

## 利用移动内点来修复传感器网络空洞的算法

杨凯<sup>1</sup>, 刘全<sup>1</sup>, 张书奎<sup>1,2</sup>, 李瑾<sup>1</sup>, 翁东良<sup>1</sup>

(1. 苏州大学 计算机科学与技术学院, 江苏 苏州 215006;  
2. 南京大学 计算机软件新技术国家重点实验室, 江苏 南京 210093)

**摘要:** 针对目标区域中存在感知空洞问题, 提出了一种空洞修复准则。在此基础上, 设计了基于移动节点无需地理信息的修复算法 SOI, 该算法通过计算空洞边缘可移动的最佳内点, 使该边缘节点到达新的位置, 实现减少目标区域中空洞面积的目的。分析与仿真实验表明, 与 VHR 算法相比, SOI 算法在部署密集传感网络中需要移动的总距离更少。

**关键词:** 传感器网络; 覆盖空洞; 空洞修复; 移动内点

中图分类号: TP393

文献标识码: A

文章编号: 1000-436X(2012)09-0116-09

## Hole recovery algorithm based on mobile inner nodes in wireless sensor networks

YANG Kai<sup>1</sup>, LIU Quan<sup>1</sup>, ZHANG Shu-kui<sup>1,2</sup>, LI Jin<sup>1</sup>, WENG Dong-liang<sup>1</sup>

(1. Institute of Computer Science and Technology, Soochow University, Suzhou 215006, China;  
2. State Key Lab for Novel Software Technology, Nanjing University, Nanjing 210093, China)

**Abstract:** To solve the problem of coverage holes in target areas, a principle for hole recovery was proposed. Based on the principle, a novel algorithm named SOI was proposed for hole recovery using mobile nodes, which does not require location information, figures out the optimal inner nodes in the edge of coverage holes, and then moves them to new positions. Through this mechanism, the area of coverage holes could be reduced. Analysis and simulation results show that, SOI has a better performance in dense deployed wireless sensor networks compared with existing VHR algorithm.

**Key words:** wireless sensor networks; coverage hole; hole recovery; mobile inner nodes

### 1 引言

无线传感器网络是当前研究的热点, 通过向目标区域部署大量价格低廉、具有移动、感知和通信能力的传感器节点构成的无线传感器网络,

可广泛应用于军事、交通、医疗和救灾等领域。随着电子技术的不断发展, 传感器节点的功能不断增强, 体积不断缩小, 使得利用无线传感网络完成对某一区域的监测成为可能。但是由于节点部署不均匀、环境外力影响或电量耗尽等因素而

收稿日期: 2011-05-05; 修回日期: 2011-10-06

基金项目: 国家自然科学基金资助项目(61070122, 61070223, 61070169, 61272005); 江苏省自然科学基金资助项目(BK2012616, BK2009116); 江苏省高校自然科学研究基金资助项目(09KJA520002); 苏州市应用基础研究计划基金资助项目(SYG201034); 中国高等学校博士学科点专项科研基金资助项目(20103201110018)

**Foundation Items:** The National Natural Science Foundation of China(61070122, 61070223, 61070169, 61272005); The Natural Science Foundation of Jiangsu Province (BK2012616, BK2009116); The High School Natural Foundation of Jiangsu Province (09KJA520002); Application Foundation Research of Suzhou of China (SYG201034); Specialized Research Foundation for the Doctoral Program of Higher Education of China (20103201110018)

产生感知区域大型空洞，影响感知信息的精确性或降低通信的效率。因此，如何探测这些空洞区域，并对之进行修复成为传感器网络研究的重要内容之一。

空洞修复算法包括2部分：空洞探测和空洞修补。在空洞探测方面，文献[1]提出一种使用 Voronoi 图来发现空洞。文献[2]提出了  $K$  覆盖( $K$ -coverage) 算法，使用计算几何中覆盖弧的相关性质进行空洞探测。在空洞探测中很少研究以地理信息作为辅助<sup>[3~6]</sup>。在空洞进行修复时，借助理信息较多。文献[7~9]中，作者利用网格对传感器网络进行划分，部分文章使用地理信息进行空洞修复。文献[10]使用节点之间洪泛信息方式对空洞进行探测，只需要节点之间的相对位置。文献[3]使用移动节点获得不精确位置信息。文献[4~6]使用相互通信方式来确定节点位置，同样需要地理信息、GPS 等设备的支持，对于大规模传感器网络来说成本昂贵，并且空洞修复问题本身就是一个 NP 问题，使用地理信息仅仅将解决问题的精度提高，仍不能得到最优解。本文不借助理信息，通过传感器节点所具有的感知、通信功能来获得不精确定位信息，利用这些不精确的地理信息确定空洞区域，并作为修复空洞的基础。

针对空洞修复问题，可有多种方式。文献[3]通过向传感器网络添加新的节点来完成空洞修复，并且提出了算法修复原则：1)加入新节点不会使空洞分裂；2)加入新节点至少能消除一段覆盖弧。文献[11]提出了一种移动空洞边缘节点以减小空洞面积的算法 VHR(vector based hole recovery)，该算法基于一种假设条件，即密集分布的无线传感器网络中，传感器节点具有一定的移动能力，网络初始化时分布节点所形成的空洞面积较小，也是非连续的，VHR 算法借助 GPS 提供的地理信息确定空洞边缘节点向空洞方向移动，在不改变已有覆盖的前提下缩小空洞面积，但是该算法只能应用于某些特定形状的空洞而且需要借助 GPS，能量消耗较大。文献[5]提出一种使用 Voronoi 图作为解决空洞覆盖的方法，该方法重新布置传感器网络的节点，最大效率利用节点覆盖，使用 Voronoi 图对传感器节点进行迭代计算，但该方法需要节点之间精确的地理信息和额外的 GPS 设备。

本文提出一种不需要地理信息的基于移动的空洞修复算法 SOI(search optimistic inner)，以文献

[4,5,11]中节点可以具有一定的移动能力为理论支持，其中文献[3]移动部分节点用于覆盖空洞边缘节点未覆盖弧，文献[3]移动空洞边缘节点朝空洞方向移动，文献[4]移动节点以填补失效节点的感知范围。本文通过移动一些特殊节点，达到使感知范围覆盖整个目标区域的目的。在空洞探测阶段，使用文献[2]中的  $K$ -coverage 算法进行探测，若目标区域为完全覆盖，算法的后续部分不进行，否则进入空洞修复阶段。SOI 算法与文献[11]相比，虽然修复的精度有所降低，但不需要增加 GPS 设备，实施代价较小。与文献[3]相比，SOI 算法不需要增加新节点，只需要移动部分节点就可以完成修复工作，尽管算法要求具有移动能力的节点提高了网络构建代价，但算法过程中移动节点数量和移动距离较小，其代价也是可以接受的。本文主要贡献如下：1)提出基于移动的空洞修复准则，并引入相关定理；2)提出基于移动内点的空洞修复算法 SOI。该算法在没有精确的地理信息时，寻找空洞边缘节点的最佳位置，最终通过移动完成修复工作。

## 2 问题描述

### 2.1 前提假设

无线传感器网络中，每一个节点都有唯一标识号(ID)，节点之间都可以正确地标识自身。每个节点都可以感知某一区域并与相邻节点进行通信，假定其感知和通信范围都为圆形，其感知圆与通信圆的半径分别为  $SR$  与  $TR$ 。假定  $TR = 2 \times SR$  (相关证明由 Bejeranp Y 完成)，这样网络的连通问题就等价于覆盖问题，一个  $K$  覆盖的网络一定是连通的。另外，传感器网络中传感器节点没有精确的地理信息，边界区域的节点都能够正确标识自身，不会将监测区域的边界误判为空洞。同时，由于节点是随机分布，假定3个节点的感知圆相交于同一点的概率为0，并且没有任意2个传感器节点位于同一个位置。

目前，大多数空洞修复的研究都是以网络连通为前提，且覆盖空洞是闭合的。本文同样是基于网络连通的，但对于空洞是否闭合没有特定的要求。本文假定每个节点都具有一定的移动能力，在目标区域随机分布节点后，通过节点的有限移动，每个节点可以获得其邻居节点的位置信息。以这些信息为基础，运行空洞探测算法可以有效探测出覆盖空洞位置与大小。由于研究的复杂性，本文只关注覆

盖空洞修复问题, 空洞探测方面使用文献[2]中  $K$  覆盖算法来探测空洞大小和位置。

### 2.2 相关术语

**定义 1 空洞边缘交点。** 如果 2 个节点都是空洞边缘节点, 且它们之间互为邻居节点, 那么这 2 个节点感知范围相交, 处于空洞相交区域的节点为空洞边缘交点。在图 1 中  $P_1$  为  $A$  节点与  $B$  节点的空洞交点,  $P_2$  为  $B$  节点和  $C$  节点的空洞交点。

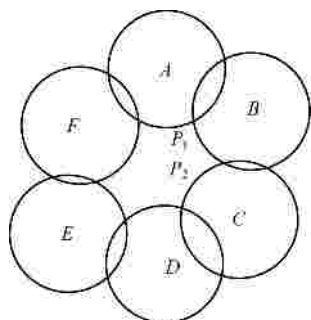


图 1 空洞示意

**定义 2 内点。** 如果若干个节点都是邻居节点, 那么一个传感器感知范围内的一点(不在该传感器感知范围的边缘上)是其他 2 个传感器感知区域的交点, 则该交点称为内点。几何学上的定义是: 3 个圆  $U, V, W$  其半径为  $r$ , 若  $P \in (C_v \cap C_w)$ ,  $P$  满足  $P \in D_u$ ,  $d(p, u) < r$  则  $P$  为圆  $U$  内点。在图 2 中  $P_3$  是传感器节点  $C$  和  $B$  的感知区域的交点, 并且  $P_3$  处于传感器  $A$  的感知范围内。

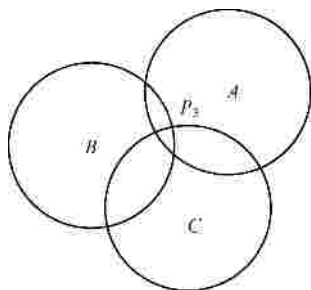


图 2 空洞边缘节点, 邻居节点示意

**定义 3 空洞内点。** 根据内点定义, 在特殊情况下如果  $A$  为空洞边缘节点, 那么  $P_3$  为空洞内点。

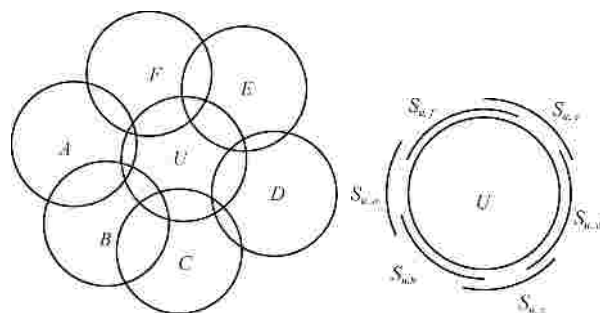
**定义 4 空洞边缘弧。** 相邻的空洞边缘交点通过圆弧相连, 节点感知区域边缘上连接空洞边缘交点的圆弧称为空洞边缘弧。

**定义 5 空洞边缘邻居。** 在传感器网络中如

果有 2 个节点互为邻居节点, 并且这 2 个节点为空洞边缘节点, 则称这 2 个节点互为空洞边缘邻居。

在本文中有如表 1 所示的符号。

符号	定义
$C_u$	传感器节点 $u$ 的感知圆的边界
$D_v$	传感器节点 $v$ 的感知范围
$S_{u,v}$	传感器 $v$ 的感知圆与传感器 $u$ 的感知圆相交, 重叠部分在 $u$ 节点边界所产生的一段弧, 即 $S_{u,v} = C_u \cap C_v$ , 如图 3 所示
$m_{u,v}$	覆盖弧 $S_{u,v}$ 的弧度, 如图 3 所示
射线 $[u,v)$	以 $u$ 为始点指向 $v$
角 $\angle uvw$	射线 $[u,v)$ 和射线 $[u,w)$ 之间的夹角
$d_{u,v}$	$u$ 距 $v$ 的距离
$\langle d_{u,v}, q_{u,v} \rangle$	$v$ 在 $u$ 节点的坐标系中的相对位置表示
$q_{u,v}$	$v$ 在 $u$ 坐标系中的角度, 指明了 $v$ 在 $u$ 中的方向
$[q_{u,v}, m_{u,v}]$	表示覆盖弧 $S_{u,v}$ , 如图 3 所示



(a) 节点  $u$  和它的邻居 (b)  $u$  的 6 个覆盖弧

图 3 覆盖弧

对于覆盖弧, 有如下性质。

**性质 1** 覆盖弧  $S_{u,v}$  与  $S_{u,w}$  相交, 当且仅当  $m_{u,w} + m_{u,v} > 2 \times \angle v, u, w$ 。

**性质 2** 覆盖弧  $S_{u,w} \subset S_{u,v}$ , 当且仅当  $m_{u,w} < m_{u,v} + 2 \times \angle v, u, w$ 。

**性质 3** 覆盖弧  $S_{u,w} \cap S_{u,v} \neq NULL$ , 且  $S_{u,w} \not\subset S_{u,v}$ ,  $S_{u,v} \not\subset S_{u,w}$ 。

$$1) \angle v, u, w > \left| \frac{m_{u,w} - m_{u,v}}{2} \right| ;$$

$$2) \angle v, u, w < \frac{m_{u,w} + m_{u,v}}{2} .$$

**性质 4** 覆盖弧  $S_{u,x} \subset S_{u,w} \cap S_{u,v}$ , 当且仅当满足以下条件:

- 1)  $\angle v, u, x + \angle x, u, w = \angle v, u, w$  ;
- 2)  $m_{u,x} / 2 < m_{u,v} / 2 + \angle v, u, x$  ;
- 3)  $m_{u,x} / 2 < m_{u,w} / 2 + \angle w, u, x$  。

### 2.3 问题描述及相关性质

在空洞修复过程中，本文采用移动空洞边缘节点的方式进行空洞修复。通过不断移动空洞边缘节点减少网络中感知圆的重叠面积，扩大网络覆盖面积，最终达到消除覆盖空洞的目标。在空洞修复过程中，引入以下准则：1)移动节点不会使其邻居节点产生新的未覆盖弧；2)移动节点必须减少覆盖空洞的面积。由于节点是随机布置的，每个节点的感知圆与周围邻居节点的感知圆不规则相交，产生若干重叠的感知区域，在节点移动的过程中，需要遵循上面提出的 2 个准则。

利用移动节点来修复空洞，其本质是移动空洞边缘节点至某一位置使该节点的感知圆恰好经过内点，即节点的感知圆和其 2 个邻居节点相交于一点。这样节点间的冗余面积减小，节点覆盖面积增大。然而，由于节点分布的随机性，一个节点必然含有多个内点，需要从这些内点之中选择一个作为移动后 3 个感知圆的交点。

简单网络模型中空洞边缘节点  $S$  的内点是有限的，从中选择一个距离  $S$  最远的内点，移动  $S$ ，使  $S$  与其邻居节点相交于该内点即可完成空洞修复。但是由于传感器节点是随机分布的， $S$  中内点位置非常复杂，不能简单的从中选择一个距离  $S$  最远的内点。如图 4 所示，若  $S$  为空洞边缘节点， $A$ 、 $B$ 、 $C$  都是  $S$  的邻居，由 2.2 节定义可知  $P_1$ 、 $P_2$ 、 $P_3$ 、 $P_4$  为  $S$  的内点，按照现有的方法，选择距  $S$  最远的内点  $P_1$  作为移动的内点，那么移动之后，虽然  $S$  与  $A$ 、 $B$ 、 $C$  的重叠面积减小了，但没有减至最小， $S$  可以再次移动。但是再次移动的计算过程极其繁琐，而且移动之前必须使用空洞探测算法确定新产生的空洞。这里，关于内点可以有以下所述性质。

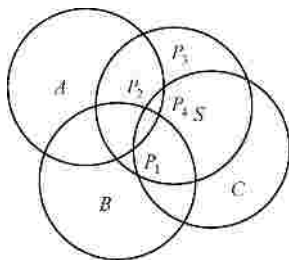


图 4 复杂网络中的拓扑结构

**引理 1** 如果一个圆的某个内点也是其他圆的内点，那么该内点的覆盖度大于同一圆的其他内点。

**证明** 假设有 4 个圆  $u$ 、 $v$ 、 $w$ 、 $x$ ，其圆周为  $C_u$ 、 $C_v$ 、 $C_w$ 、 $C_x$ ，并令  $SR = R$ ，从条件可得  $\exists p \in c_v \mid c_w$ ， $d_{p,v} = d_{p,w} = R$  且  $p \in c_u \mid c_x$ ， $p$  为内点且  $q \in (D_u \mid D_v \mid D_w \mid D_x)$ 。根据题设  $\exists q \in c_v \mid c_w$  且满足  $q \in D_u$ ，则  $q$  为内点且满足  $q \in (D_u \mid D_v \mid D_w)$ 。若根据本文算法移动圆  $u$ ，则内点  $p$  永远会处于某个圆的范围中，其覆盖度一定大于其他内点，而  $q$  不满足  $q \in (D_u \mid D_v \mid D_w)$ 。证毕。

根据引理 1，空洞边缘节点中选择被其他节点覆盖的内点作为移动的定位点，并不能使空洞边缘节点与其邻居节点的重叠面积最小(因为移动后的内点的覆盖度仍大于其他内点)。因此在选择内点时应该排除这一类特殊的内点，这里使用覆盖弧的性质来解决这一问题，并且本文把这一类空洞边缘节点的内点并且也是其邻居节点的内点称为覆盖内点。如图 4 所示， $P_1$ 、 $P_2$ 、 $P_4$  都是覆盖内点，以其中之一作为移动的定位点都会使移动算法修复空洞失败。

**定理 1** 若圆  $v$ 、 $w$  在圆  $u$  形成的覆盖弧满足  $S_{u,w} \subset S_{u,v}$ ，那么圆  $w$  产生的内点  $p$  满足  $p \subset D_v$ 。

**证明** 3 个圆  $u$ 、 $v$ 、 $w$  的覆盖范围为  $D_u$ 、 $D_v$ 、 $D_w$ ，所形成的覆盖弧为  $S_{u,v}$ 、 $S_{u,w}$  且满足  $S_{u,w} \subset S_{u,v}$ 。假设  $p_1$ 、 $p_2 \in D_u \mid D_v$  且  $d_{u,p_1} = d_{u,p_2} = R$ 。  $q_1$ 、 $q_2 \in D_u \mid D_w$  且  $d_{u,q_1} = d_{u,q_2} = R$ ，由  $S_{u,w} \subset S_{u,v}$  可知在  $D_u$  上  $\widehat{q_1 q_2} \subseteq \widehat{p_1 p_2}$ ，故可得  $(D_u \mid D_w) \subset (D_u \mid D_v)$ ，那么由  $D_u \mid D_w$  所产生的内点  $p$  均有  $p \in D_u$  覆盖，其产生的内点也为覆盖内点即存在内点  $p$  满足  $p \subset D_v$ 。证毕。

根据定理 1，在进行内点选择过程中节点会产生如图 5(c)的情况，那么此时应具体去考虑此种状况。如果  $S_{u,x} \subset S_{u,v} \cup S_{u,w}$ ， $v$ 、 $w$ 、 $x$  3 个感知圆相交， $x$  产生的内点可能会被  $v$  和  $w$  所覆盖，那么要进行下一步判断，这里对  $x$  分为如下 2 种情况。

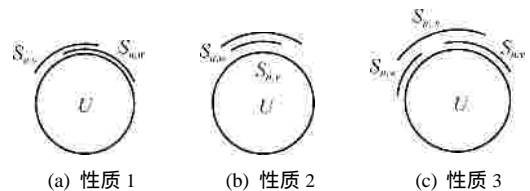


图 5 覆盖弧性质

1) 如果  $S_{x,w} \cap S_{x,v} = NULL$  ,那么  $x$  所产生的内点均被覆盖 ;

2) 如果  $S_{x,w} \cap S_{x,v} \neq NULL$  ,那么  $x$  不是覆盖内点。

**定理 2** 在圆  $u$  中若存在  $S_{u,x} \subset S_{u,w} \cap S_{u,v}$  ,且  $S_{x,w} \cap S_{x,v} \neq NULL$  ,则  $x$  在  $u$  中的内点均被覆盖。

**证明** 由覆盖弧  $S_{u,v}$ 、 $S_{u,w}$ 、 $S_{u,x}$  可得  $c_u \cap c_v$  产生的弧为  $\widehat{P_1P_2}$  ,  $c_u \cap c_w$  产生的弧为  $\widehat{Q_1Q_2}$  ,  $c_u \cap c_x$  产生的弧为  $\widehat{S_1S_2}$  ,由  $S_{u,x} \subset S_{u,w} \cap S_{u,v}$  可知  $\widehat{S_1S_2} \subset \widehat{Q_1Q_2} \cap \widehat{P_1P_2}$  。由  $S_{x,w} \cap S_{x,v} \neq NULL$  ,可知  $S_{x,w}$  与  $S_{x,v}$  覆盖弧是连续的 , 并且  $\widehat{S_1S_2} \subset \widehat{Q_1Q_2} \cap \widehat{P_1P_2}$  则  $\exists p \in (c_u \cap c_x)$  均有  $p \in (D_w \cup D_v)$  , 根据定理 1 的结论  $(c_u \cap c_x) \subset (c_u \cap c_w) \cup (c_u \cap c_v)$  , 所以  $x$  产生的内点  $p$  有  $p \in (D_w \cup D_v)$  。证毕。

结合引理 1、定理 1 和定理 2 可以逐步剔除已经被覆盖的内点,最后从剩下的内点中选择一个距离空洞节点最远的作为最佳内点。本文提出了一个寻找最佳内点的算法 (SOI)。

### 3 算法描述

本文提出了一种利用移动内点来修复空洞的算法,其中包括节点移动方向和内点选择算法。由于算法没有精确的地理信息,需要使用一种特殊的方式来确定节点移动方向。每一个节点在确定自己是空洞边缘节点时都会自动运行该算法,通过计算移动方向和移动距离将自身移动到新的位置。

#### 3.1 移动方向

本文通过移动空洞边缘节点修复空洞,首先应确定空洞边缘节点的移动方向,这里节点选择朝未被覆盖的弧方向移动,能够减少空洞的面积。如图 6(a)所示节点  $S$  为空洞边缘节点,  $P_1$ 、 $P_2$  为空洞边缘交点,弧  $\widehat{P_1P_2}$  为节点  $S$  的未覆盖弧,做向量  $\overrightarrow{SP_1}$ 、 $\overrightarrow{SP_2}$  则  $\overrightarrow{SP_1} + \overrightarrow{SP_2}$  得到向量  $\overrightarrow{SL}$  ,由图 6(a)可知  $\overrightarrow{SL}$  指向未覆盖弧中点且朝向空洞的方向,沿着  $\overrightarrow{SL}$  移动无线传感器节点  $S$  必然会减小空洞的面积,使得  $S$  与其邻居  $A$ 、 $B$  的传感圆的重叠面积缩小。需要指出的是,在确定移动方向时,可能存在例外情况,如图 6(b)所示,通过计算空洞边缘节点  $S$  与邻居所形成的空洞边缘交点  $SP_1$ 、 $SP_2$  的向量和,得到一个向量  $\overrightarrow{SL}$  ,但不能把  $\overrightarrow{SL}$  作为节点移动的方向,因为沿  $\overrightarrow{SL}$  移动只能加大空洞边缘节点与其邻居节点的重叠面积,相反地应采用  $\overrightarrow{LS}$  为节点移动方向。

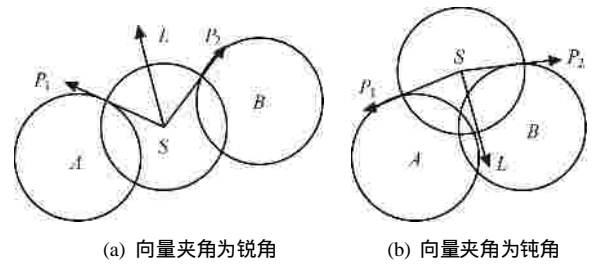


图 6 节点移动方向的确定

移动节点的目的是减小空洞面积,从移动的本身来看移动增大了空洞边缘节点未覆盖的弧长,最终会使得尽可能多的节点感知圆相交于同一点。本文 2.1 节中提到 3 个传感器节点的感知圆相交于同一点的概率为 0,并且任意 2 个传感器节点都不会位于同一个位置,本文算法的本质是移动空洞边缘节点使节点与其邻居节点相交于同一点。由于无线传感器网络中节点是随机分布的,没有精确的地理信息,故使用向量是不现实的。本文提出一个不精确的移动轨迹作为空洞边缘节点的移动方向。

如图 7 所示,  $S$  为空洞边缘节点,  $A$ 、 $B$  为  $S$  邻居节点,且  $A$ 、 $B$  恰为  $S$  空洞边缘节点 (在 SOI 算法中  $A$ 、 $B$  为覆盖弧序列中, 2 个相邻但覆盖弧不相交的节点),做  $AB$  的垂线经过  $S$  且指向  $S$  的向量  $\overrightarrow{NM}$  ,根据 2.3 节已知  $A$ 、 $B$  的位置  $A(d_{s,a}, q_a)$ 、 $B(d_{s,b}, q_b)$  计算出  $S$  的移动轨迹为:

- 1) 如果  $\cos \angle SAB > 0$  , 移动轨迹  $s_s = \frac{p}{2} - q_A + \angle SAB + 2k \times p$  ;
- 2) 如果  $\cos \angle SAB < 0$  , 移动轨迹  $s_s = \frac{p}{2} - q_A - \angle SAB + 2k \times p$  , 且  $\angle SAB = \cos^{-1} \frac{d_{s,A}^2 + d_{A,B}^2 - d_{s,B}^2}{2d_{s,A} \times d_{A,B}}$  。

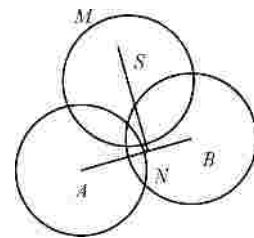


图 7 空洞边缘节点移动轨迹

#### 3.2 SOI 算法与计算移动距离

在这一部分中,算法从空洞边缘节点内部选择一个内点作为移动的终点,并计算节点移动的距离。根据 2.3 节中内点性质提出了一个选择最佳内

点的算法 (SOI 算法)。

当一个节点  $u$  被空洞探测算法计算为空洞边缘节点时, 运行下述算法( $SR=r$ )。

**Step1** 扫描  $u$  周围邻居节点构造一个分割弧队列  $Q_{all}$ ,  $Q_{all}$  中成员为逆时针遍历的邻居节点。

**Step2** 遍历  $Q_{all}$ , 删除每一个  $x$  当且仅当覆盖弧  $S_{u,x} \subset S_{u,v}$  ( $v, x$  为队列中任意邻居节点标记)。

**Step3** 遍历  $Q_{all}$ , 删除每一个  $x$  当且仅当覆盖弧  $S_{u,x} \subset S_{u,w} \cup S_{u,v}$  且  $S_{x,v} \cap S_{x,u} \neq NULL$ , 这样得到一个队列  $Q$ 。

**Step4** 将队列  $Q$  中弧对应的传感器的节点映射到队列  $Q_n$ 。

**Step5** 从  $Q_n$  中取相邻 2 个节点  $n_1, n_2$  令  $n_i = n_1, n_j = n_2$ 。

**Step6** 若覆盖弧  $S_{u,i} \cap S_{u,j} = NULL$  保存  $i$  和  $j$ , 此时  $i$  和  $j$  为空洞邻居节点, 计算节点移动的方向  $s_m, n_i = n_j, n_j = n_{j+1}$ , 转向 Step6, 否则转向 Step7。

**Step7** 计算  $n_i, n_j$  感知圆产生的交点  $o$  距  $u$  的距离, 并选择其中较小的值  $L_{i,j}$  保存, 令  $n_i = n_j, n_j = n_{j+1}$ , 如果  $n_j = n_1$  转向 Step8, 否则转向 Step7。

**Step8** 选择出  $\max(L_{i,j})$ , 并保留产生该交点的 2 个无线感知器节点  $n_i, n_j$  的相关信息。

**Step9** 以  $\max(L_{i,j}), s_m$  为条件根据图 8 所示, 计算出节点在移动方向上移动的距离  $L_o$ 。

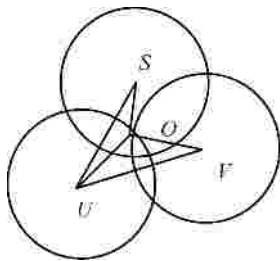


图 8 计算内点距空洞边缘节点距离,  $s$  为空洞边缘节点,  $u$  和  $v$  为队列中满足条件节点

在上述算法中, 逆时针扫描每一个空洞边缘节点的邻居节点, 根据内点性质, 剔除产生覆盖内点的邻居节点, 剩下的内点组成一个队列  $Q_n$ 。计算  $Q_n$  中节点产生的内点, 选择其中距离空洞边缘节点最远的内点为最佳内点。这样算法能同时满足空洞修复的 2 个准则, 不会产生新的空洞, 也能够使节点冗余面积最小。接下来是算法计算节点移动距离。

计算内点与空洞边缘节点距离的算法实现:

$u$  和  $v$  满足  $S_{s,u} \cap S_{s,v} \neq NULL$ , 则  $u$  和  $v$  的交点为  $s$  的内点, 设其交点为  $o$ 。  $d_{s,u}, d_{s,v}, d_{u,v}$  已知, 且  $d_{u,o} = d_{v,o} = r$ 。由三角法则得

$$\angle S, u, v = \cos^{-1} \frac{d_{s,u}^2 + d_{u,v}^2 - d_{s,v}^2}{2d_{s,u} \times d_{u,v}} \quad (1)$$

$$\angle o, u, v = \cos^{-1} \frac{d_{u,o}^2 + d_{u,v}^2 - d_{v,o}^2}{2d_{u,o} \times d_{u,v}} = \cos^{-1} \frac{d_{u,v}^2}{2r \times d_{u,v}} \quad (2)$$

$$\angle S, u, o = \angle S, u, v - \angle o, u, v \quad (3)$$

$$\text{由于 } \cos \angle S, u, o = \frac{d_{s,u}^2 + d_{u,o}^2 - d_{s,o}^2}{2d_{s,u} \times d_{u,o}} \quad (4)$$

由式(1)~式(4)可以求出

$$d_{s,o} = \sqrt{d_{s,u}^2 + r^2 - 2r \times d_{s,u} \times \cos \angle S, u, o}$$

$d_{s,o}$  有 2 个值取  $d_{s,o} < r$  即可,  $L_{u,v} = d_{s,o}$ , 最后取  $\max(L_{i,j})$ , 并求出  $o$  的相对于  $s$  的位置  $\langle d_{s,o}, q_{s,o} \rangle$  这即是所求的最佳内点。确定最佳内点后, 进行算法最后一步, 移动空洞边缘节点至最佳位置。

如图 9 所示, 在分割弧  $Q$  中找到 2 个相邻且覆盖弧相交的节点  $A$  和  $B$ , 根据 2.3 节确定移动轨迹, 在射线  $MN$  找一点  $s'$  满足  $d_{o,s'} = r$  (详细计算过程不再叙述),  $s'$  即为节点移动的最终位置。若不存在  $A$  和  $B$  则算法退出, 此时节点  $u$  已经被完全覆盖。

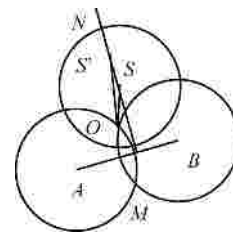


图 9 移动空洞边缘节点  $S$  至  $S'$

算法表明, 节点的移动限制在 1 跳的距离内。而且节点的移动没有影响现存覆盖, 没有对初始时节点的覆盖度产生影响, 所以算法并未对原始覆盖产生影响。设空洞边缘节点的数量为  $n$ , 则算法时间复杂度为  $O(n^2)$ , 在修复过程中空洞边缘节点按 ID 顺序依次进行修复, 能够快速收敛。尽管在修复过程中空洞边缘节点不可能全部移动到最佳位置, 但经实验证明在密集网络中, 它仍然能够到达 1 覆盖。

### 4 仿真实验

本节讨论 SOI 空洞修复算法的性能,使用节点移动的总距离和修复空洞所占面积的百分比来评估算法性能,同时与文献[3]中 VHR 算法进行比较。VHR 算法采用移动节点的方式,将空洞边缘节点沿着向量方向移动一定距离,逐渐缩小覆盖空洞面积。其中向量由边界缺口点来决定,只有缺口点向量的角度小于  $p$  时节点才能移动,而节点移动距离由邻居节点和协作节点共同决定。该算法能够保证在不会破坏已有覆盖的条件下,尽可能缩小空洞面积。

#### 4.1 仿真实验设置

仿真实验采用 C++完成算法,算法在一个  $400 \times 400$  的矩形区域中随机部署  $n$  个传感器节点,节点的感知半径为 20(其传输半径为 40),节点的个数随实验的进行不断变化。为了简化问题,仿真实验并没有实现节点通信的下层 MAC 协议,也没有关注节点在通信和移动过程中所产生的能量消耗,而是通过空洞修复算法中节点移动的总距离来衡量算法过程中的能量消耗。在算法过程中,首先对目标区域随机分布的节点进行空洞探测(使用 Bejeranp Y 的 K-coverage 算法),如果节点为空洞边缘节点则保存其 ID,否则检测下一个节点。空洞探测结束后,空洞边缘节点按照 ID 顺序依次进行修复,待所有节点执行完毕后算法终止。

在 SOI 算法中,空洞边缘节点按 ID 顺序进行移动,节点的移动只会对周围空洞邻居节点的修复产生影响并不会扩大到整个网络,并且在节点密集分布的网络中有限几个节点的移动会完全修复空洞,使得其他空洞节点运行 SOI 算法时在计算移动轨迹时自动退出,并不影响算法有效性。

#### 4.2 实验结果分析

为了说明空洞修复算法中节点的移动方向,标记出节点的移动方向。图 10(a)中显示的是  $n=30$  时,节点随机分布所形成的拓扑结构。为了使实验更为完整,设定传感区域边缘的节点不参与空洞修复,以避免移动区域边缘节点产生新的空洞。图 10(b)是运行修复算法时,每个空洞边缘节点的移动方向,图中箭头为修复算法中节点的移动轨迹。由于 SOI 修复算法中没有精确的地理信息支持,只能通过 2 个空洞边缘邻居与空洞边缘节点共同确定节点移动方向,这种移动方向是不精确的,但后续实验

表明,在节点密集分布的网络中这种粗糙的移动可以获得不错的效果。

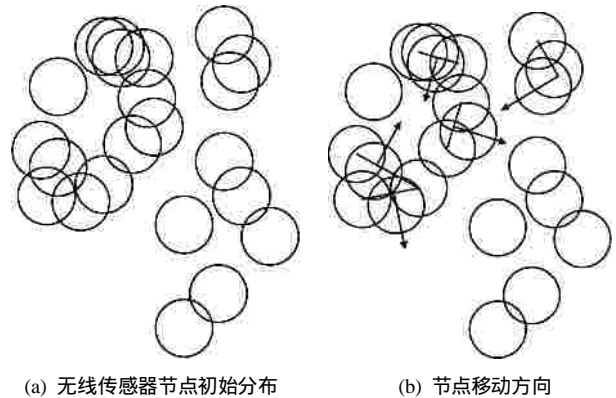


图 10 随机分布的无线传感器节点

如图 11 所示,本文分析了在目标区域中分布不同数量的节点时,运行空洞修复算法前后覆盖区域占目标区域的百分比。初始时,目标区域的覆盖率随着分布节点数量的增多不断提高。移动前后目标区域覆盖比率的变化表明修复后覆盖区域明显扩大,并且随着节点数量增多传感区域的覆盖面积增大,空洞面积相应减小,使用 SOI 修复算法的效果也越好。在节点数量较少时可移动的节点数量很少,导致移动后空洞修复效果不明显。随着节点数量增加,可移动节点也增多,使用移动修复的效果也更明显。与 VHR 算法相比,在密集网络中 SOI 算法的性能稍好于 VHR 算法,并且 SOI 算法不需要精确地理信息,算法适应性要好于 VHR 算法。

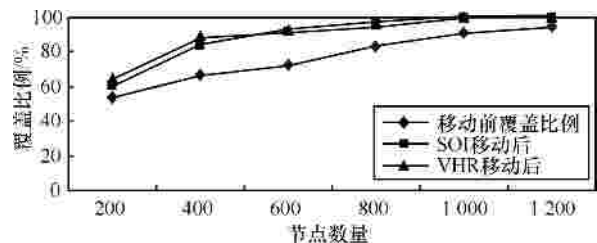


图 11 节点移动前后覆盖比率

图 12 显示了无线传感器节点数量与空洞面积之间的关系。可以看出,节点数量越大,产生空洞的面积越小。在实验过程中,因为节点是随机分布的,所以在节点数量相同的条件下,每次实验中所产生的空洞也有差异,这里的空洞是 10 次实验的结果取平均值。该图表明,在实际应用中,有限区域内密集布置的无线传感器网络所形成的空洞面积比率很小,这样空洞修复算法仅仅只需要移动有

限节点就可以完成对目标区域的完全覆盖，可以确保无线传感器网络中节点能量不会因为移动消耗大量的能量。

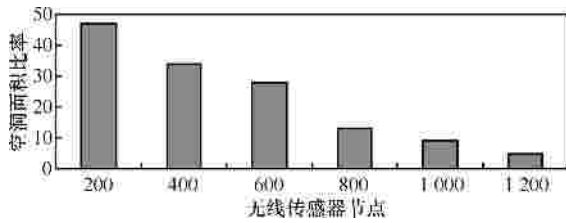


图 12 无线传感器节点数量与空洞面积的关系

本文提出的修复算法是基于节点移动，所以空洞修复算法中节点移动总距离是考查修复算法的重要条件。在无线传感器网络中，节点的能量是有限的，移动会消耗传感器节点大量能量，一旦网络中的节点因为移动消耗过多的能量，必然会产生新的空洞。因此，为了延长节点在无线网络中的生存时间，平衡网络中的负载，算法需要尽可能地减少移动所产生的能量消耗即减少移动式空洞修复算法中节点移动的距离。由于在空洞修复算法中，不同的修复算法之间因为修复的方式不同，很难比较各种方法的优越性，因此本文中只与同为移动式的 VHR 算法进行比较。图 13 中详细描述了在节点数目不同的情况下，SOI 修复算法移动传感器节点的总距离与 VHR 算法移动距离之间的比较。从图中可以看到，节点数目越大所需要移动的距离越小，因此上文中提到的修复算法的有效性得到证实。

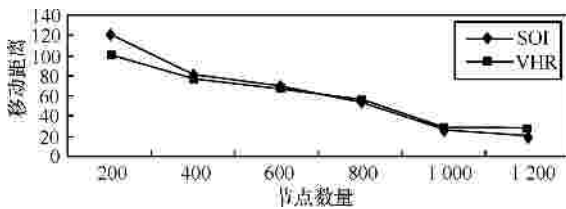


图 13 SOI 算法与 VHR 算法移动距离对比

如图 13 所示，节点数量较少时，SOI 移动距离要大于 VHR，这是因为在节点稀疏分布的网络中，与 VHR 算法相比，SOI 算法中每一个非边界的空洞边缘节点都参与移动，并且 SOI 算法不考虑移动节点与 1 跳范围内的协作节点之间的关系，也没有使用向量来精确表示移动方向，大多数节点的移动距离比 VHR 算法大。但是当节点数量超过 800 时 SOI 修复算法移动的距离要小于 VHR 算法。这是因为在密集网络中，SOI 修复算法的特点是尽可能地去移动节点，减少无线传感器节点之间重叠面

积，而 VHR 算法中节点更多地考虑保持与协作节点之间的距离和精确地移动，所以在这种情况下，SOI 算法仅仅需要移动几个节点就可以完成空洞修复任务，此时 SOI 算法移动节点的距离要小于 VHR 算法。

如图 14 所示，在修复空洞时，与 VHR 算法相比 SOI 算法需要移动更多的节点。在 VHR 算法中，只有向量夹角为锐角的节点才能移动，而 SOI 算法大部分非边界区域的空洞边缘节点都可以移动，所以 SOI 算法中可移动的节点数量要明显大于 VHR 算法。尽管如此，由于 SOI 算法所选择的移动方向是粗糙的，在稀疏网络中，SOI 算法的移动距离要明显大于 VHR 算法。空洞修复过程中，移动节点需要消耗一定能量，但如图 14 所示 SOI 算法移动节点的数量远小于节点总数量，移动总距离也较小，并且网络中的每个节点不需要额外的 GPS 设备，故该算法虽然消耗一定能量，但从整体上来说也是可以接受的。

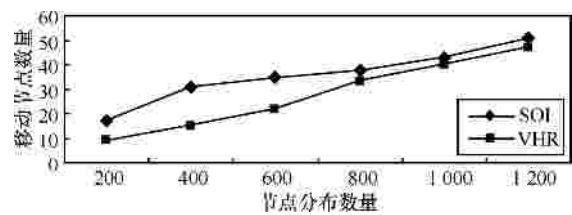


图 14 SOI 算法和 VHR 算法移动节点数量对比

本文算法是基于二维空间的，也可以将其扩展到三维空间，这样计算内点就扩展为计算空间球体相交的弧切面，SOI 算法的目标也可以演变为让多个传感器节点的弧切面相交即可。这里覆盖内点的概念也可以扩展到空间中的面，进一步讨论三维空间中的覆盖问题。

### 5 结束语

覆盖问题是无线传感器网络的重要研究问题。在前期的研究工作中，主要依靠在无线传感器网络中布置大量冗余节点来解决完全覆盖的问题。本文针对传感器网络覆盖问题，证明了网络覆盖中几个基于覆盖性质的定理。基于这些定理，以及在有地理信息的支持下，针对密集分布的无线传感器网络中空洞修复问题，提出了空洞修复算法 SOI。该算法计算出空洞边缘节点的移动方向和空洞边缘节点移动的最佳位置，通过移动减小节点感知圆之间冗余面积扩大节点的覆盖面积，以此实现传感器

网络中的空洞修复。实验表明，该算法在节点密集分布时移动所有的空洞边缘节点以较小的移动距离获得良好的空洞修复性能。本文的下一步工作重点是在没有地理信息系统的支持下提高修复稀疏网络的精确度。

参考文献：

[1] WANG G, CAO G, LA P T. On solving coverage problems in a wireless sensor network using voronoi diagrams[A]. Proceedings of the International Workshop on Internet and Network Economics(WINE)[C]. Hong-Kong, 2005. 584-593.

[2] BEJERANP Y. Simple and efficient  $k$ -coverage verification without location information[A]. Proceedings of the IEEE Conference on Computer Communications[C]. Phoenix, 2008. 291-295.

[3] 苏瀚,汪芸. 传感器网络中无需地理信息的空洞填补算法[J]. 计算机学报. 2009, 32(10): 1957-1970.

SU H, WANG Y. A self-healing algorithm without location in sensor networks[J]. Chinese Journal of Computers, 2009, 32(10): 1957-1970.

[4] SEKHAR A, MANOJ B S. Dynamic coverage maintenance algorithms for sensor networks with limited mobility[A]. Proceedings of the Pervasive Computing and Communications[C]. Hawaii, 2005.

[5] LI X, DAVID H. Distributed coordinate-free hole recovery[A]. Proc of GLOBECOM[C]. Beijing, 2006. 189-194.

[6] YOU J, LIECKFELDT D. Context-aware geographic routing for sensor networks with routing holes[A]. Proceedings of the Wireless Communications and Networking Conference[C]. Budapest, 2009. 1-6.

[7] BOYD, ALAN W. On the selection of connectivity-based metrics for wsns using a classification of application behavior sensor networks[A]. Proceedings of the Ubiquitous, and Trustworthy Computing (SUTC)[C]. California, 2010.

[8] FYS L, CHIU P L. A near-optimal sensor placement algorithm to achieve complete coverage/discrimination in sensor networks[J]. IEEE Communications Letters, 2005, 9(1):43-45.

[9] NITIN K, DIMITRIOS G. Sensor network coverage restoration[J]. CITESEER, 2008, 10(12): 21-24.

[10] KUN Y H, JANG P S. Hole detection and boundary recognition in wireless sensor networks[A]. Proceedings of the Indoor and Mobile Radio Communications[C]. Mannheim, 2009. 72-82.

[11] PRASAN K, JANG Z T. Vector method based coverage hole recovery in wireless sensor[A]. Proceedings of the Communication Systems and

Networks (COMSNETS)[C]. Bangalore, 2010. 1-9.

作者简介：



杨凯 (1985-), 男, 湖北孝感人, 苏州大学硕士, 主要研究方向为无线传感器网络。



刘全 (1969-), 男, 内蒙古牙克石人, 博士后, 苏州大学教授、博士生导师, 主要研究方向为强化学习、智能信息处理、自动推理。



张书奎 (1966-), 男, 内蒙古乌前旗人, 博士, 苏州大学教授, 主要研究方向为无线传感器网络、信息安全、分布式计算等。



李瑾 (1986-), 女, 江苏昆山人, 苏州大学硕士生, 主要研究方向为强化学习和RoboCup 机器人足球。



翁东良 (1986-), 男, 江苏苏州人, 苏州大学硕士生, 主要研究方向为可信软件。