

文章编号:1007-5321(2018)02-0075-06

DOI:10.13190/j.jbupt.2017-186

基于随机游走的无线传感器网络节点定位方法

尹雨晴¹, 高守婉¹, 王小旗², 牛 强¹

(1. 中国矿业大学 计算机科学与技术学院, 徐州 221116; 2. 河南理工大学 物理与电子信息学院, 河南 焦作 454000)

摘要: 为提高无线传感器网络中的节点定位精度,提出一种自适应随机游走模型的节点定位算法. 首先将随机游走应用于网络拓扑结构连通性中,构建节点间相对距离模型,并设计自适应算法,提高该模型有效性;然后将该模型嵌入经典定位算法 distance vector-hop (DV-Hop)中实现系统节点定位工作. 仿真和实验结果表明,该算法具有良好的鲁棒性和定位精度,误差比 DV-Hop 算法减少了 20% ~ 30%.

关 键 词: 节点定位; 无线传感器网络; 随机游走; 相对距离模型; 自适应算法

中图分类号: TN212.9

文献标志码: A

Node Localization with Random Walk for Wireless Sensor Networks

YIN Yu-qing¹, GAO Shou-wan¹, WANG Xiao-qi², NIU Qiang¹

(1. School of Computer Science and Technology, China University of Mining and Technology, Xuzhou 221116, China;

2. School of Physics and Electronic Information, Henan Polytechnic University, Henan Jiaozuo 454000, China)

Abstract: In order to improve node localization accuracy, a node localization algorithm based on adaptive random-walk module was presented for wireless sensor networks. First, a novel metric for relative distance among node sensors was modeled by applying the idea of random walk to the connectivity of system topology. Then an adaptive approach was designed to increase the validity of the metric. At last, node positions were finally obtained by embedding the metric in the classical localization algorithm distance vector-hop (DV-Hop). Simulation and outdoor environment's results show that the design achieves better robustness and positioning performance, and localization errors of the proposed method reduce by about 20% ~ 30% compared with that of DV-Hop algorithm.

Key words: node localization; wireless sensor network; random walk; a metric for relative distance; adaptive algorithm

无线传感器网络 (WSN, wireless sensor network) 中,交互信息的有效性是建立在获取节点位置信息基础上的. WSN 中节点定位技术主要着眼于定位时是否需要获得节点间的距离信息,分为基于测距的定位算法和无需测距的定位算法^[1-2]两大类. 随机游走^[3]指无规则行走者在时间间隔内以相同的概率移动到相邻位置,主要应用于研究朋友推

荐技术^[4]和链路预测^[5]问题. 此外,在生物信息学领域,Cao 等^[6]利用随机游走提出了图像扩散性质概念,对蛋白质相互作用网络中未知蛋白质进行功能预测. Chen 等^[7]将图像扩散性质用于 WSN 节点定位中,但仿真过程设定随机游走步数始终固定,不能保证不同网络环境配置的适用性. 因此,笔者在此相关工作基础上,着重讨论随机游走步数和节点

收稿日期: 2017-09-15

基金项目: 江苏省自然科学基金项目 (BK20160274); 国家自然科学基金项目 (51404258, 51674255); 中国矿业大学大学生创新项目 (DC201734)

作者简介: 尹雨晴 (1992—), 女, 硕士生; 王小旗 (1982—), 男, 讲师, E-mail: wangxiaqi_hn@163.com.

定位误差之间的关系,提出一种基于自适应随机游走距离(ARD, adaptive random-walk distance)的WSN定位算法,使得在不同网络环境配置下都可以自适应调整步数以保障提高定位精度。

1 算法设计

本节介绍节点间相对距离模型和自适应游走步数的取值算法,给出优化相对距离模型完整流程,最终获取未知节点位置坐标。

1.1 相对距离模型

根据文献[7],不失一般性,设已知全体节点数量为 N 的WSN部署在二维平面空间内,该网络拓扑结构表示为无向图 $G=(V, E)$,其中全体节点集合为 $V=\{v_1, v_2, \dots, v_N\}$,并且 $|V|=N$,而 E 为边的集合。 A 无向图 G 的 N 阶邻接矩阵,其元素 a_{ij} 表示节点对 (v_i, v_j) 的连通关系。计算两点间随机游走距离(RD, random-walk distance)共有5个步骤。

1) 计算随机游走模型一步转移概率矩阵 $\mathbf{T}^{(1)}$

$\mathbf{T}^{(1)}$ 中每一项 $t_{ij}^{(1)}$ 表示节点 v_i 随机游走1步到达节点 v_j 的概率,由式(1)求得。

$$t_{ij}^{(1)} = \begin{cases} \frac{1}{N}, & a_{ij} = 1 \\ \sum_{l=1}^N a_{il} & \\ 0, & a_{ij} = 0 \end{cases} \quad (1)$$

2) 计算两点间多步转移概率

设随机游走步数为 $s(s \geq 2)$,则两点间多跳的转移概率记为 $t_{ij}^{(s)}$,表示节点 v_i 随机游走 s 步到达节点 v_j 的概率。由于每一步随机游走事件是相互独立的,一条路径的概率应表示为每跳转移概率的乘积,如果节点 v_i 到节点 v_j 存在多条 s 步路径,则多步转移概率为每条路径概率的加和。同时,所有节点对间 s 步转移概率可组成 s 步转移概率矩阵 $\mathbf{T}^{(s)}$,有 $\mathbf{T}^{(s)} = (\mathbf{T}^{(1)})^s$ 。

3) 计算两点间 s 步累加转移概率

随机游走过程中从第1步到第 s 步产生的两点间转移概率累加和即为 s 步累加转移概率,如式(2)所示。

$$m_{ij}^{(s)} = \sum_{l=1}^s t_{ij}^{(l)} \quad (2)$$

4) 定义节点 s 步累加概率向量

对于任意节点 v_i ,将其到所有节点的 s 步累加转移概率组成 N 维行向量即 $\mathbf{M}^{(s)}(v_i)$,如式(3)所示。

$$\mathbf{M}^{(s)}(v_i) = (m_{i1}^{(s)}, m_{i2}^{(s)}, \dots, m_{iN}^{(s)}) \quad (3)$$

5) 计算两节点间RD

随机游走 s 步后两点间RD为

$$D_{RD}^{(s)}(v_i, v_j) = \|\mathbf{M}^{(s)}(v_i) - \mathbf{M}^{(s)}(v_j)\|_1 \quad (4)$$

其中 $\|\dots\|_1$ 为 L_1 范式,即求向量差中各项绝对值之和。不难得出,当步数 s 为固定值时, $D_{RD}^{(s)}(v_i, v_j) = D_{RD}^{(s)}(v_j, v_i)$,并有 $D_{RD}^{(s)}(v_i, v_i) = 0$ 。

1.2 自适应 s 取值算法

1.2.1 提出问题

相对距离模型中随机游走步数需人为给定,不同 s 取值会造成RD计算值差异,进而影响最终节点定位效果,这里讨论2种极端情况($s=1$ 和 $s \rightarrow +\infty$)下的计算影响。

1) $s=1$

随机游走1步后,计算得出的节点累加概率向量中仅仅在相连通的邻居节点位置上有数据,其余皆为0值。这在大规模不均匀网络中,不仅可能会导致各节点的累加概率向量中出现大量无意义的0值,也会造成多数节点对RD计算值相同的情况。如果一个相同的RD对应不同的实际物理距离,会导致节点定位歧义性问题。

2) $s \rightarrow +\infty$

随机游走是马尔可夫链的一个应用,每一次转移过程与上一次过程有关系,与初始状态无关。在较长时间后,马尔可夫过程会存在一个稳定状态概率。也就是说,在 s 趋向于正无穷的过程中存在一个 $k(k < s)$,有 $\mathbf{T}^{(k)} = \mathbf{T}^{(k+1)} = \dots = \mathbf{T}^{(s)}$ ($\mathbf{T}^{(k)}$ 为中间状态转移概率矩阵)。这些数值相等项在第3步骤被累加 $s-k+1$ 次后会在最后一个步骤中全部消减去掉。由此可得出,当随机游走步数趋于无限大时,相对距离模型计算出的数值是恒定值。虽说最终结果稳定,但该取值方案根本无法实现。即使人为定义了一个较大的数值,但这不仅不能证明随机取值的最优化,也会大幅增加算法的复杂度。

上述2种极端取值方法并不可行,需要折中取值方案才能优化相对距离模型。

1.2.2 自适应 s 取值算法设计

利用穷举法的思路,提出了一个基于已知锚节点的自适应 s 取值算法。综合考虑算法复杂度和上述分析, s 的自适应取值循环范围划定为 $s \in \{2, 3, \dots, N_A\}$,即从2开始到锚节点个数 N_A 为止的正整数。设锚节点集合为 X ,算法过程如下:

第1步 将 s 置为初始值2。

第 2 步 利用所提出的模型计算各个锚节点对 (v_i, v_j) 的相对距离 $D_{RD}^{(s)}(v_i, v_j)$.

第 3 步 计算锚节点单位校正距离 U 即

$$U = \frac{\sum_{i \neq j, v_i, v_j \in X} d_{ij}}{\sum_{i \neq j, v_i, v_j \in X} D_{RD}^{(s)}(v_i, v_j)} \quad (5)$$

其中 d_{ij} 为锚节点对 (v_i, v_j) 间实际距离.

第 4 步 通过相对距离与单位校正距离相乘, 计算各锚节点对 (v_i, v_j) 的估计真实距离 \hat{d}_{ij} .

第 5 步 计算所有锚节点间估计距离与实际距离的偏差和, 并存储步数和相应数值.

第 6 步 置 $s = s + 1$, 并返回第 2 步.

循环结束后取出第 5 步中存储数据, 使偏差和最小的对应步数作为自适应 s 的确定取值, 记为 s^* .

1.3 ARD

综合 1.1 节与 1.2 节内容, 给出优化相对距离模型的算法流程. 首先, 对所有已知锚节点采用自适应 s 取值算法得到可以使相对距离模型效果最优化的准确取值 s^* . 其中锚节点间的一步转移概率矩阵记为 P_1 , 多步累加转移概率矩阵记为 $P_a^{(s)}$, 由式 $P_1 + P_1^2 + \dots + P_1^s$ 计算而得, 其每一行都表示各个锚节点的累加概率向量, 时间复杂度为 $O(N_a^4)$. 其次, 对所有传感节点计算随机游走 s^* 步下优化节点间相对距离, 时间复杂度为 $O(N^3)$. 其中节点间一步转移概率矩阵记为 $T^{(1)}$, 多步累加转移概率矩阵记为 $T_a^{(s^*)}$. 为区别于 RD, 将节点对 (v_p, v_q) 的优化相对距离命名为 ARD, 记 $D_{ARD}^{(s^*)}(v_p, v_q)$. 整个算法的时间复杂度为 $O(N_A^4 + N^3)$.

伪代码如算法 1 所示.

算法 1 ARD 模型算法

输入: Adjacency matrix A

```

for  $s$  from 2 to  $N_A$  with step 1 do
  for each anchor pair  $(v_i, v_j)$  do
    Compute the  $P_1$  of all anchors
    Compute  $P_a^{(s)} = P_1 + P_1^2 + \dots + P_1^s$ 
    Compute  $D_{RD}^{(s)}(v_i, v_j)$ 
     $= \| P_a^{(s)}(i, :) - P_a^{(s)}(j, :) \|_1$ 
  end for
  Compute the  $U$ 
  for each anchor pair  $(v_i, v_j)$  do
    Estimate the distance  $\hat{d}_{ij}$ 
  end for

```

$$\varepsilon(m) = \sum_{i \neq j} |\hat{d}_{ij} - d_{ij}|$$

end for

$$s^* = \operatorname{argmin}(\varepsilon(s))$$

for each node pair (v_p, v_q) do

Compute the $T^{(1)}$ of all sensor nodes

$$\text{Compute } T_a^{(s^*)} = T^{(1)} + (T^{(1)})^2 + \dots + (T^{(1)})^{s^*}$$

Compute $D_{ARD}^{(s^*)}(v_p, v_q)$

$$= \| T_a^{(s^*)}(p, :) - T_a^{(s^*)}(q, :) \|_1$$

end for

输出: Matrix of ARD among all nodes

1.4 基于 ARD 定位算法

相似于 DV-Hop^[8] 算法, 将优化相对距离替换跳数并提出 DV-ARD 定位算法. 大体思路是将相对距离转化成绝对距离, 并根据极大似然估计法求得位置坐标. 不同于 DV-Hop 最短路径, 优化模型可直接获得未知节点到每一个锚节点的相对距离. 由此, DV-ARD 仅需要将 DV-Hop 中锚节点的单位跳数校正距离替换为单位校正距离 U , 对此式 (5) 已做出详细说明, 并使用单位校正距离与相对距离乘积估计未知节点与锚节点间实际距离, 最终利用极大似然估计法定位未知节点坐标.

2 仿真评估

从定位效果图例中评估该模型的定位效果, 并从锚节点数量、节点密度和网络大小 3 个参数不同数值对定位误差的影响评估该模型的健壮性.

2.1 网络配置

在仿真中, 基于对数衰减通信模型^[9] 建立正方形的 WSN, 记为 $W(c, N, N_A, R)$. 其中, c 为网络大小长度 (宽度), 单位为 m; N 为网络节点总数; N_A 为锚节点个数; R 为通信半径.

特别声明的是, 定义 2 种误差分析参数评估定位性能. 一种是节点定位偏移 $L_d(v_i)$, 重点分析某节点 $v_i(x_i, y_i)$ 与其估计定位坐标 (x'_i, y'_i) 间的偏移程度, 如式 (6) 所示; 另一种是系统定位误差 L_e , 重点分析整个网络所有节点的定位误差, 如式 (7) 所示.

$$L_d(v_i) = \sqrt{(x_i - x'_i)^2 + (y_i - y'_i)^2} \quad (6)$$

$$L_e = \frac{\sum_{i=1}^N L_d(v_i)}{(N - N_A) \times R} \quad (7)$$

2.2 定位表现

本节比较 DV-Hop 与 DV-ARD 定位效果. 图 1 (a) 描绘了非均匀的网络 $W(200,100,30,50)$, 星形为锚节点, 圆点为待定位的未知节点. 右上角和左下角的区域内节点密度较大, 相应网络连通也更密集. 仿真数据表明 DV-ARD 比 DV-Hop 的最大、最小以及平均 L_d 均小了 1/4 左右. 图 1(b) 和 (c) 分别展示了 2 种算法的节点定位, 线段圆点端为未知节点真实坐标, 另一端表示该节点的估计位置, 线段长短表示定位偏移. 显而易见的是, 图 1(c) 中偏移线段数量与长度均略小于图 1(b) 中线段, 尤其是在节点密度较大的区域, DV-ARD 定位优势更加明显. 值得特别关注的是, 图 1(b) 中存在多条线段交于一

点的现象, 这正是跳数距离的离散映射带来了节点定位二异性问题, 而这一问题在 DV-ARD 中得到了很好的解决. 相应地, 系统误差 L_e (DV-ARD) 比 L_e (DV-Hop) 减少了约 14%.

2.3 参数影响

为验证 DV-ARD 定位健壮性, 引入 2 种基于跳数距离的节点定位算法 DV-Hop 和鲁棒定位算法 (RPA, robust positioning algorithm)^[10] 与所提算法进行误差分析比较, 同时加入 DV-RD 算法表示 $s = 20$ 固定值下定位方法, 比较分析自适应步数算法优越性. 图 2 应用于网络 $W(700,300,70,150)$ 中, 分别展示 3 种仿真参数对定位结果的影响. 其中, 横坐标分别代表不同参数取值, 纵坐标代表 100 轮仿

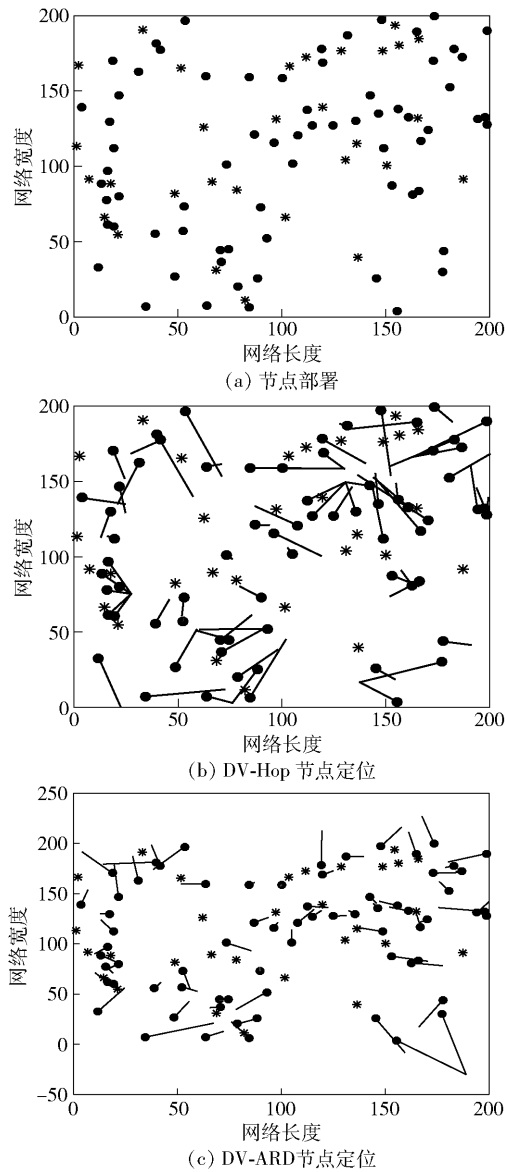


图 1 DV-Hop 与 DV-ARD 定位效果对比

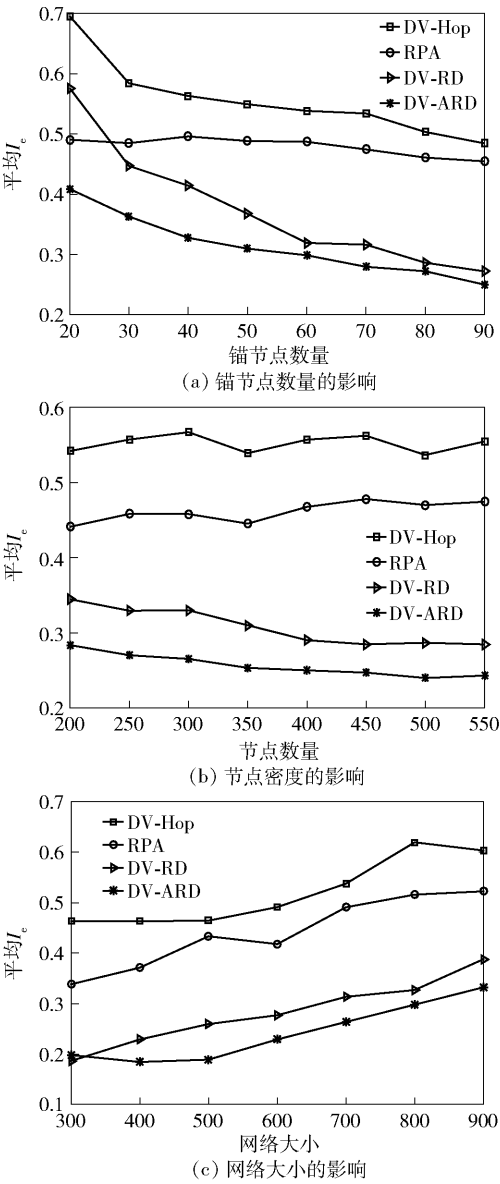


图 2 仿真参数对定位结果的影响

真结果的平均 L_e . 当一种参数取值变化时, 该网络其余参数均不变. 通过观察图 2 可得, 基于相对距离的定位算法的定位误差均小于基于跳数距离的定位算法, 这表明前者在不同网络环境下具有定位表现的优异性. 同时, 随机游走步数的自适应比固定化方法更加适用于各种 WSN.

1) 锚节点数量影响

图 2(a) 表明随着锚节点数量的增加, 各个算法都得到了定位效果的增进. DV-Hop 和 RPA 这 2 种基于跳数距离的算法增速较缓, 而基于相对距离 (RD 或 ARD) 的算法则效果改善速度较快. 相较于其他 3 种算法, DV-ARD 提高节点精度约 24%、16% 和 7.5%.

2) 节点密度影响

在图 2(b) 中, DV-Hop 和 RPA 算法的定位误差变化较平缓但有所增长, 说明节点密度的增大引发了跳数距离定位二异性的问题, 但影响较小. 反而基于相对距离定位的 DV-RD 和 DV-ARD 算法有微小的精度提高现象, 这更加验证了相对距离模型的健壮性. 相较于其他 3 种算法, DV-ARD 提高节点精度约 28%、20% 和 5.1%.

3) 网络大小影响

图 2(c) 表明随着网络大小的扩张, 由于通信半径保持不变, 各种算法可利用的网络连通信息变少, 其定位效果也相应变差. 相较于其他 3 种算法, DV-ARD 降低定位误差约 22%、18% 和 4.1%.

3 实验

3.1 实验设计

仿真部署了一个大规模非均匀的 WSN, 但无法真实模拟实际外界环境干扰, 因此室外实验只证实 DV-ARD 良好的鲁棒性. 实验在室外操场上采用 25 个 CC2530 节点部署了一个 5×5 的网格状 WSN, 横向或纵向节点间间距为 5 m, 如图 3 所示. 在实验中, 每个传感器节点会广播 80 个发送功率为 0 dBm 的数据包, 其中包含发送节点 ID 以及数据包的序列号信息. 如果接收节点接收到的信号强度大于 -100 dBm, 就把发送节点的 ID、相应的接收信号强度值、数据包序列号等信息写入到接收节点的闪存中; 相反, 此数据包就会被忽略掉. 所有节点中闪存数据将通过自身通信模块上传至服务器中, 利用 Matlab 软件即可处理获得所提算法所需的输入邻接矩阵.

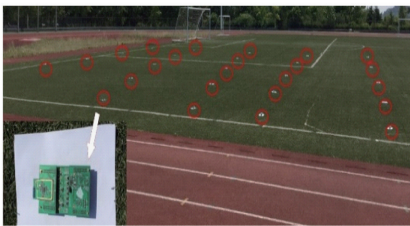


图 3 实验场景

3.2 实验结果

实验表明, 与传统定位算法 DV-Hop 相比, DV-ARD 节点定位算法具有良好的定位精度和鲁棒性特征. 表 1 比较了 2 种算法定位节点的平均、最大和最小偏移量, 计算后可得 DV-ARD 比 DV-Hop 的平均定位偏移值减少了约 26%.

表 1 定位精度比较表

偏移量	DV-Hop	DV-ARD
平均 L_d	7.21	5.29
最大 L_d	13.05	10.94
最小 L_d	1.54	0.50

4 结束语

笔者主要介绍了一种基于自适应随机游走步数的节点定位算法. 其主要思想是通过随机游走模型挖掘网络拓扑结构中连通性信息构建相对距离模型, 并设计自适应步数的方法进一步优化该相对距离模型, 最终通过嵌入经典算法替代跳数距离的方式实现节点的定位工作. 经过大量的仿真和实验验证, DV-ARD 算法在大规模网络尤其是密集区域具有良好的定位精度、健壮性和鲁棒性. 相较于其他经典的节点定位算法, 如 DV-Hop 算法, DV-ARD 算法降低了 20%~30% 定位误差.

参考文献:

[1] 谭志, 张卉. 基于节点间覆盖关系的改进 DV-Hop 算法[J]. 北京邮电大学学报, 2014, 37(1): 35-38.
Tan Zhi, Zhang Hui. Improved DV-Hop localization algorithm based on coverage of nodes[J]. Journal of Beijing University of Posts and Telecommunications, 2014, 37(1): 35-38.

[2] Zhong Ziguo, He Tian. RSD: a metric for achieving range-free localization beyond connectivity[J]. IEEE Transactions on Parallel & Distributed Systems, 2011, 22(11): 1943-1951.

[3] Fouss F, Pirotte A, Renders J M, et al. Random-walk

- computation of similarities between nodes of a graph with application to collaborative recommendation [J]. IEEE Transactions on Knowledge & Data Engineering, 2007, 19(3): 355-369.
- [4] Gong Jibing, Gao Xiaoxia, Cheng Hong, et al. Integrating a weighted-average method into the random walk framework to generate individual friend recommendations [J]. Science China Information Sciences, 2017, 60(11): 110-104.
- [5] Liu Weiping, Lu Linyuan. Link prediction based on local random walk[J]. A Letters Journal Exploring the Frontiers of Physics, 2010, 89(5): 58007-58012.
- [6] Cao M, Zhang H, Park J, et al. Going the distance for protein function prediction: a new distance metric for protein interaction networks[J]. PLoS One, 2013, 8(10): e76339.
- [7] Chen Pengpeng, Yin Yuqing, Gao Shouwan, et al. Localization with graph diffusion property [J]. Sensors, 2017, 17(7): 1636.
- [8] Niculescu D, Nath B. DV based positioning in Ad hoc networks [J]. Telecommunication Systems, 2003, 22(1): 267-280.
- [9] Zhong Ziguang, He Tian. Achieving range-free localization beyond connectivity [C] // International Conference on Embedded Networked Sensor Systems. Berkeley: ACM, 2009: 281-294.
- [10] Savarese C, Rabaey J M, Langendoen K. Robust positioning algorithms for distributed ad-hoc wireless sensor networks [C] // General Track of the Conference on USENIX Technical Conference. USENIX Association, Monterey: ACM, 2002: 317-327.