

基于智能天线和动态虚拟簇的均衡节能路由

胡青松^{1,2}, 吴立新^{1,2}, 张申¹, 丁恩杰¹

(1. 中国矿业大学 物联网(感知矿山)研究中心, 江苏 徐州 221008; 2. 中国矿业大学 环境与测绘学院, 江苏 徐州 221008)

摘要: 为 WSN 提出了一种能量节省与能耗均衡相结合的路由算法 SaDVC-Routing。该算法在辅助中继和智能天线的波束范围内构建虚拟簇, 从簇内选择中继加权值最大的节点充当路由中继。根据节点能耗情况和节点间距离, 利用波束扩展法对虚拟簇进行动态更新。如果波束宽度超过阈值或者中继节点死亡, 则依据加权距离矩阵重新发起路由由请求。仿真表明, 算法既能有效降低能量消耗, 也可以在局部和全局实现能耗均衡。

关键词: 动态虚拟簇; 智能天线; 能量节省; 能耗均衡; 中继选择; 矿山物联网

中图分类号: TP393

文献标识码: A

文章编号: 1000-436X(2013)08-0169-08

Balanced energy-efficient routing algorithm based on smart antenna and dynamic virtual cluster

HU Qing-song^{1,2}, WU Li-xin^{1,2}, ZHANG Shen¹, DING En-jie¹

(1. IoT/Perception Mine Research Center, China University of Mining and Technology, Xuzhou 221008, China;

2. School of Environment Science and Spatial Informatics, China University of Mining and Technology, Xuzhou 221008, China)

Abstract: An energy-efficient and energy-consumption-balanced routing, named SaDVC-Routing, was put forward for WSN. The algorithm constructs virtual clusters based on assistant relays in the area of smart antenna beams, and then selects the node with maximum relay weight from the cluster to act as routing relay. With the help of energy consumption and nodes distance information, SaDVC-Routing can dynamically update the virtual clusters with beam expanding. If the beam width exceeds the threshold, a routing request is re-broadcasted based on weighted distance matrix. According to the simulation, it was found that this algorithm could not only decrease the energy consumption, but also get consumption balance between nodes in the local and global area.

Key words: dynamic virtual cluster; smart antenna; energy saving; balance of energy consumption; relay selection; mine internet of things

1 引言

无线传感器网络(WSN, wireless sensor networks)在感知和监测中的地位越来越重要。近年来, 展开了一系列关于空天地一体化观测和矿井工作面活动目标定位的研究, 利用 WSN 采集和传输数据是其中一项重要研究内容。由于需要连续监测, 因此特别希望尽可能地延长网络寿命, 并且避免某个区域

的节点过早死亡而出现监测空洞。这就要求研究合适的节能方法, 同时在节点间均衡地消耗能量。为了节省能量, 一方面可以在物理层采用功耗更低的电路, 同时尽量降低发射功率^[1]; 另一方面可以在网络层设计节能路由, 降低寻路和数据传输的能耗。也可以对物理层、MAC 层和网络层进行跨层优化^[2], 利用单跳的结果求解整个网络的功率分配和能耗。本文将利用智能天线的特性, 研究网络层

收稿日期: 2012-08-27; 修回日期: 2013-03-08

基金项目: 国家重点基础研究发展计划(“973”计划)基金资助项目(2011CB707102); 国家自然科学基金资助项目(51204177); 中央高校基本科研业务费标准专项基金资助项目(2012BZB19)

Foundation Items: The National Basic Research Program of China (973 Program) (2011CB707102); The National Natural Science Foundation of China (51204177); Special Found of National Higher-education Institution General Research and Development Project (2012BZB19)

的节能路由设计方法。

在 WSN 的节能路由中, 基于分簇思想的算法是非常重要的一类, 关键是簇的划分和簇头的选择^[3]。由于簇头承担了全部数据转发任务和数据融合任务, 因此能量消耗比普通节点快得多。为了避免簇头的能量快速耗尽 (低于阈值) 而死亡, 要求各个节点轮流充当簇头, 以实现节点间的能耗均衡。

智能天线由于可以自适应地将波束对准目标节点的方向^[4], 并能按需调整波束宽度^[5], 因此将同样长度的数据发送到同样距离, 所需要的发射功率要小得多。如果能充分利用智能天线和分簇方法的优势, 将能实现物理层和网络层的跨层优化, 获得较高的能量增益。为此, 本文综合利用智能天线自适应调整波束宽度、分簇和功率调整的方法, 提出一种基于智能天线和虚拟簇的均衡节能路由算法 (SaDVC-Routing, routing based on smart antenna and dynamic virtual cluster), 以期达到能量节省和节点间均衡消耗能量的目的。

2 发送功率模型和能量消耗模型

2.1 发送功率模型

假定 WSN 分布在二维空间, 其节点坐标信息已知。每个节点都配置有一根智能天线, 其波束宽度 θ 和波束方向可以自由调整。波束宽度不同, 节点的发射距离也不一样。其规律是: 在保持发送功率不变的前提下, 波束宽度越宽, 发射距离越小。或者说, 如果保持通信距离不变, 波束越宽, 所需的发送功率越大。因此, 为了适应波束宽度的变化, 节点在发送数据之前必须确定发送功率, 在此使用自由空间路径损耗模型加以推导。在使用全向天线收发时, 有^[6,7]

$$P_r = \left(\frac{\lambda}{4\pi d} \right)^\alpha P_t \quad (1)$$

其中, P_r 为接收功率, P_t 为发送功率, λ 为传输信号的波长, α 为衰减指数。

对于智能天线而言, 有

$$P_r = \left(\frac{\lambda}{4\pi d} \right)^\alpha P_t g_m \quad (2)$$

其中, g_m 为智能天线相对于全向天线的增益^[8]。对于一个波束宽度为 θ 的智能天线而言, 其表面积 A 可以用球冠表面积计算, 为 $2\pi r^2(1 - \cos(\theta/2))$ 。于是有

$$\begin{aligned} g_m &= \frac{P_t/A}{P_t/S} = \frac{4\pi r^2}{2\pi r^2(1 - \cos(\theta/2))} \\ &= \frac{2}{1 - \cos(\theta/2)}, \quad \theta \neq 0 \end{aligned} \quad (3)$$

从式 (3) 可以看出, g_m 不受衰减指数的影响, 因此 α 值的大小不改变智能天线的相对优势, 鉴于此, 后文中只考虑 $\alpha = 2$ 的情况。显然, $g_m \in [1, +\infty)$, 且 θ 值越小, g_m 越大。当 $\theta = 2\pi$ 的时候, $g_m = 1$, 智能天线退化为全向天线模式; 当 $\theta = 0$ 的时候, 信号无法发送, g_m 没有意义。

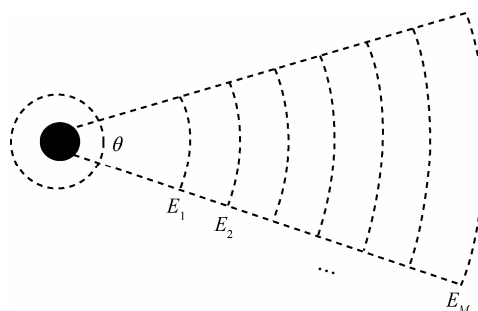


图 1 智能天线的波束及其功率等级

众所周知, 只有 P_r 大于解码阈值 P_0 , 接收节点才能对接收信号正确接收和解码。因此, 发送功率 P_t 需要满足如下条件

$$\begin{aligned} P_t &\geq P_0 \left(\frac{4\pi d}{\lambda} \right)^2 / g_m \\ &= P_0 \left(\frac{4\pi d}{\lambda} \right)^2 \frac{1 - \cos(\theta/2)}{2} \\ &= \frac{8P_0\pi^2}{\lambda^2} d^2 (1 - \cos(\theta/2)) \end{aligned} \quad (4)$$

从式(1)可以看出, 为了提高 P_r , 可以减小收发节点之间的距离 d , 也可以提高发送节点的发送功率 P_t 。显然, 节点布置好以后, 不能因为接收某个节点信号的功率弱而移动, 即 d 无法改变, 因此一般通过提高 P_t 实现。为此, 本文假设节点的发送功率可以自由调整。但是从实际的角度考虑, 发送功率一般不能连续改变, 因此将发送功率等间隔的分成 M 个等级^[2], 等级间隔为 ΔP , 如图 1 所示^[9] (其中, 虚线圆表示旁瓣, 扇形代表主瓣。现有的许多波束成形技术和旁瓣抑制技术可以在增强主瓣的同时, 将旁瓣控制在极小的范围之内, 因此本文不考虑旁瓣的影响)。在确定实际发送功率时, 取

$$P_i = \left\lceil \frac{8P_0\pi^2}{\lambda^2} \cdot d^2(1 - \cos(\theta/2)) \right\rceil = P_i, \quad i = 1, 2, \dots, M \quad (5)$$

其中， P_i 表示发送节点的第 i 个功率等级； $\lceil x \rceil$ 表示大于等于 x 的最小整数。

2.2 能量消耗模型

下面推导本文使用的能量消耗模型（简称能耗模型）。假设发送节点的射频放大部分(transmit amplifier)的能耗为 E_{fs} /bit，其他电路部分的能耗为 E_{elec} /bit，发送能耗为 E_{Tx} /bit，接收能耗为 E_{Rx} /bit。为了区别，分别用 d_o 和 d_s 表示全向天线与智能天线在功率 P_i 下的发射距离，那么智能天线发送一个长度为 k bit 的数据分组的能耗为^[3]

$$\begin{aligned} E &= E_{Tx}(k, d_s) + E_{Rx}(k) \\ &= (kE_{elec} + kE_{fs}d_s^2) + kE_{elec} \\ &= 2kE_{elec} + kE_{fs}d_s^2 \end{aligned} \quad (6)$$

在本文中， d_s 表示直接使用节点间的物理距离。式(6)中，对于给定种类的节点而言，其 E_{elec} 和 E_{fs} 是固定不变的，因此，发送一个数据分组的能耗 E 只与收发节点之间的距离有关。在同样的发射功率下，根据式(1)和式(2)可以推出

$$d_s^2 = g_m d_o^2 \quad (7)$$

因此，如果采用全向天线将同样的 k bit 数据发送到相同节点，它所需要的能耗为

$$\begin{aligned} E &= 2kE_{elec} + kE_{fs}d_s^2 g_m \\ &= 2kE_{elec} + 2kE_{fs}d_o^2 / (1 - \cos(\theta/2)) \end{aligned} \quad (8)$$

3 虚拟簇的动态构建与更新

SaDVC-Routing 的基本思想是综合运用智能天线自适应波束宽度调整、功率控制和分簇技术，为收、发节点对选取合适的中继节点，使数据传输所需的能耗尽可能小，节点间的能耗能够均衡。算法的关键是构建以辅助中继为圆心、簇边界与智能天线波束边界相切的簇。这种簇只是用于辅助确定数据传输所需的中继节点，其簇头并不承担一般意义下的簇头职责（比如数据汇聚和中转），是一种虚拟簇。这种虚拟簇会随着数据的发送和网络节点能量的消耗而动态更新或重建，因此称为动态虚拟簇。本节在第 2 节的功率模型和能耗模型的基础上，介绍动态虚拟簇的构建和更新方法。

3.1 虚拟簇的构建方法

算法运行之初，先利用现有的简单路由算法（比如 AODV、Floyd，这里采用 Dijkstra 算法）寻找从源节点 S 到目标节点 D 的路由作为辅助路由，路由上的中间节点称为辅助中继，用 R_i 表示。如果找不到辅助路由，算法放弃数据发送，直接退出。

假定 Dis_i 为本跳节点与 R_i 之间的距离。以 R_i 为圆心、 $Dis \cdot \sin(\theta/2)$ 为半径画圆，构成虚拟簇的边界，这样可以保证所构造的圆与上一跳的波束边界相切，如图 2 所示。簇边界内的节点（簇内节点）组成一个簇，记为 C 。由于辅助中继也是网络中的普通节点，拥有数据收发能力，因此它也有可能成为中继节点。为了区别，后文将除辅助中继外的簇内节点称为成员节点。

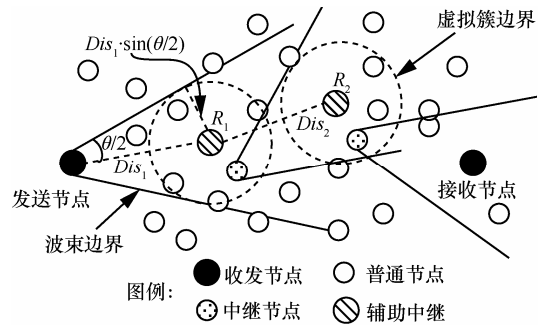


图 2 虚拟簇的构建

令智能天线的最大波束宽度为 θ_{max} 。如果 C 内没有任何节点（称为“簇内真空”）且 $\theta < \theta_{max}$ ，就将波束宽度值更新为 $\theta = \theta + \Delta\theta$ （ $\Delta\theta$ 表示智能天线的波束宽度变化增量），然后以 R_i 为圆心、 $Dis_i \cdot \sin(\theta/2)$ 为半径重新构建虚拟簇，即圆心位置不变，半径增大，使虚拟簇的覆盖范围更广，从而令原来处于覆盖范围外的节点有机会成为簇内节点，这种方法称为波束扩展法。

注意，确定好下一跳后，如果要发送数据给它，必须根据式(5)确定发送功率，公式中所需的距离可以根据节点的坐标直接计算得出，不过更快速的方法是查询矩阵索引。为此，假定每个节点维护有一个“距离矩阵” D_m 和一个“死亡节点向量” V_{dead} 。“距离矩阵”的作用体现在两方面：一是在发送数据时辅助确定发送功率等级；二是与死亡节点向量一起更新网络拓扑。令网络的初始节点数量为 N ，那么“距离矩阵”的初始大小为 $N \times N$ ，其元素 $d_{i,j}$ 为节点 i 和 j 之间的欧式距离。

为了避免始终选择某个簇内节点，在寻找中继节点时，最好将距离和剩余能量这 2 个因素结合起来考虑，这里采用文献[3]提出的指标作为中继加权值^[3]，有

$$W = E_r(1 + \beta \cdot \exp(-d/d_{\max})) \quad (9)$$

其中， E_r 表示簇内节点的剩余能量； d 为簇内节点与辅助中继的距离， d_{\max} 为 d 的最大值， $\exp(-d/d_{\max})$ 表达了距离因素所占的比重； β 为权重调节因子，用于调节节点剩余能量与节点距离在中继加权值中的比重。选择中继时，将以中继加权值最大的节点作为下一跳，实现局部（虚拟簇）能耗均衡。

在选择中继时，是由本跳节点在下一虚拟簇的簇内节点中选择的，因此它必须知道下一虚拟簇的簇内节点。为此，要求虚拟簇构建完毕以后，由辅助中继将该簇的簇内节点 ID 号存储在一个 $m \times 2$ 的“簇身份矩阵” C_{id} 中，它的每一行表示一个节点的[簇 ID, 节点 ID]，共有 m 个簇内节点。随后，将“簇身份矩阵”传递给上一辅助中继，由它将来自下一跳的“簇身份矩阵”广播给其自身的簇内节点。这样，虚拟簇的所有簇内节点都知道了下一虚拟簇的簇内节点组成情况。因此，无论本簇的哪个簇内节点当选为中继，都有能力在下一虚拟簇中寻找中继节点。

显然，当辅助中继为目标节点时，不需要构建虚拟簇。

3.2 虚拟簇的动态更新

3.1 节中的“簇内真空”可能是如下原因导致的：1) 随着数据转发的进行，簇内所有节点的剩余能量都低于阈值而死亡；2) 物理损毁。无论哪种原因，只要出现“簇内真空”，都意味着所在簇的节点全部死亡，在该区域形成网络空洞和监测盲区。

为此，要求中继节点每发送一个数据分组后就检查自己的剩余能量，如果低于阈值，就利用自己的仅有能量将节点死亡消息沿着数据传递的反向路径报告给源节点（节点死亡报告）。否则，继续判断波束宽度是否超过阈值，如果没有超过，就采用波束扩大法重建虚拟簇；如果超过，要求源节点重新发起一次 Dijkstra 路由请求，以便重新寻找一条辅助路由和一系列辅助中继，用新的辅助中继为圆心构建虚拟簇，称为源头更新法。无论何时，只

要源节点收到节点死亡报告，就需要使用源头更新法重建虚拟簇。

源节点在收到节点死亡报告（假定死亡节点的 $ID = x$ ）的时候，将自己的“距离矩阵”中的元素 $(x, :)$ 和 $(:, x)$ 删除，其中 $(x, :)$ 表示第 x 行的所有元素， $(:, x)$ 表示第 x 列的所有元素。同时，源节点还需要将死亡节点 ID 记录到“死亡节点向量”中，并在新的 Dijkstra 路由请求中包含“死亡节点向量”的内容，以便其他节点知晓网络的最新拓扑结构。其他节点收到该请求后，按照相似的方法更新自身的“距离矩阵”和“死亡节点向量”。

波束扩展法是在原位置通过增大覆盖范围的方式实现的，它实现了局部节点能耗均衡，但是无法跳出局部搜索的陷阱。而源头更新法则在新找到的辅助中继位置周围构建虚拟簇，跳出了原搜索区域，克服了局部搜索的缺陷。不过，在源头更新法中，Dijkstra 算法只根据距离最短的原则搜索辅助中继，如果节点坐标不变且没有节点死亡，新的请求所得到的辅助路由与上一次必然相同，仍然跳不出局部陷阱，还是会导致该区域的节点过早死亡。为此，在源头更新法中用节点剩余能量去调节距离矩阵，使得剩余能量越小的节点，其加权距离越大，被 Dijkstra 算法选择为辅助中继的可能性越小。“加权距离矩阵”中的元素为

$$d(w)_{i,j} = d_{i,j} \exp(E_r/E_o - 1) \quad (10)$$

其中， $d(w)_{i,j}$ 表示节点 i （路由请求发起节点或者中继节点）和节点 j （收到路由请求的节点）之间的加权距离， E_r 为节点 j 的剩余能量， E_o 为其初始能量。从式(10)可以看出， E_r 越小， $d(w)_{i,j}$ 越大，被 Dijkstra 算法选择为辅助中继的概率越小。

有了这样的更新过程之后，在簇内没有节点死亡的情况下，一个簇在随后的 M 次数据发送过程中只需要扩大虚拟簇范围，实现简单，计算容易，减小了寻路开销。一旦出现死亡节点或者波束宽度超过阈值，算法会马上启动源头更新，让网络内的其他剩余能量更大的节点充当数据转发中继，降低了现有簇内节点继续死亡甚至出现大片网络空洞的几率。

4 SaDVC-Routing 的数据转发流程

第 3 节阐述了 SaDVC-Routing 的关键步骤，从中可以看出，波束扩展法实现了局部能量消耗均衡，而源头更新法则保证了全局能量消耗均衡，二者的结合形成了全网的能耗均衡。另外，第 2.1 节的推导过程充分说明，智能天线的使用能够有效地节省能量。综合这 2 个因素可知，动态虚拟簇的方法能够实现能量节省和能耗均衡的联合优化。

需要指出的是，对于智能天线和定向天线，节点在通信之前必须进行波束对准，否则会出现“耳聋问题”，这可以通过 MAC 层的 CTS 和 RTS 数据分组的到达方向进行估算。本文采用 DMAC 协议达到这一目的^[4]。

SaDVC-Routing 的数据转发流程如图 3 所示，分步骤解释如下。

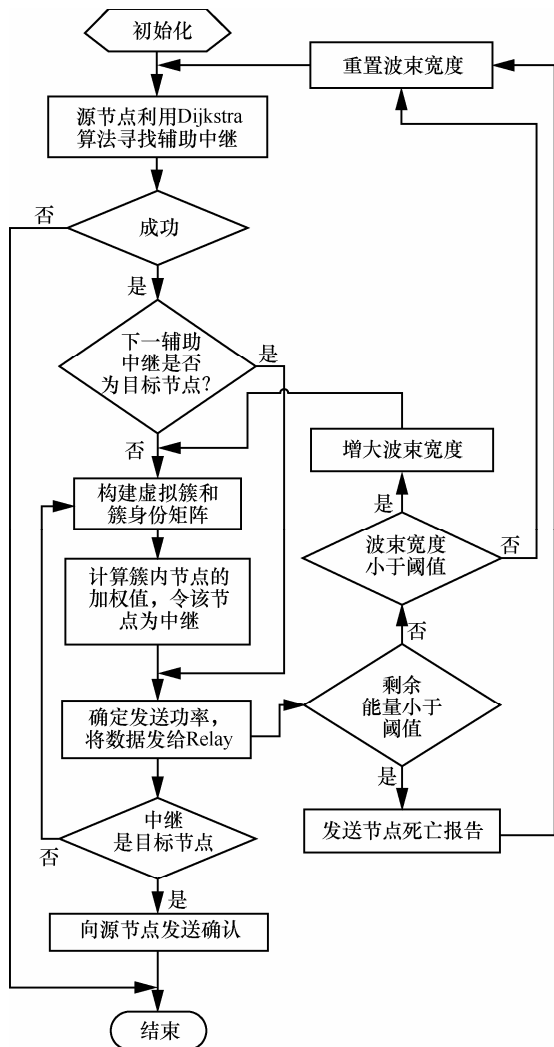


图 3 SaDVC-Routing 的算法流程

1) 初始化：设定 θ 、 $\Delta\theta$ 、 θ_{max} 、 P_0 的初始值，根据网络各节点的坐标设置距离矩阵 D_m 、簇身份矩阵 C_{id} 和死亡节点向量 V_{dead} 为空，设置最小能量等级 P_{min} 、最大能量等级 P_{max} 和 ΔP ，给定 E_{elec} 、 E_{fs} 和 E_0 ，并为数据分组大小 k 和权重调节因子 β 给定初值。

2) 源节点利用 Dijkstra 算法寻找辅助中继和辅助路由，如果失败，就直接退出，否则进入第 3) 步。

3) 如果下一跳就是目标节点，直接发送数据给它；否则以 R_i 为圆心、 $Dis_i \cdot \sin(\theta/2)$ 为半径构建虚拟簇，同时更新簇身份矩阵。

4) 利用式(9)计算中继加权值，选择中继加权值最大的簇内节点为下一跳，记为 Relay。

5) 本跳节点查询“距离矩阵”，确定到达下一跳（即 Relay）的距离，然后根据式(5)确定本次数据的发送功率，将数据发给 Relay。

6) 发送节点发完数据、接收节点收到数据以后，分别监测自己的剩余能量，若低于死亡阈值，转向第 7) 步；否则转向第 8) 步。

7) 本跳节点向源节点发送节点死亡报告，转向第 10) 步。

8) 判断波束宽度是否超过阈值，若超过阈值，转向第 11) 步；否则转向第 9) 步。

9) 将波束宽度值更新为 $\theta = \theta + \Delta\theta$ ，重新构建虚拟簇。

10) 源节点删除距离矩阵中与死亡节点相关的行和列，将死亡节点的 ID 加入死亡节点向量。

11) 重置波束宽度为初始值，发起新的 Dijkstra 路由请求。

12) 如果是目标节点收到数据，就向源节点发送一条确认；否则转到第 3) 步。

5 数值仿真

5.1 仿真设置

在 $1\ 000\ m \times 1\ 000\ m$ 的二维空间上随机部署 150 个节点，随机选择一对源、目标节点，后文的图中用五角星绘制、分别标注为 src 和 dst。仿真中所用到的参数及其初始化值如表 1 所示，其中， $Trans_{max}$ 表示节点在全向模式下的最大发射距离，它决定了能够成为辅助中继的候选节点数量，其他参数在前文中均已交代。与能耗计算无关的参数没有列出。

为了集中研究中继节点的能耗数值和能耗分布情况，假定源节点和目标节点的能耗没有限制。令源节点向目标节点发送 50 000 个数据分组，每发一个数据分组称为一轮。

表 1 仿真参数及其初始值

参数	初值	参数	初值
E_o	0.5 J	$Trans_{max}$	300 m
E_{elec}	50 nJ/bit	θ_{min}	20°
E_{fs}	10 pJ/bit·m ⁻²	θ_{max}	90°
E_{dead}	0.2 J	$\Delta\theta$	10°
β	1	k	500 bit

5.2 结果与分析

为了对比，本文仿真了 3 个算法，即采用智能天线的 SaDVC-Routing、采用全向天线的 Dijkstra-Routing 和采用智能天线的 YANG-Routing^[10]，在此分别称之为场景 1、场景 2 和场景 3。

场景 1: 仿真从第 5 012 轮开始出现节点死亡，到算法结束为止，共死亡 26 个节点，50 000 个数据全部成功发送到了目标节点。运行结束后，死亡节点、存活节点的分布情况如图 4 所示，其中存活节点和死亡节点分别用小圆圈和小十字表示。为了进一步验证场景 1 到底能够成功发送多少数据分组，继续增大发送轮数，发现可以成功发送 107 643 个数据分组。

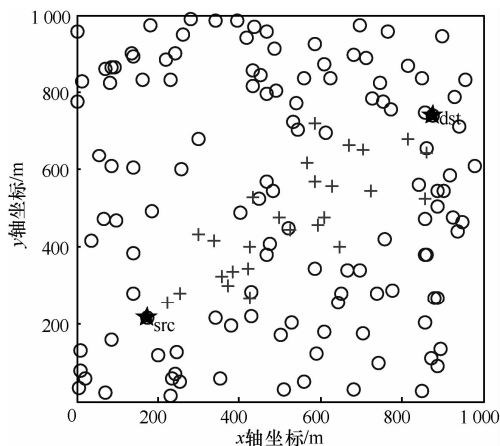


图 4 SaDVC-Routing 运行 50 000 轮后的拓扑

场景 2: 第一个死亡节点出现在第 796 轮，到仿真结束为止，共发送成功 18 780 个数据分组，比场景 1 少得多；共有 62 个节点死亡，是场景 1 的 2.4 倍。这说明，由于采用了全向天线，场景 2 比

场景 1 的能耗大了很多。但是，Dijkstra 路径上的中继节点在能量耗尽之后，能够重新在源、目标节点之间寻找新的路径，不会因为一出现中继节点死亡便无法发送数据的情况，这一点与 SaDVC-Routing 相似。但是，场景 2 必须等到有 Dijkstra 路径节点死亡时才会重新发起新的路径请求，也就是说，一旦找到路径，便要将该路径上的某个节点能量用尽为止，能耗在节点之间很不均匀。场景 2 的死亡节点、存活节点的分布情况如图 5 所示，从中可以看出其死亡节点较多。

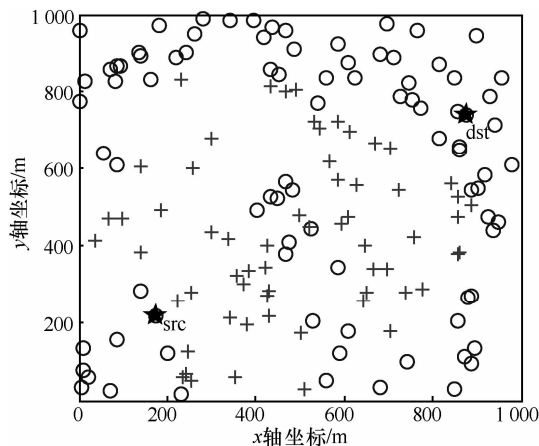


图 5 Dijkstra-Routing 运行 50 000 轮后的拓扑

场景 3: 运行到 4 275 轮时出现第一个节点死亡，随后就不能成功发送数据（即成功发送了 4 275 个数据分组），后续数据全部丢失。到仿真结束时为止，总共只有 5 个节点死亡，如图 6 所示。之所以只有 5 个节点死亡，是因为网络中虽然还有大量节点可用，但是场景 3 却无法在源节点与目标节点之间找到一条可用的路径，相当于网络已经死亡，因此浪费了大量的资源，网络利用率很低。

之所以会有这种结果，根本原因是 YANG-Routing 假定源节点和目标节点的连线上等间距的分布着 $h-1$ 个虚拟中继（图 6 中的小菱形， h 的大小是根据能耗公式对跳数求导得到的），以这些虚拟中继为圆心，寻找距离虚拟中继最近的节点作为真实中继。如果在智能天线的波束范围内找不到真实中继，就换成全向天线模式。很显然，无论是智能天线模式或者全向天线模式，有可能成为真实中继的节点被严格限制在虚拟中继范围内，一旦某个虚拟中继范围内的节点全部死亡，就再也无法寻找可行通信路径。

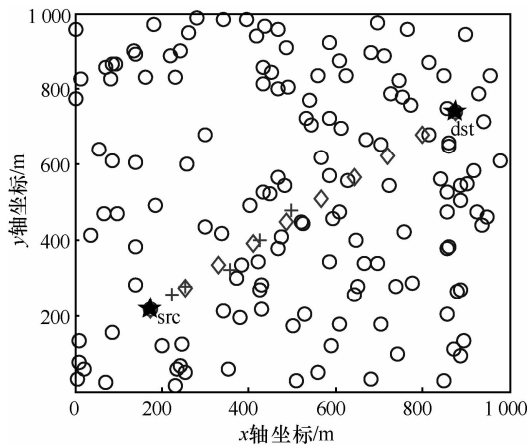


图 6 YANG-Routing 运行 50 000 轮后的拓扑

为了比较，将上述 3 种场景的关键结果用表列出，如表 2 所示。

表 2 3 种场景的部分运行结果

场景	出现死亡轮数	总共死亡节点	总共发送数据	最大能发数据
1	5 012	26	50 000	107 643
2	796	62	18 780	18 780
3	4 275	5	4 275	4 275

下面分析 3 个场景的能耗随运行轮数的变化情况，如图 7 所示。停止点 1 表示场景 2 无法在源节点、目标节点之间找到路径时的运行轮数，发生在第 18 781 轮，从此轮开始，后续数据都无法发送。因此，停止点 1 以后，场景 2 的能耗将不再增长。与此相似，停止点 2 以后，场景 3 的能耗也不再增长。这并不是说明这 2 种算法节省能量，而是说明这 2 种场景情况下的网络死亡得比场景 1 的早，效率较低。

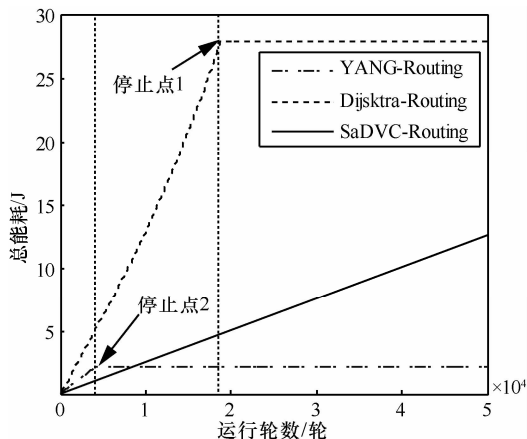


图 7 运行 50 000 轮的能量消耗情况

如果撇开效率不谈而单独研究能耗的话，那么只有对比 3 种场景都能工作时的能量消耗才有意义。为此，分别过停止点 1 和停止点 2 画两条垂直点划线，3 种场景都能工作的部分位于图 7 中过停止点 2 的垂直点划线左边。可以看出，SaDVC-Routing、YANG-Routing 和 Dijkstra-Routing 所对应的能耗曲线越来越陡峭，说明三者的能耗依次增大。Dijkstra-Routing 的能耗曲线近乎于指数上升，而其他 2 种采用了智能天线的算法能耗要小得多，这说明智能天线的能量节省效果非常显著。

通过上面的仿真分析，可以得出如下结论：1) 由于可以动态重构动态虚拟簇以便寻找新的路径，SaDVC-Routing 可以成功发送的数据分组个数远远高于其他 2 个算法；2) 由于在簇内采用了中继加权值，在重新发起路由请求时采用了加权距离，因而实现了局部搜索和全局搜获的结合，使能量在网络节点间均匀消耗；3) 采用智能天线的路由算法，发送同样数据所消耗的能量明显低于采用全向天线的路由算法。

6 结束语

智能天线的功率模型和能耗模型说明，采用智能天线可以明显降低 WSN 能耗。本文引入动态虚拟簇机制，根据网络节点的存活情况动态选取中继节点，在节省能量的同时，实现了能耗在节点之间的均匀分布，明显延长了节点的中继寿命。随着 WSN 从当前的纯全向天线网络向纯智能天线网络过渡，将会产生部分天线是智能天线的异构网络，在后续研究中，将对这种网络的路由方法进行探讨。

参考文献:

[1] ALAWIEH B, ASSI C, MOUFTAH H. Power-aware ad hoc networks with directional antennas: models and analysis[J]. Ad hoc Networks, 2009, 7(3):486-499.

[2] SHI L, FAPOJUWO A O. Cross-layer optimization with cooperative communication for minimum power cost in packet error rate constrained wireless sensor networks[J]. Ad Hoc Networks, 2012, 10(7): 1457-1468.

[3] 胡青松,张申,陈艳等. 一种基于 voronoi 图的能量均衡分簇路由协议[J]. 小型微型计算机系统, 2012, 33(3):457-461.

HU Q S, ZHANG S, CHEN Y, et al. An energy balanced clustering routing based on voronoi-graph[J]. Journal of Chinese Computer Sys-

tems, 2012, 33(3):457-461.

- [4] SPYROPOULOS A, RAGHAVENDRA C S. Capacity bounds for ad-hoc networks using directional antennas[A]. IEEE International Conference on Communications[C]. Anchorage, USA, 2003. 348-352.
- [5] QUINTERO A, LI D Y, CASTRO H. A location routing protocol based on smart antennas for ad hoc networks[J]. Journal of Network and Computer Applications, 2007, 30(2):614-636.
- [6] FONTAN F P, ESPINEIRA P M. Modelling the Wireless Propagation Channel: A Simulation Approach with MATLAB[M]. Chichester:Wiley, 2008.
- [7] HEINZELMAN W B, CHANDRAKASAN A P, BALAKRISHNAN H. An application-specific protocol architecture for wireless microsensor networks[J]. IEEE Transactions on Wireless Communications, 2002, 1(4): 660-670.
- [8] RAMANATHAN R. On the performance of ad hoc networks with beamforming antennas[A]. The 2nd ACM International Symposium on Mobile Ad hoc Networking & Computing[C]. Long Beach, USA, 2001.95-105.
- [9] CAPONE A, MARTIGNON F, FRATTA L. Directional MAC and routing schemes for power controlled wireless mesh networks with adaptive antennas[J]. Ad hoc Networks, 2008, 6(6):936-952.
- [10] 杨光松, 耿旭. WSN 中基于定向天线的节能寻路机制[J]. 计算机工程 2010, 36(22):91-93.
- YANG G S, GENG X. Energy-efficient route finding mechanism based on directional antenna in wireless sensor network[J]. Computer Engineering, 2010, 36(22):91-93.

作者简介:



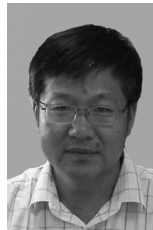
胡青松 (1978-), 男, 四川岳池人, 博士, 中国矿业大学副教授, 主要研究方向为矿山物联网和认知无线电。



吴立新 (1966-), 男, 江西宜春人, 博士, 中国矿业大学教授、博士生导师, 主要研究方向为数字矿山与感知矿山、灾变遥感与灾害观测、空间信息理论与方法、开采沉陷与矿区环境。



张申 (1957-), 男, 江苏兴化人, 博士, 中国矿业大学教授、博士生导师, 主要研究方向为矿山物联网和 CPS 技术。



丁恩杰 (1962-), 男, 山东青岛人, 博士, 中国矿业大学教授、博士生导师, 主要研究方向为矿山物联网和矿山自动化。