

基于目标覆盖的异构有向传感器网络分布式节点调度策略

王力立, 吴晓蓓, 黄成, 丁旭, 王槐元

(南京理工大学 自动化学院, 南京 210094)

摘要: 针对无线传感器网络服务质量会随着网络运行而下降的现象, 研究随机部署的有向传感器网络的节点调度问题, 提出分布式的节点感知方向调节算法, 各节点利用相邻节点间的信息交换, 计算出各自的最佳感知方向, 从而使得网络在满足覆盖需求的同时减少活跃节点数目, 进而达到降低能耗、提高通信质量的目的. 为均衡网络能耗, 进一步设计了冗余节点调度协议, 周期性地重构网络拓扑. 仿真结果表明了所提出算法的有效性.

关键词: 无线传感器网络; 分布式算法; 拓扑重构; 覆盖; 网络生命周期

中图分类号: TP393.02

文献标志码: A

Node scheduling strategy based on target coverage for heterogeneous directional sensor networks

WANG Li-li, WU Xiao-bei, HUANG Cheng, DING Xu, WANG Huai-yuan

(School of Automation, Nanjing University of Science and Technology, Nanjing 210094, China. Correspondent: HUANG Cheng, E-mail: hearthc@163.com)

Abstract: Aiming at the phenomenon that the QoS of the wireless sensor network always decreases with the running of the network, the node scheduling problem of the random deployment directional sensor network is studied. The distributed node orientation adjustment algorithm is proposed to make the network meet the coverage requirement while reducing the number of active sensor nodes, further reducing the energy consumption and improving the communication quality. In the algorithm, each node calculates its best sensing direction in a distributed manner by exchanging coverage information with adjacent nodes. Moreover, distributed redundant node scheduling for network topology reconfiguration is designed to balance the network energy consumption. Simulation results show the effectiveness of the proposed strategy.

Keywords: wireless sensor networks; distributed algorithm; topology reconfiguration; coverage; network lifetime

0 引言

覆盖问题是无线传感器网络中的一个基本问题^[1-2], 继全向传感器网络的覆盖问题得到了广泛研究之后^[3-5], 有向传感器网络的覆盖问题也成为了国内外新的研究热点^[6-7]. 网络的覆盖性能与网络的部署方式密切相关, 相比较于确定性部署方案, 随机部署方案的适用范围更广, 但是也带来了不少应用挑战.

一方面, 因为采用随机投放的部署方法, 随机网络节点呈现随机分布, 节点的利用率远不如确定性网络, 很难一次性覆盖所有目标, 造成信息采集不完整. 因此, 如何改善初始部署后的网络性能, 是一个具有研究意义的问题. 对于随机部署的有向传感器网络, 通过旋转传感器的感知方向改善网络覆盖性能是

一种常见的措施^[6-12]. 针对有向传感器网络目标覆盖问题, 文献[12]引入质心和虚拟力的概念, 规定节点的感知方向会根据其质心所受虚拟力旋转调节, 从而将覆盖问题转化为质心点均匀分布的问题. 由于该方法没有考虑覆盖重叠区域所带来的影响, 网络覆盖性能没有得到充分提高. 为了减少节点的计算量, 文献[13]在文献[12]的基础上进一步提出了成对的切点排斥机制, 但是该机制只关注最邻近的节点, 牺牲了网络覆盖性能. 文献[14]则提出了一种基于虚拟向心力模型旋转调节的覆盖算法, 该算法考虑了覆盖重叠区域的影响, 但是需要持续计算来自所有相邻节点的虚拟力, 计算量较大. 与上述方法不同, 文献[9]利用遗传算法解决有向传感器网络的目标覆盖问题, 该

收稿日期: 2015-10-10; 修回日期: 2016-01-26.

基金项目: 江苏省自然科学基金项目(BK2012803); 江苏省高校优势学科建设工程二期基金项目.

作者简介: 王力立(1987-), 女, 讲师, 博士, 从事无线传感器网络的研究; 吴晓蓓(1958-), 女, 教授, 博士生导师, 从事网络控制、无线传感器网络等研究.

方法虽然可以得到问题的最优解,但需要节点掌握全局信息,计算复杂度高,算法可行性较弱。

另一方面,随着网络的持续运行,会出现网络性能下降,生命周期缩短的问题。文献[14]试图通过关闭冗余节点达到延长生命周期的目的,但是同时关闭多个冗余节点可能会导致覆盖盲区的出现。文献[10]则提出了一种基于线性规划思想的启发算法——Progressive算法,通过寻找一系列传感器方向集合,并为各集合分配工作时间以最大化网络生命。该算法为集中式算法,并不十分适用于计算、通信能力受限的传感器网络。

为此,本文围绕有向传感器网络目标覆盖问题展开研究。针对随机部署的感知能力异构的网络,提出了分布式的节点感知方向调节(DNOA)算法,并在DNOA算法的基础上,提出了分布式的冗余节点调度(DRNS)协议,周期性地调度节点状态,从而均衡网络能耗,延长网络生存时间。最后,通过仿真结果表明了本文算法和协议的有效性,并给出了在部署网络时的一些参数的取值建议。

1 预备知识

1.1 前提假设

传感器的感知能力是指传感器节点感知范围的大小。不同于目前普遍流行的圆盘形全向传感器感知模型,有向传感器的感知角度受限,感知区域为扇形。采用文献[15]的感知模型,用三元组 $\langle r, \mathbf{V}, \alpha \rangle$ 表示,其中: r 是该扇形的半径,表示节点的感知距离; \mathbf{V} 是单位向量,位于该扇形的中轴线上,表示传感器的主感知方向(简称感知方向); α 是该扇形的张角,表示传感器的感知角度。

为了便于研究,这里对待监测目标、传感器节点及网络环境进行假设。

首先,对待监测目标的假设如下:待监测的平面目标区域内随机地分布着 m 个离散目标点,记作集合 $T = \{t_1, t_2, \dots, t_m\}$,假设这些目标点的位置已知,且固定不变。

其次,类似于文献[12-14],对网络环境作如下假设:

1) 网络中共有 L 种感知能力不同的有向传感器节点,即对于任意节点 i ($r_i, \mathbf{V}_i, \alpha_i$),有 $\langle r_i, \alpha_i \rangle \in \{\langle R_j, A_j \rangle, j = 1, 2, \dots, L\}$;

2) 在云台等转向设施的帮助下,传感器节点可以以固定步长 $2\pi/D$ 进行旋转调节,也就是说,传感器节点共有 D 个感知方向可供选择;

3) 传感器节点采用全向通信模型,所有节点具有

相同的通信距离 r_c ;

4) 传感器节点可以通过某种定位算法或者定位装置获得自身的位置信息,并且有能力知道及控制自己的感知方向;

5) 初始网络是连通性的,即所有节点都可以通过多跳转发的方式与sink通信。

1.2 评价标准

本文利用网络总覆盖面积和活跃节点数目评价确定性部署的网络的性能,用覆盖率、活跃节点数目和网络生存时间评价随机部署的网络性能。

1) 覆盖率 c_r 。

覆盖率,又称目标覆盖率,是指被传感器节点覆盖到的监测区域中的目标数与监测区域中所有待监测目标数的比值,即

$$c_r = \frac{\text{card}\left(\bigcup_i T_c^i\right)}{m}. \quad (1)$$

其中:在被监测区域中节点 i 的感知范围覆盖到的目标构成集合 T_c^i ; m 为待监测目标总数;函数 $\text{card}(\cdot)$ 表示对集合求势,也就是集合中元素的个数。覆盖率从整体上直接反映了网络的覆盖性能。

2) 活跃节点数目 n_a 。

传感器节点有活跃和休眠两种状态,活跃节点需要承担网络中的监测和通信工作。同一网络中不同时刻活跃节点的数目可能不同。

对于能量受限的无线传感器网络而言,在满足应用需求的前提下关闭一部分冗余节点,减少活跃节点数量是一项降低网络能耗、延长网络运行时间的有效措施。有时,为使网络的生存时间更长,在不同时间段设置不同的活跃节点是十分有意义的。

3) 网络生存时间 T_{ex} 。

网络生存时间是指网络在能满足特定应用背景中性能需求下的最长运行时间。这里的性能需求可以指失效节点数目、覆盖率、连通情况等。

2 节点感知方向调节算法

2.1 问题描述与相关定义

相比于确定性部署,随机部署网络很有可能出现节点分布不均的现象,导致有的目标无法被覆盖。因此,需要研究如何在云台等辅助设备的协助下,通过重新调节有向传感器节点的感知方向改进初始网络的覆盖性能。针对这一情况,现将该重构问题描述如下。

问题1 已知待监测的平面目标区域内有 m 个离散目标点,记作 $T = \{t_1, t_2, \dots, t_m\}$ 。 n 个感知能力

异构的方向传感器节点被随机布撒在该区域中, 记作 $S = \{s_1, s_2, \dots, s_n\}$. 节点的初始感知方向集合记作 $s_o = \{V_{o1}, V_{o2}, \dots, V_{on}\}$. 节点可旋转, 且旋转步长均为 $2\pi/D$. 求解一个新的节点感知方向集合 $s = \{V_1, V_2, \dots, V_n\}$, 使得被覆盖的目标数 $\text{card}\left(\bigcup_{i=1}^n T_c^i\right)$ 取得最大值.

此外为了提高通信可靠性, 节约能耗, 在问题 1 的基础上引出如下冗余节点的休眠问题.

问题 2 在满足被覆盖的目标数最大的条件下, 寻找 S 的活跃节点子集 S_a , 并确定相应的感知方向集合 s_a , 使得 $\text{card}(S_a)$ 最小.

为了方便说明, 本文引入如下几个相关定义.

定义 1(覆盖邻居) 对于节点 $u, v \in S$, 如果节点 u 与节点 v 的感知范围可以在某个方向上产生重叠, 则称节点 u 和节点 v 是覆盖邻居. 节点 u 的全部覆盖邻居可以用集合表示为

$$N_s(u) = \{v \in S | d(u, v) \leq r_u + r_v, v \neq u\}, \quad (2)$$

其中 $d(u, v)$ 表示节点 u 与 v 之间的欧氏距离.

定义 2(可能覆盖目标矩阵 PCTM) 任一节点的可能覆盖目标矩阵可以用如下 $D \times m$ 维的矩阵表示:

$$A = \begin{bmatrix} a_{11} & \dots & a_{1m} \\ \vdots & \ddots & \vdots \\ a_{D1} & \dots & a_{Dm} \end{bmatrix}. \quad (3)$$

其中: 元素 a_{ij} 是布尔变量, 它表示当节点的感知方向向量处在第 i 个可旋转方向上时, 对目标 j 的覆盖情况, 该值可以利用式 (1) 和 (2) 求得. 如果目标 j 没有被该节点覆盖, 则 $a_{ij} = 0$; 否则 $a_{ij} = 1$.

定义 3(覆盖目标矩阵 CTM) 任一节点在当下的覆盖目标矩阵可以用如下 $D \times m$ 维的矩阵表示:

$$B = \begin{bmatrix} b_{11} & \dots & b_{1m} \\ \vdots & \ddots & \vdots \\ b_{D1} & \dots & b_{Dm} \end{bmatrix}. \quad (4)$$

其中: 元素 b_{ij} 是布尔变量, 规定当且仅当该节点在当前时刻下的感知方向向量正好是第 i 个可旋转方向, 并且目标 j 被该节点覆盖时, b_{ij} 的值取 1. 与 PCTM 不同, CTM 反映了节点当下真实的覆盖情况.

定义 4(覆盖目标行向量 CTRV) 将任一节点在当前的覆盖目标行向量表示为

$$X = (x_1 \ x_2 \ \dots \ x_m), \quad (5)$$

其中元素 $x_j = \sum_{i=1}^D b_{ij}$. 由元素 b_{ij} 的定义可知, x_j 也是布尔向量. 覆盖目标行向量反映了节点在该时刻对

各个目标的覆盖情况, $x_j = 0$ 表示在该时刻目标 j 没有被该节点覆盖, $x_j = 1$ 则表示在该时刻目标 j 被该节点覆盖.

定义 5(可能覆盖目标列向量 CTCV) 将任一节点的可能覆盖目标列向量表示为

$$Y = (y_1 \ y_2 \ \dots \ y_D)^T, \quad (6)$$

其中 $y_i = \sum_{j=1}^m a_{ij}$. 可能覆盖目标列向量反映了节点处在不同可旋转感知方向上时, 可以覆盖到的目标数. 比如, $y_i = 2$ 表示该节点处在第 i 个可旋转感知方向上时, 有两个目标点能被覆盖到. 特殊地, $y_i = 0$ 表示该节点处在第 i 个可旋转感知方向上, 没有任何目标被覆盖到.

2.2 分布式节点感知方向调节算法

为了解决问题 1 和问题 2, 本文提出一种分布式的节点感知方向调节算法——DNOA 算法. 通常, 集中式算法的运行结果要优于分布式算法, 但是分布式算法具有良好的可扩展性, 通信开销较小, 因此更适用于拓扑动态变化的大规模随机网络.

DNOA 算法由初始化阶段和决策阶段组成. 在初始化阶段里, 相邻节点会互相交换一些基本信息, 对于任意两个节点而言, 只有它们互为覆盖邻居才可能覆盖到相同的目标, 所以在初始化阶段, 为了减少通信量, 节点只需要接收覆盖邻居的信息. 初始化算法只在算法刚开始运行时执行一次, 随后进入决策阶段. 在决策阶段, 各节点会根据收集到的覆盖邻居信息选择最佳的感知方向. 具体地规定, 如果某个目标只可能被该节点覆盖到, 则可以覆盖到该目标的方向即为最佳感知方向; 如果不存在这样的目标, 则选择可以覆盖最多有效目标的方向为最佳感知方向, 此处有效目标表示尚未被其他任何节点覆盖到的目标; 如果节点没有最佳感知方向, 则进入休眠状态. 下面以节点 u 为例, 阐述利用 DNOA 算法实现节点感知方向调节的具体过程.

初始化阶段:

1) 节点 u 广播一则 Probe 消息给一跳通信可达的节点. Probe 消息的格式为 (ID, Location, r_u , Timestamp), 分别表示节点的标识号、位置、感知半径和消息发出的时间.

2) 节点 u 接收到别的节点发送来的 Probe 消息, 利用式 (4) 判断出哪些节点是其覆盖邻居, 并组成覆盖邻居集 $N_s(u)$.

3) 节点 u 以自己目前的位置为初始位置, 计算自己在 D 个不同旋转方向上的覆盖情况, 得到 PCTM

矩阵, 记为矩阵 $A(u)$. 用 $B(u)$ 表示 CTM 矩阵, 并且令 $B(u) = A(u)$. 计算相应的 CTRV 向量与 CTCV 向量, 分别记为 $X(u)$ 和 $Y(u)$. 对 $X(u)$ 向量进行简单的归一化处理, 将其中大于 0 的元素改成 1.

4) 节点 u 发送 Coverage 消息给其覆盖邻居集 $N_s(u)$ 中的所有节点. Coverage 消息的格式为 (ID, Status, CTRV, Timestamp). 其中: Status 用于指示节点是否已经做出决策, 值为 1 表示节点已经确定了最终的决策结果, 值为 0 表示尚未进行决策; CTRV 是 u 的 CTRV 向量, 并且动态维护一张覆盖邻居信息表, 用来记录和更新覆盖邻居节点最新发送来的 Coverage 消息.

5) 给节点 u 分配一段等待时延 $\tau_u = T_d \xi_u$. 其中 T_d 是时间常数, 其值远大于决策阶段算法运行和节点传输单位数据所需的通信时间; ξ_u 为取值范围在 $[0, 1]$ 之间的随机因子, 以保证 u 的等待时延及其覆盖邻居的等待时延各不相同. 可见时延短的节点有优先决策权.

决策阶段:

6) 等待时延结束后, 节点 u 根据其在初始化阶段最终生成的覆盖邻居信息表, 计算这些节点的 CTRV 向量之和, 记为 $Z(u)$, 即

$$Z(u) = X(u) + \sum_{v \in N_s(u)} X(v).$$

7) 节点 u 遍历其覆盖邻居信息表, 找出所有 Status = 1 的节点以及它们所覆盖的目标; 然后据此计算出节点 u 处在不同感知方向上时, 可以覆盖到的有效目标数, 用列向量 $Y'(u)$ 表示. $Y'(u)$ 中的元素依次表示节点在各个方向上的有效覆盖目标数.

8) 依次遍历 $Z(u)$ 中的所有元素, 找出值为 1 的元素; 然后, 分成以下 3 种情况进行处理:

① $Z(u)$ 中只有一个元素等于 1. 举例来说, 假设 $z_l = 1 (1 \leq l \leq m)$, 显然, 这代表目标 t_l 只可能被唯一的一个节点的某个或者多个感知方向覆盖到, 此时需要进一步查看 $X(u)$ 中第 l 个元素 x_l 是否非零. 如果 $x_l = 0$, 说明节点 u 无法覆盖目标 t_l , 则找出 $Y'(u)$ 中最大的元素. 假设该元素位于第 g 行, 则节点 u 的最佳感知方向为第 g 个方向. 此时, u 选择了可以覆盖最多有效目标的方向为感知方向, 转入步骤 9). 如果 $x_l = 1$, 说明节点 u 可以覆盖目标 t_l , 则 u 遍历 $A(u)$ 矩阵的第 l 列元素, 找到值为 1 的元素, 然后转入步骤 8).

② $Z(u)$ 中有多个元素等于 1. 依次核对位于 $X(u)$ 中相应位置上的元素是否非零. 如果 $X(u)$ 中不存在满足要求的非零元素, 则节点 u 选择可以覆盖最

多有效目标的方向为最佳感知方向. 如果 $X(u)$ 中不存在满足要求的非零元素, 则说明某些目标只能被 u 唯一覆盖. 任选其中一个目标, 并将节点 u 可以覆盖该目标的任一方向作为最佳方向. 此时问题也就变得与 ① 中的一样, 所以转入 ①).

③ $Z(u)$ 中没有元素等于 1. u 选择可以覆盖最多有效目标的方向为最佳感知方向, 转入步骤 9).

9) 如果 $A(u)$ 的第 l 列中只有一个元素的值为 1, 不妨假设它为 $a_{kl} (1 \leq k \leq D)$, 则节点 u 的最佳感知方向为第 k 个方向. 这表示 u 只有在该方向上才能覆盖到目标 t_l . 如果 $A(u)$ 的第 l 列中有多个值为 1 的元素, 这表示 u 可以在不止一个方向上覆盖到目标 t_l , 则比较各方向在 $Y'(u)$ 中对应行上的元素大小. 节点 u 的最佳感知方向即为对应元素值最大的方向, 也就是可以覆盖有效目标最多的方向, 转入 10).

10) 如果节点 u 在最佳感知方向上可以覆盖的有效目标数为 0, 则认为 u 没有最佳感知方向, 被认为是冗余节点, 准备进入休眠状态.

11) 根据最终的决策结果, 更新 CTRV 向量和状态值 Status, 然后向覆盖邻居发出新的 Coverage 消息. 退出 DNOA 算法.

在上述算法中, 因为节点在不同的旋转方向上的覆盖范围可能会有重叠, 为了能够在其后的步骤中快速找到只有节点 u 才能覆盖到的目标, 必须对 $X(u)$ 进行归一化处理, 而其收到的覆盖邻居的 CTRV 向量则无需归一化.

步骤 5) 中, 给节点 u 设置一个随机等待时延相当于给各个节点分配了不同的优先级. 等待时延越短意味优先级越高, 节点就可以优先地作决策. 而等待时延长的节点则要在等待时延短的节点作完决策并发出新的 Coverage 消息后才能进入决策阶段. 为了确保等待时延比 u 短的覆盖邻居节点已经作好决策, 同时等待时延比 u 长的覆盖邻居节点必须在 u 作好决策之后才能进入决策阶段, 从而使得 u 的覆盖邻居信息表在其决策阶段保持固定不变, 这里简单地规定 T_d 的值远大于决策阶段算法运行时间和节点传输单位数据所需的通信时间. 当 T_d 不满足此条件时, 可以采用回退决策的机制确保等待时延比 u 短的节点先作决策. 具体而言, u 在 τ_u 结束后并不直接进入决策阶段, 而是先向覆盖邻居发出 Apply 消息进行决策申请, Apply 消息的格式为 (ID, τ_u , Timestamp). 如果有等待时延小于 τ_u 的节点在收到 Apply 消息时尚未完成决策, 则该节点否决 u 的决策申请, 并告知 u . u 要等待一个固定的回退时延后重新发出决策申请.

只有通过决策申请, u 才能进入决策阶段.

2.3 DNOA 算法分析

定理 1 DNOA 算法可以在有限的时间内结束.

定理 2 不带回退机制的 DNOA 算法的信息复杂度为 $O(n)$, 带回退机制的 DNOA 算法的信息复杂度的下限为 $O(n)$, 而它们的时间复杂度在最坏情况下均为 $O((D+n+nD)m)$.

此处, 定理 1 和定理 2 的证明不再赘述.

3 分布式冗余节点调度协议

异构方向传感器网络被随机地部署之后, 各个节点通过运行一次 DNOA 算法确定自己的最佳感知方向, 并判断自己是否是冗余节点, 即有效覆盖目标数为零的节点. 如果是冗余节点, 则在发消息通知覆盖邻居节点之后进入低功耗休眠状态; 非冗余节点则旋转至各自的最佳感知方向, 并保持活跃的工作状态. 然而, 随着网络的持续工作, 这些经由 DNOA 算法选择出来的活跃节点将会面临能量耗尽而失效的问题. 如果在出现节点失效时, 能够充分调度休眠的冗余节点来重构网络拓扑, 则可以延长网络的工作时间, 有效提高网络的容错性和自适应能力. 基于以上考虑, 本节基于 DNOA 算法提出一种分布式的冗余节点调度协议——DRNS 协议.

DRNS 协议将网络运行时间划分为若干间隔相等的周期, 称为轮. 每一轮由重构阶段和运行阶段构成. 在重构阶段, 每个节点都会被设置为活跃状态, 并且运行 DNOA 算法确定自己的状态和最佳感知方向. 在运行阶段, 依据 DNOA 算法的运行结果, 活跃节点会在最佳感知方向上继续工作, 而休眠节点则会一直保持低功耗休眠状态直至下一轮重构阶段的到来.

具体而言, 在 DRNS 协议的重构阶段, 为了能兼顾节点能耗的均衡性, 避免部分节点因连续工作时间过长、能耗过大而提前失效, 引入 DNOA 算法步骤 5) 中的等待时延

$$\tau_u = T_d \left(\beta_1 \frac{1}{E_u} + \beta_2 \xi_u \right). \quad (7)$$

其中: T_d 为时间常数, 其值远大于于节点传输单位数据所需的通信时间; E_u 为节点 u 当前的剩余能量; ξ_u 为取值范围在 $[0, 1]$ 之间的随机因子, 以防覆盖邻居中多个节点时延相同; β_1 和 β_2 为可调权重参数. 可见剩余能量越多的节点等待时延越短, 它被唤醒的优先级就越高. 这是因为让剩余能量多的节点优先选择最佳感知方向, 可以使得更多的目标被剩余能量多的节点覆盖到, 从而大大减少了剩余能量少的节点需要覆

盖的目标数. 通过这样的时延设置, 剩余能量多的节点很可能被 DNOA 算法选择为活跃节点, 负责在当期轮中完成网络的监测任务; 而剩余能量少的节点则更有可能被判定为冗余节点, 在该轮进入休眠状态来节约能量. 在下一轮的重构阶段里, DNOA 算法会根据最新的节点能量消耗情况重新分配等待时延, 从而实现网络节点能耗的动态均衡.

在 DRNS 协议中, 网络运行周期(即每一轮的时间长度)是一个非常重要的时间参数. 一轮的时间如果设置得太短, 则网络重构会过于频繁, 这反而加大了网络能耗. 但是, 每一轮的时间也不可以设置得过长, 它必须远小于节点的生命周期, 否则将失去网络重构的意义. 所以, 轮的长度会影响到 DRNS 协议的使用效果. 假设节点的初始能量为 E_{in} , 重构一次的平均能耗为 \bar{E}_{re} , 正常工作时单位时间的平均能耗为 \bar{E}_T , 则 E_{in}/\bar{E}_{re} 的值越小, 轮的长度需要设置得越长; E_{in}/\bar{E}_T 的值越小, 轮的长度需要设置得越短.

4 仿真分析

本文通过以下几组仿真实验来检验上述算法的性能, 并与已有的算法进行比较. 首先, 分析随机部署下 DNOA 算法的参数选择对其性能的影响; 然后, 验证 DRNS 协议的有效性. 本文仿真均基于 Matlab 平台, 为了更好地反映统计规律, 所有结果都是多次仿真结果的平均值.

4.1 DNOA 算法

在下面的仿真中, 假设被监测区域是一个 $(200 \times 200) \text{ m}^2$ 的二维平面, 上面随机地分布着 450 个离散目标点, 以及 n 个有向传感器节点, n 越大意味着网络规模越大. 节点感知距离为 $10 \sim 20 \text{ m}$ 之间的随机值, 感知角度为 $\pi/3, 2\pi/3, \pi$ 中的任意值. 图 1 和图 2 分别表示当节点的可旋转方向数 $D = 1, 2, 3$ 时, 运行 DNOA 算法所得的覆盖率 c_r 以及活跃节点数 n_a 随初始部署节点数的变化情况.

由图 1 可以看出, 在 $D = 1$ 的情况下, n 达到 300 之前 c_r 随着 n 的增加快速增大, 但是 n 达到 300 之后 c_r 趋近饱和, 接近最大值 1. 类似地, 在 $D = 2, 3$ 的情况下, c_r 会分别在 n 达到 250 和 200 之后趋近饱和. 显然, 可旋转方向数越多, DNOA 算法的覆盖性能越好. 此外, 利用 Matlab 优化工具箱, 图 1 中还将 DNOA 算法所得的覆盖率与使用集中式整数规划方法 (IP) 求得的最优解进行了对比. 可以看到, DNOA 算法的分布性并没有对问题 1 的求解造成太大的影响.

由图 2 可以看出, $D = 1$ 时所需的活跃节点最多, $D = 3$ 时所需的活跃节点最少. 具体而言, 在 D

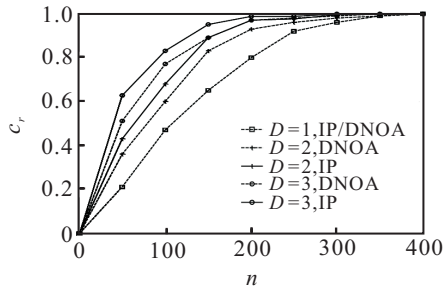


图 1 覆盖率与总节点数的关系

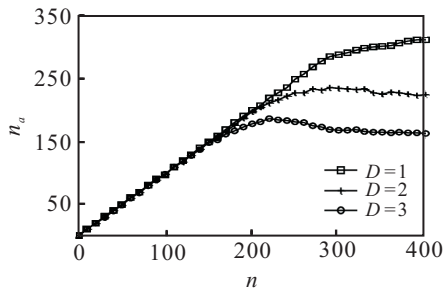


图 2 活跃节点数与总节点数的关系

= 1 的情况下, 当 $0 \leq n \leq 240$ 时, 活跃节点数 n_a 会随着 n 的增加而线性增加, 而且 $n_a \approx n$. 也就是说, 几乎所有的节点都是活跃节点. 此后, n_a 的增加开始变缓, 并且 $n_a < n$. 在 $D = 2, 3$ 的情况下, 分别当 $0 \leq n \leq 180$ 以及 $0 \leq n \leq 140$ 时, n_a 会随着 n 的增加而线性增加, 此后 n_a 的增加变缓, 甚至会有所下降, 直至最终收敛. 这主要是因为, 此时已经不需要所有的节点工作, 更多的初始部署节点反而可以令算法作出更优的决策, 从而使得 n_a 减小.

由上面的仿真研究可以看出, 在相同的初始部署条件下, 节点的可旋转方向数越多, 网络的覆盖率越高, 活跃节点数越少. 而节点的可旋转方向数不变时, 则可以通过调整初始部署节点数 n , 在得到满意的覆盖性能的同时降低活跃节点数.

4.2 DRNS 协议

本节主要研究 DRNS 协议对网络生存时间的影响, 并与文献 [10] 中的 Progressive 算法进行对比. Progressive 算法是基于线性规划思想的一种集中式启发算法, 通过寻找一系列传感器方向集合, 并为各集合分配工作时间以最大化网络生命. 仿真中, 规定节点的可旋转方向数 $D = 3$, 单个节点的初始生存时间为 10, 每一轮的时间长度设置为 2, 初始部署节点数 $n = 200, 400, 600, 800, 1000$. 为方便研究, 假设重构节点所需的能耗远小于正常运行时的能耗, 可忽略不计; 且将网络从初始运行到有一半的节点失效的时间视为网络生存时间. 其他仿真参数与 4.1 节相同.

图 3 给出了 DNOA 算法以及等待时延取不同值的 DNOA 算法 +DRNS 协议运行时的网络生存时间. 其中

$$\tau_{u1} = T_d \left(0 \times \frac{1}{t_{ex}(u)} + \xi_u \right),$$

$$\tau_{u2} = T_d \left(0.8 \times \frac{1}{t_{ex}(u)} + 0.2\xi_u \right),$$

这里 $t_{ex}(u)$ 表示节点 u 的剩余生存时间. 显然, τ_{u1} 没有考虑节点的剩余生存时间, 而 τ_{u2} 考虑了节点的剩余生存时间. 由图 3 可以看出, 由于缺少调度协议, 运行一次 DNOA 算法得到的网络生存时间始终不变, 等于单个节点的生存时间. DNOA 算法 +DRNS 协议运行的网络生存时间则随着 n 的增加而显著增加; 并且, 等待时延等于 τ_{u2} 时的 DNOA 算法 +DRNS 协议运行的网络生存时间要长于等待时延等于 τ_{u1} 时的网络生存时间.

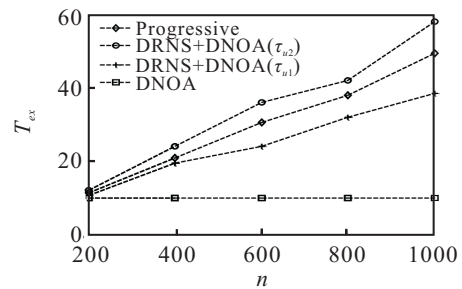


图 3 网络生存时间与节点数的关系

总之, 通过在每一轮的重构阶段里实施冗余节点调度, 尤其是根据节点能量状况进行调度, 显著延长了网络的生存时间, 而且网络初始部署的节点越多生存时间提高的也越多. 此外, 通过对比发现, 在相同网络规模下, 本文方法所获得的网络生存时间总体长于 Progressive 算法, 这主要是因为本文算法充分考虑了节点剩余能量.

4.3 网络连通性

本节主要分析 DRNS 协议对网络连通性能的影响, 并与 Progressive 算法进行对比. 设置初始部署节点数 $n = 400$, 等待时延为 τ_{u2} , 节点通信半径 $r_c = 30$ m, 其他参数与 4.2 节相同. 表 1 显示了在两种不同算法下, 可与 sink 通信的节点的比例随时间的变化情况. 不难发现, DRNS 协议对维护网络的连通性能有一定的优势, 这是 DRNS 可以均衡节点生存时间的必然结果.

表 1 连通节点所占比例的比较 %

算法	轮 数			
	10	20	30	40
DRNS	96.4	62.8	41.8	0.5
Progressive	97.1	57.3	29.6	0.4

5 结 论

考虑到随机网络的拓扑特点和部署条件的限制, 本文提出了分布式的节点感知方向调节算法——DNOA 算法, 并且分析了该算法的相关性质. 该算法

利用节点的覆盖目标矩阵求得各自的最佳感知方向, 以及判断自己是否需要休眠. 为了避免部分节点因为持续工作而能量耗尽, 又在 DNOA 算法的基础上, 提出了分布式的 DRNS 协议, 用来周期性地调度节点状态, 从而均衡网络能耗, 延长网络生存时间. 最后通过具体仿真案例对本文提出的算法和协议进行了分析和验证. 如何在满足覆盖性能的同时, 提高网络的连通性能将是下一步的研究方向.

参考文献(References)

- [1] Wang B. Coverage problems in sensor networks: A survey[J]. ACM Computing Surveys (CSUR), 2011, 43(4): 32.
- [2] Song Y, Liu L, Ma H, et al. A biology-based algorithm to minimal exposure problem of wireless sensor networks[J]. IEEE Trans on Network and Service Management, 2014, 11(3): 417-430.
- [3] Mohamed Y, Sookyoung L, Gupta, S, et al. A localized self-healing algorithm for networks of moveable sensor nodes[C]. 2008 IEEE Global Telecommunications Conf(GLOBECOM 2008). New Orleans, 2008: 1-5.
- [4] Neelofer Tamboli, Mohamed Y. Coverage-aware connectivity restoration in mobile sensor networks[J]. J of Network and Computer Applications, 2010, 33(1): 363-374.
- [5] Sahoo P K, Liao W. HORA: A distributed coverage hole repair algorithm for wireless sensor networks[J]. IEEE Trans on Mobile Computing, 2014, 14(7): 1397-1410.
- [6] Guvensan M A, Yavuz A G. On coverage issues in directional sensor networks: A survey[J]. Ad Hoc Networks, 2011, 9(7): 1238-1255.
- [7] 孙泽宇, 伍卫国, 王换招, 等. 无线传感器网络基于参数可调增强型覆盖控制算法[J]. 电子学报, 2015, 43(3): 466-474.
(Sun Z Y, Wu W G, Wang H Z, et al. An enhanced coverage control algorithm for wireless sensor networks based on adjustable parameters[J]. Acta Electronica Sinica, 2015, 43(3): 466-474.)
- [8] Zhibo W, Qing C, Hairong Q, et al. Achieving k-barrier coverage in hybrid directional sensor networks[J]. IEEE Trans on Mobile Computing, 2013, 13(7): 1443-1455.
- [9] Joon-Min G, Youn-Hee H. A target coverage scheduling scheme based on genetic algorithms in directional sensor networks[J]. Sensors, 2011, 11(2): 1888-1906.
- [10] Yanli C, Wei L, Minglu L, et al. Target-oriented scheduling in directional sensor networks[C]. The 26th IEEE Int Conf on Computer Communications(IEEE INFOCOM 2007). Barcelona, 2007: 1550-1558.
- [11] Wang Y C, Chen Y F, Tseng Y C. Using rotatable and directional (R&D) sensors to achieve temporal coverage of objects and its surveillance application[J]. IEEE Trans on Mobile Computing, 2012, 11(8): 1358-1371.
- [12] Tao D, Ma H D, Liu L. Virtual potential field based coverage-enhancing algorithm for directional sensor networks[J]. J of Software, 2007, 18(5): 1152-1163.
- [13] Huang H, Sun L, Wang R, et al. A novel coverage enhancement algorithm for image sensor networks[J]. Int J Of Distributed Sensor Networks, 2012, 9(3): 184-195.
- [14] Jing Z, Jian-Chao Z. A virtual centripetal force-based coverage-enhancing algorithm for wireless multimedia sensor networks[J]. IEEE Sensors J, 2010, 10(8): 1328-1334.
- [15] 王力立, 徐志良, 吴晓蓓, 等. 感知能力异构的无线传感器网络优化部署机制[J]. 南京理工大学学报: 自然科学版, 2012, 36(3): 408-413.
(Wang L L, Xu Z L, Wu X B, et al. Optimal Sensor Deployment Scheme in sensing-heterogeneous Wireless Sensor Networks[J]. J of Nanjing University of Science and Technology: Nature Science, 2012, 36(3): 408-413.)

(责任编辑: 孙艺红)