1. 介绍

蓝色部分的字为2016/9/8日新加的内容

本文主要讨论了一种在FDD系统中采用TDOA定位系统外的转发机的方法。在转发机（与发射机，接收机处于不同的系统中）转发了发射机发出的信号的情况下，并且已知发射机，接收机的位置（以及少量监听机的位置）以及信号到达每个节点的时间，就可以计算出转发机的位置。

1. 系统模型

系统中，一个发射机正在与一个接收机在FDD系统中通信，该通信使用了CLPC（闭环功率控制）来保证接收机接收到的信号强度。此时假设有一个转发机接收到了这个信号，并将这个信号进行了增益并转发。那么接收机将会接收到两个信号。如果这两个信号满足了下列两个条件，就可以参与定位转发机的过程：

1. 两个信号到达的时间差大于接收机的分辨阈值。
2. 转发机转发过来的信号强度大于发射机直接发送的信号强度。

条件（1）是为了确保接收机能够接收到两个信号并分辨它们；条件（2）是因为接收机只会选择信号强度更大的信号（由于接收机与发射机位置已知，它们通信的正常到达时间会包含在数据包中，由此可以判断信号是否来自发射机），具体选择过程由以下框图表示：

接收机接收到两个信号

到达时间差大于阈值

到达时间差小于阈值

无法用于定位

通过数据包中的基准时间判断两个信号的归属

判断信号强弱

转发信号强

直发信号强

可以用于定位

无法用于定位

当两个条件都满足时，就可以列出以下公式：

*式（1）*

其中，为发射机位置，为转发机位置，为接收机位置。上式中，*C*为光速，*t*为接收机测得的两个信号的到达时间差，由此可见，式中唯一的未知量为转发机的位置坐标。

由于一个*式（1）*不足以解出转发机位置，因此我们引入了监听机：监听机在发射机与接收机的通信信道上监听，记录下两个信号的到达时间差与强度。但监听机不向发射机发射任何信号，也就没有CLPC控制。一个符合条件（1）、（2）的监听机就可以再列出一个*式（1）*。

由于*式（1）*实际形状为一个椭圆，理论上3个椭圆交于一点即可解出确定的方程解（在此我们不考虑3个椭圆交与2个点或者更多点的情况，因为这种情况的几率太小，由于这种情况导致的误差会被算入系统误差中，并且这种误差可以采用使用更多监听机的方法消除），因此只要有1个接收机和2个监听机或者3个监听机能用于定位就可以解出转发机位置。

由于模型采用了FDD系统，则接收机与监听机的功能是可以互换的。如在时隙1中，A为接收机，则B、C、D为监听机；在时隙2中，B为接收机，A、C、D为监听机，以此类推。这样就可以完成对所有节点的通信。

1. 公式及算法推导

计算部分主要分为两部分：

1. 计算接收机接收到的两个信号的信噪比。
2. 根据（1）的计算内容，进一步计算转发机位置。

计算部分（1）实际为信道模型的建模。

信道建模采用以下公式：

接收机收到的功率即为，分别代表经过转发机延时后的，即：

计算部分（2）实际为寻找二元齐次方程组最佳解的问题。我们使用一种遍历的方法找到最佳解：先把*式（1）*整理为一个表示误差的方程：

*式（2）*

由*式（2）*可见，任意一组x，y都会对应一个误差值*Error*，这个值越小，则代表x，y越接近真实值。

由于实际仿真中存在多个接收机，因此*式（2）*可以转化为：

*式（3）*

上述式中的最小值所对应的就是转发机位置，对此可以用三种方法找到最小值。（由于成功率并不是定值，导致函数的项数不确定，Matlab自带的寻找最小值算法无法使用）

方法1：撒网法

在地图上生成一个二维的网格，计算每个格点所对应的函数值，在函数值最小的地方重新生成一次网格并重复上述过程，多次迭代后即可得到函数最小值。

这种方法有固定的收敛速度，并且准确性较高。

之前初步的计算时间约为0.8s左右（求出最小值）（100个RX都可用的情况下（成功率与计算所花费时间呈线性增加））

后来对这个算法进行了优化，调整了一些参数，最终把时间缩短在0.1s以下。但是算法的整体思路没有变。

方法2：随机数撒网法

在地图上随机生成多个随机点，计算这些点对应的函数值，在函数值最小的点附近减小随机数的产生半径并再次产生多个随机点，多次迭代后即可得到函数的最小值。

这种方法需要特定的参数来确保能够得到正确的结果（由于求解过程中使用了随机数求解，因此最终能否得到正确的结果也是一个概率不为1的事件，正确结果的定义：每次新产生的随机数范围包含了实际的最小值点，如果某次迭代时最小值对应的点不在其中，则表明前次迭代是错误的），并且需要在速度与准确度上折中。

方法3：求导法

（之前邮件中提到的问题已经被解决了）

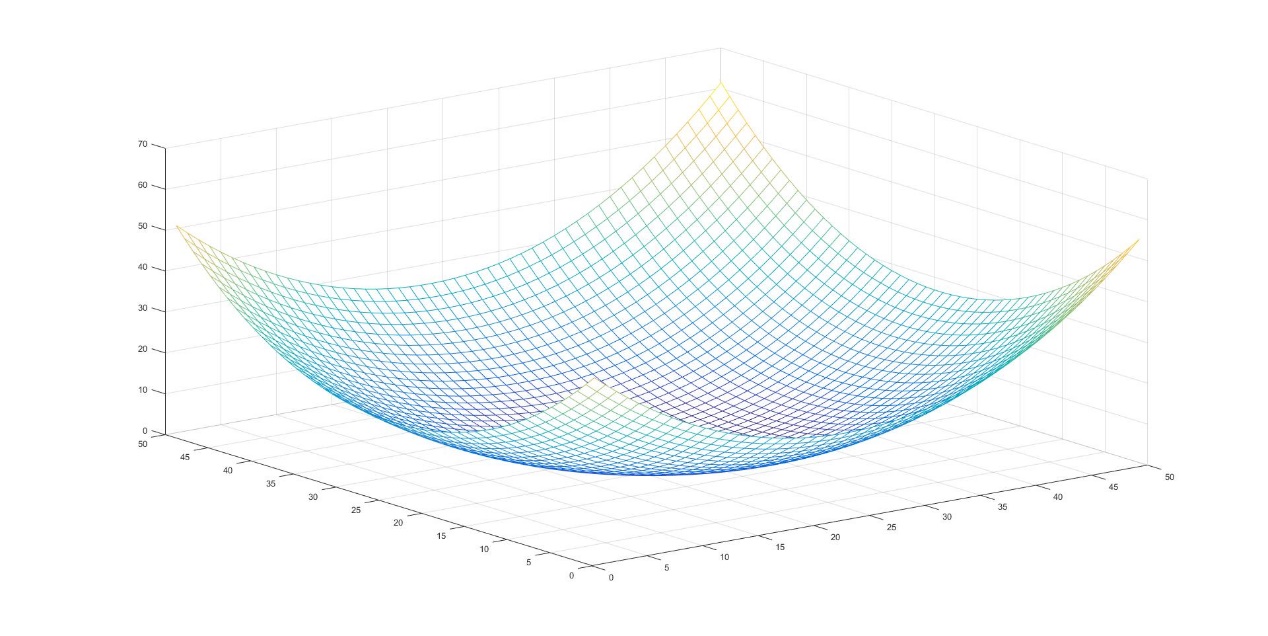
最早先的版本：我们采用先随机找点，然后求偏导和此点的函数值算出下一个点的方法进行寻找。只要在任何一次迭代后发现点超出了地图边界，就重新生成一个随机点。

迭代过程的公式表示：

理论上，一个合适的起始点和足够多的迭代就可以算出最小值，并且这个算法的效率会高于方法1，但实际上这个算法效率很低，平均约为方法1的1/5。并且效率非常不固定，特殊情况下能算上10多秒。

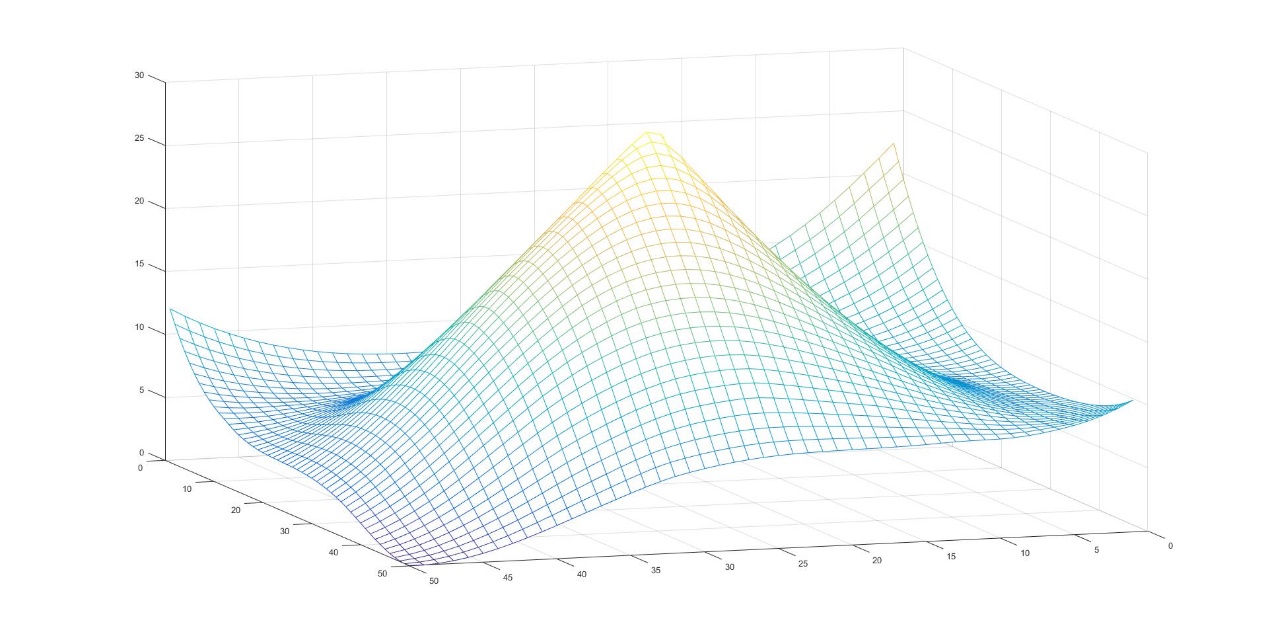
最终我们去找原因，发现是函数的形状不是固定的，并且总是在变化。

比如，当转发机和发射机很近时，函数图像呈现下面的形式



*发射机位于（0，0） 转发机位于（5.1，5.1）*

当转发机和发射机相距很远时，会出现完全不同的函数图像：



*发射机位于（0，0） 转发机位于（99.1，99.1）*

实际的解应该在左下角位置（函数值最接近0）。但是由图不难看出，如果随机找点的话，大部分的点都不能求出正确的最小值。由此造成了求导的算法效率并不高。

求导法优点：可以迅速增加精确度，迭代速度极快。

求导法缺点：对起始点的要求很高，只有少部分起始点能算出最小值。

方法四：求导与撒网相混合

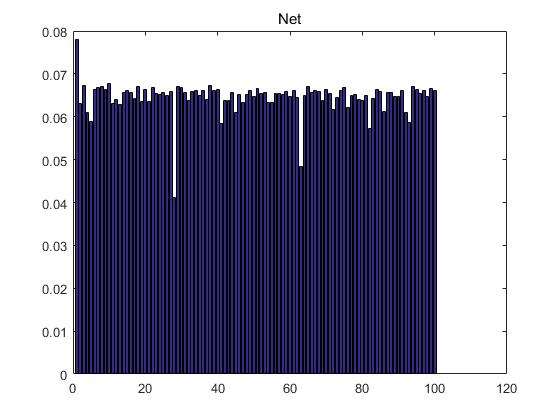
这是我们目前所写出来的效率最高的算法了。

同样，由本页的上图可以发现，如果要使用求导法的话，把最小值附近的点作为起始点是可以找到最小值的。

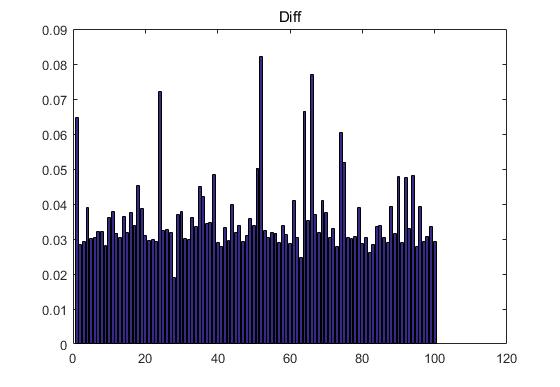
因此，我们把撒网法和求导法相混合。先使用撒网找到一个最小值的范围区间，再在这个区间内取随机点进行求导找到最小值。

效率比较：（我们采用随机生成转发机，固定发射机的方法进行测试）

撒网法：由下图可见，撒网法所花的时间平均约为0.06s~0.07s，并且方差较小。（Y轴是时间，单位秒；X轴是循环次数，共循环100次）



求导与撒网法混合：由下图可见，这种方法所花的时间平均约为0.03s，并且方差较大。这一点上和我们之前的推断是相吻合的。

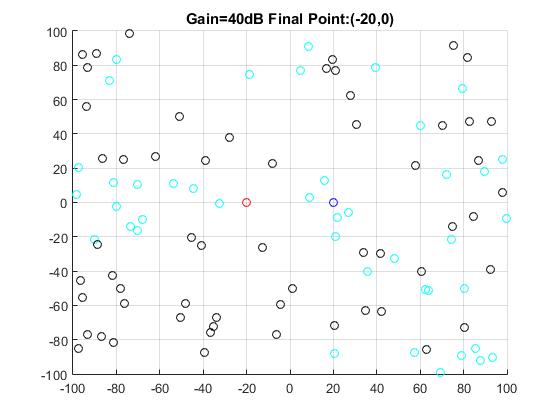


1. 仿真结果

仿真部分总共分为4部分，分别为（1）方法可靠性的验证；（2）成功率；（3）定位的误差RMSE；（4）地图大小对定位的影响。

* 仿真结果（1）：方法可靠性验证

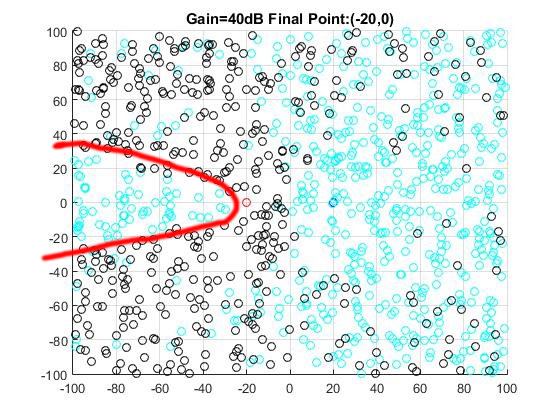
我们假定地图边界为100m（注意，边界100m是指-100m到+100m，实际边长为200m，下文均用边长），总共存在100个监听机，转发机增益为40dB，最小分辨时间为5ns。接收机与转发机位置为固定值（设定为固定值是为了方便观察图像性质，后面有关成功率与定位误差的图会改为随机确定接收机与转发机位置）。



结果中，黑色的点为能够用于定位的监听机位置，浅蓝色的点为无法用于定位的监听机位置，蓝色的点为发射机（位置固定为20，0），红色的点与绿色的点重合，其中红色的点代表计算所得的转发机位置，绿色的点为实际转发机位置。

图中，无法用于定位的监听机（浅蓝色）是由于不满足条件（1）（2）导致的。其中，不满足条件（1）的部分会集中在地图左侧中间部分（理论上由*式（1）*可得，分界线会是一条双曲线），不满足条件（2）的会集中在地图右半部分（由于监听机距离转发机太远导致信号衰弱过于明显）。

更加明显的比照可以由下图得知：



图中，红线所圈出来的部分就是由于不满足条件（1）而导致数据无法用于定位的监听机。红线之外的监听机则是不满足条件（2）的部分，由于噪声是采用的随机数生成，因此并没有严格的分界线。

* 仿真结果（2）：成功率

成功率的定义是在某个特定条件下，有多大的可能性使得系统能够定位出转发机位置。

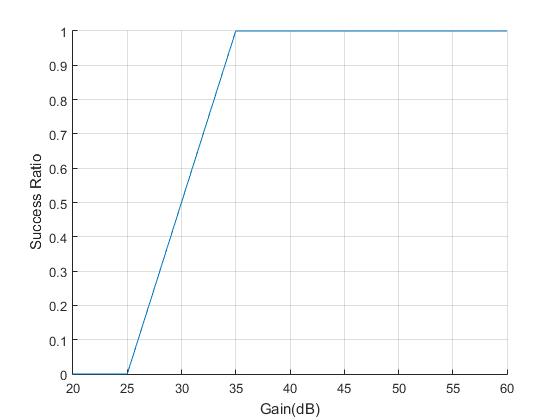
我们假定地图边长为200m，地图中随机存在100个监听机，接收机、转发机、发射机位置均固定，如果有3个（或者更多）监听机能够用于定位，则我们认为这种情况是成功的。（假定接收机不参与定位，即使它的数据能够用于定位）

同样的参数，重复多次后就可以得到这个参数下的成功率为多大。

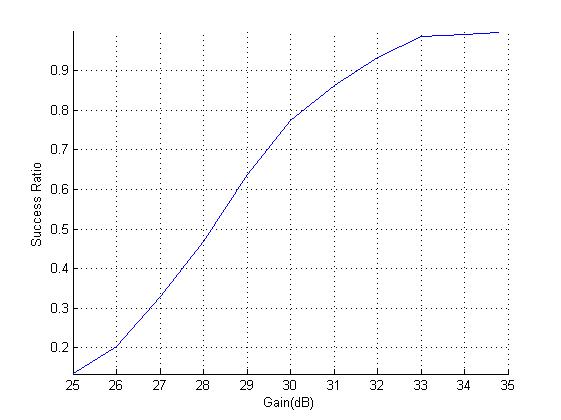
为了更加符合实际，仿真中考虑了转发机的延时，模型为一个正态分布，实际公式如下：。参数选取为，，实际为加在正态分布前的系数，经验证，当取为0.1时，生成的100000个随机数中不存在负数。同时相对应的，在最终的求解算法中加入了转发机延时这一部分，即公式变为：

即为延时的均值，代为16ns。

最终仿真结果如下：

**

由于在25-35dB之间成功率变化最大，因此我们增加了重复次数后重新计算得到了更加精确的图像：



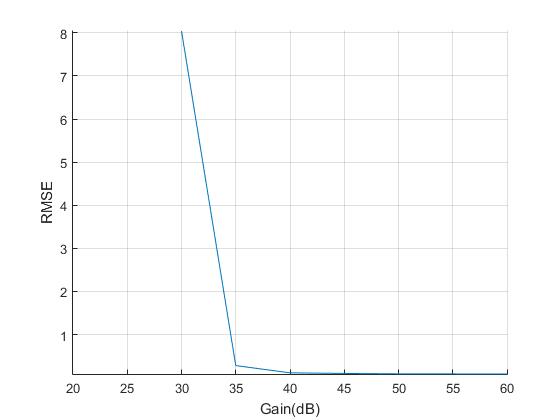
由图可见，当转发机增益达到35dB时，就可以确保成功率接近1 。相对应的，下图中的RMSE也接近于0，这与成功率接近1是相吻合的。

* 仿真结果（3）：定位的误差RMSE

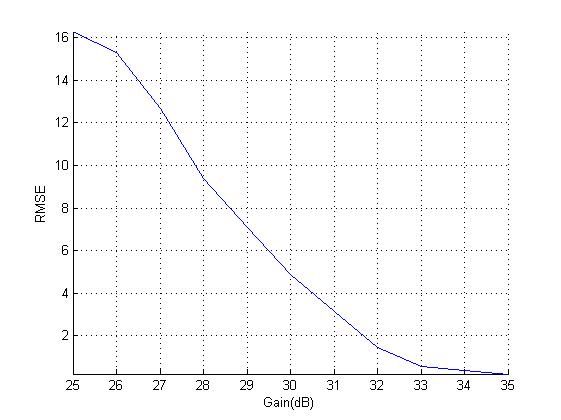
误差来源于两种，大部分是由于转发机的延时导致的，小部分是由于解方程的算法中出现了多个椭圆交于多个点的情况。但这两种误差都会随着可用监听机的增多而减少。

注意：如果满足条件（1）（2）的监听机数量少于3个。则RMSE会被记为正无穷，不会在结果中表示。

结果如下图：



同样，由于x轴跨度较大，精确度不高，所以后来进行了更精确的仿真：



此图循环了500次的结果。

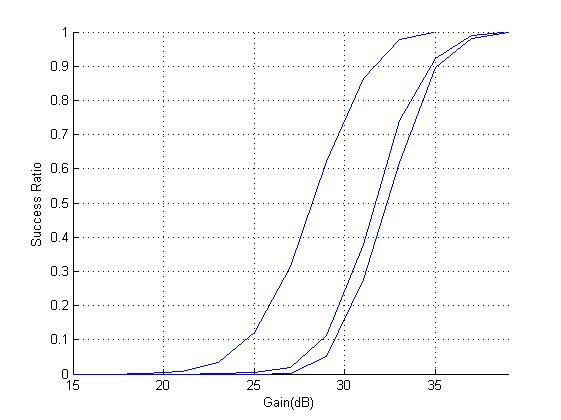
由仿真结果（2）（3）可见，当在边长200m的范围内，转发机增益大于35dB时，就可以获得很好的定位效果。

注意：上述两幅图中，30dB点处所对应的RMSE并不相等，并且相差较大。这是由于第一幅图中只随机生成了10个地图，而第二幅图是随机生成了500个地图，换言之，第一幅图的RMSE偏高是由于循环次数太少导致的。

* 仿真结果（4）：地图大小对定位的影响

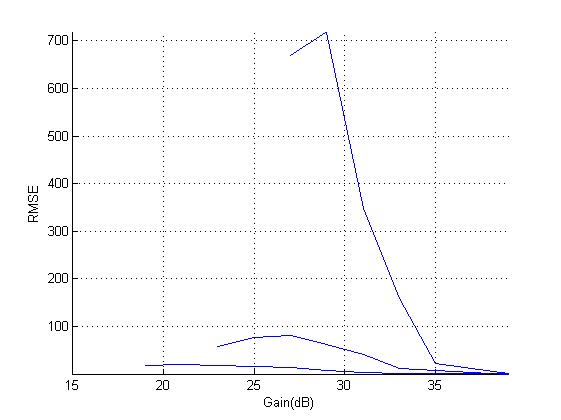
由于发射机，接收机，转发机的位置均固定，并且相对距离在此次仿真中相比于地图太小，可能会导致较大的其他误差，这一部分还需要更加深入的研究。（在边长为2000m的地图中，接收机，发射机，转发机只相距了20m）

关于成功率的图：



从左到右，依次为边长200m，1000m，2000m的地图。

关于定位误差RMSE的图：



从上到下，依次为边长2000m，1000m，200m的地图。

由这幅图可以看出，地图变大时，成功率，RMSE均有影响。但当转发增益逐渐变大时，同样可以提高定位效果。同时，在同一增益情况下，地图越大，定位的误差也越高。

注意：RMSE的图中，曲线的趋势（先小幅上升再下降）与预期不符，这一问题可能是由于随着Gain的减少，成功率越来越低，用于计算的样本（满足条件1，2的情况）也越来越少，从而产生了误差。（后来发现，求最小值的算法的参数与地图边界长度有可能有联系，不适当的参数可能会造成相对应的误差）

未来的计划和已经遇到的问题：

1. 加入多径干扰的仿真代码。目前我们仍然停留在读论文的阶段，现在的问题是大部分关于多径干扰的文章都是加入了滤波器和通行协议来分析多径干扰效应的。但是在我们的模型里面完全没有加入有关滤波器的仿真。并且很多论文提出了更好的通信协议来减少多径效应带来的误差，这个在我们的模型里面也是不可取的。
2. 研究更多的关系，比如地图大小与转发机转发功率的关系（接着仿真结果4做）。这一点其实并没有修改核心的代码，所以要做的话，速度会非常快，但感觉实际意义不大。