

# 基于曲线拟合的改进 DV-Hop 定位算法<sup>①</sup>

李云飞, 江明, 葛愿, 高文根

(安徽工程大学 检测技术与节能装置安徽省重点实验室, 芜湖 241000)

**摘要:** 针对传统的 DV-Hop(Distance Vector-Hop)定位算法存在定位精度不高的问题, 提出基于曲线拟合的改进定位算法. 在改进算法中, 以锚节点的实际直线距离和估计距离之间存在的误差作为样本之一, 结合误差所对应的节点间跳数, 采用曲线拟合找出两者存在的关系. 曲线拟合的结果, 用于估计和预测锚节点和未知节点之间的距离误差, 并对两者之间距离进行修正, 从而减小两者之间的距离误差, 降低未知节点定位误差. 仿真结果表明, 改进后的 DV-Hop 算法能够较好地修正了算法中关键的估计距离, 显著提高了对于未知节点的定位精度.

**关键词:** 平均跳距; 估计距离; 曲线拟合; DV-Hop 定位算法; 无线传感器网络

## Improved DV-Hop Localization Algorithm Based on Curving Fitting

LI Yun-Fei, JIANG Ming, GE Yuan, GAO Wen-Gen

(Anhui Key Laboratory of Detection Technology and Energy Saving Devices, Anhui Polytechnic University, Wuhu 241000, China)

**Abstract:** Aiming at relative inaccuracy in positioning accuracy of traditional DV-Hop (Distance Vector-Hop) localization algorithm, the improved DV-Hop localization algorithm based on curve fitting is proposed. The improved algorithm regards the error between the actual straight-line distance and the estimated distance as one of samples, and combines the corresponding hops to search the relationship between the hops and the corresponding distance error based on curving fitting. The results of curve fitting are employed to estimate and forecast the distance error between unknown nodes and the anchor nodes, then to correct the estimated distance and decrease the positioning error. The simulation results show that the improved DV-Hop algorithm can preferably estimate the vital distance, and significantly improve the positioning accuracy.

**Key words:** average hop distance; estimated distance; curving fitting; DV-Hop positioning algorithm; wireless sensor network

在无线传感器网络(Wireless Sensor Network, WSN)应用中, 位置信息对于网络中的节点来说, 具有十分重要的、基础的作用, 例如在传感器网络技术广泛应用的环境监测、远程医疗、国防军事等领域<sup>[1]</sup>, 传感器节点获得、传输的信息, 在不能明确表示信息发生位置的情况下, 是不具备实用性的<sup>[2]</sup>. 因此, 传感器网络节点的位置信息是不可或缺的.

现有的无线传感网络定位算法中, 主要存在两类定位算法, 其主要区别在于, 如何获得节点之间的距离信息. 由此, 主要可以分成基于测距和无需测距(Range-Free)的两类定位算法. 基于距离的定位算法常

用测距方法有 RSSI(Received Signal Strength Indicator), TOA (Time of Arrival), TDOA (Time Difference on Arrival), AOA (Angle of Arrival), 且定位精度相对较高, 同样要求节点硬件达到一定的条件. 无需测距的定位算法不依赖节点之间的测距和方位, 在成本上, 更加适用于大规模的网络. 基于无需测距的定位算法主要有质心算法、DV-Hop 算法、APIT 算法、MDS-Map 算法等<sup>[3]</sup>, 在现有技术中, DV-Hop 算法应用较为广泛.

文献[4]中给出了 WSN 定位中运用锚节点的一些主要定位算法的性能, 以及性能的比较. 文献[5]中提出的算法, 采用加权最小二乘法, 来减少锚节点在估

<sup>①</sup> 收稿时间:2014-09-03;收到修改稿时间:2014-10-15

计阶段的误差, 以此提高定位的精度. 文献[6]中在原算法的基础上, 提出了重新估算平均跳距的方法, 并且给出了在 C 形分布下的节点定位. 文献[7]中采用局部的平均跳距来代替全局的平均跳距, 提出了 PHDV-Hop, 降低了定位误差. 文献[8]针对原算法由于平均跳距带来的估计距离误差, 本文建立了数学模型, 采用曲线拟合的方法, 修正未知节点到锚节点的估计距离, 以期提高定位精度.

### 1 DV-Hop算法

美国路特格斯大学 (Rutgers University) 的 Niculescu 等人<sup>[9,10]</sup>最早提出 DV-Hop 定位算法, 主要通过已知的网络平均跳距来估计锚节点和未知节点之间的距离, 再结合数学计算方法完成定位过程<sup>[11,12]</sup>.

DV-Hop 算法主要由 3 个阶段组成. 第 1 阶段, 完成节点间信息的交换, 锚节点将信息传递至出自己外的所有节点, 使之获得与锚节点的跳数. 第 2 阶段, 网络平均跳距的估计和传播. 网络中锚节点在获得其他锚节点的相应信息后, 计算网络平均每跳距离, 然后将其作为校正值得广播至网络中. 所有网络节点只记录接收到的第一个校正值得, 并通过邻居节点转发, 接着根据记录的最小跳数, 计算到附近各个锚节点的跳段距离. 第 3 阶段, 完成坐标定位. 最后当未知节点获得与 3 个或更多锚节点间的距离时, 采用数学方法计算自身坐标.

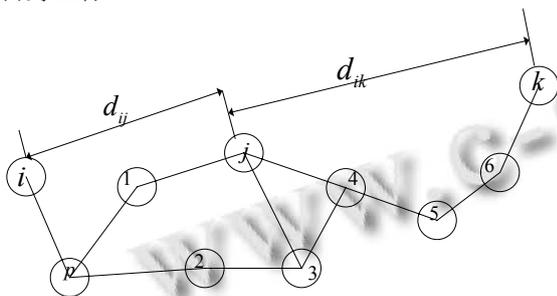


图 1 网络拓扑联通示意图

第  $i$  个锚节点  $(X_i, Y_i)$  的校正值得(平均跳距)  $avg_i$  可用(1)式表达:

$$avg_i = \frac{\sum \sqrt{(X_i - X_j)^2 + (Y_i - Y_j)^2}}{\sum hops_{ij}} \quad i \neq j \quad \forall j \quad (1)$$

式中  $(X_j, Y_j)$  表示锚节点  $j$  的坐标,  $hops_{ij}$  表示锚节点  $i$  与锚节点  $j$  之间的最短跳数. 如图 1 所示, 锚节点  $i, j, k$ , 锚节点  $i$  到  $j$  和  $k$  的距离已知, 分别为

$d_{ij}, d_{jk}$ , 跳数  $hops_{ij}, hops_{jk}$  分别为 3 跳和 4 跳, 那么:

$$avg_i = (d_{ij} + d_{jk}) / (hops_{ij} + hops_{jk}) \quad (2)$$

如图 1 中所示, 未知节点  $P$  从最近的锚节点  $i$  处获得校正值得, 则未知节点  $P$  与三个锚节点之间的估计距离分别为  $1 \times avg_i, 2 \times avg_i, 6 \times avg_i$ , 然后用三边测量定位法确定  $P$  的位置.

假设某一未知节点  $q$  的坐标为  $(x, y)$ , 测得能与  $n$  个锚节点通信, 且获得与它们的距离, 第  $i$  个锚节点的坐标为  $(x_i, y_i)$ , 到它的距离为  $d_i$ , 可以得到下面的式(3)所示的系统方程:

$$\begin{cases} (x_1 - x)^2 + (y_1 - y)^2 = d_1^2 \\ (x_2 - x)^2 + (y_2 - y)^2 = d_2^2 \\ \vdots \\ (x_n - x)^2 + (y_n - y)^2 = d_n^2 \end{cases} \quad (3)$$

将上述二元二次方程组转换为  $AX=b$  的形式. 其中:

$$A = \begin{bmatrix} 2(x_1 - x_n) & 2(y_1 - y_n) \\ \vdots & \vdots \\ 2(x_{n-1} - x_n) & 2(y_{n-1} - y_n) \end{bmatrix} \quad (4)$$

$$b = \begin{bmatrix} x_1^2 - x_n^2 + y_1^2 - y_n^2 + d_n^2 - d_1^2 \\ \vdots \\ x_{n-1}^2 - x_n^2 + y_{n-1}^2 - y_n^2 + d_n^2 - d_{n-1}^2 \end{bmatrix} \quad (5)$$

$$X = \begin{bmatrix} x \\ y \end{bmatrix} \quad (6)$$

方程  $AX=b$  可用标准的最小二乘法解出:

$$X = (A^T A)^{-1} A^T b \quad (7)$$

### 2 DV-Hop算法的改进

本文对上述 DV-Hop 算法进行了改进, 改进算法的主要特点在于采用曲线拟合方法, 对锚节点之间的实际直线距离和估算距离之间存在的误差, 称为距离差, 以及所对应的节点间跳数进行分析处理, 并将拟合曲线结果估计未知节点和锚节点之间的距离误差, 并修正两者之间的距离, 提高定位精度.

在工程技术中常常需要根据实验数据求变量间的函数关系, 或根据测量的坐标求出某条曲线的方程. 曲线拟合的方法主要分为: 多项式拟合和非线性曲线拟合<sup>[13]</sup>. 在本文中, 由于跳数的重复性, 即每个节点会有不同的跳数, 多个节点跳数重复, 跳数与距

离差的关联特性可以看作线性模型. 因此, 本文采用多项式拟合. 在多项式拟合中, 常用的方法为最小二乘法. 在偏差足够小的条件下, 最小二乘法不要求拟合的曲线严格的经过描绘的数据点, 且很好的反应数据的趋势, 能够满足本文中后续数据的估计和修正需求.

### 2.1 DV-Hop 定位算法问题描述

传统 DV-Hop 算法中, 平均跳距的估计是基于锚节点之间的物理直线距离. 如图 2 所示, 锚节点  $j$  的平均跳距的获得是基于锚节点之间的物理直线距离, 以及相应的跳数, 并将获得的平均跳距用于估计锚节点  $i$  其他节点之间的距离.

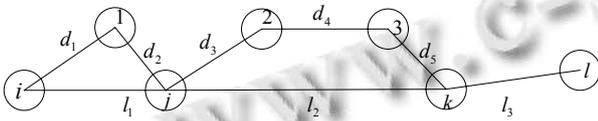


图 2 网络拓扑结构图

如图 2 所示, 由上述情况, 计算出来锚节点  $i$  的平均跳距:

$$AHD_i = \frac{d_{AB} + d_{AC} + d_{AD}}{2 + 5 + 6} = \frac{3l_1 + 2l_2 + l_3}{13} \quad (8)$$

假设未知节点 1 获得锚节点  $i$  的平均跳距, 则未知节点 1 在估计到锚节点  $i$  和锚节点  $j$  的距离时:

$$d_{est} = AHD_i \cdot Hops = \frac{6l_1 + 4l_2 + 2l_3}{13} \quad (9)$$

假设  $l_2 > l_1 > l_3 > R$ , 则有:

$$\begin{aligned} d_{error} &= |l_1 - d_{est}| \\ &= \left| l_1 - \frac{6l_1 + 4l_2 + 2l_3}{13} \right| \\ &= \left| \frac{7l_1 - 4l_2 - 2l_3}{13} \right| > \left| \frac{7l_1 - 6l_3}{13} \right| > \frac{l_3}{13} \end{aligned} \quad (10)$$

可以看出这种估计方法, 所产生的估计距离, 必然会与物理直线距离之间产生误差, 称为距离差. 从 (10) 式可以看出, 产生的距离差与所经过跳数成一定的相关性. 如果未知节点到锚节点  $j$  的跳数增大到 2, 则会有:

$$\begin{aligned} d_{error} &= |l_1 - d_{est}| \\ &= \left| l_1 - \frac{9l_1 + 6l_2 + 3l_3}{16} \right| \\ &> \left| \frac{7l_1 - 9l_3}{13} \right| > \frac{2l_3}{13} \end{aligned} \quad (11)$$

随着定位阶段跳数的增多, 必然会带来距离差的增大, 使得定位误差更大. 为了在定位过程中, 最大程度的减少这种情况带来的误差, 本文提出了基于曲线拟合的改进方法. 通过曲线拟合, 逼近的呈现出跳数与距离差的相应的关联特性, 并作为参考, 修正未知节点和锚节点之间的距离误差.

### 2.2 改进 DV-Hop 定位算法

在 DV-Hop 定位算法第二步中, 在锚节点获得网络平均跳距后, 每个锚节点计算与其他锚节点之间的估计距离之间的误差, 即距离差, 如下所示:

$$dist_{ij} = avg_i \times hops_{ij} - d_{ij} \quad i \neq j \quad (12)$$

式(10)中,  $i, j$  分别表示锚节点  $i$  和  $j$ ,  $d_{ij}$  表示两者之间的已知实际距离. 由于锚节点的数量和分布情况不同的原因, 同一个跳数会对应多个不同的距离差, 因此, 将不同跳数对应的各个距离差, 分别进行分类, 如下:

$$p_n = \{dist_1, dist_2, \dots, dist_m\} \quad n = 1, \dots, hops \quad (13)$$

式(11)中,  $m$  表示该集合中距离差的个数. 针对一个跳数中的距离差进行处理, 即对每个分类集中的距离差进行平均加权处理, 如下所示:

$$s_n = \frac{\sum_{i=1}^m dist_i}{m} \quad n = 1, 2, \dots, hops \quad (14)$$

这样就获得了每个跳数对应的距离差的平均值, 能有效的体现距离差和相应跳数之间的相关性. 将  $p_n, s_n$  分别作为函数类以及目标函数进行曲线拟合,

$$s(x) = a_0 \varphi_0 + a_1 \varphi_1 + \dots + a_n \varphi_n \quad (15)$$

使得函数

$$J(a_0, a_1, \dots, a_m) = \sum_{i=1}^m \omega_i^2 = \sum_{i=1}^m \omega(x_i) [s(x_i) - f(x_i)]^2 \quad (16)$$

成最小值.  $\omega(x)$  为权函数. 在权函数  $\omega(x) = 1$  时, 问题转变成  $\{a_0, a_1, \dots, a_n\}$  使  $J(\varphi_0, \varphi_1, \dots, \varphi_m)$  最小.  $\{a_0, a_1, \dots, a_n\}$  即在本文

中所求得的函数系数:

$$s(x) = \sum_{i=1}^n p_n \varphi_n \quad (17)$$

采用曲线拟合找出跳数和相应距离误差之间存在的近似高阶关系. 在 DV-Hop 定位算法第三步中, 在未知节点进行定位, 确定与锚节点之间距离时, 利用已经获得的曲线拟合关系式, 来修正估算的距离, 如下所示:

$$Dist_{ij} = avg \times hops_{ij} - s(hops_{ij}) \quad (18)$$

式(14)中,  $hops_{ij}$  表示未知节点  $i$  与锚节点  $j$  之间的跳数,  $avg$  表示离未知节点  $i$  最近的锚节点的平均跳距. 在完成修正后, 将修正的估算距离计算节点的位置坐标.

表 1 改进 DV-Hop 算法的流程

<p>第 1 阶段, 完成节点间信息的交换, 锚节点将信息传递至出自己外的所有节点, 使之获得与锚节点的跳数;</p> <p>第 2 阶段, 网络平均跳距的估计和传播.</p> <p>1. 网络中锚节点在获得其他锚节点的相应信息后, 计算网络平均每跳距离;</p> <p>2. 在锚节点获得网络平均跳距后, 每个锚节点计算与其他锚节点之间的估计距离之间的误差, 针对同一跳数中的距离差进行处理, 即对每个分类集中的距离差进行平均加权处理;</p> <p>3. 将上述处理结果分别作为函数类以及目标函数进行曲线拟合, 采用曲线拟合找出跳数和相应距离误差之间存在的近似高阶关系;</p> <p>第 3 阶段, 完成坐标定位. 利用已经获得的曲线拟合关系式, 来修正锚节点与未知节点之间估算的距离, 最后当未知节点获得与 3 个或更多锚节点间的距离时, 采用数学方法计算自身坐标.</p>
---

本文对曲线拟合采用多次仿真的结果进行比较, 进行残差分析, 筛选结果最优的拟合曲线, 即最逼近的拟合结果. 如图 3 所示, 是十次仿真的  $p_n, s_n$  分布散落以及曲线拟合情况, 从图中可以看出,  $p_n, s_n$  的分布散落是没有明显直接的观测性的, 拟合出来的结果表明该拟合曲线有效地覆盖  $p_n, s_n$  所散落的分布点. 如图 5 所示, 是百次仿真的  $p_n, s_n$  分布散落以及曲线拟合情况.

从图中可以看出,  $p_n, s_n$  的分布散落不具有任何明显特性, 但拟合结果曲线能有效地覆盖出现的样本点, 在排除孤立锚节点造成的非有效点后, 通过两种不同次数的曲线拟合结果的对比, 在获得数据样本

较充分的条件下, 可以得到本文所提出的距离差和相应的跳数之间存在近似高阶关系, 验证了数学模型的正确性.

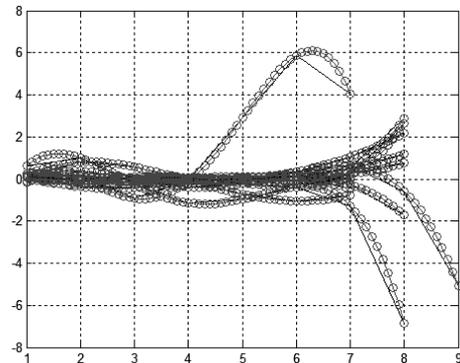


图 3 10 次  $p_n, s_n$  分布散落及曲线拟合图

表 2 10 次分布散落图曲线拟合的曲线系数

系数序号	$p_1$	$p_2$	$p_3$	$p_4$	$p_5$	$p_6$	$p_7$	$p_8$	$p_9$
1	0	0	0	0.02	-0.2	1.18	-2.3	1.5	
2	0	0	0	0	0	-0.1	0.15	0.04	
3	0	0	0.01	-0.2	1.03	-2.1	1.4		
4	0	0	0	0	0	0.4	-2.2	4.9	-3.2
5	0	0	0.04	-0.1	0.05	0.12			
6	0	0	0	0.05	-0.2	0.4	-0.3		
7	0	0	0	0.01	-0.2	1.3	-2.7	1.7	
8	0	0	0	0	-0.1	0.89	-2.5	2.04	
9	0	0	0	0	0.13	-0.6	0.93	-0.1	
10	0	0	0	0.1	-1.0	3.01	-1.9		

在表 2 中, 分别给出了十次曲线拟合的系数, 其中 0 因表达不便, 均为近似值. 为了尽可能的减少偏差, 所有的曲线拟合的阶数均有最大跳数决定, 即:

$$f(x) = \sum_{i=1}^n a^i x^i \quad (19)$$

其中  $n$  为当次曲线拟合的最大跳数, 即最大阶数.

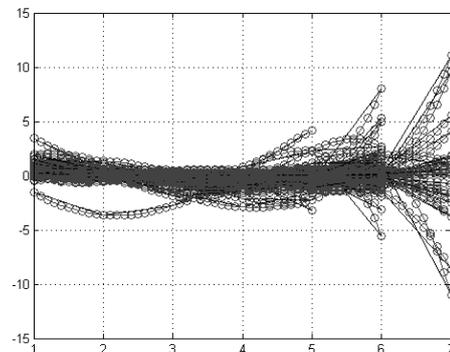


图 4 百次  $p_n, s_n$  分布散落及曲线拟合图

### 3 算法仿真及分析

为了检验算法的性能和先进性, 在 MatLab 中同时对三种算法进行仿真, 并对所出现的仿真结果进行评估和分析. 其中, ‘改进 DV-Hop’是本文提出的改进算法. 网络中所有节点都随机分布在特定的区域 (100×100). 两种算法性能的评估, 都是基于相同条件下, 呈现不同锚节点密度、节点通信半径和不同节点数量对两种算法的性能影响. 每种性能的仿真基于 100 次仿真结果的平均值.

定义节点定位误差为  $\delta$ , 如下所示:

$$\delta = \frac{\sqrt{(x_{real} - x_{estd})^2 + (y_{real} - y_{estd})^2}}{R} \quad (20)$$

式中  $(x_{real}, y_{real})$  是未知节点实际的坐标,  $(x_{estd}, y_{estd})$  是估计出来的未知节点的坐标, R 表示节点的通信半径. 节点平均定位误差为已定位为未知节点的误差和与被定位的未知节点个数比值. 如图 6 所示, 是随机部署的节点分布图, 其中实心点表示为未知节点, 星形表示锚节点.

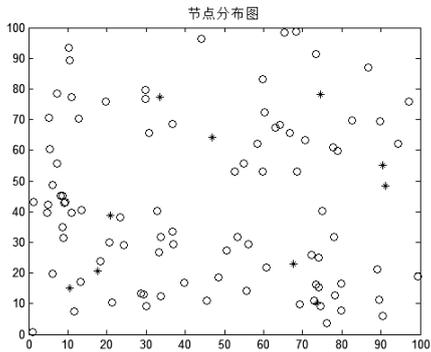


图 5 WSN 随即部署的节点分布图

#### 3.1 结果分析

下面图 6 和图 7 仿真在监测区域内随机部署了 100 个节点, 节点通信半径变化范围均为 20-45, 锚节点的个数分别是 8、20、30、.....、40, 两算法在不同的锚节点密度条件下的定位性能. 横轴为锚节点的个数, 纵轴为定位误差.

如图 7 中所示, 随着锚节点数目和通信半径的增加, 两种算法的平均定位误差均变化基本一致, 且均呈现向一侧减小倾斜的趋势.

在同一的锚节点数目下, 随着通信半径的增大, 两种算法的定位误差均下降, 且改进算法的曲线要低于原算法的曲线, 而且, 随着通信半径的增大, 两种

算法之间差距不断减少. 随着半径的增大, 使得节点间跳数相对减少, 样本数相对减少, 曲线拟合所获得数据量减少, 曲线拟合对距离差和相应跳数之间的关联特性描述不足, 导致对修正的程度有限, 优化程度不再大幅度增加. 因此, 在通信半径增加到一定程度, 优化算法的定位精度不在明显增加.

在同一通信半径下, 随着锚节点数目的增大, 两种算法的定位误差均下降, 且改进算法的曲线要低于原算法的曲线. 当锚节点个数增多时, 锚节点分布密度增大, 所获得的距离差和跳数样本逐渐增多, 曲线拟合能获得更多数据, 拟合结果能够很好地描述距离差和相应跳数之间的关联特性, 符合前面建立的模型, 对估计距离修正效果明显增强, 因此, 改进的算法会优化与原算法, 且随着锚节点的增多而增加.

如图 8 所示, 随着锚节点增加, 两条曲面之间差距逐渐增大, 正好说明这个问题.

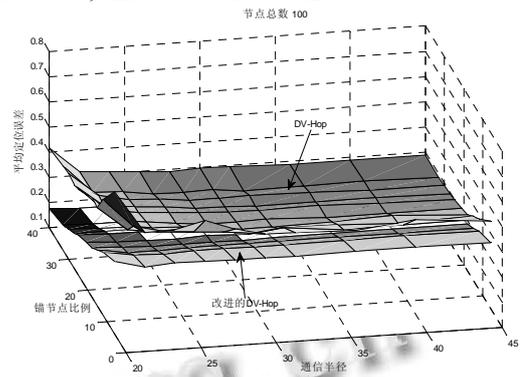


图 6 平均定位误差随通信半径和锚节点比例变化曲面上视图

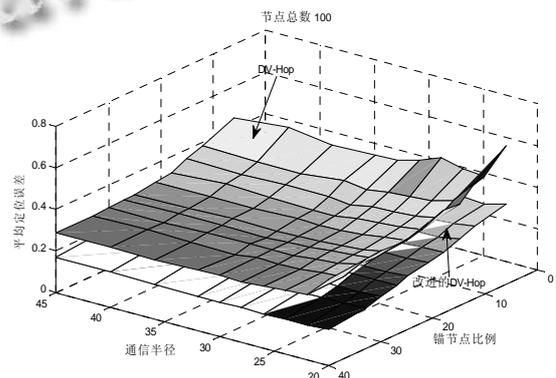


图 7 平均定位误差随通信半径和锚节点比例变化曲面前视图

如图 8 和图 9 所示, 节点总数变化范围在 50-200, 通信半径分别为 20, 22, ..., 45, 锚节点的比例保持 20%不变.

如图 8 所示, 两种算法在通信半径和节点总数变化的条件下, 定位误差均呈现减少倾斜的趋势, 且改进算法要优于原算法。

如图 8 所示, 在节点总数很小的条件下, 在节点的通信半径比较小时, 原算法定位误差很大, 如图出现的峰值, 同时, 由于通信半径的相对较小, 增大了节点跳数, 使得样本数增大, 在此处对估计距离的修正效果明显, 使得优化算法明显优于原算法; 在节点总数很大的条件下, 随着节点的通信半径的增大, 使得曲线拟合所获得数据样本足够多, 甚至基本不影响拟合曲线的方程和偏差, 导致优化算法所需的样本数量的增大不明显, 如图所示, 在节点总数很大时, 随着通信半径的增大, 两种算法之间的差距只有很小的变化。

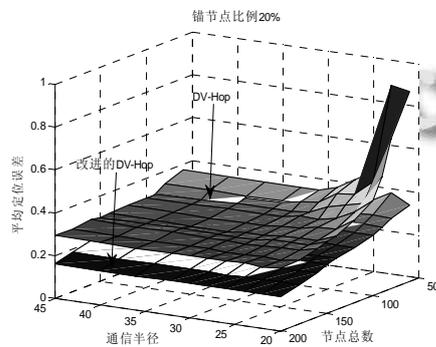


图 8 平均定位误差随节点通信半径和节点总数变化曲面上视图

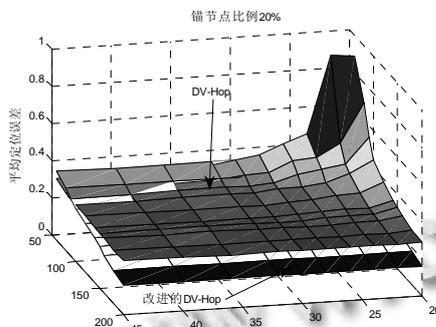


图 9 平均定位误差随节点通信半径和节点总数变化曲面左视图

多个仿真结果表明了本文所提出的理论模型的可行性、有效性和先进性, 同时, 在锚节点稀疏和通信半径较大的情况下, 还可以对改进算法和原算法的性能进行进一步的探索。

#### 4 结论

本文在分析传统 DV-Hop 算法的缺陷的基础上, 提出一种基于曲线拟合改进的 DV-Hop 定位算法。在

明确的理论分析和仿真结果条件下, 优化的算法, 不增加额外硬件设备和算法复杂度, 引入合理、可靠地估计模型, 有效地估计和预测原算法中的误差, 与原定位算法和现有改进算法对比, 在继承原算法简单、低成本优点的前提下, 明显降低了定位误差。

#### 参考文献

- 1 孙利民, 李建中, 陈瑜, 等. 无线传感器网络. 北京: 清华大学出版社, 2005.
- 2 Yick J, Mukherjee B, Ghosal D. Wireless sensor network survey. *Computer Networks*, 2008, 52 (12): 2292–2230.
- 3 Niculescu D. Positioning in Ad Hoc sensor networks. *IEEE Network*, 2004, 18 (4): 24–29.
- 4 唐弢. 基于锚节点的无线传感器网络定位技术研究[学位论文]. 哈尔滨: 哈尔滨工业大学, 2013.
- 5 Kumar S, Lobiyal DK. An advanced DV-Hop localization algorithm for wireless sensor networks. *Wireless Personal Communications*, 2013, 71: 1365–1385.
- 6 Tomic S, Mezei I. Improved DV-Hop localization algorithm for wireless sensor networks. *IEEE 10th Jubilee International Symposium on Intelligent Systems and Informatics*. September 20–22, 2012. 389–394.
- 7 Wang RJ, Zhang B, Shen Y, Qin ZG, Wang JH. PHDV-Hop: A more accurate DV-Hop positioning algorithm in WSN. *International Journal of Digital Content Technology and its Applications*, 2012, 6(13): 89–97.
- 8 Zhang J, et al. node localization algorithm based on DV-HOP in wireless sensor networks. *Applied Mechanics and Materials*, 2013: 440, 289.
- 9 Niculescu D, Nath B. DV based positioning in Ad Hoc networks. *Journal of Telecommunication System*, 2003, 22 (14): 267–280.
- 10 Yu XD., Zhang MY., Zhu MQ., Xu KH. and Xiang QC. Improved DV-Hop algorithm for enhancing localization accuracy in wsn. *Applied Mechanics and Materials*. 2014, 543–547: 3256–3259.
- 11 Niculescu D, Nath B. DV based positioning in Ad Hoc networks. *Journal of Telecommunication Systems*, 2003, 22(14): 267–280.
- 12 Niculescu D, Nath B. Ad Hoc positioning system (APS) using AOA. *Proc. of the 22nd Annual Joint Conference of the IEEE Computer and Communications Societies*, March 30–April 3, 2003. Rutgers University, Piscataway. 1734–1743.
- 13 刘建勇, 刘华. 曲线拟合分类与求解. *知识丛林*, 2007, 23: 144–145.