结 论

本文在兼顾网络拥塞避免和能耗均衡的情况下,提出了一种基于网格的多路径路由算法(GECR)。在路径建立阶段,根据“相对位置坐标”为每个节点建立多条路径;而在数据转发阶段,每个节点会根据路径权重值选择转发路径。GECR.Balance 会根据主节点的“队列长度”和“剩余能量等级”进行主节点的更替即路径更新。仿真结果表明,GECR 能够降低网络丢包率和增加网络吞吐量,同时也能够均衡 sink 节点邻居节点的能量消耗。

## 参考文献(References)

[1] Rahman K C. A survey on sensor network[J]. J of Computer and Information Technology, 2010, 1(1): 76-87.

- [2] 牛玉刚,甘峰浩,胡源.基于压缩感知的拥塞控制机制[J].控制与决策,2015,30(2):246-250.  
(Niu Y G, Gan F H, Hu Y. Congestion control mechanism in WSN based on compressive sensing[J]. Control and Decision, 2015, 30(2): 246-250.)
- [3] Ghaffari A. An energy efficient routing protocol for wireless sensor networks using a-star algorithm[J]. J of Applied Research & Technology, 2014, 12(4): 815-822.
- [4] Singh D, Kuila P, Jana P K. A distributed energy efficient and energy balanced routing algorithm for wireless sensor networks[C]. Int Conf on Advances in Computing, Communications and Informatics. New Delhi: IEEE Press, 2014: 1657-1663.
- [5] 董国勇,彭力,吴凡,等.一种采用蚁群优化的WSN能量均衡非均匀分簇路由算法[J].小型微型计算机系统,2015,36(7):1565-1568.  
(Dong G Y, Peng L, Wu F, et al. Energy-balanced uneven clustering routing protocol based on ant colony optimization for wireless sensor networks[J]. J of Chinese Computer Systems, 2015, 36(7): 1565-1568.)
- [6] Liu Y, Wang Z. Maximizing energy utilization routing scheme in wireless sensor networks based on minimum hops algorithm[J]. Computers & Electrical Engineering, 2012, 38(3): 703-721.
- [7] Li J, Mohapatra P. Analytical modeling and mitigation techniques for the energy hole problem in sensor networks[J]. Pervasive and Mobile Computing, 2007, 3(3): 233-254.
- [8] Mohajerzadeh A H, Yaghmaee M H, Eskandari Z. Tree based energy efficient and congestion aware routing protocol for wireless sensor networks[C]. IEEE Singapore Int Conf on Communication Systems. Guangzhou: IEEE Press, 2008: 1707-1711.
- [9] Peng Q, Enqing D, Juan X, et al. Multipath routing protocol based on congestion control mechanism implemented by cross-layer design concept for WSN[C]. The 17th Int Conf on Computational Science and Engineering. Chengdu: IEEE Press, 2014: 378-384.
- [10] Marina M K, Das S R. Ad hoc on demand multipath distance vector routing[J]. Wireless Communications and Mobile Computing, 2006, 6(7): 969-988.
- [11] Stojmenovic I, Olariu S. Data-centric protocols for wireless sensor networks[M]. New Jersey: A John Wiley & Sons, Inc, 2005: 417-456.

(责任编辑: 闫 妍)