第 34 卷 第 3 期 计 算 机 工 程 2008 年 2 月
Vol.34 No.3 Computer Engineering February 2008
·博士论文· 文章编号：1000—3428(2008)03—0021—03 文献标识码：A 中图分类号：TP393

无线传感器网络中的新型入侵轨迹算法

秦宁宁[1]，张 林[2]，徐保国[1]
(1. 江南大学通信与控制工程学院，无锡 214122；2. 清华大学电子系，北京 100084)

摘 要：在无线传感器网络中，通过为进攻的入侵者寻找入侵轨迹来实现栅栏覆盖控制是比较有效的方法。在兼顾安全性能和时效性能，并且无需全网拓扑信息的支持的前提下，提出一种新型的入侵轨迹算法：SS(Security & Speed)算法。该算法形成的轨迹能动态地反映入侵者对上述两种性能的需求差异。和传统的 Worst-coverage 方法相比，该模型对网络中节点密度的变化有更低的敏感性，计算复杂度降低的同时仍能较好地接近理想结果。

关键词：无线传感器网络；覆盖控制；栅栏覆盖

Novel Attacking Locomotion Trajectory in Wireless Sensor Networks

QIN Ning-ning[1], ZHANG Lin[2], XU Bao-guo[1]
(1. School of Communication and Control Engineering, Southern Yangtze University, Wuxi 214122;
2. Department of Electronic Engineering, Tsinghua University, Beijing 100084)

【Abstract】 Searching the attacking locomotion trajectory is a valid method for the barrier-coverage control in wireless sensor networks. Without any information of topology, an algorithm called as SS(Security & Speed)algorithm, is presented to build attacking locomotion trajectory. This trajectory can image the different of above two demands. Compared with the traditional worst-coverage algorithm, the proposed SS algorithm can match the results of the optimal more closely and has smaller complexity than the optimal approach.

【Key words】 wireless sensor networks; coverage control; barrier-coverage

1 概述

由于无线传感器网络[1]的布撒方式和拓扑结构具有相当灵活的特点，使得该领域中覆盖控制[2]的研究备受学者们关注。传统方法是从传感器节点布撒的角度出发来研究如何布撒，能更有效地实现对入侵者的监视与跟踪。近年来新涌现出的栅栏覆盖[3]控制则从入侵的角度研究：如何在已有的传感器网络中寻找网络中的薄弱环节，建立一条合理的入侵轨迹。

文献[4]给出了一个生成理想安全入侵轨迹的算法——Grid-based方法，该方法在剖分阶数m_1和m_2的指导下，将感应区域高度剖分，再通过全路径搜索得到入侵轨迹，但计算复杂度高达$O(|V_G|^3)$(其中，$|V_G|=2m_1m_2^2+2m_1m_2+m_2-1$)。Worst-coverage[4-5]策略是栅栏覆盖控制中的传统算法，该策略生成了一个基于Voronoi模型的安全入侵轨迹。但通过试验发现，Worst-coverage策略对传感器节点的分布情况过于敏感，在节点数量较少或者节点分布不均匀时，与Grid-based方法生成的理想安全入侵轨迹相比，平均误差为 64.09%，甚至最大误差可达 80%以上。文献[6]曾对Voronoi图模型进行改进，但误差降低有限。

文献[7-8]介绍的入侵算法都是以单一的安全性能为衡量指标而建立的入侵轨迹。这种单纯考虑安全性能，忽略时间指标的入侵轨迹，在很多场合中并不能真正满足应用的需求，尤其在对时间要求严格的战场环境中，如侦查兵沿着过长的入侵轨迹前进，将会导致战机的贻误。虽然可以改进Grid-based 方法使其生成的入侵轨迹同时受安全和时效两个参数的约束，计算出满足两种性能指标的理想入侵轨迹，但是计算复杂程度依然保持为一个很高的程度，不易于工程推广。

为改善原有算法过于片面性的情况，本文从兼顾安全(security)性能和到达目的地的时效(speedy)性能两方面出发，提出一种不依赖于整个网络的拓扑信息，仅利用传感器与入侵者的距离值，计算出当前入侵者在网络中安全与时效性能的折衷值，从而自发地调整入侵轨迹的算法，即 SS(Security & Speedy)算法。

2 网络模型

为了便于分析，本文提及的无线传感器网络基于以下假设：

(1)感应区域 F 内随机布撒 n 个传感器节点 $s_i \in S, i=1,2,\cdots,n$，并且不失一般性地假设这 n 个传感器节点具有一致的物理特性。

(2)入侵者 *object* 配有 RSSI(Received Signal Strength Indicator)技术的收发机设备，可实现对传感器 S 距离 d 的估计计算，$d=d_1,d_2,\cdots,d_n$。其中，$d_i=d_i(s_i,object)$是传感器节点 s_i 与 *object*，在二维空间上的欧基里德距离。

(3)入侵者在感应区域 F 内，从起点 *Ps* 以均匀速度向终点 *Pd* 前进。

(4)存在传感器感应临界距离：感应半径r和通信半径R[9]，并根据网络连通性的设计原则[10]，设定$R \geq 2r$。

基金项目：国家自然科学基金资助项目(60672107)
作者简介：秦宁宁(1980－)，女，博士研究生，主研方向：传感器网络覆盖控制；张 林，讲师、博士；徐保国，教授、博士生导师
收稿日期：2007-02-10 **E-mail：**ningning801108@163.com

3 两种性能的定义

3.1 安全性能

根据对入侵者考查的时间片段的差异，安全性能的衡量分为安全概率、轨迹暴露量两种。安全概率衡量了入侵者暂态或者行进了某一轨迹片断的安全性能；但针对于长时间段或者长轨迹的安全性能的衡量，则需要通过对感应量在时间或空间上的累积来表示，即轨迹的暴露量。

(1)安全概率

定义 1 传感器节点 s_i 对入侵者的发现概率为

$$p_i=\begin{cases}0 & d_i\geq R\\ \dfrac{R^k-d_i^k}{R^k-r^k} & r<d_i<R\\ 1 & d_i\leq r\end{cases} \tag{1}$$

其中，$i=1,2,\cdots,n$ ；$k=1,2$ 是距离影响系数。区域内的所有传感器节点对入侵者的未被发现概率，入侵者的安全概率为

$$P_{\text{safe}}(S,object)=\prod_{i=1}^{n}(1-p_i) \tag{2}$$

(2)轨迹暴露量

传感器节点对入侵者的感应量随着距离的增长而降低，则某个传感器节点 s_i 对入侵者的感应量可以定义如下：$S(d_i)=\dfrac{\lambda}{d_i^{\,k}}$ [11]，其中，$\lambda>0$ 是感应系数。

那么入侵者在时间$[t_1,t_2]$内，移动了 Γ 长度的暴露量计算公式可定义[12]如下：

$$E(\Gamma)=\int_{t_1}^{t_2}\sum_{i=1}^{n}S(d_i)\cdot\left|\frac{\mathrm{d}\Gamma(t)}{\mathrm{d}t}\right|\mathrm{d}t \tag{3}$$

其中，$\Gamma(t)$ 是入侵者在 t 时刻的位置。在二维空间内，存在 $\Gamma(t)=(x(t),y(t))$ 是连续可导的，那么时间上累计的暴露量可转换为在轨迹长度上累计的暴露量。其线积分形式推导如下：

$$E(y)=\int_{t_1}^{t_2}\sum_i S(d_i)\sqrt{\left(\frac{\mathrm{d}x(t)}{\mathrm{d}t}\right)^2+\left(\frac{\mathrm{d}y(t)}{\mathrm{d}t}\right)^2}\,\mathrm{d}t=\int_{x_1}^{x_2}\sum_i S(d_i)\sqrt{1+y'^2(x)}\,\mathrm{d}x=\int_{\Gamma}\sum_i S(d_i)\,\mathrm{d}l \tag{4}$$

其中，$x\in[x(t_1),x(t_2)]=[x_1,x_2]$；$\sum_i S(d_i)$ 是所有传感器节点对入侵者的感应量之和。

3.2 时效性能

因为入侵者是匀速运动前进，故入侵者沿轨迹移动的时效性能可影射为对轨迹长度有效性的度量，定义有效前进距离来考察目标的时效性能。

定义 2 设目标从当前位置 A 前进到位置 B ，且位置 B 在 $\overrightarrow{A,Pd}$ 上的投影点为 $B`$，那么轨迹 (A,B) 的有效前进距离应该是该轨迹在 $\overrightarrow{A,Pd}$ 上的投影距离：

$$Progess(A,B)=\begin{cases}l, \left\langle\overrightarrow{AP_d},\overrightarrow{AB`}\right\rangle\geq 0\\ -l, \left\langle\overrightarrow{AP_d},\overrightarrow{AB`}\right\rangle\leq 0\end{cases} \tag{5}$$

其中，$|\overrightarrow{AB`}|=l$ 。

4 SS 算法

4.1 建立模型

算法的目的是寻找一条从起点 Ps 到终点 Pd 的轨迹，使得在此轨迹上移动的入侵者 $object$ 被发现的程度尽可能地降低，同时行进过程中所消耗的时间尽可能地缩短，即同时实现安全性能与时效性能。对入侵者的轨迹进行单步分析，如图 1 所示。设入侵者当前行进到某一点 O ，准备以步长 dl 前进到 O' 且 $\angle O'OP_d=\theta$ 。则入侵轨迹的寻找过程可以等价为：入侵者每一步只朝着，安全概率和时效性能均较大的方向 θ 前进的过程。

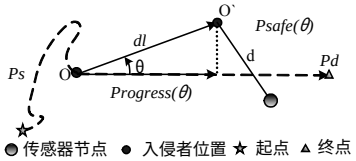

图 1 轨迹单步分析效果

入侵者前进到位置 O' 时的安全概率为 $P_{\text{safe}}(S,O')$，根据式(1)和式(2)，P_{safe} 是一个与 d 有关的函数。显然，d 是关于 θ 的函数，故 $P_{\text{safe}}(S,O')=P_{\text{safe}}(\theta)$ 。

入侵者每一步的时效性的提高，直接反映为每一步有效前进距离 $Progress(dl)$ 的增大。根据图 1 存在 $Progress(dl)=dl\cdot\cos(\theta)$，因此 P_{safe} 与 $Progress$ 均是 θ 的函数。因此，可得到入侵者的单步优化方程式(6)。

$$\begin{aligned}&\max\ \Lambda(\theta)=\alpha\cdot P_{\text{safe}}(\theta)+\beta\cdot Progress(\theta)\\ &\text{s.t.}\quad -\pi\leq\theta\leq\pi\quad \alpha,\beta\in(0,1)\ \alpha+\beta=1\end{aligned} \tag{6}$$

上式将入侵轨迹的生成问题转化为对每一步求使 $\Lambda(\theta)$ 最大化的 θ 值的问题，其中 α,β 分别为安全性能和时效性能的权重参数，它们反映了用户对两种性能关切的程度。

4.2 算法描述

入侵者从起点出发，每一步以步长 dl 朝向使得 $\Lambda(\theta)$ 达到最大的角度 θ 前进，直到终点为止，具体算法流程图见图 2。算法复杂度为 $O(|\theta|)$，其中，$|\theta|=\dfrac{2\pi}{d\theta}$ 是单步可选角度数量。

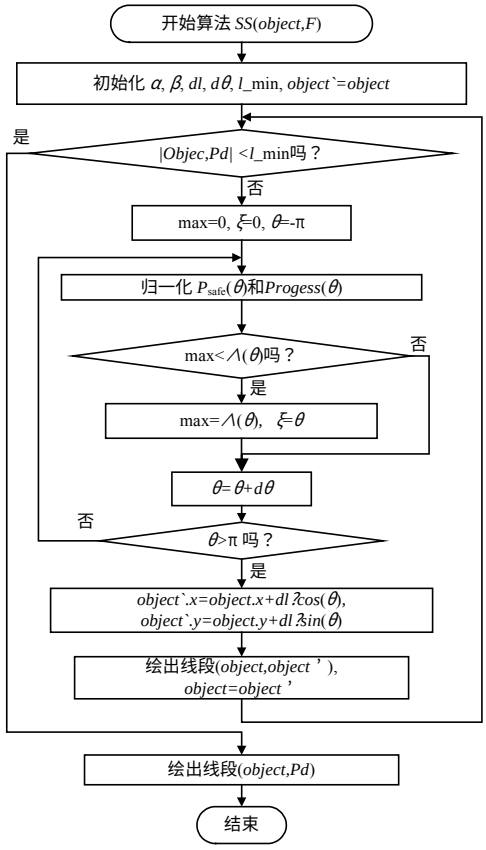

图 2 算法流程

通过实验验证，算法对 θ 的寻找间隔 $d\theta$ 的敏感性比较低，工程应用中的 $d\theta$ 值可根据入侵者的计算能力设定；步长

dl 直接决定算法的准确程度，值越小，形成的轨迹将越理想；定义常量 $l\_\min$，以避免算法陷入死循环，一旦入侵者十分接近目的地时，即 $|\overrightarrow{object, Pd}| < l\_\min$ 时，入侵者将直接迈向目的地。应注意，$P_{\text{safe}}(\theta)$ 和 $Progress(\theta)$ 进行带权重相加时，必须先进行归一化处理。

4.3 轨迹的综合增益

定义 3 连接起点与终点的直线线段为 L，长度 $length(L)$，L 的暴露量 $E(L)$；入侵者的运动轨迹 $track$，长度为 $length(track)$，$track$ 的暴露量为 $E(track)$。那么：

$$\text{安全增益：}\ SG = \frac{E(L) - E(track)}{E(L)}$$

$$\text{时效增益：}\ LG = \frac{length(L) - length(track)}{length(L)} \tag{7}$$

$$\text{综合增益：}\ IG = \alpha \cdot SG + \beta \cdot LG$$

其中，α, β 与式(6)中的权重参数意义相同。入侵轨迹是一个长时间片段量，安全增益的考量不宜直接利用安全概率，应采用对安全性能的累计值，即轨迹的暴露量进行计算；由于模型假设入侵者采取匀速运动，因此时效增益体现为对路程长度的衡量。

5 仿真

仿真平台采用 Matlab，仿真场景设定如下：二维平面 F=100 m×100 m。SS 算法参数设置：r=3 m, R=15 m, λ=1, $k=2$, dl=1, $d\theta = 0.001\pi$。由于本试验为计算机仿真，故步长和角度间隔均选择了较小的值。

5.1 权重值调整试验

权重值调整试验结果如图 3 和图 4 所示。

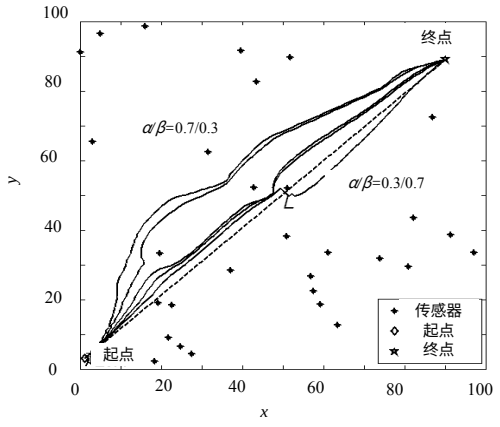

图 3 进攻轨迹随着安全和时效性能要求的变化

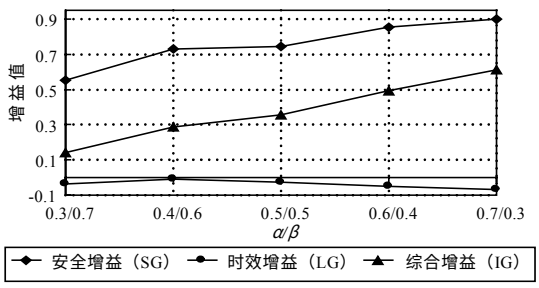

图 4 3 种增益相应的变化曲线

本次试验中随机布撒 30 个传感器节点。比较在该场景下，安全性能与时效性能的要求变化时(即 α/β 变化)，入侵轨迹自发调整的情况。α/β 的值从 0.3/0.7 升高到 0.7/0.3，这种对安全性能的要求逐渐提高的趋势，在图 3 中明显地反映为轨迹自发地远离传感器节点分布密集的右下方区域。图 4 中绘制出 3 种增益 SG，LG，IG 随 α/β 相应变化的曲线。

5.2 SS 算法与其它两种算法的比较

比较 Worst-coverage、改进型 Grid-based 方法和 SS 3 种算法的综合增益。设与改进型 Grid-based 方法的参数 $m_1 = 4$，$m_2 = 16$。安全性能和快速性能等量考虑，即 α=0.5, β=0.5。依照蒙特卡洛方法随机进行 20 次试验，每次实验随机布撒 15 个传感器节点，并且拓扑结构、轨迹的起点与终点位置均随机选择。Worst-coverage, Grid-based, SS 3 种方法的平均综合增益分别标记如图 5。显然，SS 优于 Worst-coverage，并且和理想轨迹(即改进型 Grid-based 方法生成的轨迹)之间的误差较小。这主要是由于 Worst-coverage 算法是一种贪婪的追求安全性能的方法，它完全舍弃对入侵轨迹消耗时间的考虑，以过长的时间代价换取安全代价，从而导致整条轨迹的综合增益被严重拉低。

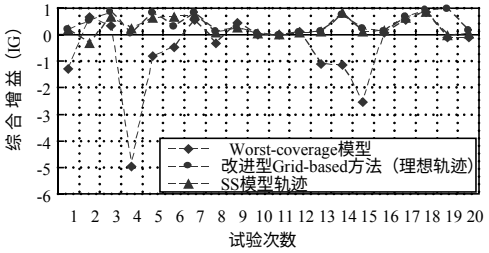

图 5 3 种算法轨迹的综合性能比较(α/β=0.5/0.5)

5.3 SS 算法对传感器节点密度的敏感性试验

提高传感器节点的布撒密度为上一轮试验的 2 倍以上，即在区域 F 内，随机布撒 40 个传感器节点，进行 20 次随机试验，考察 SS 算法对网络密度的敏感程度。鉴于改进型 Grid-based 方法的计算复杂度在 40 节点的网络中过大，故未参与本次试验。这也是本算法提出意义所在，以较小的计算代价获得与理想轨迹较为接近的结果。如图 6 所示，在较高的节点密度分布，经过多次的试验，SS 算法的综合增益较 Worst-coverage 方法相比，优势依然明显。可见 SS 算法对区域内传感器节点分布密度的敏感程度较低，提高了该模型的通用性。

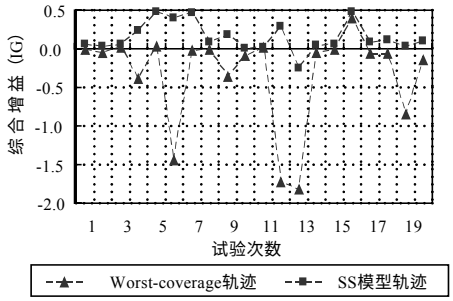

图 6 传感器节点随机高密度分布下的综合性能比较(α/β=0.5/0.5)

6 结束语

本文提出的 SS 入侵轨迹算法，可以根据入侵者当前所处位置，无需全局拓扑信息的指导，自发地朝着安全和时效折衷效果最好的方向前进。该算法具有相对于理想算法更小的计算复杂度，但能获得比 Worst-coverage 方法更加全面增益。此外，通过对模型安全和时效参数的调整，能有效地满足应用者对安全和时效性能的不同的需求。（下转第 61 页）