

无线传感网络实验报告

作者：17130120116 李云水

一、实验背景

在无线多跳自组织网络环境下，高效地将关键数据从网络的一端扩散至全网是多跳自组织网络中非常重要的重要之一，广泛用于网络控制、公共数据广播、时间同步等，这种数据广播协议一般称为数据分发协议。

无线传感器网络是由许多具有感知、计算和通信能力的低成本低功耗的微型传感器组成，用于实时感知和采集网络覆盖区域内的感知对象的信息，常常由成百上千个传感器协同工作。源节点通过内置的一个或多个物理传感器，可对观察范围内的感知对象产生原始数据。这些节点靠不能补充能量的电池供电，具有有限的能量。网关节点，也称为sink节点(或基站)，这些节点用于实现传感器网络Internet的连接，往往个数有限。但能量能够得到补充。无线传感器网络的主要目的是从监测环境中收集用户感兴趣的数据，发送给sink节点。因此，数据分发是无线传感网络的一个基本功能，而带宽和能量是传感器网络中最为缺乏的资源，为了延长网络的生存时间。设计能量有效的数据分发方案是至关重要的。

有效可靠的数据分发关键在于建立数据分发的路径，通常要考虑节点的剩余能量、所处的地理位置、网络拓扑、当前环境和节点状态等因素。数据分发方案主要用于优化数据分发路径，使其尽量接近最佳路径，有效延长网络寿命。

分发协议目前存在的挑战：

1. 应实现总体节能，分发协议应尽可能减少无线发送的次数，延长网络的工作寿命；
2. 应实现个体节能，应降低对某个特定节点的发送次数，当此节点失效时，可能会影响整个网络的工作；
3. 应尽量减小网络跳数，网络跳数过多，传输成功率会下降，同时传输时延会上升；

二、实验要求

1. 在中模拟N ($N > 100$) 个节点的多跳传感网络, 该N个节点随机分布在 $100m \times 100m$ 的正方形2维平面。每个节点的通信半径r符合正态分布; 进一步, 假设当两个节点距离为d时, 通信成功率为 $t = 1 - d^2 / (r_1 * r_2)$ 。每个节点一些基础物理信息: id, 总电量, 单次发射耗电量等;

2. 设计自己的数据分发协议:

1. 基本要求: 在不考虑电量的情况下, 可实现从任意选定的一点将信息分发至全网

2. 进阶要求: 在考虑电量的情况下, 实现从任意选定的一点将信息分发至全网

3. 高阶要求: 在考虑电量的情况下, 提出一种最优的数据分发策略

4. 可视化多跳网络, 并通过简单的过程动画展示分发过程

三、实验设计

此无线传感网络选用基于外部存储的数据分发方案ES结合LEACH协议以及BFS算法进行数据分发模拟一个有效的无线传感网络。考虑所有节点的自有不可充电量, 将信息发布至全网, 在数据分发过程中, 经比较决定使用基于LEACH协议以及BFS算法的优化策略进行数据的分发。

此无线传感网络分发系统的软件架构设计为分层结构, 共分为四层, 分别为网络静态描述层、策略层、动态模拟层、可视化层。

层内设计:

(1) 静态描述层

网络组成结构由传感器节点序号、节点坐标、节点能量、每次传输耗能、通信半径、节点传输次数组成, 下图定义网络组成并进行初始化

```

class node(object):
    """

    def __init__(self, id, x, y, total_power, per_power, r, times=0):
        """
        id : id
        x,y : 坐标
        total_power : 总电量
        per_power : 单次发射耗电量
        r : 通讯半径
        times : 发射几次了
        """
        self.id = id
        self.x = x
        self.y = y
        self.total_power = total_power
        self.per_power = per_power
        self.r = r
        self.times = times

    # 传输消耗能量
    def transmit(self):
        self.total_power -= self.per_power
        self.times += 1

```

经过初始化后得到Node_list, 里面存储着N个节点的信息。

```

def init_node(N, u, sigma, total_power, per_power):
    # 节点的x, y坐标 和 节点的发射半径
    x = np.random.randint(0, 100, size=N)
    y = np.random.randint(0, 100, size=N)
    R = np.random.normal(u, sigma, N)
    Node_list = []
    for i in range(N):
        tmp_node = node(i, x[i], y[i], total_power, per_power, R[i])
        Node_list.append(tmp_node)

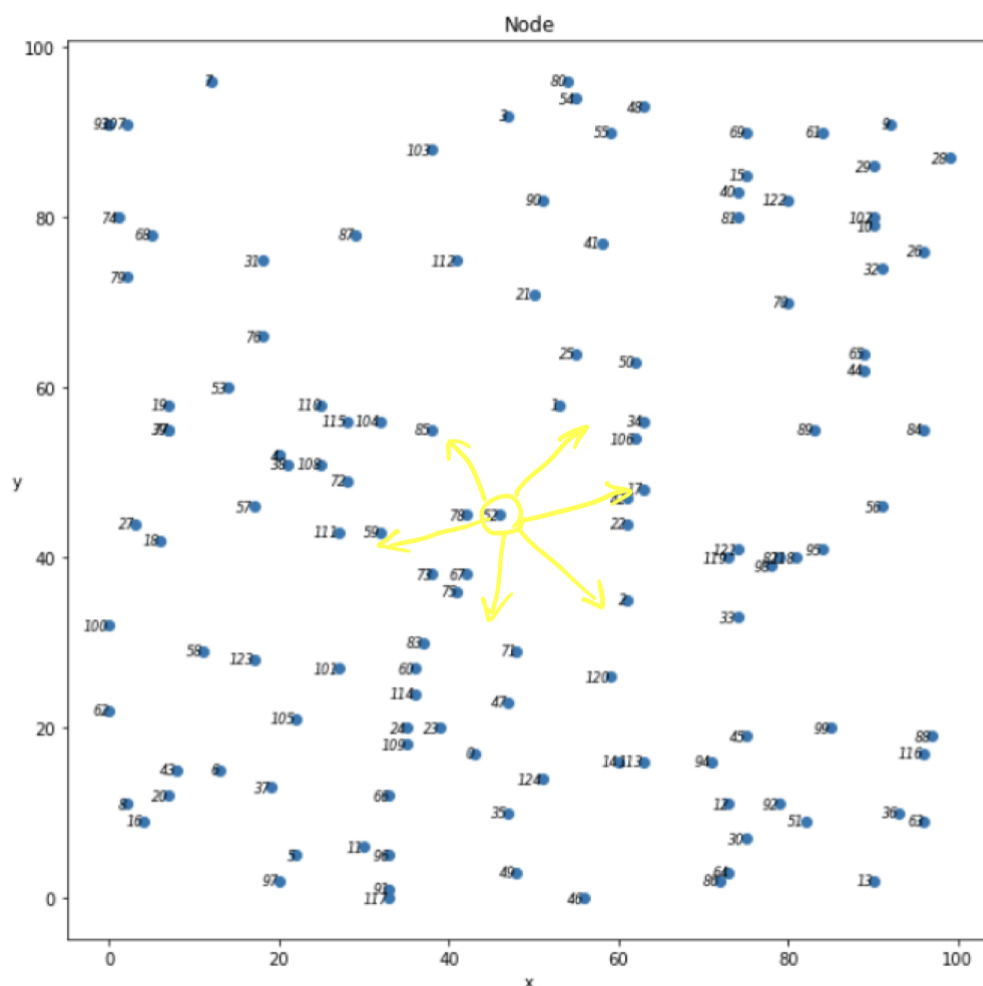
    return Node_list

```

(2) 策略层

数据分发策略采用基于外部存储的数据分发方案ES, 这是使用最广泛的一种方案, 它依赖于一个位于传感器网络内的集中式基站, 该基站用于收集和存储感知数据。在网络中, 选择随机任意坐标点作为本次网络的基站。针对静态描述层中定义的网络组件, 经比较, 本次设计选择了“最近优先”算法进行数据分发。

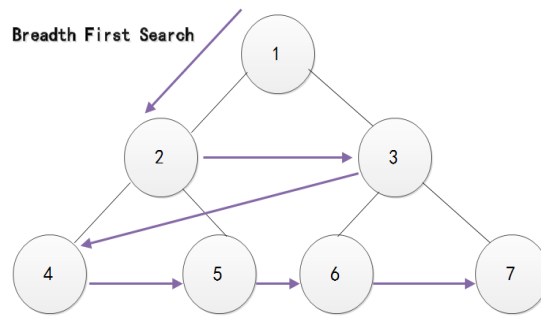
首先是计算所有节点之间的距离，得到距离矩阵Distance_list。“最近优先”算法的策略即传播源节点根据距离矩阵找到离它最近的接收到消息的节点群，在我们的实验中，一个节点群包含10个节点。然后传播源对它最近的节点群进行传播，顺利收到消息的节点，成为新的传播源，将消息传播给它最近的节点群，直到所有的节点都收到消息。



(3) 动态模拟层

策略层的分发机制，利用基站转发普通节点的信息至每个簇头节点，簇头节点对接受信息的转发模拟，我们采用BFS算法进行转发。

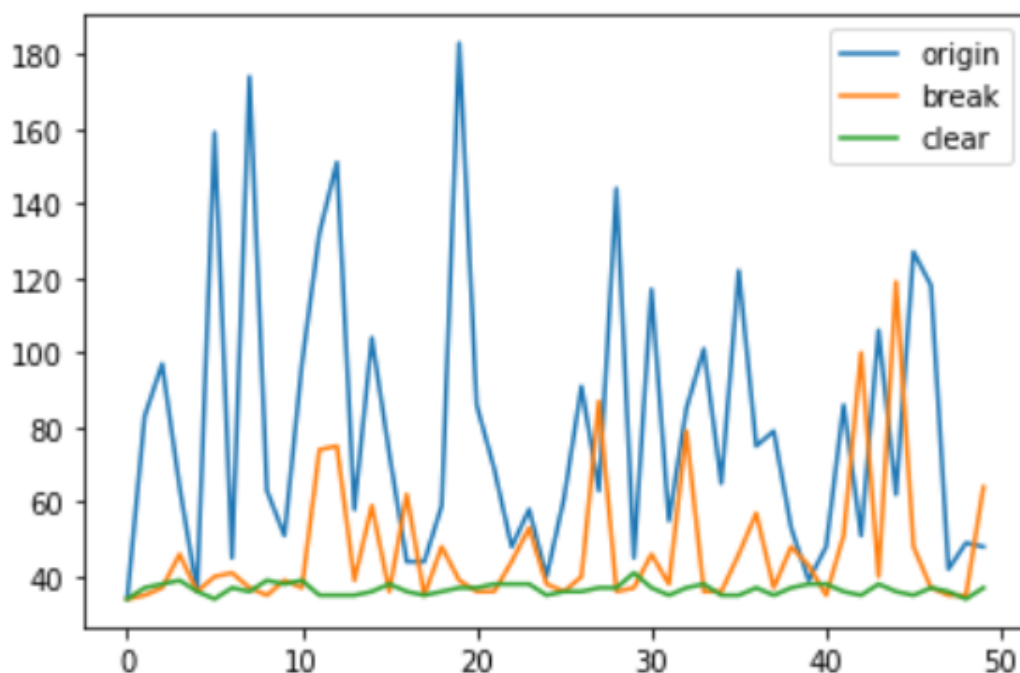
宽度优先搜索算法（又称广度优先搜索）是最简便的图的搜索算法之一，这一算法也是很多重要的图的算法的原型。Dijkstra单源最短路径算法和Prim最小生成树算法都采用了和宽度优先搜索类似的思想。其别名又叫BFS，属于一种盲目搜寻法，目的是系统地展开并检查图中的所有节点，以找寻结果。换句话说，它并不考虑结果的可能位置，彻底地搜索整张图，直到找到结果为止。



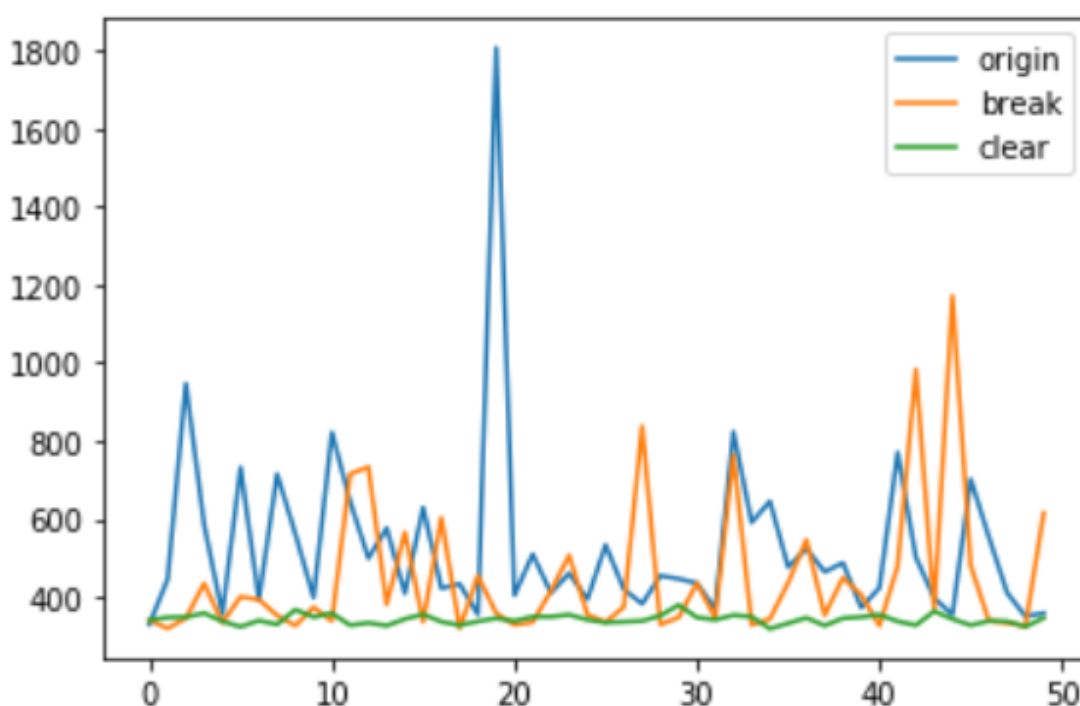
我们在实验中发现，这个策略在传播的前中期效果很好，但是当大部分的节点都接受的消息后，最后极少的一些节点却需要较长的时间进行传输，原因是使用BFS算法时，当大部分节点已经接受到时，在队列中还有不少节点还没作为传输源传输过消息，然而未收到消息的节点往往与在队列中传播源相距较远，所以传输的成功率很低，做了很多无用的工作。于是我们加入了“break”策略，即当未收到消息的节点达到某个阈值时（实验中阈值为10），我们跳出BFS的搜索，使用单点寻找的方法，即我们找到未接收消息的节点的最近传播源，由它直接向未接收到消息的节点传输消息。

“break”的策略提高了节点传输的效率，但是还是具备很大的不稳定性，原因是在节点传输的过程中，常常出现带有方向性的传播，使得某个方向的节点都传输完后，队列中的节点又陷入了无谓的搜索，大大降低了传输效率。于是我们又在“break”的基础上加入了“clear”策略，即在某一次队列传输源向其节点群传输后，未接受的节点数量没有下降时，我们清空队列中的所有节点，转而寻找一个未接收消息的节点的最近传播源，并将其加入队列中，继续进行BFS搜索。

我们进行了50次独立重复实验，从图中可以明显地看看传输效率的提升：



上图纵坐标表示我们实验传输所需要的循环周期，横坐标表示实验的序号，可以明显看到经过两次改进后，传输效率大大提高，而且非常稳定。



上图纵坐标表示整个网络消耗的能量，在包含300个节点的网络中至少需要300次传输，可以看到我们的方法十分接近最优解。横坐标表示实验序号，下表是实验中我们的方法消耗的能量次数。

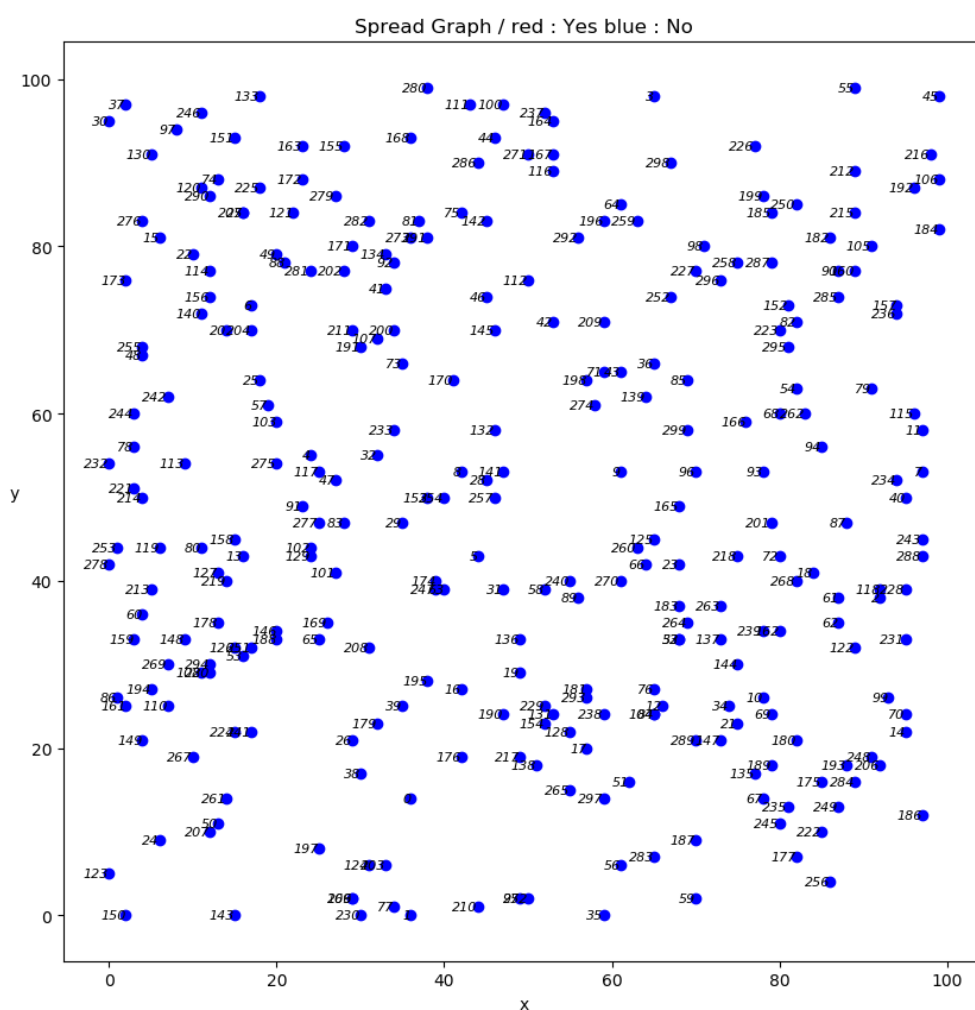
num	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
cost	340	347	348	358	337	323	339	329	367	348	358	327	333	326	343	355	336	327	337	345	338	349	349	354	341
num	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50
cost	334	336	338	352	379	347	341	353	349	318	332	346	326	345	348	354	337	327	363	343	327	338	337	323	346

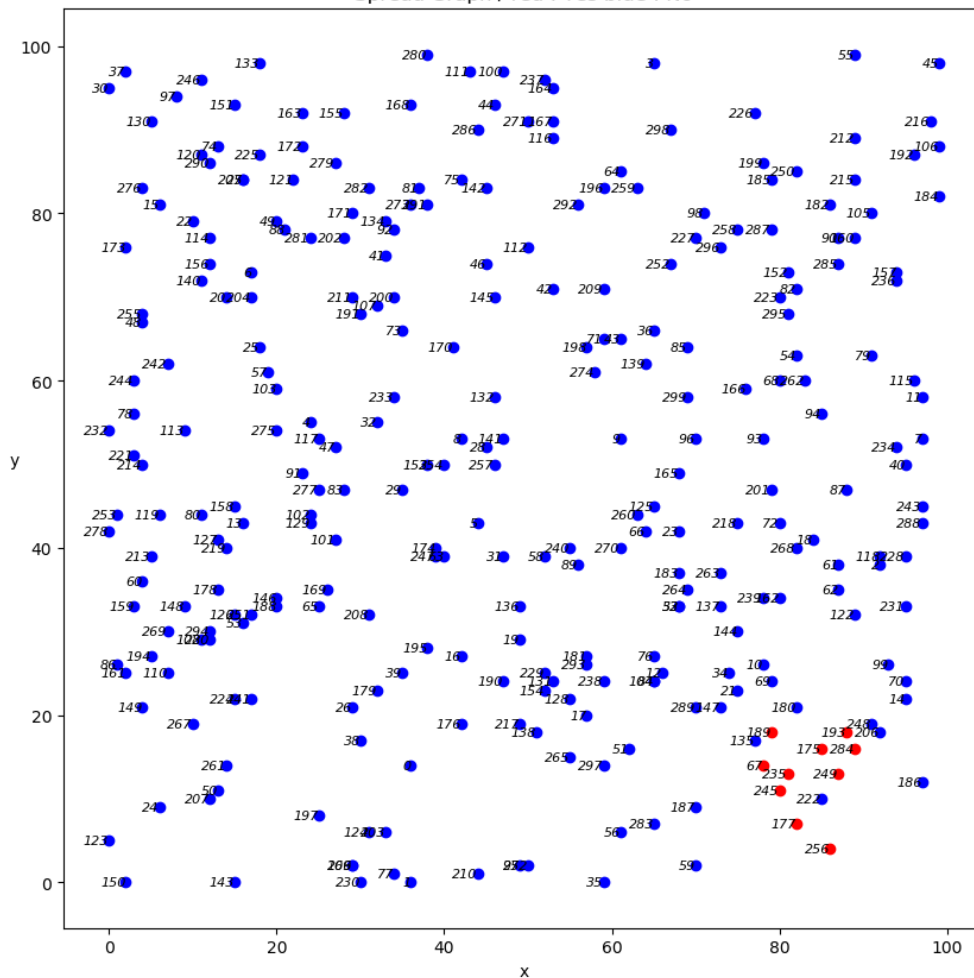
(4) 可视化层

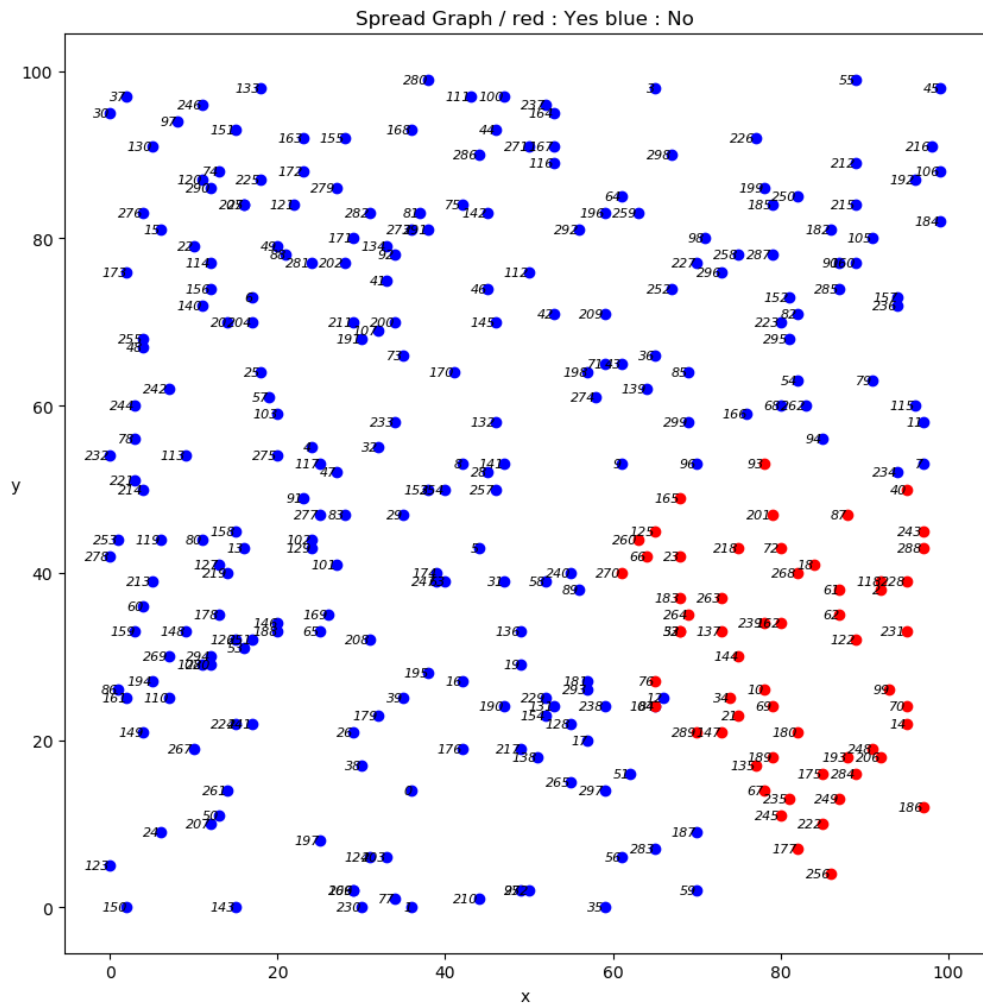
采用python的plot函数绘制数据分发策略和动态模拟的无线传感网络传输过程，整个过程经历三个过程，初始化阶段的可视化，传输模拟的可视化，传输完成的可视化。具体过程可视化结果见实验结果。

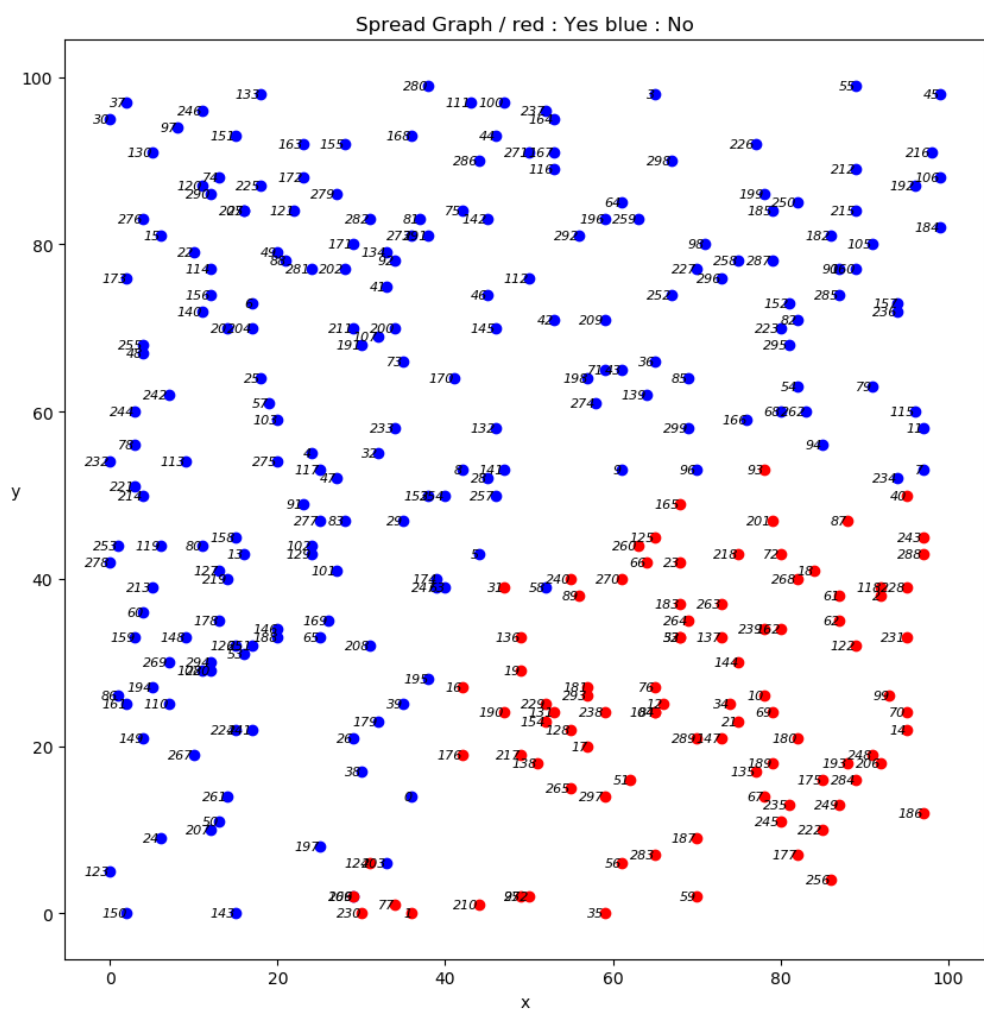
四、实验结果

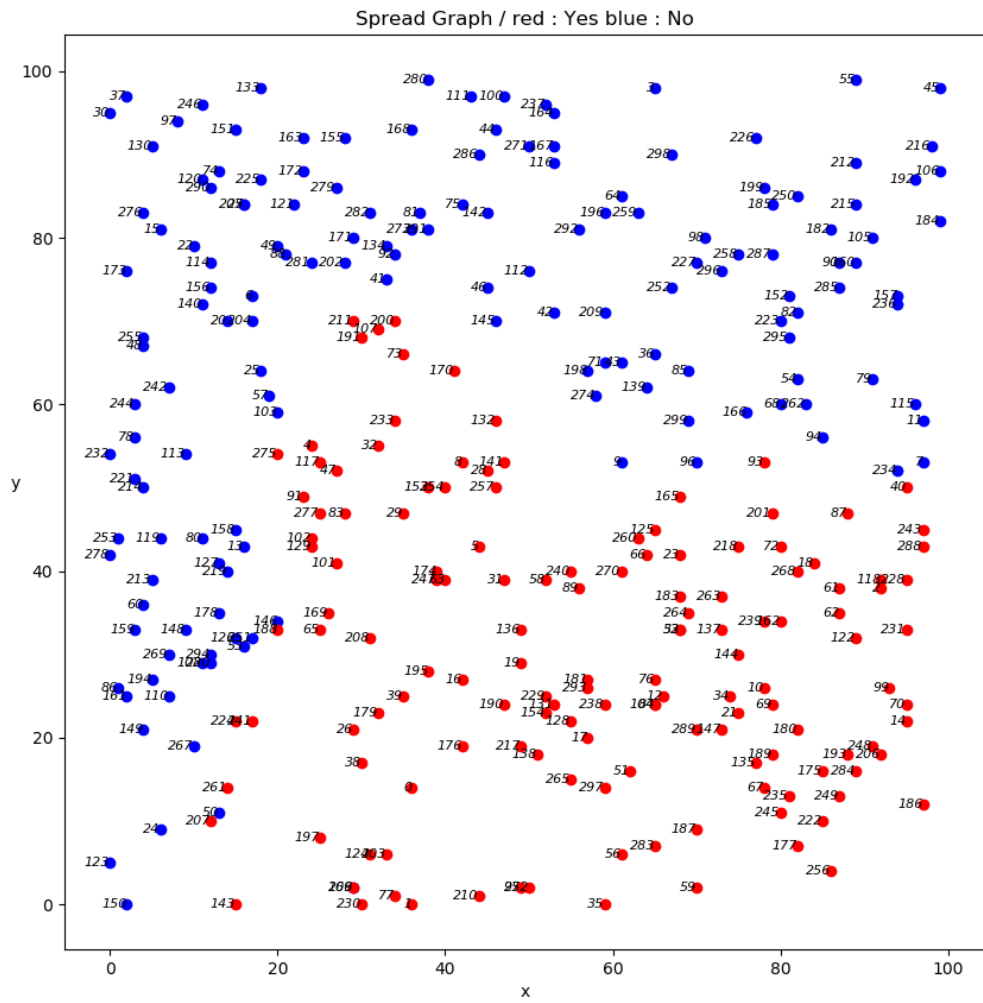
其中蓝色为未接受消息节点，红色为已接收信息节点。模拟过程中可以观察到初始化全部为蓝色节点，由传播源向其他节点传播，其他周围蓝色节点逐渐变为红色，变化结果如下所示。

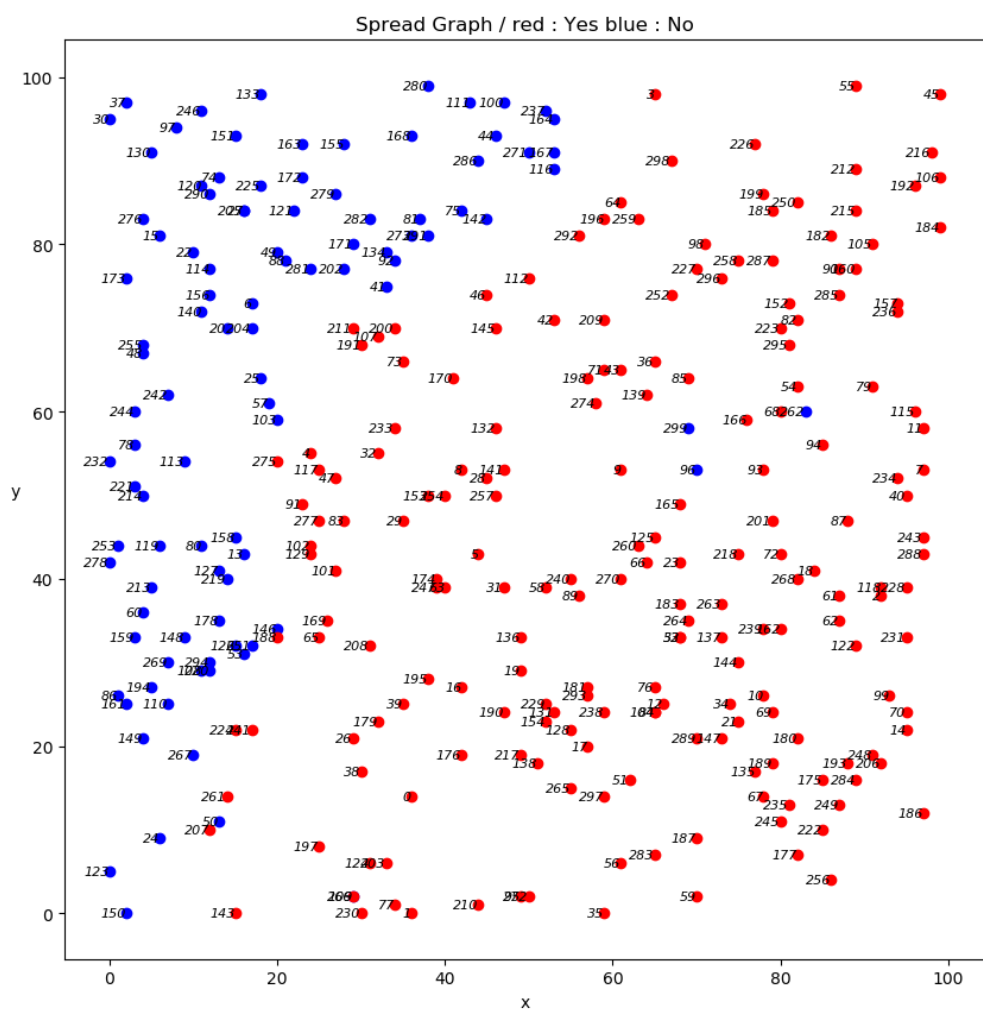


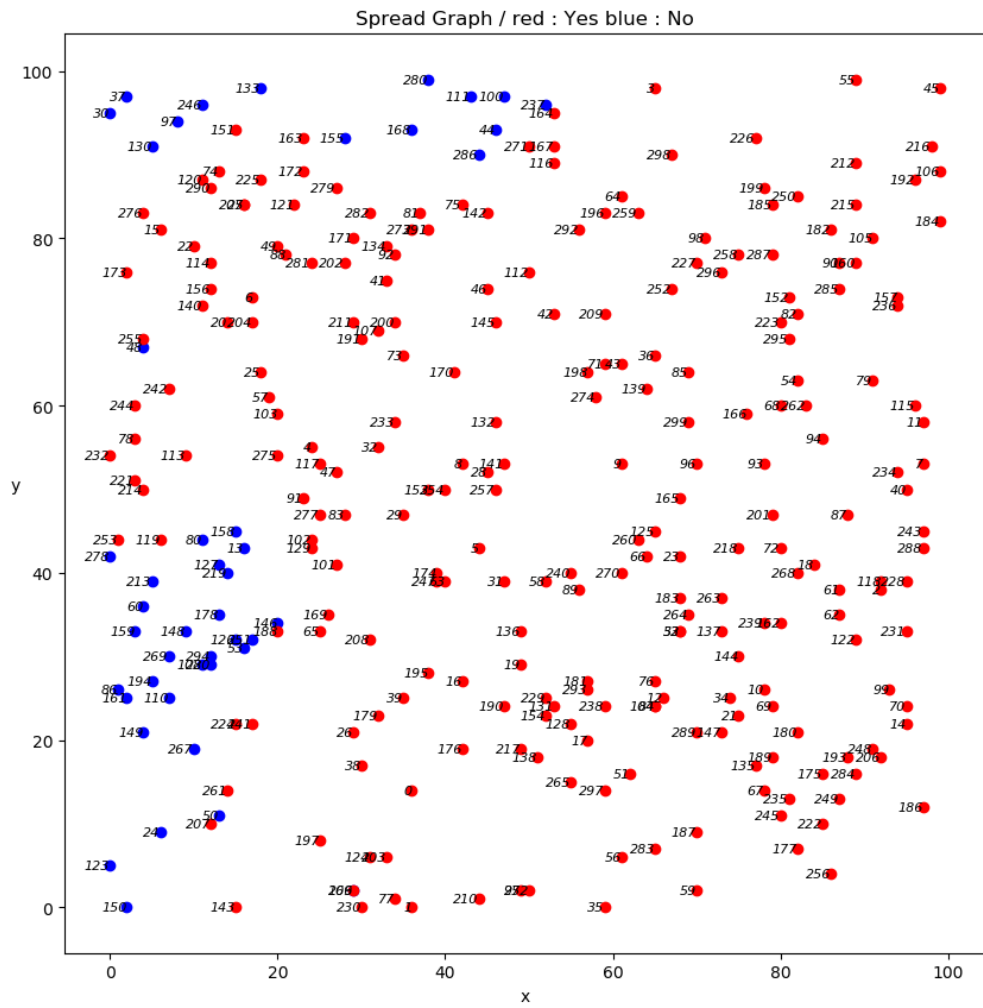


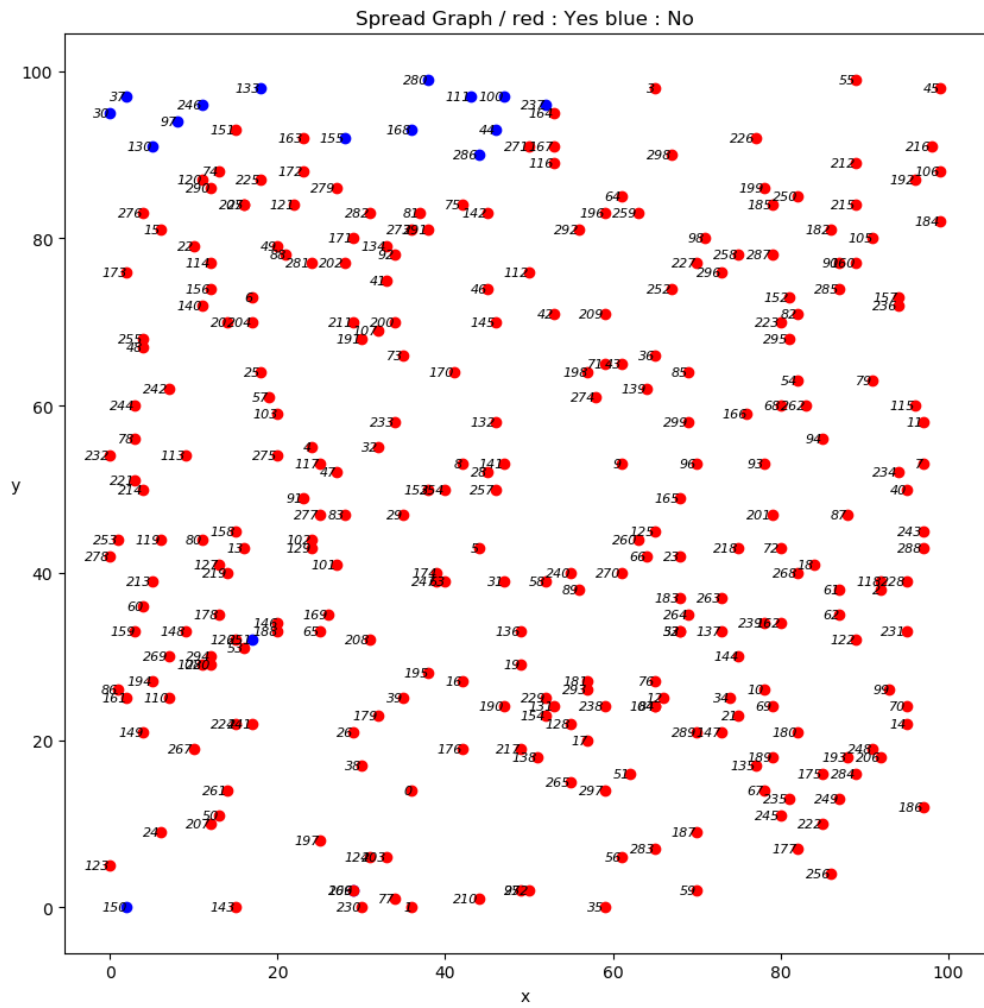


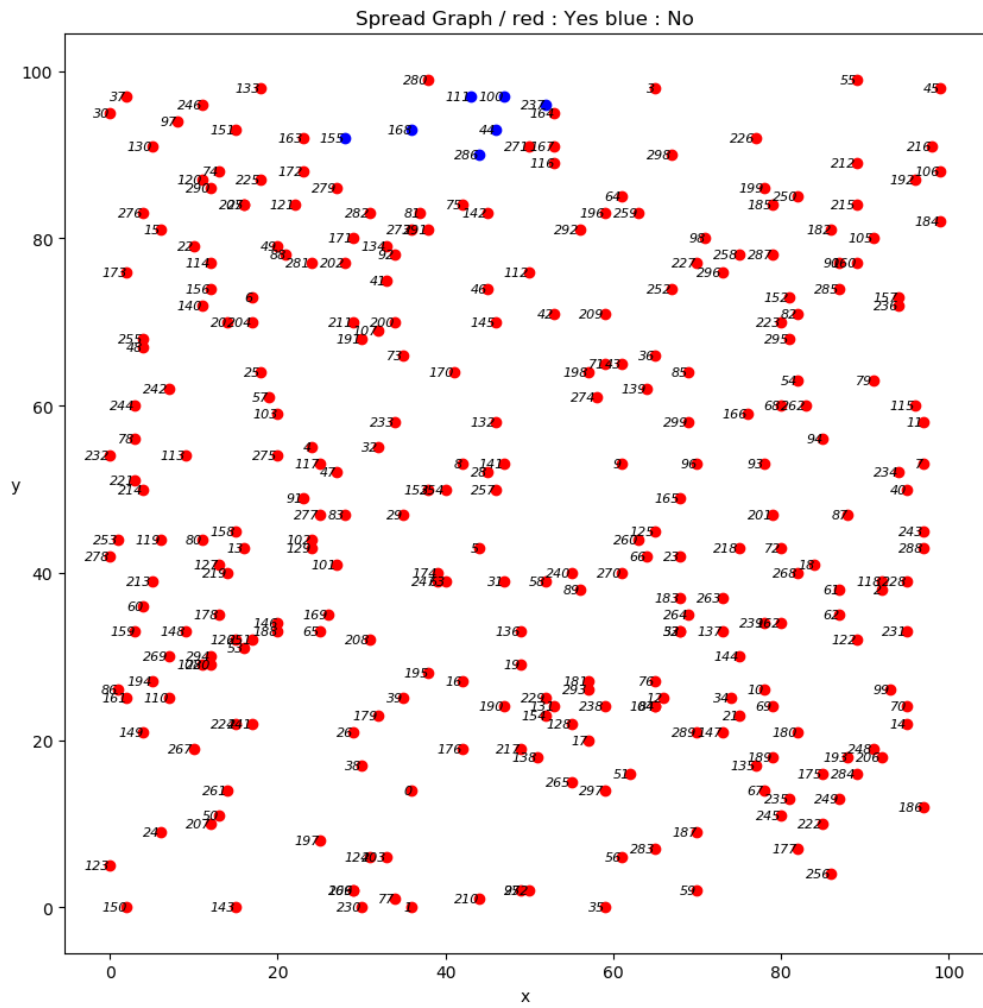


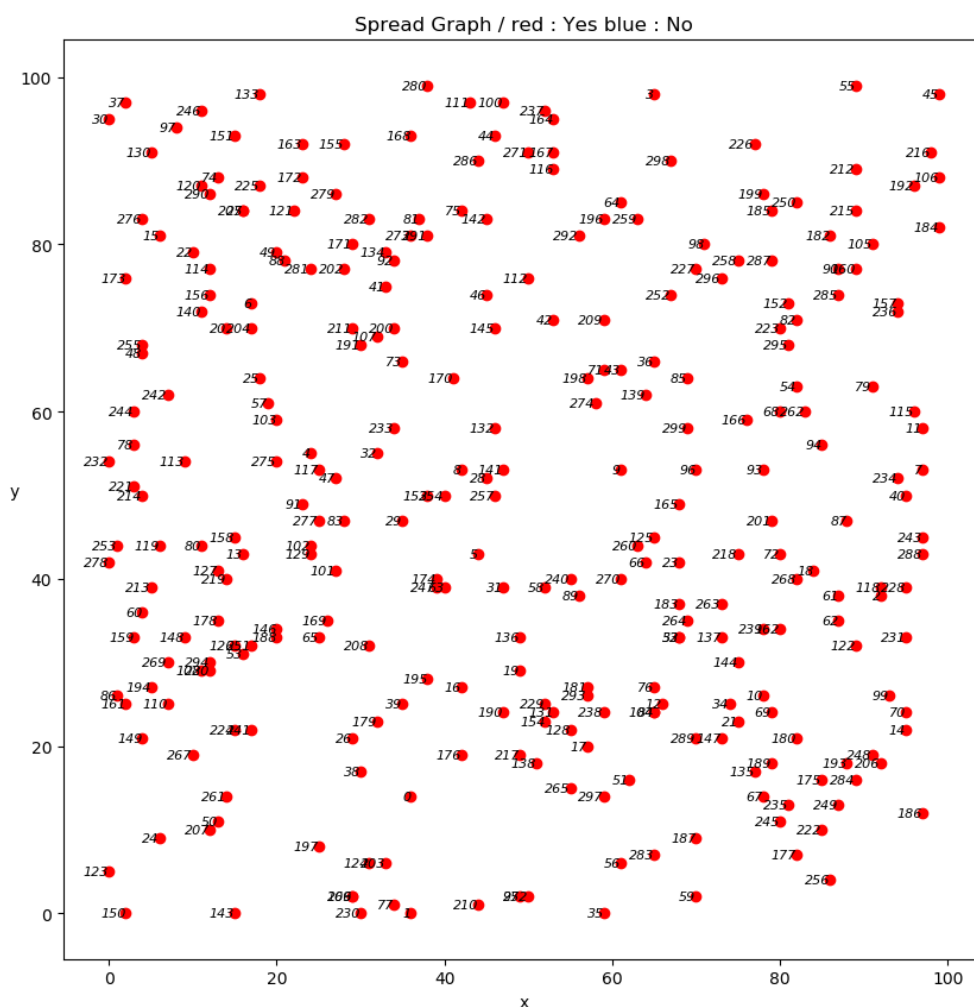












五、实验总结

本次实验的无线传感网络选用ES方案结合“最近优先”算法以及BFS算法进行数据分发，成功地模拟了一个有效的无线传感网络。

使用ES方案，简化设计，省去设计成本，适用于规模小，查询频率较稳定的网络。使用BFS搜索算法，解决最短或最少问题特别有效，寻找深度较小，每个结点只访问一遍，结点总是以最短路径被访问，所以第二次路径确定不会比第一次短，节省转发时间。

在实验中从结构的设计，算法的结构到算法的实现，是一个从模糊慢慢走向清晰的过程，一次次在实验中找到问题，找到原因，分析问题。代码结构渐渐清晰，实验的结果也如我们设想的在变好起来，当一切都结束的时候，看到那几张折线图出现的时候，心中的喜悦溢于言表，当你认真

去做一件事的时候，你会发现过程十分有趣，甚至为了追求一个问题的答案而忘了吃饭。

实验过程中也是收获颇多，掌握了无线传感网络的数据分发机制，掌握了Markdown等文档的常用写作方法。