

题目：基于 Taylor 展开的多边测距定位及其优化

队员 1 信息

姓名： 楼家澍

专业： 数理金融

学号： 2019191138

上课时间： 周四下午6-8

队员 2 信息

姓名： 邱文宣

专业： 数理金融

学号： 2019191137

上课时间： 周四下午6-8

队员 3 信息

姓名： 卢泽海

专业： 数理金融

学号： 2019193018

上课时间： 周三下午6-8

基于 Taylor 展开的多边测距定位及其优化

摘要

在众多应用中例如目标跟踪，环境监测，智能建筑等都需要对应采集位置技术，而较为成熟的 GPS 技术成本较高，且不适用于室内定位，故无线传感器网络应定位是一个非常重要且具有挑战性的实际问题，本文基于无线传感器网络原理，构建无线传感器网络定位技术模型。

无线传感器网络中的节点定位需要我们根据网络中少数已知节点的位置信息，通过一定的定位技术确定网络中其他节点的位置信息。在现实情况下，由节点以及锚点构成的网络是连通而非完全连通的，故首先我们采用了锚点广播算法对锚点与节点之间的邻接矩阵进行扩展，使得对于的 SA 矩阵，其 L-0 范数等于矩阵的元素个数。此时的拓补网络是一个完全连通的网络。我们通过比较包括 DV-hop、Amorphous、Shortest-Path 算法在内的几种常见锚点广播算法。最终通过比较他们预测的平均误差，选择了 DV-hop 作为后续使用的锚点广播算法。

基于上述所得到的 SA 矩阵，利用一阶 Taylor 展开所得到的多边测距定位迭代算法进行估计计算。首先通过预实验确定迭代的停止条件与迭代最大轮数，使得算法的稳定性与鲁棒性更好。之后，我们得到了基于 4 组数据的误差值。与市场上准确度最高的竞品相比较，发现还存在一定的优化空间。

为了对模型进行优化，我们取 data1 各点的误差数据进行了分析。发现只与一个锚点相连且距离很近；与其他节点相隔甚远的节点存在较大的误差。我们猜测这是由于多边测距定位算法初始点的选择所导致的。故我们在锚点坐标的基础上将横纵坐标偏移一个正态随机噪声，在此基础上进行迭代。结果发现各组数据的预测准确度都上升了 20% 左右，说明模型被显著优化了。

最后，我们随机生成约 5000 组数据，对模型进行压力仿真测试。模型预测的成功率为 93.7%，平均误差为 5.432%。模型的稳定性良好，存在应用价值。随后我们对 4000 个用户的坐标进行预测，共耗时 60.08 秒，耗时较长。经计算发现，该模型的时间复杂度为 $O(n^3 + n^2)$ ，时间复杂度较差，存在进一步优化的空间。距离投入使用还有一段距离

关键词：Taylor 展开 锚点广播 多边测距定位

一、问题重述

无线传感器网络是信息通讯中的核心攻坚技术，具有十分广阔的应用前景。但是资源和能量的受限使得无线传感器网络对定位的算法和定位技术都提出了较高的要求。基于此，我们将问题重述为下列 3 个子任务：

- 基于所给数据的特征，给出一个准确度较高，稳健性好的节点预测模型
- 将所给数据代入，计算并分析误差
- 模拟大量数据的应用场景下模型的稳定性

二、符号说明

符号	含义
\tilde{r}_{ai}	节点到锚点的距离
(x_{pi}^p, y_{pi}^p)	第 <i>i</i> 个锚点的坐标
$(\hat{x}_{pa}^{p-}, \hat{y}_{pa}^{p-})$	用户坐标的预测值
H_R^P	几何矩阵
$\begin{pmatrix} \hat{x}_{pa}^{p+} \\ \hat{y}_{pa}^{p+} \end{pmatrix}^T$	迭代后用户坐标
SA	节点与锚点的邻接矩阵
SS	节点和节点的邻接矩阵

三、问题分析

数据分别给出了锚点坐标，节点与锚点的邻接矩阵以及节点和节点的邻接矩阵。由于我们不知道待预测节点的真实坐标，所以我们采用题目中给出的距离，以及估计出来的坐标到锚点的距离作为我们的优化目标。

无线传感器网络中的节点定位需要我们根据网络中少数已知节点的位置信息，通过一定的定位技术确定网络中其他节点的位置信息。一般来说当我们获得未知节点到三个已知节点的距离和已知节点的坐标时，我们就可以通过多边测量法估计出未知节点的坐标，且已知节点数量越多，所估计出来未知节点的坐标就越精确。

而初始数据并不能提供足够的信息，即数据中并不存在某一未知节点与三个或三个以上的锚节点存在通信，故采用多边测量法估计坐标时，我们需先将 SA 矩阵扩展成一个不含 0 元素的矩阵。随后利用得到的新矩阵进行迭代计算。

在此基础上得到的预测结果可能存在一定误差，我们将根据高误差节点的具体特征进行具体分析，并尝试做一定优化。

考虑到这是一个实用度较高的模型，我们还将用大量数据做压力测试，探究该模型的稳定性以及时间复杂性。

四、模型假设

- 1、锚点和未知节点、未知节点和未知节点构成连通图。
- 2、不探究伪距的误差，不考虑由于大气折射等因素产生的误差，认为伪距可以代表真实距离。
- 3、锚点与节点相对较多，不出现过于离散的情况
- 4、用户在一定时间内不移动，即我们预测的是静止的节点

五、模型的建立

5.1 三边定位原理

5.1.1 锚点和节点及相关概念

如果将信号发射源视作一个一个的点，那么这些点可以被分为锚点和节点。所谓锚点是指坐标确定，且不会任意移动的信号发射源；而节点一般指用户，即坐标未知，但需要通过节点与节点、节点与锚点之间的沟通确定自己位置的点。

每个发射源具有一定的发射半径，在该半径内的其他发射源将可以接收到彼此的信号，并返回给彼此一个各自的距离。这个距离我们称为伪距。伪距是一个测量值，而非真实值。我们假设无线电信号以光速 c 传播，点A在 t_0 时刻发出信号，点B在 t_1 时刻接收到信号，那么可以求得伪距 $d = c * (t_1 - t_0)$ 。

伪距会由于大气折射、时钟误差等因素而产生与真实值的偏差。但此处我们暂时不探究伪距的误差，只是将其认为成一个定值。

5.1.2 通过伪距确定节点坐标

给定一个已知坐标的锚点 a_1 ，其坐标为 (x_1, y_1) 。又给定 a_1 到某节点 s_1 的距离为 d_1 。那么 s_1 的坐标 (x, y) 满足圆方程：

$$(x - x_1)^2 + (y - y_1)^2 = d_1^2$$

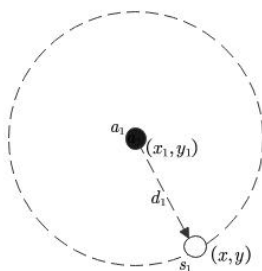


图 1 圆方程

如果此时 s_1 与另一锚点 a_2 也能通信，那么坐标 (x, y) 也应满足圆方程：

$$(x - x_2)^2 + (y - y_2)^2 = d_2^2$$

此时这两个圆交于点A、B。如果不考虑误差，那么 s_1 的位置必然应该是点A或是点B。但是如果仅有两个锚点的参与，我们并无法确定到底是点A还是点B，如果采取取中点的方式则必然存在误差。为了解决定位的模糊性，我们必须引入第三个锚点 a_3 ，当然由数学理论我们可以得出 a_3 所在圆应该交于A点或B点，但由于测距过程中存在一定噪声， a_3 产生的通信圆大致将区域AB分为两个部分，我们将面积更小的那一部分的质心确定为点 s_1 的坐标。可用的锚点越多，该区域的面积就越小，定位就更精准。

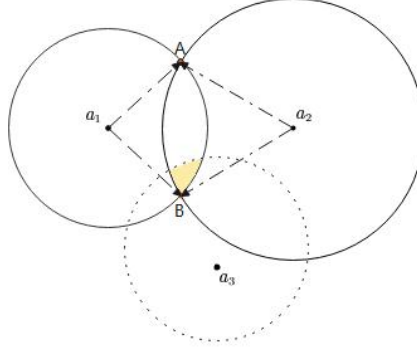


图 2 三边测距定位

我们称该算法为多边测距定位算法。接下来将介绍该算法的具体实现。

5.1.3 多边测距定位算法的实现

我们令节点到锚点的距离为 \tilde{r}_{ai} ,

$$\tilde{r}_{ai} = \sqrt{(x_{pi}^p - \tilde{x}_{pa}^{p-})^2 + (y_{pi}^p - \tilde{y}_{pa}^{p-})^2} \quad i \in 1, 2, 3$$

其中第 i 个锚点的坐标为 (x_{pi}^p, y_{pi}^p) , 用户坐标的预测值为 $(\hat{x}_{pa}^{p-}, \hat{y}_{pa}^{p-})$, 在迭代初始, 令用户坐标为 $(0, 0)$ 。

$$(1) \quad \hat{r}_{ai} = \sqrt{(x_{pi}^p - \hat{x}_{pa}^{p-})^2 + (y_{pi}^p - \hat{y}_{pa}^{p-})^2} \quad i \in 1, 2, 3$$

由于 \hat{r}_{ai} 对于 $\forall i$, 存在 \hat{r}_{ai}' 故利用

$$f(x) = \sum_{n=0}^{\infty} \frac{f^n(a)}{n!} (x-a)^n$$

对(1)左右泰勒展开, 得

$$(2) \quad (\tilde{r}_{ai} - \hat{r}_{ai}) = H_R^P \begin{pmatrix} \tilde{x}_{pa}^p - \hat{x}_{pa}^p \\ \tilde{y}_{pa}^p - \hat{y}_{pa}^p \end{pmatrix} + \begin{pmatrix} \sigma_{r_{l1}} \\ \sigma_{r_{l2}} \end{pmatrix} \text{ 其中 } H_R^P \text{ 为}$$

$$H_R^P = \left(\begin{array}{cc} \frac{\partial r_{a1}}{\partial x_{pa}^p} & \frac{\partial r_{a1}}{\partial y_{pa}^p} \\ \frac{\partial r_{a2}}{\partial x_{pa}^p} & \frac{\partial r_{a2}}{\partial y_{pa}^p} \end{array} \right) \bigg|_{(x_{pa}^p, y_{pa}^p) = (\hat{x}_{pa}^{p-}, \hat{y}_{pa}^{p-})} = \left(\begin{array}{cc} -\frac{x_{p1}^p - \hat{x}_{pa}^{p-}}{\hat{r}_{a1}^-} & -\frac{y_{p1}^p - \hat{y}_{pa}^{p-}}{\hat{r}_{a1}^-} \\ -\frac{x_{p2}^p - \hat{x}_{pa}^{p-}}{\hat{r}_{a2}^-} & -\frac{y_{p2}^p - \hat{y}_{pa}^{p-}}{\hat{r}_{a2}^-} \end{array} \right)$$

称为几何矩阵

整理 (2) 式得：

$$\begin{pmatrix} \hat{x}_{pa}^{p+} \\ \hat{y}_{pa}^{p+} \end{pmatrix} = \begin{pmatrix} \hat{x}_{pa}^{p-} \\ \hat{y}_{pa}^{p-} \end{pmatrix} + H_R^{p-1} \begin{pmatrix} \tilde{r}_{a1} - \hat{r}_{a1}^- \\ \tilde{r}_{a2} - \hat{r}_{a2}^- \end{pmatrix}$$

其中 $\begin{pmatrix} \hat{x}_{pa}^{p+} \\ \hat{y}_{pa}^{p+} \end{pmatrix}^T$ 即为在本次迭代后用预测用户坐标，对于 $n+1$ 轮迭代，只需令

$$\begin{pmatrix} \hat{x}_{pa}^{p-} \\ \hat{y}_{pa}^{p-} \end{pmatrix} = \begin{pmatrix} \hat{x}_{pa}^{p+} \\ \hat{y}_{pa}^{p+} \end{pmatrix}$$

重复上述计算即可。

5.2 锚点广播

多边测距定位算法需要在可用锚点数量大于等于 3 的时候才相对准确，而现实生活中锚点的无线传感器具有一定的无线电信范围，如果同时可以借助节点之间的通信，测量出节点之间的距离就可以更好的帮助我们实现定位。将节点与锚点视为一个拓扑网络，在假定该图是连通图的情况下，对于任意节点 s_k 与任意锚点 a_k 中必然存在一条路径，使得他们连通，记该路径的长度为 p 。根据三角形两边之和大于第三边，必然有分段路径长度 p 大于锚点与节点的直线距离。

所以，对于锚点与节点的邻接矩阵 SA ，我们可以利用节点与节点的邻接矩阵 SS ，将 SA 中的零元素通过分段路径长度 p 估算出来。

5.2.1 DV-hop 及其衍生算法

我们采用 DV-hop 算法来估计未知节点与锚点之间的直线距离

DV-Hop 定位算法可以分为以下 3 个阶段：

- 计算未知节点与每个信标节点的最小跳数。

信标节点向邻居节点广播自身位置信息的分组，其中包括跳数字段，初始化 0。接收节点记录具有到每个信标节点的最小跳数，忽略来自同一个信标节点的较大跳数的分组。然后将跳数值加 1，并转发给邻居节点。通过这个方法网络中的所有节点能够记录下到每个信标节点的最小跳数。

- 计算未知节点与信标节点的实际跳段距离。

每个信标节点根据第 1 阶段中记录的其他信标节点的位置信息和相距跳数，利用式 (1) 估算平均每跳的实际距离：

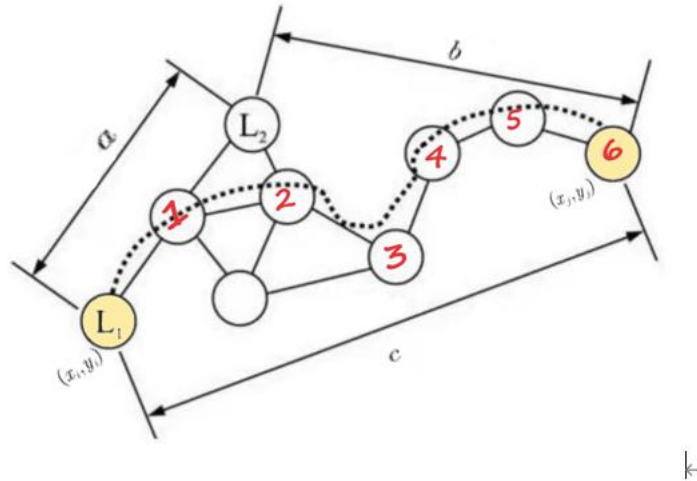


图 3 DV-hop

其中， (x_i, y_i) 、 (x_j, y_j) 是信标节点 i 、 j 的坐标， h_{ij} 是信标节点 i 与 j ($i \neq j$)之间的跳段数。

然后，信标节点将计算的每跳平均距离用带有生存期的字段的分组广播到网络中，未知节点仅记录接收到的第1个每跳平均距离，并转发给邻居节点。这个策略可以确保绝大多数未知节点从最近的信标节点接收每跳平均距离。未知节点接收到平均每跳距离后，根据记录的跳数，计算到每个信标节点之间的距离。

● 未知节点计算自身位置。

未知节点利用第2阶段中记录的到各个信标节点的跳段距离，利用三边测量法或极大似然估计法计算出自身坐标。

除此之外，我们还有 *Amorphous* 算法，将传感器的最大通信范围代替平均每跳距离；也可以采用 *dijkstra* 算法计算出两点之间的加权最短路径长度，并用该长度替代平均每跳距离。在下文中，我们将对每组数据进行三种方法的讨论与比较，采用估计坐标离传感器的距离与真实距离之间的均方误差作为我们的评判标准，并对方法进行选择。

5.3 模型性能与健壮性分析

模型可能产生的误差主要来源于锚点广播与三边测距法两方面。

在锚点广播中，我们利用折线距离来近似直线距离，由三角形有关性质可知：对于一个1跳的路径而言，当两条路径的夹角恰好为60度时，这种估计的误差最大，为50%。这是对于 Shortest-Path 算法而言的，对于 DV-hop 算法，由于其本质是将折线段之和乘以一个特定的系数 a ， $0 < a < 1$ ，故其误差将小于50%。

在三边测距法中，误差主要来源于三边测距法迭代轮数、初值条件以及终止条件的选取，在后续模型求解中我们将对这些内容进行优化，以求达到最佳的模型效果。

六、模型求解——以 data1 为例

6.1 数据分析与清洗

将数据读入 python，首先研究 anchor.mat，即传感器的位置。坐标如下：

表 1 锚点坐标

	x	y
A1	-0.3	-0.3
A2	-0.3	0.3
A3	0.3	-0.3
A4	0.3	0.3

我们发现四个传感器位于平面的四个角落，这提示我们越接近平面中心的节点可能拥有更差的预测效果。

对于 netsa 矩阵与 netss 矩阵，我们发现不存在某一行全为 0，即待预测的节点至少与一个锚点存在通信连接，这可以在某种程度上提高预测的准确度。当然后续我们也会遇到 netsa 矩阵存在全零行，即图中存在孤立点的情况。我们将 netsa 矩阵可视化，其中红色的点为锚点，蓝色的点为节点。得到下图：

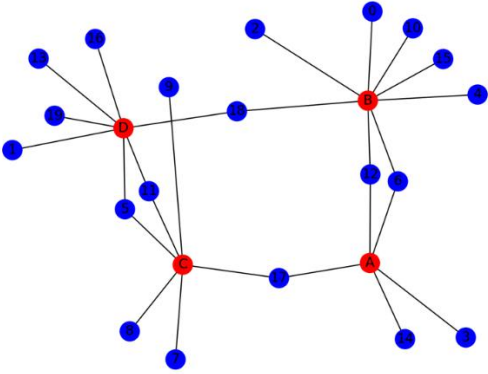


图 4 拓扑网络 1

接下来在图中加入节点与节点之间的联系，我们可以得到下图：

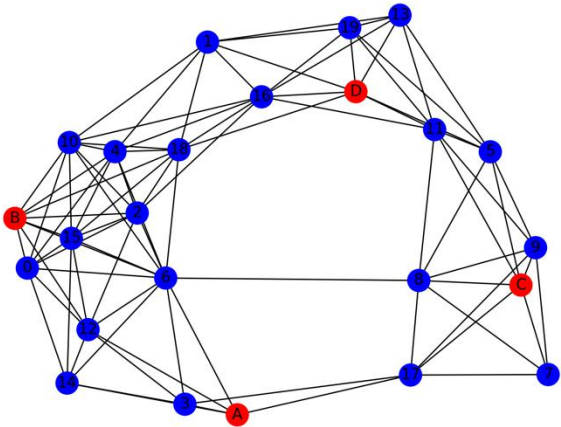


图 5 拓扑网络 2

需要指出的是，此处的图只表示节点与锚点之间的拓扑关系，还未考虑具体位置的因素。从图中我们大致可以发现，点与点之间的联系相对比较密切，但是没有一点与三个或三个以上的锚点存在通信。故接下来的三边测距定位法的准确性很大程度上依赖于锚点广播方法的选取。为了便于对比准确度，接下来先展示三边测距定位法的步骤，随后再对比不同广播方法对于测距定位准确度的影响。

6.2 三边测距定位算法与参数选择

由于三边测距定位算法本质上是一个迭代算法，所以我们必须先确定算法迭代停止的条件。又因为我们无法得知节点的真实坐标，所以实际上也无法划分训练集与测试集。我们唯一的已知条件是节点到某些锚点的距离，故我们就利用节点估计坐标到锚点的距离的均方误差作为损失函数，表达式如下：

$$f = (\tilde{r} - \hat{r})^2$$

当 $f(n+1) - f(n) < \epsilon$ 的时候，我们认为算法收敛；或是当 n 大于一个事先给定的 ϵ 时，我们认为达到最大迭代次数。这两种情况下，停止迭代。对于 data1，以 20 个节点的平均均方误差作为参考标准，不同的 ϵ 取值如下图所示：

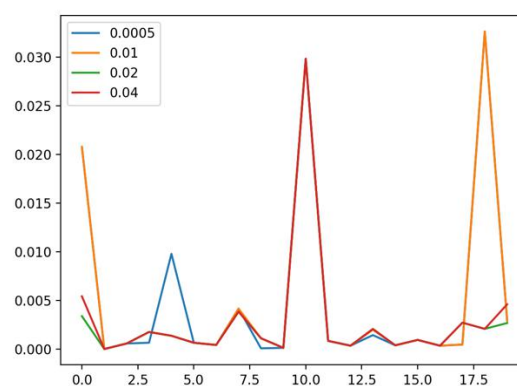


图 6 不同参数对比

由图中可以看出，当 $\epsilon = 0.02$ 时，对于不同数据的鲁棒性较好，故我们将 ϵ 确定为 0.02。

6.3 三种锚点广播方法对比

基于以上参数选择与 data1，三种锚点广播方法对比如下表所示：

表 2 三种广播算法对比

	Shortest-Path	Amorphous	DV-hop
<i>Mean</i>	0.00508	0.009246	0.004656
<i>std</i>	0.010029	0.01198	0.005964
<i>Min</i>	0.000001	0.000003	0.000057
0.25 分位数	0.00041	0.000553	0.000638
0.5 分位数	0.000888	0.005823	0.001923
0.75 分位数	0.002212	0.012221	0.00665
<i>Max</i>	0.032644	0.042833	0.020487

其每个节点估计值的均方误差折线图如下图所示：

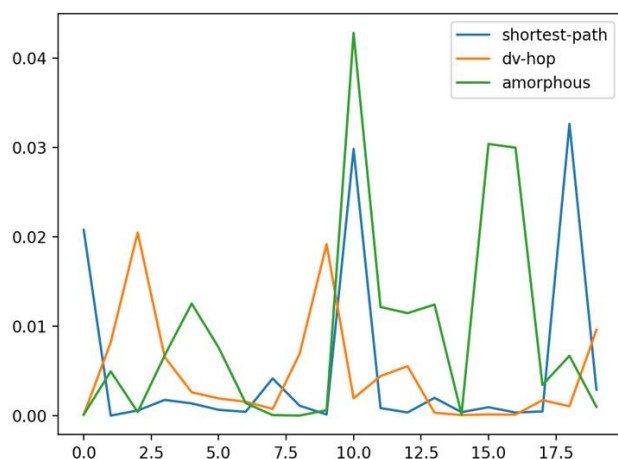


图 7 三种广播方法对比

6.4 结果与性能分析

综合考虑均值与标准差等因素，我们认为 dv-hop 算法是最为准确与鲁棒的。故对于 data1，我们采用 dv-hop 算法进行锚点坐标广播。随后使用三点测距定位算法进行坐标估计。得到结果，如下图所示：

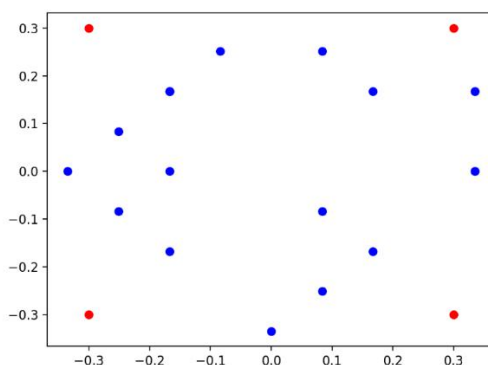


图 8 data 1 预测结果

其平均误差为 0.06823，即在一个 $[-0.5, 0.5] \times [-0.5, 0.5]$ 米的平面中，该算法的误差在 0.06823 米左右，误差率约为 6.823%。对比 Apple 公司最新推出的 Airtag 防丢器，该产品宣称可以在 5 米范围内实现 5-10 厘米的定位，即误差率在 1%-0.5% 左右。可见该算法性能还存在一定的优化空间。

对于不同的数据，我们得到结果如下：

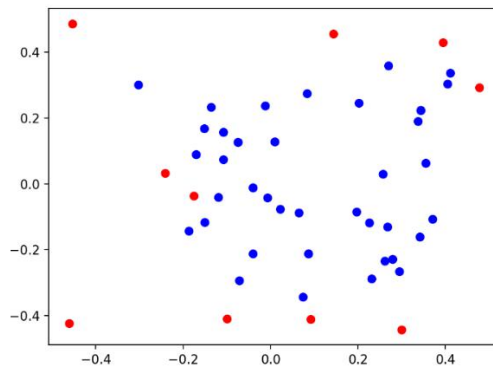


图 9 data 2 预测结果

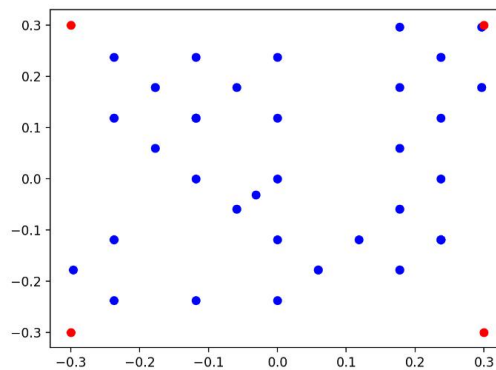


图 10 data 3 预测结果

在三维场景下，该模型依然适用，结果如下：

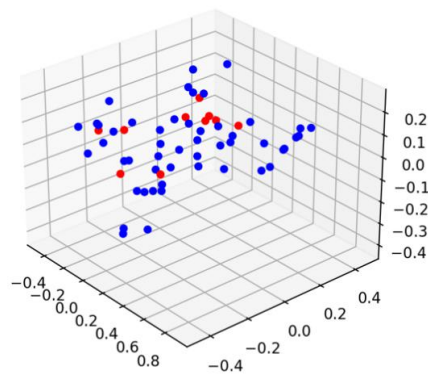


图 11 data 4 预测结果

对于 data1-4，模型性能如下图所示：

表 3 误差

	data1	data2	data3	data4 (3D)
误差率	6.82%	12.15%	8.34%	7.05%

可见模型对于不同数据存在较差的稳定性，这是由于点的分布具有较强的随机性，这种随机性导致容易存在一些异常点，下文将探讨异常点的处理与优化。

七、针对孤立点的模型优化

7.1 误差探究

我们取基于 dv-hop 方法与 data1 下的误差曲线，如下图所示：

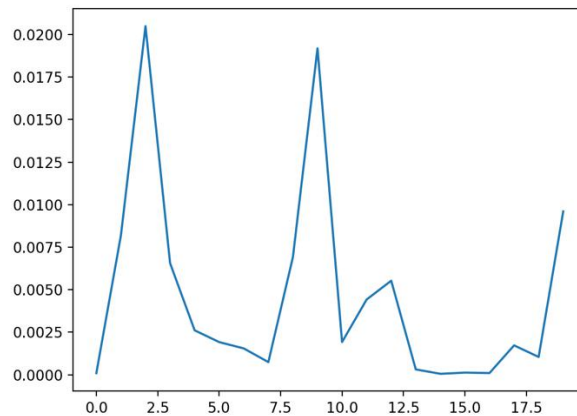


图 12 data 1 误差曲线

我们发现 s2 与 s9 这两个点的误差相当高，一个为 0.02048，一个为 0.01918。而 data1 误差的均值仅为 0.004656，相差近一个数量级。说明原模型对于某些特殊情况下的节点具有较差的预测效果。

随后我们从 SA 与 SS 矩阵中找到这两个节点的原始数据。在 SA 矩阵中，我们发现 s2 与 s9 都仅仅与一个锚点存在通信，且他们之间的距离一个为 0.04422，一个为 0.04884。而 SA 矩阵所有非 0 数据的平均值为 0.236914。再看 SS 矩阵，点 s2 与 8 个节点存在联系，而 s9 在 20 个节点中仅仅与 4 个节点存在联系。

综上，我们发现两个误差较大的节点存在如下特征：仅与一个锚点相连且距离很近；与其他节点相隔甚远。

7.2 误差原因分析

由于未知节点的坐标由三边测距定位法确定，而由于存在度小于 3 的点，我们采用锚点广播的方式令图变成完全图，所以误差就存在于锚点广播的过程中。在锚点广播的过程中，我们总是用某种大于两点直线距离的折线距离来替代直线距离，这使得我们所估算的距离相较于真实值总是偏大。而又因为这些异常点离某一锚点特别近，由于锚点分布比较分散，意味着它离其他的锚点相对较远，那么估计产生的误差就会随着跳段数的上升而上升。

7.3 基于随机噪声的迭代初值条件优化

由于基于 Taylor 展开的三边测距定位算法的精确度很大程度上取决于初值条件的选择，如果单纯将(0,0)作为迭代的初值条件的话，会使得上文所提到的特别接近锚点的坐标估计效果变差，在距离真实位置较远的地方算法容易收敛至局部极小值。所以我们需要在迭代开始之前先对于初值条件进行一次预估计。

对于那些离锚点距离小，且只与一个锚点相连的点，我们称之为孤立点。对于孤立点，我们本

可以选取离他最近的锚点作为迭代初值条件，但由于对于迭代方程而言，锚点坐标是它的不动点，将会使得算法无法收敛，故我们在锚点坐标的基础上将 xy 坐标偏移一个以 \tilde{r}_{ai} 为均值，标准差为 0.01 的正态分布噪声，该噪声代表在伪距测量中的误差，偏移后的初值条件 $(\hat{x}_{pa}^p, \hat{y}_{pa}^p)$ 如下：

$$(\hat{x}_{pa}^p, \hat{y}_{pa}^p) = (x_{pa}^p + X, y_{pa}^p + X), \text{ 其中 } X \sim N(\tilde{r}_{ai}, 0.01)$$

基于该初值条件，保持其他条件不变，估计误差与之前的变化如下图所示：

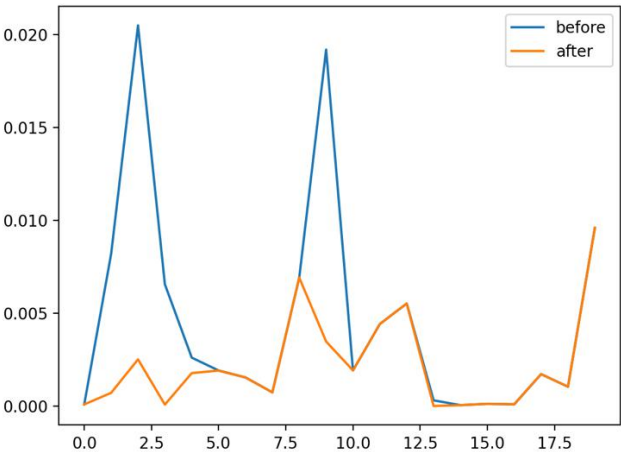


图 13 优化误差对比

可以看出，该改进模型对于孤立点的估计要远好于原模型。在该条件下，其平均误差为 0.04873，即在一个 $[-0.5, 0.5] \times [-0.5, 0.5]$ 米的平面中，该算法的误差在 0.04873 米左右，误差率约为 4.873%。相较于原模型性能提升 28.58%。

基于 data1-4，模型性能与提升如下：

	data1	data2	data3	data4 (3D)
原误差率	6.82%	12.15%	8.34%	7.05%
优化后误差率	4.87%	9.06%	6.89%	5.68%
性能提升	28.58%	25.40%	17.35%	19.49%

由表知该优化算法对模型性能有显著的提升。

八、模型稳定性分析与测试

为了测试模型面对不同数量、位置的节点与锚点，我们将随机生成 5000 组测试数据进行测试，其中我们令节点与锚点的 x 、 y 坐标均服从 $X \sim N(0,0.5)$ ，令锚点与节点的通信半径为 1，讨论 4 个锚点与 20 个节点的情况，得到的误差结果如下：

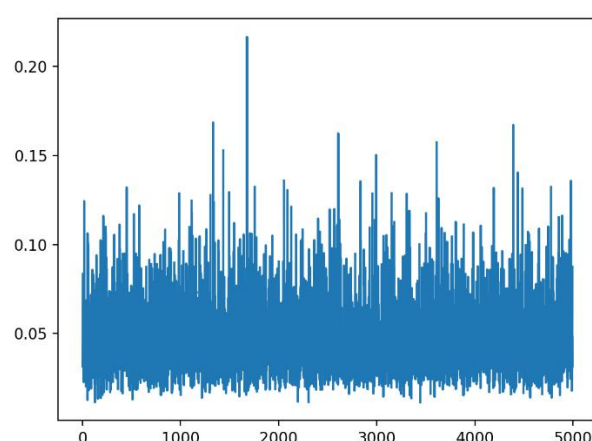


图 14 5000 组数据误差测试

其中平均值为 0.05432，最大值为 0.2376，误差相对稳定，说明该模型可以适用于随机性较强的数据，可用性强。预测成功率为 93.7%，存在 315 组预测失败的数据。通过调取历史记录我们发现，预测失败的原因是这些数据中存在与任何节点或锚点都不相连的纯孤立点，使得在迭代过程中出现了奇异矩阵，使得迭代无法正常进行。基于此，我们进一步改进了算法，使得检测到奇异矩阵的时候抛出一个异常，提高了模型未来的可拓展性与用户友好性。

接下来我们改变节点和锚点的数量，用于模拟不同的使用环境。我们采用节点数与锚点数分别为 10、20、40 进行交叉组合，每个组合实验 500 组数据，得到的误差热力矩阵如下：

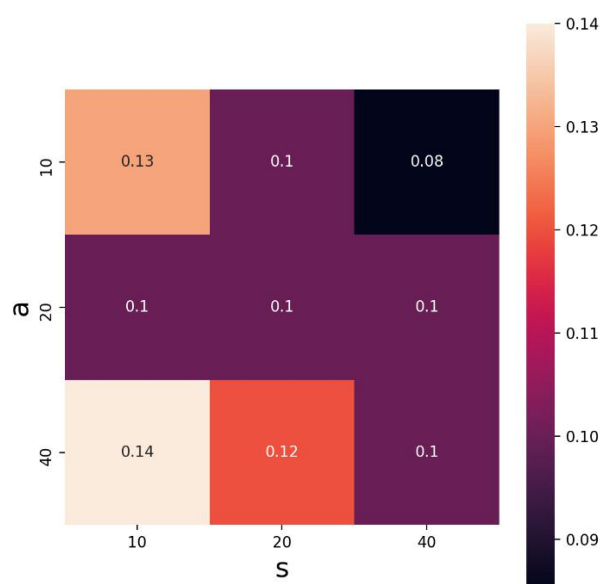


图 15 误差热力图

从图中可以看出，由于节点与节点之间的通信可以帮助修正预测准确性，所以在一定范围内，待预测节点数目的上升反而有利于预测精度的上升。当锚点数量过多是，在本模型条件下，反而不

利于坐标的预测。这是由于在三边测距定位中，到 40 个锚点的距离都会被使用，而其中的误差与噪声累积即造成了最终误差的上升。

最后我们探究模型的性能，由于三边定位测距算法的时间复杂度是 $O(n^3)$ ，除此之外的算法最高的时间复杂度为 $O(n^2)$ ，故该模型的总时间复杂度为 $O(n^3 + n^2)$ 。在数值试验中，对 100 组数据进行预测，每组数据有 40 个锚点与 40 个节点，即总共 4000 个用户，共耗时 60.68 秒，耗时较长，对于算法的时间复杂度还有待优化。

九、参考文献

- [1] 曾晶. 基于量子遗传算法和混沌粒子群的无线传感器网络节点定位算法研究[D].江西理工大学,2013.
- [2] 刘德亮. 室内环境下无线定位关键技术研究[D].天津大学,2015.
- [3] 程超,钱志鸿,付彩欣,刘晓慧.一种基于误差距离加权与跳段算法选择的遗传优化 DV-Hop 定位算法[J].电子与信息学报,2015,37(10):2418-2423.
- [4] 刘士兴,黄俊杰,刘宏银,易茂祥.基于多通信半径的加权 DV-Hop 定位算法[J].传感技术学报,2015,28(06):883-887.
- [5] 王磊,李鹏涛,贾宗璞.基于全质心-Taylor 的 UWB 室内定位算法[J].传感器与微系统,2017,36(06):146-149.
- [6] DV-HOP 算法.CSDN <https://blog.csdn.net/wangh0802/article/details/7355075>
- [7] 多边测距定位.CSDN https://blog.csdn.net/weixin_39641257/article/details/