

基于节点健康度的无线传感器网络 冗余通路控制方法*

宋佳[†] 罗清华 彭喜元

(哈尔滨工业大学自动化测试与控制研究所, 哈尔滨 150080)

(2014年1月9日收到; 2014年3月10日收到修改稿)

广泛应用于各种物理参数测量领域的无线传感器网络, 因其节点具有能量供应有限、硬件资源有限、数目众多、自组织和动态拓扑等特点, 使得网络极易发生故障, 从而高可靠、低故障是其运行的基本要求. 本文针对多冗余通路设计的无线传感器网络故障预防方法存在工作状态冗余节点过多、能量大量浪费的问题, 提出一种基于节点健康度的冗余通路控制方法. 该方法利用汇聚节点收集网络内所有节点能量状态, 计算节点健康度等相关参数, 使用 A-Star 算法选择最优工作通路, 控制其余冗余通路分批轮流休眠, 从而达到减少和均衡网络工作过程能量消耗、预防某些节点能量提前耗尽导致网络能量故障发生的目的. 仿真实验和实际节点实验的结果表明, 在保证网络适当冗余通路的前提下, 与其他相关方法比较, 该方法可以显著均衡网络能量消耗, 有效预防节点能量故障提前发生, 明显延长网络寿命.

关键词: 无线传感器网络, 节点健康度, 冗余通路控制, 故障预防

PACS: 84.40.Ua, 06.30.-k

DOI: 10.7498/aps.63.128401

1 引言

随着科技水平的迅速发展, 对各种物理参数的无线测量和传输技术越来越受到人们的重视. 无线传感器网络 (wireless sensor networks, WSNs) 具有低成本、低功耗、多跳路由和无线通信的特点, 能够实时监测和收集网络分布区域中各种环境和对象的信息, 可以让用户在任意时间、地点和环境条件下获取详实可靠的物理世界数据, 因此在参数测量领域得到极大关注^[1,2], 并已从国防军事应用广泛扩展到环境监测^[3]、目标追踪^[4]、智能农业^[5]等各种领域. 近年来, 有大量文献研究了 WSNs 的路由技术^[6]、容错协议^[7]、网络安全^[8]等方面的内容.

文献^[9]在 WSNs 领域大量的理论研究与实际应用中发现, WSNs 有易于发生能量问题的情况. 设计者希望 WSNs 节点数量多、体积小、造价低、甚至是使用后可被抛弃. 于是所设计的节点一般都是

计算能力、存储能力比较低, 且使用电池供电, 从而导致节点能量非常受限, 比较脆弱, 容易发生故障, 其中因电池电量耗尽而出现节点失效的状况时有发生. 由于 WSNs 部署时节点数量庞大, 且经常处于无人值守的环境, 人为更换电池通常是无法实现的, 于是就会引起整个或者局部网络故障. 节点因能量耗尽而死亡是一类典型的能量故障^[10], 虽然 WSNs 设计之初即为多冗余的容错系统, 通过布置大量冗余节点来保证系统低故障运行, 但依然无法避免系统能量故障的发生. 因此, WSNs 针对能量故障的故障预防 (fault prevention) 方法非常重要, 需要设计相应方法降低和均衡网络工作过程中的节点能量消耗, 从而有效预防故障发生、延长网络寿命.

目前已有的 WSNs 故障预防方法主要有三类: 预先规划网络部署^[11]、网络状态监测^[12]和设计冗余通路. 其中, 冗余通路的设计是最常用且效果

* 国家自然科学基金 (批准号: 61170262, 61102038) 和国家自然科学基金青年科学基金 (批准号: 51307033) 资助的课题.

[†] 通讯作者. E-mail: songjia@hitwh.edu.cn

很好的方法, 已有大量的文献资料进行研究. 文献[13]提出的 directed diffusion 方法, 主要用于网络的初始阶段构建多条路由通路, 并在节点故障发生时进行路径修复. 源节点广播请求数据包, 邻居节点产生应答数据包, 若请求数据包逐级传递到达汇聚节点, 且汇聚节点的应答数据包回传到源节点, 则建立一条通信路径. 按照此过程工作, 可同时建立多条路由. EAR (energy aware routing) 方法[14]设计次优路径与主路由交替工作, 从而避免某些关键节点因能量提前耗尽而导致网络故障. 冗余通路设计方法在保证 WSNs 成功获得目标区域感知数据的同时, 有效地预防了因某些节点失效而引起的网络故障, 提高了网络的整体工作质量. 但此类方法需要布置大量冗余节点, 且冗余节点处于空闲侦听状态, 甚至与工作节点一样处于工作状态, 消耗了大量能量, 这对于能量非常珍贵的 WSNs 来说是很大的浪费. 于是, 后续有学者研究了节点休眠控制方法, 对冗余节点进行休眠控制.

节点休眠控制方法的核心思想是保证网络连通性的同时, 控制尽量多的节点进入休眠模式, 从而达到节省能量避免节点过早失效的目的. GAF (geographical adaptive fidelity) 方法[15]是根据节点的 GPS 所得到的地理位置划分出多个网格, 每个网格中保留一个工作节点, 命令其他节点休眠; Span 方法[16]是研究节点自决策工作还是休眠的方法. 节点若发现至少有 2 个邻居节点不能直接通信, 就把自身设置为工作节点候选人, 最终选择剩余能量值高和邻居数量多的节点加入网络工作, 其余节点休眠; DSSP (dynamic sleep scheduling protocol) 方法[17]则是把节点剩余能量作为评价节点是否休眠的参数, 先由汇聚节点通过其位置信息计算确定哪些节点是冗余节点, 然后再将剩余能量值低的冗余节点优先控制进入休眠模式. 目前, 已有的节点休眠控制方法也存在缺点. 首先, 选择进行休眠控制的对象是节点而不是一条通路, 在某些极端情况下, 这会破坏网络的连通性; 其次, 通常是对网内所有节点进行控制, 但实际上网络中距离汇聚节点较远的节点能量消耗较小, 对其进行休眠控制意义不大, 反而会增加网络的通信负担.

因此, 本文提出了基于节点健康度的 WSNs 冗余通路控制 (fault prevention technique of controlling redundant routes into sleeping based on health degree, FPCR) 方法, 利用节点工作时间和能量消耗情况计算能耗度参数, 然后融合邻居数目

和能耗度计算得到节点健康度, 使用 A-Star 算法控制汇聚节点两跳范围内的冗余通路轮流休眠和工作. 该方法基于网络能量均衡消耗的原则, 对冗余通路而不是单个节点进行优化分析、选择和控制休眠, 以达到均衡能量消耗、预防节点过早失效导致网络故障的目的.

2 网络模型及分析

2.1 网络模型及相关定义

WSNs 的故障管理体系结构主要包括集中式结构和分布式结构两类, 其中分布式结构因为其优越的性能而得到广泛的应用. 该结构将整个网络分割成多个子区域, 鼓励普通节点参与本地决策, 把故障管理工作量均匀分配到整个网络中, 拥有非常好的网络扩展性和能量使用效率. 文献[18]提出的网格化网络结构是分布式结构的一种特殊形式, 它将网络的子区域规划成虚拟单元格的形式, 可以方便地进行网格分割与合并, 易于自适应 WSNs 网络拓扑的动态变化, 因此其体系结构更清晰, 规则性更好, 网络的可扩展性更强.

本文的理论分析采用与文献[18]类似的网格型分簇网络模型, 如图 1 所示, 假定 n 个传感器节点随机部署在面积为 $L \times L$ (m^2) 大小的区域内, 节点周期性地采集数据发送给汇聚节点 H , 汇聚节点 H 位于监测区域之外, 负责接收感知数据并进行分析处理.

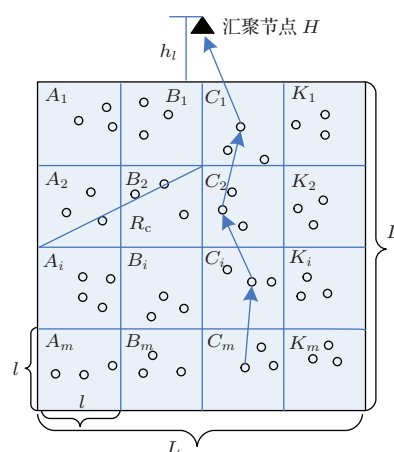


图1 无线传感器网络网格模型

为了便于模型分析和描述, 给出如下假设:

- 1) 传感器节点同构, 初始时在网络中地位、作用相同, 部署后不再移动;

2) 网络中只有一个汇聚节点 H , 固定在节点部署区域之外的某一个点上, 且其能量不受限;

3) 将整个监测区域按照地理位置相对于汇聚节点由近及远均匀分成 m 行 k 列个网格, 每个网格都是边长为 l (m) 的正方形, 距离汇聚节点最远处的网格边长可小于 l ;

4) 节点只能与相邻的网格节点通信, 用 R_c 表示通信半径, 由图可知, 要求 $R_c \geq \sqrt{5}l$;

5) 汇聚节点与最近的 C_1 网格距离为 h_l , 要求 $h_l < l$;

6) 网内存在冗余节点, 汇聚节点通过控制命令可实现分批工作轮流休眠.

2.2 通信数据量分析

2.2.1 最少网格数路径传输模式

最少网格数路径传输模式是指节点数据按照最少网格数路径传输, 如图 2 所示. 在这种模式下, C 区源节点与汇聚节点 H 进行交互数据包传输过程中, 基本沿直线路径、在同一个列网格区域中逐级传递, 不出现向左右两侧的 B 区网格和 D 区网格跳转的情况, 此时路由通路经过的网格数目最少.

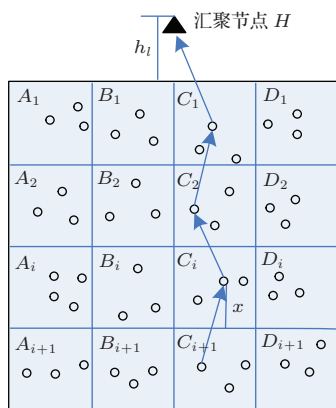


图2 最少路径数传输数据网络结构

位于距汇聚节点第 i 行某个网格中的某个节点 n_i , 距离第 $(i+1)$ 行网格边界距离为 x (m), 则 n_i 与 H 的数据传输距离 D_{is} (m) 可表示为

$$D_{is} = il - x + h_l \quad (i \in [1, m]). \quad (1)$$

节点 n_i 既要接收相邻网格节点传送的数据, 又要把这些数据转发给汇聚节点, 同时还要加入自身的数据. 此处假设节点通信时传送的皆为 b bit 数据.

节点 n_i 可接收数据的区域面积用 S (m^2) 表示, 于是有

$$S = l \cdot \{(l+x) + (2l+x) + \dots + [(m-i)l+x]\} \\ = l \cdot \left[\frac{1}{2}(m-i)(m-i+1)l + (m-i)x \right]. \quad (2)$$

定义 ρ 为网络中节点密度 (m^{-2}), 可知

$$\rho = \frac{n}{L \times L}. \quad (3)$$

定义 $Data_{n_i}^r$ 为节点 n_i 接收的数据量 (bit), 于是有

$$Data_{n_i}^r = S \cdot \rho \cdot b \\ = l \times \left[\frac{1}{2}(m-i)^2 l + \frac{1}{2}(m-i)l \right. \\ \left. + (m-i)x \right] \cdot \rho \cdot b. \quad (4)$$

定义 $Data_{n_i}^t$ 为发送数据量 (bit), 于是有

$$Data_{n_i}^t = b + Data_{n_i}^r \\ = \left\{ 1 + \left[\frac{1}{2}(m-i)^2 l + \frac{1}{2}(m-i)l \right. \right. \\ \left. \left. + (m-i)x \right] \cdot l \cdot \rho \right\} \cdot b. \quad (5)$$

所以, 最少网格数路径传输情况下, 节点的通信数据表达式可写成

$$\begin{cases} Data_{n_i}^r = \left[\frac{1}{2}(m-i)^2 l + \frac{1}{2}(m-i)l + (m-i)x \right] \cdot l \cdot \rho \cdot b \\ Data_{n_i}^t = \left\{ 1 + \left[\frac{1}{2}(m-i)^2 l + \frac{1}{2}(m-i)l + (m-i)x \right] \cdot l \cdot \rho \right\} \cdot b \end{cases} \quad (i \in [1, m], x \in (0, l]). \quad (6)$$

2.2.2 复杂网格数路径传输模式

复杂网格数路径传输模式是指节点按照比较复杂的路由通路传输数据, 如图 3 所示. 在这种模式下, 源节点与汇聚节点进行交互数据过程中, 只经过 1 个列网格区域无法到达目的节点, 会出现向

左侧或者右侧区域网格跳转的情况, 路由通路经过折线传递, 中间经历多于 1 个列网格区域, 路由通路比较复杂.

位于第 i 行 j 列网格中的某个节点 n_{ij} , 距离第 $(i+1)$ 行网格边界距离为 x , 距离第 $(j+1)$ 列网格

边界距离为 y , 则 n_{ij} 与汇聚节点 H 的距离 D_{is} 可表示为

$$D_{is} = (il - x + h_l) + (jl - y) \quad (i \in [1, m], j \in [1, k]). \quad (7)$$

节点 n_{ij} 可接收数据的区域面积亦用 S 表示, 于是有

$$S = l \times \left[\frac{1}{2}(m-i)(m-i+1)l + (m-i)x \right] + l \times \left[\frac{1}{2}(k-j)(k-j+1)l + (k-j)y \right]. \quad (8)$$

采用与最少网格数路径传输情况相同的推导方法, 可得到复杂路径传输情况下, 节点的通信数据表达通式为

$$\begin{cases} \text{Data}_{n_{ij}}^r = \left\{ \left[\frac{1}{2}(m-i)^2l + \frac{1}{2}(m-i)l + (m-i)x \right] + \left[\frac{1}{2}(k-j)^2l + \frac{1}{2}(k-j)l + (k-j)y \right] \right\} \cdot l \cdot \rho \cdot b & (i \in [1, m], j \in [1, k]) \\ \text{Data}_{n_{ij}}^t = \left\{ 1 + \left[\frac{1}{2}(m-i)^2l + \frac{1}{2}(m-i)l + (m-i)x \right] \cdot l \cdot \rho \right. \\ \left. + \left[\frac{1}{2}(k-j)^2l + \frac{1}{2}(k-j)l + (k-j)y \right] \cdot l \cdot \rho \right\} \cdot b & (x, y \in (0, l)). \end{cases} \quad (9)$$

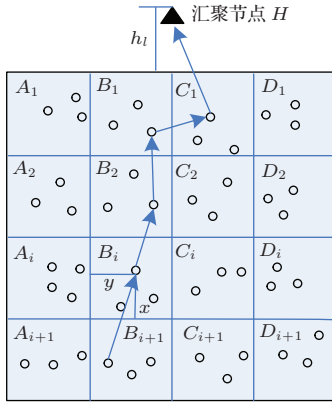


图3 复杂路径传输数据网络结构

如表 1 所示.

表 1 节点通信模型经验参数

参数	数值
$E_{\text{elec}}/\text{nJ} \cdot \text{bit}^{-1}$	50
$\varepsilon_{\text{fs}}/\text{pJ} \cdot \text{bit}^{-1} \cdot \text{m}^{-2}$	10
$\varepsilon_{\text{amp}}/\text{pJ} \cdot \text{bit}^{-1} \cdot \text{m}^{-4}$	0.0013
d_0/m	100

2.3.2 各种传输模式下的节点能量消耗分析

最少网格数路径传输情况下, 将节点 n_i 需要转发的数据量 Data 代入能量消耗公式 (10) 和 (11) 式, 可得节点 n_i 在一轮数据通信过程中的能量消耗:

$$\begin{aligned} E_{\text{Rx}}(b) &= \text{Data}_{n_i}^r \cdot E_{\text{elec}} \\ &= \left[\frac{1}{2}(m-i)^2l + \frac{1}{2}(m-i)l + (m-i)x \right] \cdot l \cdot \rho \cdot b \cdot E_{\text{elec}}, \end{aligned} \quad (12)$$

$$\begin{aligned} E_{\text{Tx}}(b, d) &= \text{Data}_{n_i}^t E_{\text{elec}} + \text{Data}_{n_i}^t \varepsilon_{\text{fs}} d^2 \\ &= \left\{ 1 + \left[\frac{1}{2}(m-i)^2l + \frac{1}{2}(m-i)l + (m-i)x \right] \cdot l \cdot \rho \right\} \\ &\quad \times b \cdot (E_{\text{elec}} + \varepsilon_{\text{fs}} d^2). \end{aligned} \quad (13)$$

将 (12) 式对 i 求偏导数, 可得

$$\frac{dE_{\text{Rx}}}{di} = \left[-(m-i)l - \frac{1}{2}l - x \right] \cdot l \cdot \rho \cdot b \cdot E_{\text{elec}}. \quad (14)$$

2.3 通信消耗能量分析

2.3.1 节点能量消耗模型

本文采用文献 [19] 的 WSNs 节点能量消耗公式, 如 (10) 和 (11) 式所示:

$$\begin{cases} E_{\text{Tx}}(b, d) = bE_{\text{elec}} + b\varepsilon_{\text{fs}}d^2 & (d \leq d_0), \\ E_{\text{Tx}}(b, d) = bE_{\text{elec}} + b\varepsilon_{\text{amp}}d^4 & (d > d_0), \end{cases} \quad (10)$$

$$E_{\text{Rx}}(b) = bE_{\text{elec}}, \quad (11)$$

式中 $E_{\text{Tx}}(b, d)$ 为向距离为 d 的节点发送 b bit 数据消耗的能量; $E_{\text{Rx}}(b)$ 为接收 b bit 数据消耗的能量; E_{elec} 为发射电路损耗能量.

若传输距离小于阈值 d_0 , 功率放大损耗采用自由空间模型; 若传输距离大于阈值 d_0 , 功率放大损耗采用多路径衰减模型. ε_{fs} 和 ε_{amp} 分别为这两种模型中功率放大所需的能量. 各个参数的经验取值

由前述定义可知, $(m-i)$, l , x , ρ , b , E_{elec} 皆为正数, 于是可得

$$\frac{dE_{Rx}}{di} < 0, \quad (15)$$

从而, E_{Rx} 是单调减函数, i 的取值越小, E_{Rx} 值越大, 即距离汇聚节点越近, 接收的数据量越大, 消耗的能量越高.

同理, 将 (13) 式对 i 求偏导数, 可以知道转发数据能量消耗 E_{Tx} 亦为单调减函数, 即距离汇聚节点越近, 转发的数据量越大, 消耗的能量越高.

最少网格数路径传输情况下节点到汇聚节点距离与能量消耗的关系曲线如图 4 所示. 图中横坐标是节点距离汇聚节点的网格数, 纵坐标是能量消耗量, 采用表 1 的参数设置.

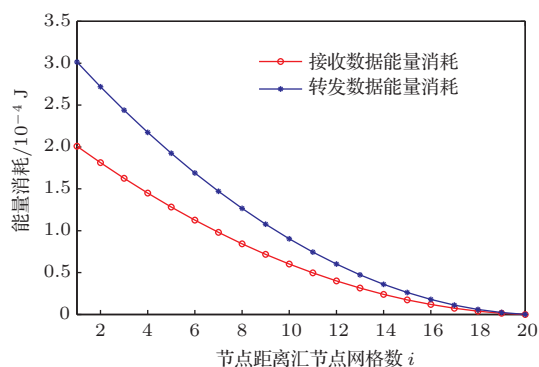


图 4 节点到汇聚节点距离与消耗能量关系 ($m = 20$)

采用相同的分析方法和思路, 可得复杂路径传输情况下节点到汇聚节点距离与能量消耗的关系曲线, 如图 5 所示. 水平方向两坐标表示的是节点距离汇聚节点的网格列数和网格行数, 垂直方向坐

标是能量消耗量, 采用表 1 的参数设置.

2.4 能量消耗对 WSNs 网络寿命影响分析

节点的能量消耗主要包括空闲侦听 $E_{\text{空闲}}$ 、节点休眠 $E_{\text{休眠}}$ 、感知数据 $E_{\text{感知}}$ 、数据处理 $E_{\text{数据处理}}$ 、发送数据 E_{Tx} 和接收数据 E_{Rx} , 由于通信部分能量消耗远大于其他部分, 此处为了分析方便, 做简化处理, 将前四项能量消耗忽略不计.

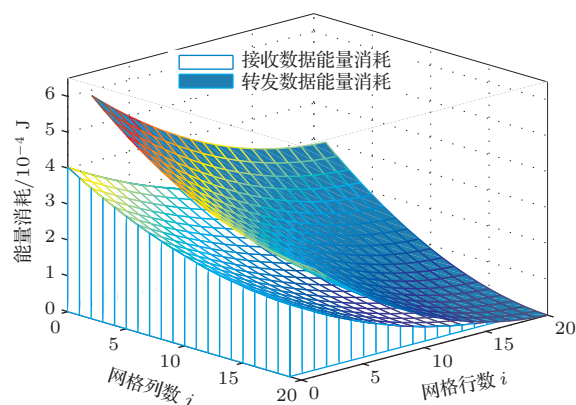


图 5 (网刊彩色) 节点到汇聚节点距离与消耗能量关系 ($m = k = 20$)

定义 T_i 为节点 n_i 的存活寿命, 节点的初始能量为 e_0 (网络中节点同质, 所有节点的初始能量相同), 于是有

$$T_i = \frac{e_0}{E_{Tx}^i + E_{Rx}^i} \quad (i \in [1, m]). \quad (16)$$

以最少网格数路径传输情况为例, 将与汇聚节点距离不同的两个节点的寿命进行比较, 将 (12) 和 (13) 式代入 (16) 式, 可得

$$\frac{T_i}{T_j} = \frac{3 \times \left[\frac{l}{2}(m-j)^2 + \frac{l}{2}(m-j) + (m-j) \cdot x \right] + 2}{3 \times \left[\frac{l}{2}(m-i)^2 + \frac{l}{2}(m-i) + (m-i) \cdot x \right] + 2}. \quad (17)$$

取 $m = 10$ 的网格型网络, 比较距离汇聚节点最近与最远节点的寿命, 即 $i = 1$, $j = 10$, l 和 x 按照之前的约定条件取值, 通过计算得到

$$\frac{T_1}{T_{10}} \approx \frac{1}{3 \times 10^3}, \quad (18)$$

也就是说, 距离汇聚节点最远处的网格中的节点寿命, 是距离汇聚节点最近网格中节点寿命的 10^3 倍以上. 同理, 分析复杂传输路径情况下的节点寿命, 也会得到类似的结果.

由上述分析可以知道, 网络中最先死亡的是距

离汇聚节点最近的节点, 这些冗余节点寿命延长一倍, 就可以将整个网络的寿命延长一倍. 因此, 对汇聚节点较近区域内的节点进行控制, 降低和均衡其能量消耗, 防止过早死亡, 可有效预防网络能量故障的提前发生, 提高整个网络的生存周期. 但是在整个网络死亡时候, 远端节点必然会剩余大量能量, 这些能量即使节省下来也无法有效利用, 因此对远端节点进行休眠控制意义不大, 反而会增加故障管理过程的通信负担和能量消耗. 本文将在此理论分析的基础上进行 WSNs 故障预防方法研究, 设

计相关方法对汇聚节点两跳范围内冗余通路进行轮流休眠控制, 达到故障预防目的.

3 基于节点健康度的WSNs冗余通路控制方法

3.1 参数定义

定义节点能耗度 c , 表示节点 n_i 在空闲侦听和工作状态下, 单位时间内能量消耗与总能量的比值, 其定义式为

$$c_i = \left(\frac{e_0 - e_{ri}}{t_{ci} - t_{si}} \right) \times 10^6 / e_0, \quad (19)$$

式中, c_i 为节点 n_i 的能耗度 (s^{-1}), 且 $c_i \in (0, 1)$; e_0 为节点初始能量 (J); e_{ri} 为节点 n_i 当前剩余能量 (J); t_{ci} 为节点 n_i 当前工作时间 (s); t_{si} 为节点工作期间的休眠时间 (s).

需要说明的是, 节点单位时间内收发数据过程中消耗的能量一般在 10^{-6} J 级别, 本文对 c_i 进行规划处理, 让定义式的分子多乘了 10^6 .

定义节点健康度 h , 表示在规定的条件下和规定时间内, 节点能够保持一定可靠性并稳定持续完

成预定功能的能力, 其定义式为

$$h_i = \alpha \cdot \frac{n_{in}}{N} + \beta \cdot \left(\frac{e_{ri}}{c_i} / e_0 \right), \quad (20)$$

式中, h_i 为节点 n_i 的健康度, 且 $h_i \in (0, \infty)$; n_{in} 为节点 n_i 的所有邻居数; N 为汇聚节点一、二跳范围内所有节点数; α, β 为权衡系数, 且 $\alpha + \beta = 1$.

3.2 FPCR算法原理

FPCR方法是一种集中式控制方法, 其原理如图6所示, 具体描述如下:

- 1) 在网络创建时保证有多条路由通路, 且把全网路由表保存在汇聚节点中, 如图6(a)所示;
- 2) 汇聚节点计算两跳范围内每个节点的能耗度、健康度参数, 然后选择节点健康度之和最大的通路作为工作通路, 命令其他通路休眠, 如图6(b)所示;
- 3) 节点接收到控制指令进入休眠状态, 网络确定最终路由通路开始收集感知数据, 如图6(c)所示;
- 4) 经过设定的休眠时间之后, 所有节点唤醒, 重新计算相关参数确定休眠冗余通路.

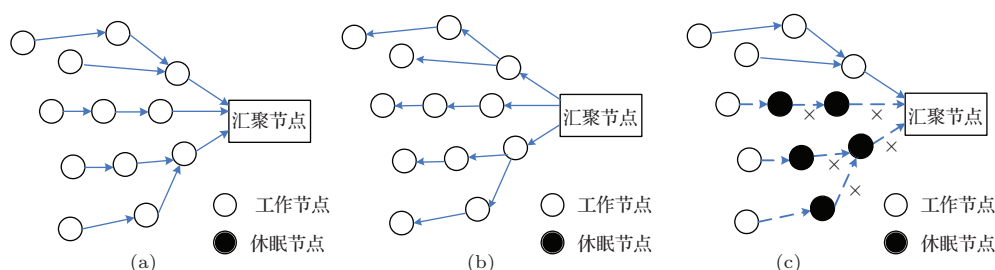


图6 FPCR原理示意图 (a) 源节点发送数据; (b) 汇聚节点发送控制指令; (c) 冗余通路休眠

3.3 问题的数学描述

FPCR算法在进行冗余通路选择时, 希望让性能最好的通路进行工作, 也就是期望在汇聚节点两跳范围之内, 选择节点健康度之和为最大的通路作为数据传输路径. 于是, 这个问题就归结为数学中典型的“最短路径问题”, 只是我们的问题将“最短”取反变成了“最大”.

本文的问题可以描述成如图7所示的一个带权重的图 $G = (V, E)$, 其中, 顶点集合为

$$V(G) = \{1, 2, 3, 4, 5, 6, 7, 8, H\}, \quad (21)$$

集合中的元素为顶点的标号. 边的集合为

$$E(G) = \{(1, 2), (3, 2), (4, 5), (6, 7), (8, 7),$$

$$(2, H), (5, H), (7, H)\}. \quad (22)$$

边的权重定义为节点的健康度 h_i , 则一条路径 $p = n_1 \rightarrow n_2 \rightarrow \cdots n_k$ 的权重为该路径上每条边的权重之和:

$$w(p) = \sum_{i=1}^{k-1} w(h_i, h_{i+1}), \quad (23)$$

也就是说, 求任意一个节点 u 到汇聚节点 H 节点健康度和值最大的路径, 就是求所有路径中权重最大的路径, 即

$$\delta(u, H) = \max\{w(p)\}. \quad (24)$$

A-Star算法是人工智能领域中一种典型的启发式搜索方法, 该方法于1968年提出, 是静态路

网中求解最短路径问题非常有效的方法^[20], 其公式为

$$f(n) = g(n) + h(n), \quad (25)$$

式中, $f(n)$ 为从初始点经由节点 n 到达目标节点的估价函数; $g(n)$ 为从初始节点到节点 n 的代价, 即实际过程节点健康度值; $h(n)$ 为从节点 n 到汇聚节点的估计代价, 本文约束成到汇聚节点的物理空间距离, 算法运行过程中使用节点跳数进行约束.

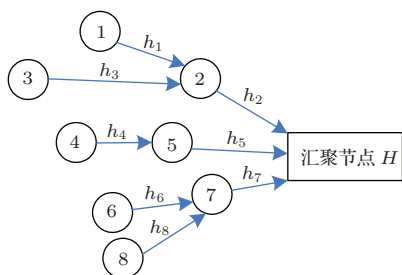


图7 冗余通路选择的图模型

3.4 FPCR算法流程

FPCR算法由如下步骤组成.

步骤1 汇聚节点 H 读取全网路由表, 掌握各条路由通路, 所有节点将剩余能量值、当前时间等数据传送给汇聚节点, 汇聚节点根据 (19) 和 (20) 式, 计算出两跳范围内所有节点的能耗度、健康度参数.

步骤2 汇聚节点确定控制休眠的冗余通路:

1) 汇聚节点创建两个表格 OPEN 和 CLOSED, OPEN 表用于保存所有已生成而未考察的节点, CLOSED 表用于记录已访问的节点;

2) 将汇聚节点 H 作为起点, 放入 OPEN 表中, 令 $g(n) = 0$, $f(n) = 0$, 并将 CLOSED 表设置为空;

3) 选取 OPEN 表中 $f(n)$ 值最大的节点为最佳节点 n_{BEST} , 将其从 OPEN 表中删除, 放入 CLOSED 表中;

4) 若 n_{BEST} 为目标节点, 则搜索成功;

5) 若 n_{BEST} 不是目标节点, 则将与 n_{BEST} 相邻的所有节点扩展开接受检验, 设置节点为 n_{SUC} ;

6) 取一个邻居节点 n_{SUC} , 若其在 CLOSED 表中, 则不进行任何操作, 继续检验下一个节点;

7) 若 n_{SUC} 不在 CLOSED 也不在 OPEN 表中, 则将其添加到 OPEN 表中, 并将其父节点设置为 n_{BEST} , 同时保存 $g(n_{\text{SUC}})$ 值和 $f(n_{\text{SUC}})$ 值;

8) 若 n_{SUC} 在 OPEN 表中, 则按 (26) 式计算

$g(n)$:

$$g(n) = g(n_{\text{BEST}}) + g(n_{\text{SUC}}), \quad (26)$$

并将 $g(n)$ 与原来保存的 g 值比较, 若小于或等于, 则不扩展 n_{SUN} 的邻居节点, 而继续检验下一个已扩展节点; 若大于, 则将 n_{SUC} 的父节点设置为 n_{BEST} , 并保存当前 $g(n_{\text{SUC}})$ 值和 $f(n_{\text{SUC}})$ 值;

9) 循环执行步骤2)—8), 直到 OPEN 表为空, 即无可以添加的新节点, 表明搜索结束, 则结束循环;

10) 从目标节点开始沿父节点遍历, 并保存整个遍历到的节点坐标, 遍历所得节点就是最优路径.

步骤3 汇聚节点确定节点健康度之和最大通路作为工作通路, 广播命令其他冗余通路休眠. 相应节点接收到控制指令后, 中断当前状态进入休眠模式, 网络确定主路由通路, 开始工作收集感知数据.

步骤4 经过设定的休眠时间 t_F 之后, 所有节点唤醒, 重复执行步骤1—步骤3

需要说明的是, 本文使用 A-Star 算法解决的是汇聚节点两跳范围内的最优冗余通路选择, 实际上使用该方法并不局限在解决两跳范围内, 它可以解决任意跳数的节点健康度和值最大的路径选择问题.

4 实验验证与结果分析

4.1 FPCR算法的NS2仿真实验设计

将 FPCR 方法在 Cygwin 环境下的 NS2 2.34 软件平台上进行仿真, 使用 TCL 语言编程实现. 假设有 25 个节点在 $50 \text{ m} \times 50 \text{ m}$ 的区域范围内随机分布, 节点部署后不再移动. 网络使用 AODV 路由协议, Mac 协议使用 802.15.4 协议, 信道类型使用 Wireless Channel, 队列管理机制使用 Drop Tail 机制, 无线传播模型使用 TwoRayGrand 模型, 天线模型使用 OmniAntena 模型, 节点通信模型使用 (12), (13) 式所示的模型. 仿真过程中节点的一些关键参数的设置如表 2 所示.

在相同的网络环境设置下, FPCR 算法、DSSP 方法和无任何节点控制算法 NC (non-control) 的仿真重复运行 30 次, 获得的数据取平均值, 比较这三种算法在网络寿命、第一个节点死亡时间、网络瘫痪后全网剩余能量等方面的表现.

表2 节点关键参数设置

参数	数值
初始能量 e_0/J	1.0
数据包长度/Byte	1024
通信距离 d_0/m	15
感知半径 R_s/m	10
休眠时间 $t_{\text{sleep}}/\text{ms}$	200
工作时间 t_a/s	1.0
算法设计控制时间 t_F/s	10.0
$E_{\text{elec}}/\text{nJ}\cdot\text{bit}^{-1}$	50
$\varepsilon_{\text{fs}}/\text{pJ}\cdot\text{bit}^{-1}\cdot\text{m}^{-2}$	10
$\varepsilon_{\text{amp}}/\text{pJ}\cdot\text{bit}^{-1}\cdot\text{m}^{-4}$	0.0013
α	0.3
β	0.7

4.2 FPCR算法的NS2仿真实验结果与分析

对NC, DSSP和FPCR3种情况进行仿真实验, 最终的效果如图8—10所示. 图中黑色圈形符号表示健康存活的节点, 红色和浅色圈形符号表示濒临死亡和已经死亡的节点.

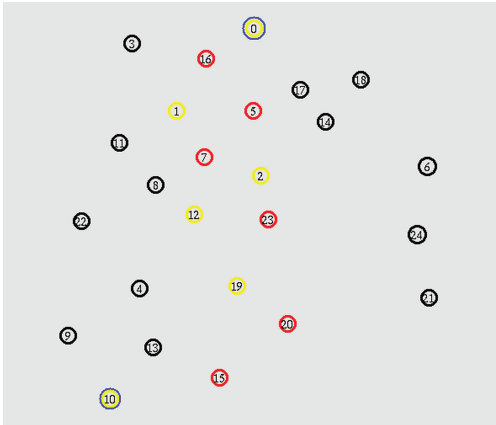


图8 (网刊彩色) NC最后效果图

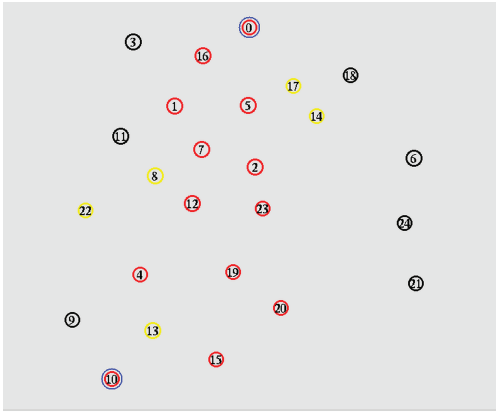


图9 (网刊彩色) DSSP算法效果图

4.2.1 网络寿命分析

仿真实验结果数据列于表3中, 第一个节点死亡时间方面, FPCR方法比NC推迟25.91 s, 性能提高19.88%, 比DSSP方法推迟10.36 s, 性能提高7.10%; 在网络寿命方面, FPCR方法比NC延长110.69 s, 性能提高47.61%, 比DSSP方法延长39.96 s, 性能提高13.18%, 故障预防效果明显. 各种方法性能比较如图11所示, 图中每种方法的左侧柱形表示网络中第一个节点死亡的时间, 右侧柱形表示网络的最终寿命, FPCR方法显示出明显的优势. 网络中第一个节点死亡时间是一个比较重要的评价指标, 通常情况下, 一旦出现节点死亡, 网络中随之会产生“覆盖空洞”问题, 导致网络的覆盖质量QoC (quality of coverage) 下降. 极端情况下, 死亡节点可能会导致整个网络陷入瘫痪, 此时的第一个节点死亡时间就变成了最终网络寿命.

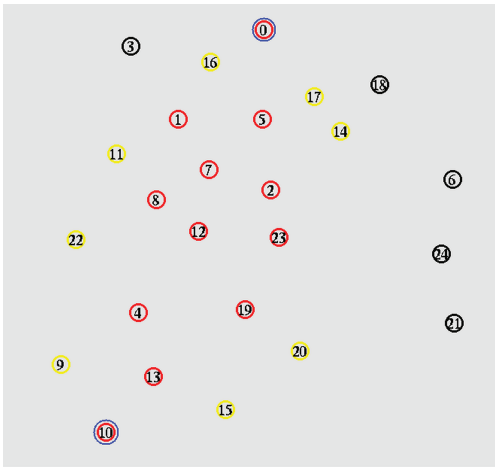


图10 (网刊彩色) FPCR算法效果图

表3 3种方法网络仿真结果比较

	第一个节点死亡时间/s	网络寿命/s
FPCR	156.23	343.17
DSSP	145.87	303.20
NC	130.32	232.48

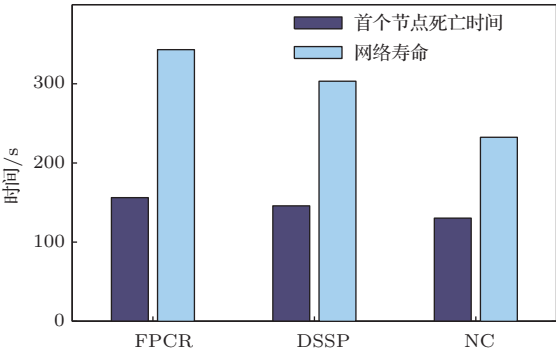


图11 不同算法网络参数比较示意图

4.2.2 剩余能量分析

从前面的理论分析可以知道,网络瘫痪时,网络中应该还有大量存活的节点,仿真实验也验证了这一点,网络瘫痪后各个节点还存在大量能量,对网络内所有节点的剩余能量进行累加计算发现,FPCR方法下全网剩余能量为7.28 J, DSSP技术下全网剩余能量为7.36 J, NC技术下全网剩余能量为8.14 J. FPCR方法对NC方法的能量优化比率为13.64%. 由此可见, FPCR方法使得网络能量消耗更加均衡.网络瘫痪时剩余能量比较情况如表4所示.

表4 几种方法全网剩余能量比较

	全网剩余能量/J	能量优化比率/%
FPCR	7.2756	13.64
DSSP	7.3644	12.59
NC	8.4247	—

FPCR方法中两跳范围内关键节点的剩余能量随时间的变化过程如图12所示,图中横坐标表示网络工作时间,纵坐标表示节点的剩余能量.从图中可以看到,节点3,节点14,节点17和节点18这几个节点的位置靠近边缘区域,邻居数目少,健康度低,被控制休眠的时间比较长,进入网络主路由通路工作的时间比较短,因此剩余的能量值比较高.而节点1,节点2,节点5和节点8这几个节点在网络中间区域,邻居数多,工作时间长,因此死亡较早.

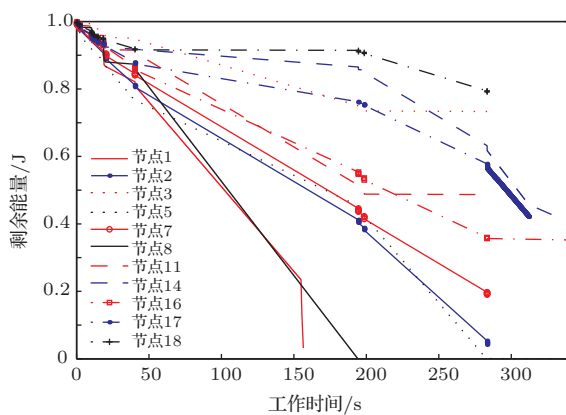


图12 (网刊彩色) 节点剩余能量随时间的变化

4.2.3 节点健康度分析

FPCR方法运行过程中需要定期计算汇聚节点两跳范围内节点的健康度,然后根据此参数使用A-Star算法选择路由通路,本仿真过程中设置的时

间间隔为10 s. 各个关键节点健康度随时间变化的过程如图13所示,图中横坐标表示网络运行时间,纵坐标表示节点的健康度数值.其中被控制休眠冗余通路上的节点,在休眠期间被认为健康度数值不变,于是会出现平行于 x 轴的情况.

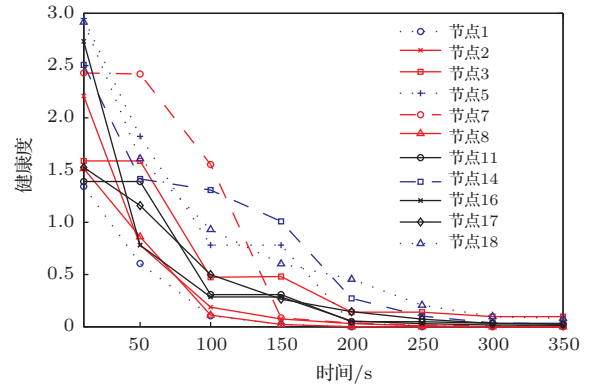


图13 (网刊彩色) 健康度随时间的变化

4.2.4 β 取值对FPCR算法的影响分析

FPCR方法中的节点健康度参数直接决定了控制算法的效果,根据定义式我们知道,需要综合考虑节点能耗度和邻居数目来确定健康度,根据需求选择不同的权值.最理想的情况是选择能耗度最低、邻居数目最多的节点作为路由工作节点,让能耗度高、邻居少的节点休眠.图14所示为随着系数 β 取值的不同,网络寿命的变化情况.图中横坐标表示 β 的取值,纵坐标表示网络寿命.当 β 取0时,表示只考虑邻居数目作为节点健康度参数,效果非常差,网络的寿命比较短;当 β 取1时,表示只把节点能耗度作为健康度计算标准,这与DSSP方法非常类似,取得的效果也不是很理想;当 β 取0.7时,较好地融合了能耗度和邻居数目情况,能耗度所占的比例稍大些,最后的网络寿命比较长,表明控制效果较好.

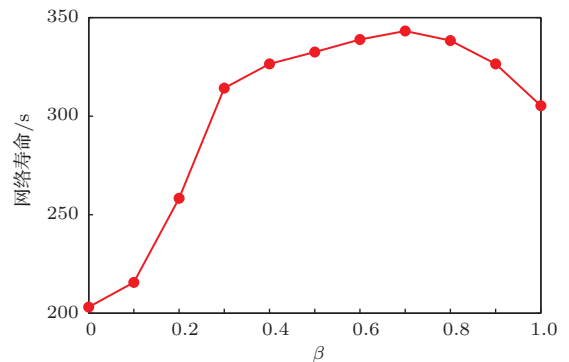


图14 网络寿命随 β 取值不同的变化

4.3 FPCR算法的实际节点实验设计

使用8个节点组成一个3跳的小型网络,其网络结构如图15所示.其中源节点与汇聚节点之间部署6个节点,保证有3条以上的路由通路,路由节点使用电池供电,源节点和汇聚节点都用直流电源供电.

实验中采用的节点是自研的CC2530模块,它是基于TI公司SOC架构的无线传感器网络节点解决方案,如图16所示.网络部署在面积为20 m × 10 m的室内实验室进行,实验现场为室内,如图17所示.实验条件如表5所示.

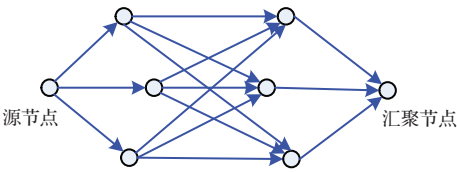


图 15 无线传感器网络结构示意图

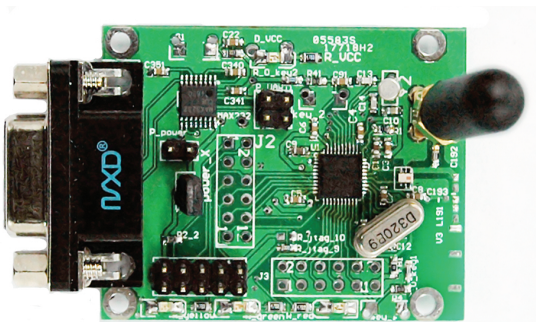


图 16 节点实物

表 5 室内环境下实验参数设置

参数	数值
节点类型	CC2530 节点
温度/°C	26.0
节点放置高度/m	0.75
节点间距/m	5.0



图 17 实验现场

为了减少实验时间,采用加速实验的方式,网络不设置休眠时间,源节点一直处于工作状态,以5 s的时间间隔持续发送数据包,路由节点只有接收到汇聚节点的控制指令才休眠,否则一直处于转发数据状态.路由节点使用2节5号GP超霸碳性电池,初始电压2.8 V,当电压值低于1.5 V时不再传输数据,此时认为节点死亡.

实验中网络节点传送数据包共分两类:源节点发出的感知数据包和路由节点汇报自身状态数据包,其通信格式分别如表6和表7所示.

需要说明的是,为了加快能量消耗,数据包中的感知数据位设计得比较长,实验时未挂接传感器,感知数据位用固定数值代替.当汇聚节点接收不到某路由节点信息时,说明该路由节点已死亡;当接收不到源节点数据时,则说明传输通路已断.根据接收到的路由节点最后一次数据的第5至第8字节可以计算得到工作时间,第9至第10字节可以得到节点当前电压值.

在上述相同的网络结构和实验条件设定下,分别重复进行FPCR, DSSP和NC算法的实验各10次,下面对得到的实验结果进行详细分析.

表 6 源节点发送数据包通信格式

1—2 字节	3—4 字节	5—8 字节	9—39 字节	40 字节
帧头	节点标示	计数位	感知数据	帧尾

4.4 FPCR算法的实际节点实验结果与分析

对重复实验获得的数据取平均值,路由节点死亡时间和死亡顺序如表8所示.当网络中第4个节点死亡时,汇聚节点收集不到源节点的数据,网络陷入瘫痪,这个时间即为网络寿命.

4.4.1 网络寿命分析

根据前面的实验数据,将3种算法的第一个节点死亡时间和网络最终瘫痪时间进行汇总比较,如表9所示.由表中数据可知,第一个节点死亡时间方面,FPCR算法比NC推迟754.33 min,性能提高57.44%,比DSSP方法推迟180.32 min,性能提高9.55%;在网络寿命方面,FPCR 算法比NC延长760 min,性能提高55.48%,比DSSP算法延长175.74 min,性能提高8.99%,故障预防效果明显.

表 7 路由节点转发数据包通信格式

1—2 字节	3—4 字节	5—8 字节	9—10 字节	11—47 字节	48 字节
帧头	节点标示	计数位	节点电压	转发数据	帧尾

表 8 3 种算法中各路由节点死亡时间

NC 算法		DSSP 算法		FPCR 算法	
节点号	死亡时间/min	节点号	死亡时间/min	节点号	死亡时间/min
02	1313.25	05	1887.26	03	2067.58
06	1362.00	06	1909.51	05	2098.08
05	1368.92	03	1949.75	04	2099.49
04	1369.75	02	1954.01	01	2129.75
03	—	04	—	06	—
01	—	01	—	02	—

表 9 3 种方法网络仿真结果比较

	第一个节点死亡时间/min	网络寿命/min
FPCR	2067.58	2129.75
DSSP	1887.26	1954.01
NC	1313.25	1369.75

4.4.2 剩余能量分析

FPCR 算法中各个节点剩余能量随时间的变化过程如图 18 所示, 图中横坐标表示网络工作时间, 纵坐标表示节点的剩余能量. 当节点休眠时, 能量消耗极小, 近似认为剩余能量不变, 因此图中会出现曲线平行于 x 轴的情况. 同时从图中也可以看到, 各个节点的剩余能量变化曲线比较接近, 表明它们的能量消耗比较均匀, 说明 FPCR 算法取得了较好的控制效果, 很好地达到了均衡网络能量消耗、延长网络寿命的目的.

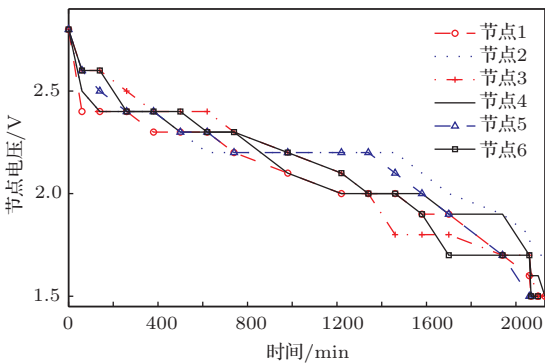


图 18 节点剩余能量随时间的变化

4.4.3 节点健康度分析

FPCR 算法会定时计算二跳范围内关键路由节点的健康度, 然后据此决定控制哪几条冗余通路休眠. 实验中设置健康度的计算时间间隔为

120 min, 使用电池电压值近似代替能量, 得到的各个节点健康度随时间变化的过程如图 19 所示. 图中横坐标表示网络运行时间, 纵坐标表示健康度数值, 6 个节点的健康度数值随时间变化曲线如图所示.

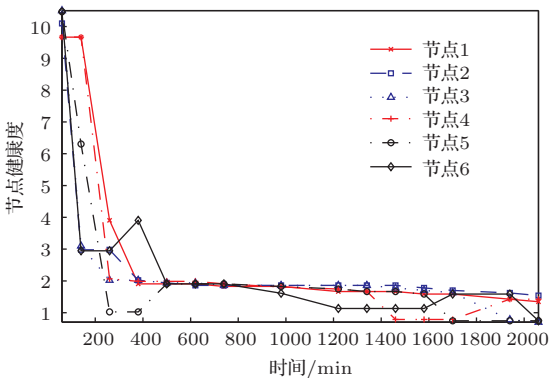


图 19 健康度随时间的变化

另外, 由于实验中采用的节点数目较少, 网络瘫痪时一般只剩 2 个节点仍然存活, 因此整个网络的剩余能量在 3 种算法中差别不大, 在此情况下再进行整体网络剩余能量的比较已经意义不大, 所以此处不再对此项参数进行分析.

5 结 论

本文分析了利用冗余通路进行无线传感器网络故障预防的优缺点, 在此基础上, 提出了一种基于节点健康度的无线传感器网络冗余通路控制方法 FPCR. 该方法利用节点工作时间和能量消耗计算能耗度参数, 并选择适当的权值融合邻居数目和能耗度计算得到节点健康度, 通过 A-Star 算法选择

汇聚节点二跳范围内的最优通路, 控制其他冗余通路轮流休眠, 从而达到控制冗余通路数目、均衡能量消耗的目的, 能够防止由于某些关键节点过早死亡而造成的网络能量故障. 经过NS2仿真实验结果表明, 在相同的网络参数设置下, 相对于DSSP方法和NC方法, FPCR方法可将网络寿命分别延长13.18%和47.61%; 同时, 实际节点实验表明, 与DSSP方法和NC方法相比, FPCR方法可将网络寿命分别延长8.99%和55.48%.

本文研究的FPCR方法面向的是网格型WSNs网络监测场景, 目前的研究还比较理想, 在实际的应用中需要将监测目标区域规范成该数学模型, 要求监测区域形状规则, 而且要掌握节点的位置信息, 由汇聚节点和普通节点协同工作完成划分虚拟单元格工作, 此规划工作比较复杂, 尤其在网络规模较大的时候, 情况会更加困难, 这也是我们今后要进一步详细深入研究的工作.

参考文献

- [1] Wang Y Q, Yang X Y 2012 *Acta Phys. Sin.* **61** 090202 (in Chinese) [王亚奇, 杨晓元 2012 物理学报 **61** 090202]
- [2] Qi H, Wang F B, Deng H 2013 *Acta Phys. Sin.* **62** 104301 (in Chinese) [祁浩, 王福豹, 邓宏 2013 物理学报 **62** 104301]
- [3] Chintalapudi K, Fu T, Paek J 2006 *Internet Comput.* **10** 26
- [4] Zhang C, Fei S M, Zhou X P 2012 *Chin. Phys. B* **21** 1201011
- [5] Vellidis G, Tucker M, Perry C 2008 *Comput. Electron. Agricul.* **61** 44
- [6] Wang Z, Wang Q, Wei D B 2012 *Acta Phys. Sin.* **61** 120505 (in Chinese) [王翥, 王祁, 魏德宝 2012 物理学报 **61** 120505]
- [7] Wang Y Q, Yang X Y 2013 *Chin. Phys. B* **22** 010509
- [8] Tong X J, Zuo K, Wang Z 2012 *Acta Phys. Sin.* **61** 030502 (in Chinese) [佟晓筠, 左科, 王翥 2012 物理学报 **61** 030502]
- [9] Lilia P, Qi H 2007 *J. Network Syst. Manag.* **15** 171
- [10] Ma C, Liu H W, Zuo D C, Wu Z B, Yang X Z 2011 *J. Tsinghua Univ. (Nat. Ed.)* **51** 1418 (in Chinese) [马闯, 刘宏伟, 左德承, 吴智博, 杨孝宗 2011 清华大学学报 (自然科学版) **51** 1418]
- [11] Isler V, Kannan S, Daniilidis K 2004 *IEEE/RSJ International Conference on Intelligent Robots and Systems* Sendai, Japan, September 28–30, 2004 p1780
- [12] Yong J Z, Ramesh G, Deborah E 2002 *Wireless Communications and Networking Conference* Orlando, USA, March 17–21, 2002 p356
- [13] Intanagonwiwat C, Govindan R, Estrin D 2000 *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking* Boston MA, USA, August 6–11, 2000 p56
- [14] Mahfoudh S, Minet P 2010 *Proceedings of the 6th International Wireless Communications and Mobile Computing Conference* Caen, France, May 11–14, 2010 p1126
- [15] Xu Y, Heidemann J, Estrin D 2001 *Proceedings of the 7th International Conference on Mobile Computing* Rome, Italy, July 16–21, 2001 p70
- [16] Chen B J, Jamieson K, Balakrishnan H 2002 *Wireless Networks* **8** 481
- [17] Bulut E, Korpeoglu I 2007 *Proceedings of the 21st International Conference on Advanced Information Networking and Applications Workshops* Niagara Falls, Canada, May 21–23, 2007 p725
- [18] Asim M, Mokhtar H, Merabti M 2008 *International Wireless Communications and Mobile Computing Conference* Crete Island, Greece, August 6–8, 2008 p779
- [19] Heinzelman W B, Chandrakasan A P, Balakrishnan H 2002 *IEEE Trans. Wirel. Commun.* **5** 660
- [20] Keyur R, Mukesh Z 2011 *Commun. Comput. Inform. Sci.* **197** 232

Faultprevention technique of controlling redundant routes into sleeping based on health degree^{*}

Song Jia[†] Luo Qing-Hua Peng Xi-Yuan

(Automatic Test and Control Institute, Harbin Institute of Technology, Harbin 150080, China)

(Received 9 January 2014; revised manuscript received 10 March 2014)

Abstract

Wireless sensor networks (WSNs) are widely used in various measurement fields. However, their nodes are characteristic of constrained energy supply, limited hardware resource, large number of nodes, self-organization and dynamic topology, which make networks prone to malfunction. As a result, high reliability and low fault are the basic requirements of WSNs. To prevent too many active nodes and wasting more energy in redundant routing method for WSNs, a fault-prevention technique of controlling redundant routes into sleeping based on health degree is proposed. This technique uses sink node to collect the residual energy of each node, and to calculate energy consumption degree parameter and health degree parameter, and adopts a-star algorithm to control the redundant routes into sleeping mode. The simulation results of NS2 and actual node experimental results show that the present health degree based technique can balance the networks energy consumption and prolong the network lifetime significantly.

Keywords: WSNs, health degree, redundant routes control, fault prevention

PACS: 84.40.Ua, 06.30.-k

DOI: [10.7498/aps.63.128401](https://doi.org/10.7498/aps.63.128401)

^{*} Project supported by the National Natural Science Foundation of China (Grant Nos. 61170262, 61102038) and the Young Scientists Fund of the National Natural Science Foundation of China (Grant No. 51307033).

[†] Corresponding author. E-mail: songjia@hitwh.edu.cn