无线传感器网络中基于分簇的节点定位异常检测 *

张玉琴, 秦 拯
(湖南大学 软件学院, 长沙 410082)

摘 要: 在引入 WSN 分簇结构基础上,提出一种分布式的节点定位异常检测方法,利用聚类拓扑减少通信量,同时降低以往集中式检测存在的单点风险。该方法不需要任何已知的部署知识或额外的硬件,每个簇的簇头节点只需根据该簇节点报告的位置和邻居表信息进行过滤计算,更新权值,即可确定和撤销定位异常的节点。通过理论分析和仿真模拟验证了这种基于分簇的节点定位异常检测方法的正确性和有效性。

关键词: 无线传感器网络; 节点定位; 异常检测; 位置验证; 分簇
中图分类号: TP393 **文献标志码**: A **文章编号**: 1001-3695(2010)03-1139-03
doi:10.3969/j.issn.1001-3695.2010.03.092

Node localization anomaly detection based on clustering in wireless sensor networks

ZHANG Yu-qin, QIN Zheng
(*College of Software*, *Hunan University*, *Changsha* 410082, *China*)

Abstract: On top of a clustering WSN architecture, this paper proposed a distributed scheme to detect localization anomaly. The clustering network could reduce the communication overhead between sensor nodes by utilizing clustered topology, and didn't have the risk of single point of failure. This method didn't need any deployment knowledge or hardware. According to compute the location and neighbor list from the sensor node, updated its weight by the cluster head, this scheme could filter the anomaly node. The simulation verifies the correctness and efficiency of this scheme.

Key words: wireless sensor networks(WSNs); node localization; anomaly detection; location verification; clustering

0 引言

在无线传感器网络(WSNs)中,节点的位置信息至关重要,它不仅是提供监测事件或目标位置信息的前提,也是实现传感器网络各项功能的基础[1]。许多应用都是基于传感器的位置信息,如环境监测、目标跟踪、地理路由等。然而,由于传感器未带防窜改的硬件和传感器网络的开放性、无人看护性,使得节点定位过程极易受到来自恶意节点或被俘获节点的攻击,从而产生一些错误的位置信息[2]。此类信息如果不能被及时检测,将会产生错误的位置报告,给基于位置的应用带来严重的后果,可能导致基于位置的路由协议陷入混乱,或者恶意节点通过编造位置信息发动进一步的攻击,包括 Sinkhole 攻击、Wormhole 攻击等,对网络造成更大的危害。因此,应用软件在使用传感器的位置信息之前,需要先对节点报告的位置信息进行判断,检测传感器定位是否存在异常,这是传感器网络近年来的一个重要研究领域[3~5]。

目前,已提出的定位异常检测方法有很多,但大部分都依赖于可用的已知部署信息或额外的硬件,不适用于一般的 WSNs,而且很多方法都是通过基站来进行整个网络的集中式检测,存在通信量大、单点易失效等问题,一旦基站被损坏,整个检测系统将失效。针对这种集中式检测存在的问题,本文提出了一种基于分簇的节点定位异常检测方法(BCLAD)。该方法引入了基于分簇的网络拓扑结构,这种结构相当于一个分布式的信息聚合系统,一部分的 SN 被组织在一起,由簇头控制,簇头负责收集本簇内的数据进行整合处理,将结果上报给 AP,AP 负责处理簇头上传的数据,并与有线网络进行通信。这种网络结构可以有效降低网络开销,而且不存在单点失效问题。簇内的检测采用 MFWT(基于矩阵过滤的权重信任)算法,它不需要任何已知部署信息或额外硬件,只须根据传感器报告的位置 L_i 和邻居表信息来计算差异矩阵。利用矩阵过滤出位置异常的节点,再根据过滤结果更新每个传感器的权值,定位异常节点的权值将不断减少,而正常节点的权值保持不变。通过一定次数的反复后,权值低于阈值的节点即为异常节点,会被簇头检测出来,继而上报给基站。通过理论分析和模拟实验证明,这种异常检测方法可以有效降低通信量,并能保证较好的检测率。

1 相关工作

定位异常检测是指通过位置验证方法来检测一个节点是否确实在它所声称的位置,这是近年来的一个热点安全问题。Sastry 最先在文献[6]中解决该问题,他采用计算超声波和无

收稿日期: 2009-07-10; **修回日期**: 2009-08-17 **基金项目**: 国家"973"项目子课题(2007CB310702);湖南省科技资助项目(7007730)
作者简介: 张玉琴(1986-),女,湖北襄樊人,硕士研究生,主要研究方向为无线传感器网络中的异常检测(xiaomonv_sophie@126.com);秦拯(1969-),男,湖南祁东人,教授,博士,主要研究方向为网络安全、无线传感器网络等.

线电的传播时间差的方法验证节点是否在特定的区域内(如一个房间),但这种方法只能实现有限的判断(在区域内/不在区域内),而且超声波传播距离较短,容易受到干扰,不适合在野外大面积部署传感器网络。Capkun 等人[7]提出了基于距离界定协议的 VM(verifiable multilateration)机制。该机制借助一个授权节点和若干信标节点协作实现网络中未知节点的安全定位和对定位结果的验证,但该方法没有考虑参考点被俘获的情况,而且该方案依赖高速硬件测定无线电在空气中传播的时间,成本较高。Bahl 等人[8]使用信号强度测试的方法估计移动节点在室内的位置,这种方法需要事先知道室内的信号强度分布地图,这对于在野外大规模部署的传感器网络(特别是战场监视等场合)是不可行的,而且,它是一种集中式的检测方法,会给基站带来很大的计算和存储开销。Du 等人[9]提出了 LAD 算法,该方案借助在许多传感器网络应用中可以事先获知的节点分布信息以及邻居节点间的组关系,检测节点的估计位置是否与它的观测位置相一致。然而,LAD 方案依赖于节点的分布信息,如果无法获得精确的分布概率, LAD 的检测结果将会受到极大影响。

2　基于分簇的节点定位异常检测

2.1　网络模型与假设

在这里引入一个基于分簇的网络拓扑结构,在这种网络中,整个传感器网络形成分层结构,传感器节点通过由基站指定或者自组织的方法形成各个独立的簇(cluster),每个簇选出相应的簇头(cluster head),由簇头负责簇内节点的控制,并对簇内所收集的数据进行整合、处理,随后转发给基站。这种分层式网络结构既通过簇内控制减少了节点与基站远距离的信令交互,降低了网络建立时的复杂度,减少了网络路由和数据处理的开销,同时又可以通过数据融合降低网络负载,减少了网络的能量消耗。

本文采用的分簇式网络拓扑结构如图 1 所示,它包括三种类型节点:

a)传感器节点(sensor node,SN)——网络最底层,负责监测环境的数据收集;

b)簇头节点(cluster head,CH)——负责将该簇中所有 SN 收集到的数据传递给上层;

c)接入点(access point,AP)——整个网络的最高层,负责无线网络与有线网络之间数据的发送与接收。

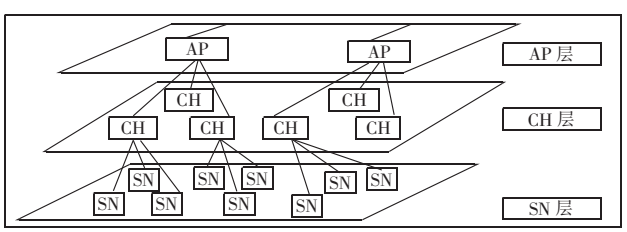

图 1　本文采用的分簇式网络结构

这种基于分簇的网络模型相当于一个分布式的信息聚合系统,每个传感器有惟一的 ID 号。一部分的 SN 被组织在一起,由高层的簇头 CH 控制,因此,每个 SN 只需要负责与它的簇头 CH 通信,CH 位于 SN 的上层,负责与它的上层 AP 和其他 CH 通信。AP 负责处理簇头上传的数据,并与有线网络进行通信。这种网络结构相比传统的网络来说,只是增加了一些簇头节点的数目,但是使得通信开销大大减少,而且即使在一部分的节点受到损害后,整个网络还是有足够的健壮性。在这里假设簇头 CH 和 AP 都是可信的。

2.2　基于分簇的节点定位异常检测原理

在传感器初始化部署后,整个网络采用成簇协议选出簇头并将网内节点分簇,在簇内,每个传感器广播它的 ID 号给邻居,获得它的邻居 ID 表,然后向它的簇头节点 CH 报告自己的位置 L_i 和邻居表信息。每个簇的簇头节点 CH 收到该簇中所有节点上报的信息后,执行 MFWT 算法(见 2.3 节)对本簇所收到的位置信息进行检测,把没有通过检测的传感器 ID 号上报给 AP,AP 把该 ID 号广播给整个传感器网络,通知将该传感器从网络中排除,不再使用其收集的信息。

2.3　基于矩阵过滤的权重信任算法(MFWT)

针对以前的定位异常检测需要硬件或已知部署知识的问题,本文提出了基于矩阵过滤的权重信任算法 MFWT,它只需要根据传感器报告的位置 L_i 和邻居表信息来计算差异矩阵,利用矩阵过滤出来位置不一致的节点,根据过滤结果更新每个传感器的权值,对于位置异常的节点其权值将不断减少,通过一定次数的反复后,权值低于阈值的节点被检测出来上报给 AP。

a)计算差异矩阵 M_d

首先计算邻居矩阵 M_n。它由传感器的邻居表信息构成,传感器的总数是 n,矩阵 M_n 是一个 $n\times n$ 矩阵,里面的元素 $M_n(i,j)$ 满足下列情况:

$$M_n(i,j)=\begin{cases}1 & \text{若 } j \text{ 在 } i \text{ 的邻居表内}\\ 0 & \text{若 } j \text{ 不在 } i \text{ 的邻居表内}\end{cases} \tag{1}$$

接着是通信矩阵 M_c。矩阵 M_c 是通过比较传感器报告位置之间的距离与通信范围构成的,里面的元素 $M_c(i,j)$ 满足下列情况:

$$M_c(i,j)=\begin{cases}1 & \text{若 } d_{ij}\leqslant R\\ 0 & \text{若 } d_{ij}>R\end{cases} \tag{2}$$

其中:(x_i,y_i) 和 (x_j,y_j) 是传感器报告的位置;d_{ij} 是 i 和 j 之间的欧式距离;R 是传感器之间的通信范围。

在构建完 M_n 和 M_c 后,差异矩阵 M_d 通过异或它们的元素组成:

$$M_d=M_n\oplus M_c \tag{3}$$

在 M_d 中,若元素为 0,则代表邻居的观察值与所报告的位置一致,如 $M_n(i,j)=1$,$M_c(i,j)=1$,异或后 $M_d(i,j)=0$,代表传感器 i 可以观察到 j,i 与 j 之间的距离也在通信范围内,这是正常的。若元素非零,则代表邻居的观察值与所报告的位置存在矛盾。例如,$M_n(i,j)=1$,$M_c(i,j)=0$,异或后 $M_d(i,j)=1$,代表传感器 i 可以观察到 j,但是根据它们的报告位置计算可知它们不是邻居。只要这些矛盾存在,可以用矩阵 M_d 来检测异常位置。

b)利用 M_d 过滤可疑异常节点

定义三个变量 AD、PD 和 AS,利用这几个变量来过滤可疑的节点。具体定义如下:

$$AD_i=\sum_{k=0}^{n}M_d(i,k) \tag{4}$$

它是差异矩阵 M_d 中第 i 行的元素总和,量化了所得到的邻居关系与 i 本身的观察之间的不一致性。

$$PD_i=\sum_{k=0}^{n}M_d(k,i) \tag{5}$$

它是差异矩阵 M_d 中第 i 列的元素总和,量化了所得到的邻居关系与其他节点对 i 的观察之间的不一致性。

$$AS_i = \sum_{k=0}^{n} |M_d(i,k) - M_d(k,i)| \quad (6)$$

它量化了两个传感器之间通信的不对称程度。

检测开始,CH 计算所有传感器的矩阵 M_d 和变量 AD、PD 和 AS,然后比较它们与相应的阈值。如果有任何传感器的变量值超过阈值,CH 将它的 ID 号记录下来(如节点 k),并设置矩阵 M_n 和 M_c 的第 k 行和第 k 列元素为 0。在接下来每一轮中,CH 重建 M_d,重新计算这些变量,并过滤掉当前轮中最不正常的传感器,直到所有变量值低于阈值。被过滤出来的这些传感器都是可疑的异常节点,剩下的都通过验证,为正常节点。

c)更新权值并确定异常节点

在这里加入权重信任机制,是为了减轻由于定位所产生的误差。传感器在部署后,会给每个传感器 i 设置一个权值初值 w_i,在经过矩阵过滤以后,每个节点的权值更新基于式(7):

$$W_i = \begin{cases} w_i - \theta & \text{若 } i \text{ 为可疑节点} \\ w_i & \text{若 } i \text{ 为正常节点} \end{cases} \quad (7)$$

其中:θ 为权重处罚值,当一个节点被确定为可疑节点,它的权值就减少 θ。权值更新以后,CH 就可以根据权值是否低于门限值来确定位置异常的节点。CH 把没有通过检测的传感器 ID 号上报给 AP,AP 把 ID 号广播给整个传感器网络,通知将该传感器从网络中排除,不再使用其收集的信息。

3　实验与分析

下面采用 OMNET ++ 作为实验平台对本文提出的基于分簇的定位异常检测方法进行仿真模拟。仿真分析的网络模型的主要参数如下:

a)测试区域大小为一个 300 ×300 的平面区域,500 个节点随机分布在该区域范围内,传感器的通信距离为 20 m;

b) 节点部署后,采用 LEACH 协议对网络中节点分簇,簇个数为 5%;每一个节点都有单独的 ID 号,以区别于其他的节点;

c)模拟异常节点的行为如下:它们距离自己真实位置的距离为 D = 30 m,为了迷惑检测者,它们不但报告错误位置,同时报告的邻居表也与错误位置一致。

每个实验结果根据超过 100 次的独立模拟后的平均值计算得到。

定义以下指标评估检测方法的性能:

响应时间(response time,RT),正确检测出异常节点的平均周期,代表一个异常节点如何被快速地检测出来。

检测率(detection rate),$DR = Ndf/Nf$。其中:Ndf 是被检测出来的异常节点的数目;Nf 是整个网络中异常节点的总数目。

误判率(false positive rate),$FP = Ndt/Nt$。其中:Ndt 为正常节点但被错判为异常节点的数目;Nt 是整个网络中正常节点的总数目。

3.1　网络密度的影响

设置异常节点比率为 10%,异常程度为 30 m,当权重处罚值分别为 0.01 和 0.05 时,本文评估了节点数目变化时对本方法的影响,如图 2、3 所示。

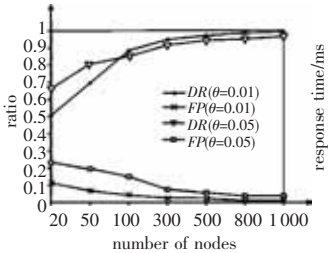

图 2　网络密度变化时的检测结果

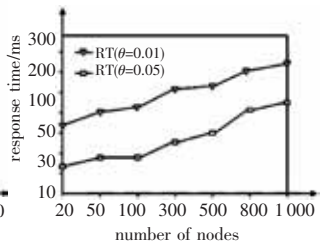

图 3　网络密度变化时的响应时间

从图 2 中可以看出,随着节点数目增加,网络密度和检测率都提高了。当节点数目从 20 增加到 1 000 时,检测率和误判率都相对稳定,尤其在节点数目超过 100 后,检测率保持在 80% 以上,误判率低于 5%。这意味着,该方法具有很好的扩展性,在各种不同的网络状态下运行良好,特别是在网络规模较大时,可以保持很好的性能。图 2 和 3 也显示了权重处罚值设置的影响,当处罚值较大时,响应时间较小,检测速度更快,但同时也导致误判率 FP 较高,因此在实际中要根据具体的网络需求来选择合适的处罚值。图 4 显示了节点数目为 500 时,权重处罚值的设置对方法性能的影响。从图中可以看出,当权重处罚值增加时,检测率比较稳定,但误判率 FP 有所提高。为了在检测率和误判率之间保持平衡,检测权重处罚值取为 0.01 ~0.1 的值比较好,可以保证算法的性能。图 5 是节点密度变化时本方法与文献[5]提出的 GFM 算法进行的比较,两种算法在节点密度较大时都能保持稳定的性能,本文的方法检测率稍高一些。

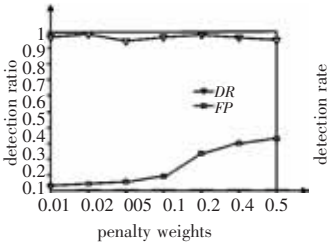

图 4　权重处罚值对方法的性能影响

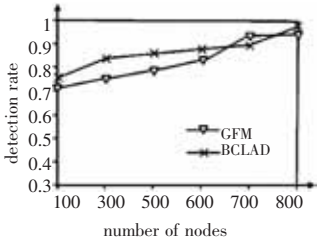

图 5　网络密度变化时的方法性能比较

3.2　异常程度的影响

设置节点总数目为 500 个,本文评估了异常程度对方法性能的影响,如图 6 所示。

节点的异常程度包括两个方面:a)异常节点占整个网络节点数目的比例;b)某个节点的异常程度 D(D 为报告位置距其真实位置的距离)。图 6 中,本文评估了在 D 分别为 10 m,30 m,50 m,100 m 时,异常节点比例从 0.1 增加到 0.7 的本方法的性能。从图 6 可以看出,当异常比例增加时,检测率有稍微的下降,这是因为更多异常节点的出现意味着聚合结果将会被更多的错误结果影响,但是总体看来,检测率大都保持在 80% 以上,由此可见,该方法针对大规模的异常也能保持较好的检测率,性能稳定。

图 7 是本方法与文献[5]提出的 GFM 算法在不同异常程度下的性能比较。文献[5]的算法是一个集中式的检测算法,从图中可看出,当异常节点比率低于 30% 时,GFM 算法还可保持较好的性能,随着异常节点比率的增加,它的检测率也快速下降,因此它只能适用于轻度异常的网络,当网络异常程度严重时,算法性能远不如本文方法。

(下转第 1153 页)

题,从而降低算法的时间复杂度。

3　实验结果

为了验证算法的可靠性和实用性,用 OPNET 网络仿真工具来构建模拟网络。参数配置情况为:节点具有独立均匀的错误概率 P,针对维数为 10、20、30 的超立方体网络,分别设置 P 为 0.5%、10%、20% 和 30% 四种情况,随机选择 20 对正确节点进行消息传递。仿真实验结果如表 1 所示。其中:Path F 为算法找到的正确路径数与网络无错误时可用总路径的比率;Min P 为算法找到的最短路径数与网络无错误时可用总路径的比率;Path L 为算法构建的路径长度与两节点间汉明距离的比率。实验结果表明:当节点的错误概率为 10% 时,算法有 93% 以上的概率路由成功;当节点的错误概率为 0.5% 时,算法有 99% 以上的概率路由成功。通过与文献[7]的模拟实验结果比较可以看出,算法构造的路径长度更接近于最优路径长度。对算法进行分析可以看出,死锁可能发生在存在 V_j^{**} 不为空的节点 v_i,却不存在到达 v_i 的可行路径时。由于为当前节点添加了局部标记变量,且算法步骤 5 的循环结构定序地选取 v_i,算法能够正确回朔来避免死锁,模拟实验的结果也表明算法可以预防死锁。

表 1　算法寻径实验结果

n	P/%	Path$F$/%	Min$P$/%	Path$L$/%	n	P/%	Path$F$/%	Min$P$/%	Path$L$/%
10	0.5	99.6	93.6	108.8	20	20	98.9	87.1	125.4
10	10	94.5	74.5	114.3	20	30	96.1	76.3	133.6
10	20	97.2	81.1	127.1	20	0.5	99.3	80.2	109.5
10	30	93.4	78.2	137.4	30	10	99.2	73.6	114.1
10	0.5	99.1	79.0	105.6	30	20	97.6	95.5	123.7
10	10	93.5	94.7	112.5	30	30	94.8	77.0	131.3
20	20	91.8	85.3	121.0	30	0.5	99.7	84.7	106.0
20	30	98.4	79.9	138.7	30	10	99.0	73.9	118.4
20	0.5	99.7	75.4	104.2	30	20	91.3	74.8	127.2
20	10	95.3	82.8	116.9	30	30	94.9	92.4	136.8

4　结束语

本文提出了相邻节点集合类的概念和在 n 维超立方体网络中求解相邻节点集合的方法,解决了在相邻子立方体中寻找一对连通的可达节点的问题。利用子连通性的概念减小了求解路由问题的规模,有效降低了算法的时间复杂度。在实际应用中由于很少碰到 k 接近于 n 的特殊情况,该算法在根据网络的错误节点情况选取一个合适的 k 后是很高效的。新提出的概念也可以推广到其他层次型拓扑结构的网络中去。

参考文献:

[1] DAY K, TRIPATHI A. A comparative study of topological properties of hypercubes and star graphs[J]. IEEE Trans on Parallel and Distributed Systems, 1994, 5(1): 31-38.

[2] AKERS S B, KRISNAMURTHY B. A group-theoretic model for symmetric interconnection networks[C]//Proc of International Conference on Parallel Processing. 1986: 216-223.

[3] ESFAHANIAN A H. Generalized measures of fault tolerance with application to N-cube networks[J]. IEEE Trans on Comput, 1989, 38(11): 1586-1591.

[4] LATIFI S, HEDGE M. Conditional connectivity measures for large multiprocessor systems[J]. IEEE Trans on Comput, 1994, 43(2): 218-222.

[5] BRUCK J, CYPHER R, SOROKER D. Tolerating faults in hypercubes using subcube partitioning[J]. IEEE Trans on Comput, 1992, 41(5): 599-605.

[6] CHEN Jian-er, WANG Guo-jun. Locally subcube-connected hypercube networks: theoretical analysis and experimental results[J]. IEEE Trans on Comput, 2002, 51(5): 530-540.

[7] 王雷,陈治平. 故障超立方体网络中的高效容错路由算法研究[J]. 计算机应用, 2005, 25(1): 4-6.

(上接第 1141 页)

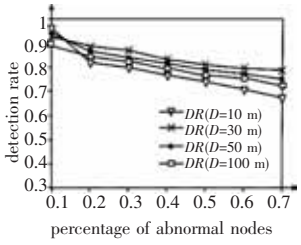

图 6　异常程度变化时的检测效果

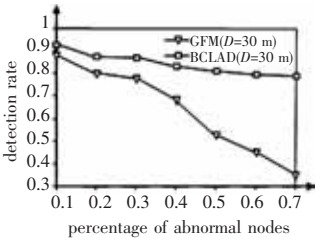

图 7　异常程度变化时的算法性能比较

4　结束语

本文研究定位异常检测,旨在发现由于错误的定位过程或恶意节点所造成的异常位置信息。针对传统检测方法存在的问题,本文引入了分簇式网络结构,不再由基站进行集中式检测,节点部署下去组合成簇后,由簇头进行本簇内的检测。这种分布式方法可以有效降低通信开销,而且不存在单点失效的风险。簇头只需根据传感器报告的位置 L_i 和邻居表信息对簇内节点采用 MFWT 算法检测定位是否存在异常,并确定异常节点上报给基站,不需要任何其他信息,实现简单。仿真实验表明,这种分布式的检测方法在各种网络环境中可以保证很好的性能,对研究传感器如何安全地定位具有一定的意义。

参考文献:

[1] 李建中,高宏. 无线传感器网络的研究进展[J]. 计算机研究与发展, 2008, 45(1): 1-15.

[2] 王福豹,史龙,任丰原. 无线传感器网络中的自身定位系统和算法[J]. 软件学报, 2005, 16(5): 857-868.

[3] 曹晓梅,俞波,陈贵海,等. 传感器网络节点定位系统安全性分析[J]. 软件学报, 2008, 19(4): 879-887.

[4] LIU Dong-gang, NING Peng, DU Wen-liang. Detecting malicious beacon nodes for secure location discovery in wireless sensor networks[C]//Proc of the 25th Conference on Distributed Computing Systems. Washington DC: IEEE Computer Society, 2005: 609-691.

[5] WEI Ya-wen, YU Zhen, GUAN Yong. Location verification algorithms for wireless sensor networks[C]//Proc of the 27th International Conference on Distributed Computing Systems. 2007: 70-77.

[6] SASTRY N, SHANKAR U, WAGNER D. Secure verification of location claims[C]//Proc of the ACM Workshop on Wireless Security (WiSe). New York: ACM Press, 2003: 1-10.

[7] CAPKUN S, HUBAUX J P. Secure positioning of wireless devices with application to sensor networks[C]//Proc of the INFOCOM 2005. 2005: 1917-1928.

[8] BAHL P, PADMANABHAN V N. RADAR: an in building RF-based user location and tracking system[C]//Proc of the INFOCOM 2000. 2000: 775-784.

[9] DU Wei-liang, FANG Lei, NING Peng. LAD: localization anomaly detection for wireless sensor networks[C]//Proc of the 19th IPDPS. Orlando, FL: Academic Press, 2006: 874-886.