# 1 MESH网络中的基于定向天线的全双工调度算法

## 1.1 基础模型与问题抽象

### 1.1.1 网络传播模型

本文以装备有定向天线的无线MESH网络作为研究基础。不同于全向天线的全方位的接收或者发送信号，定向天线可以根据自身的需求选择性的获取某个方向的信息，或者定向的发送信号。目前定向天线有两类：传统定向天线和智能天线系统。不同于传统定向天线，智能天线不仅有辐射单元还包括划分网格和控制单元。控制单元通过数字信号处理器实现，可以说是智能天线的大脑。定向天线的优势在于其可以扩大传输范围，提高空间重用，阻止多径效应，有效的降低了干扰从而提高了网络容量。

天线增益是天线定向性的度量。与理想的全向天线在各个方向所产生的输出相比，天线增益定义为在特定方向上的功率输出。天线的增益与天线的有线面积等因素相关。

在这里我们为了研究的简单化，天线具有固定方向和定向角度，同时天线只考虑主瓣，对天线所存在的副瓣暂时不做考虑，同时天线的增益在天线范围内保持一致。

在建立无线网络模型时一般需要考虑三种主要的传播影响，分别是确定性路径损耗、大规模衰落和小规模衰落。下面简单介绍下三种影响。

（1）确定性路径损耗：模拟了信号从发射天线到接收天线间的传输衰减。普遍假设路径损耗与距离的次方成正比，即：接收功率与发送功率的比值与距离的次方成反比。

（2）大规模衰落：对于同样的发送接收点距离，大型障碍物的阻挡会使得平均接收功率不同。这种变化在测量中通常被定义为由外界传输环境所决定的常数。

（3）小规模衰落：在典型的无线通信环境下，信号从发送节点传播到接收节点的是有多条路径的，这些多径的叠加会对信号产生波动影响，即多径效应。

在本次研究中我们不考虑传播中的那些复杂情况，使用一般的确定性功率传播模型来预测接收信号强度：

其中和分别是传输功率和接收功率，和是分别是发送天线和接收天线的天线增益，是由天线高度和传播信号波长等等所决定的常数，一般情况下，在之后的计算中可能会将该值忽略，是发送节点和接收节点之间的相对距离，表示传输过程中的路径损耗指数，一般情况下该指数在2和6之间。

在本文研究中，无线网络的拓扑结构一般情况下是固定不变的，网络是由静止的具有收发功能的节点组成。同时在本文中，不考虑地理环境的影响，将所有节点分布在同一个平面中进行研究。

在这里一个很重要的方面需要考虑，在建立模型时我们应该如何描绘或者表示对于给定信号在被接收时的干扰情况。换句话说，给定的接收终端，假设有两个或者多个发送节点同时发送数据，那么这个接收终端在接收信号时会发生什么情况？经过长久的分析研究下，对于上述的的问题渐渐有了两种描述，也就是下面将介绍的两种干扰模型。

文中，网络的拓扑结构可以通过有向图来整体表示。表示节点的全部集合，表示链路集合，同时单个链路由发送点和接收点向量表示为，其中。所有的链路都是有向链路，不考虑链路的双向性，即链路和表示不同的链路。链路存在表示在网络中，节点可以直接发送数据至节点。

我们将节点分布于固定大小的空间中。在密集的随机网络中，节点随机分布在欧式空间中，即：节点间相互独立并且均匀分布在给定空间中。每个节点只参与一个链路，即：节点对某一特定节点发送或者只参与单个发送节点数据的接收。在密集的任意网络中，节点分布在任意位置，可以构成任意特殊情况的网络拓扑结构。

对于给定的一组链路集合，一个单独的链路可以成功通信的必要条件是该链路接收节点的信噪比（SNR）需要超过硬性条件所规定的临界值，

其中是外界环境确定的常数，是发送功率，表示外界环境噪声，是发送节点到接收节点之间的距离。

### 1.1.2 干扰模型

协议干扰模型是根据IEEE 802.11MAC层协议来提出的，该模型把干扰当作一种碰撞来进行处理。当前所说的协议干扰模型就是一种基于MAC层协议来判断任意的两个节点是否有邻居关系，之后再判断这两个邻居节点各自构成的链路能否在同一时刻进行数据传输。两条链路之间存在着两种关系，干扰或者不干扰。在模型中每个节点都一个相对固定的圆形传输范围，在该区域以外的节点不会受到当前信号传输所产生的干扰。如图2.1所示，链路1进行数据传输会对链路2产生干扰，但是对传输范围以外的链路3不会造成干扰。

协议干扰模型可以说是一种仅仅考虑一对链路之间的相互影响的情况。在协议干扰模型下对于给定链路与其他链路的相互干扰，给定链路可以成功通信的条件是链路的接收节点满足一下条件：（其中表示接收节点接收到的有效信号功率，表示接收节点接收到的的干扰功率）。

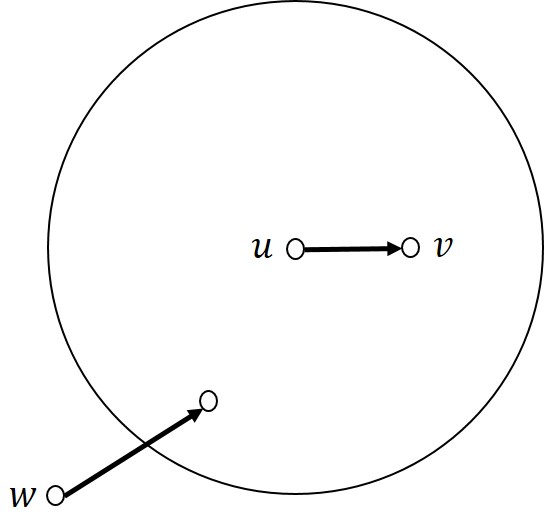


图1.1 协议干扰模型

由上述可知协议干扰模型中对于给定链路，为了使该链路可以成功传输，必须满足以下两种情况：

（1）该链路的最大距离不能超过一个给定节点通信范围，即：成立。

（2）其他同时传说的链路的发送节点w需要满足下列条件：

其中参数是协议干扰模型中为了保护使得邻居节点不相互传输冲突而定义的临界值，同时该值可以允许一定程度上的误差存在。表示该空间中发送节点的最大传输距离。

协议干扰模型中最大的缺点就是没有描述出一个真实的物理传输情况，只是很简单的认为不在其他链路的干扰范围内的任意链路都可以进行随意传输。事实上，对于某个链路的干扰也不应该仅仅分开考虑每个链路对该链路的影响，对于一个不足与产生影响传输的链路我们不用对其有太多的考虑，但是这么一个微不足道的干扰的数量变得很大的时候，我们便不能继续对其进行忽视，故而，更为现实的情况应该是将所有的链路的干扰综合在一起进行判断。虽然从实际出发，协议干扰模型有一定的局限性，但是我们不能对该协议进行想当然的否认，协议干扰模型将现实情况进行了简化，提高了模型的可操作性。

物理干扰模型采用了一种比较符合实际情况的方法，链路能否准确传输数据由该链路接收节点的信号干扰噪声比（SINR）所决定的。用表示可以同时进行通信传输的链路的发送节点的集合，表示网络中节点的传输功率。那么对于该链路集中链路成功传输的条件为：

其中的表示链路的距离，表示外界环境对接收节点的噪声影响，表示在链路成功接收情况下的最小的信号干扰噪声比。注意在大多数情况下有成立。如下图2.2所示。

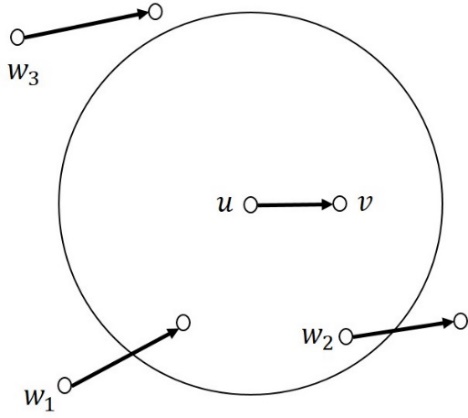


图1.2 物理干扰模型

### 1.1.3 基于定向天线的干扰模型的建立

在这里我们给出本次研究所利用到的模型的建立，我们研究的天线传输方式为发送节点定向传输，接收节点全向接收。当前对于给定的链路集合，其中每个链路表示从发送节点到接收节点发出的通信请求。

我们假设所有的节点随机分布在欧式空间中，并且任意两个节点和之间的距离表示为，链路的长短表示为。同时我们所采用的定向天线简单的简化为只有一个定向主波束，同时天线的方向固定。

对于两对同时发送的链路和，全向天线下这两条链路之间会或多或少的产生一些干扰。但对于节点使用定向天线的情况下，发送节点存在一个干扰区间，只有在该干扰范围内链路才能受到来自该发送节点的影响。

下面给出基于定向天线下的干扰模型的定义，在该模型中对于链路成功传输条件如下：

（1）在没有其他链路存在的情况下，满足：

（2）在有其他同时传输的链路集中，满足：

其中上述条件中，表示链路发送节点的传输功率，是路径损耗指数，表示链路可以成功传输所需要的最小的阈值，表示外界环境对接收节点的噪声干扰，表示同时传输的其他链路，和分别表示发送节点和接收节点的天线增益，由天线的硬件条件所确定。对于我们在下面再做详细的解释。

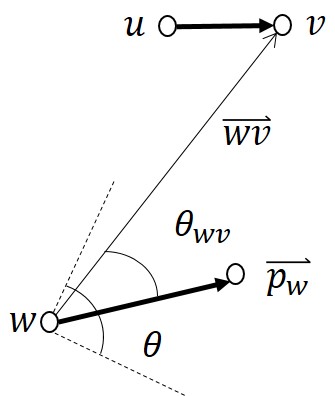


图1.3 定向角度

对于定向天线所使用的公式相对比较复杂，由于定向的加入使得链路的干扰范围变小，对干扰的计算中我们需要确定发送节点能否传输至某个接收节点上。这里我们使用来判断发送节点能否干扰到接收节点，同时定义发送节点定向天线的定向方向由向量来表示，天线的定向角度由表示，判断条件如下：

其中是天线方向与发送接收节点连续所形成的角度，通过该夹角的大小判断接收节点是否在传输范围中，如图2.3所示。计算方法为：。此外我们定义发送节点与接收节点之间的干扰距离。从公式中我们可以知道，只有当一条链路的接收节点在其他链路的发送节点的传输覆盖范围以内时，该链路才受到链路的干扰影响。否则干扰距离趋于无限大，使得两个节点之间的实际干扰趋于零。

在这里我们假设所有的发送节点都是使用相同的传输功率来进行数据传输，同时算法的分析是在所有的传输功率固定不变的情况下进行的。

我们还给出一些需要使用的定义，用来表示信号从发送节点到接收节点的接收功率，用表示接收节点收到的来自同一时刻调度的发送节点的信号干扰。

## 1.2 调度算法与算法分析

在本次研究中，我们的主要目的是解决在单个时隙中可以同时调度的最多的链路数量。该问题是一种求最大值问题，对于给定输入的链路集合，我们需要得出可以被同一时隙调度的链路子集。下面我们将介绍针对该问题而设计的基于定向天线的调度算法，并且给出该算法性能的数学分析。

### 1.2.1 半双工调度算法

为了解决所谓的最大无干扰链路集合的问题，我们给出了下面算法，首先我们定义一些使用的参数。我们用表示链路对链路所所产生的影响，在这里称之为相关干扰，。因而一个链路集合对链路产生的相关干扰就可以表示为该链路集中所有的链路对单独产生的相关干扰以及外界环境噪声的总和，我们称这个总相关干扰为链路所受到的影响，用表示，即：

由上式可以看出对于链路而言，当所受到的影响满足，相当于受到的同时调度的来自链路集中所有链路的信号干扰噪声比满足，也就是说在满足这种条件下，链路是可以成功发送的。

表3.1 基于定向天线的链路调度算法

Tab. 3.1 Link scheduling algorithm based on directional antenna

|  |
| --- |
| **算法1** 单时隙调度算法 |
| **输入**：链路集合：；  **输出：**单时隙调度链路集合；  设置参数（该参数在公式（3.5）中给出）；  迭代重复  选择链路集合中最短的链路加入到集合中，即：；  将选择链路从链路集合中删除；  迭代重复  在链路集合中选择会干扰到链路的链路加入到集合中，  即：；  直到链路集遍历结束；  将集合中满足的链路，从链路集合中删除；  集合重置为空；  将集合中满足的链路，从链路集合中删除；  直到集合为空；  返回集合。 |

在上面的单时隙调度算法中，可以说是用一种穷举的方法，对于给定输入链路集合通过遍历每个链路并做出选择的方式来得到可以在一个时隙中同时传输的链路集。首先，输入集合需要用链路的长度做一个简单的递增排序来判断各条链路的优先顺序，由SINR的定义我们可以看出，当传输功率不变的情况下，链路自身长度越小则该条链路的抗干扰性越好。当我们将链路加入到调度集合中后，我们需要通过以下几个步骤来保证调度集合是安全的，也就是保证调度集合中的链路可以同时成功的传输。第一步，我们通过距离来进行判断，将每条链路附件的发送节点屏蔽，以免得对链路产生太大的干扰，同时也可以相对的增加同时调度的链路的数量。删除过程中，首先确定链路传输时能否干扰到链路，将所有对产生干扰的链路至于集合中，在链路中的发送节点在链路的接收节点半径为（这里的是一个常数，在之后会做解释，并且有）的圆形区域内的链路都从链路集合中删除。第二步，对每个链路受到的影响值进行一定的限制，删除链路集合中受到调度链路的影响值超过的链路，即对于，满足的链路，从中删除。最后调度集合中的链路表示当前可以在同一个时隙中调度的所有链路，也就是我们所需要知道的链路集合。

在算法的设计中，我们要尽可能的保证任何一个合法的链路不能被无辜抛弃，并且不能让不合法的链路进入所求的调度集合中。最关键的问题还是，在给定数量的链路集中，选择出可以同时调度的最多的链路。下面我们将继续针对上述算法进行分析，证明该算法的正确性，也就是保证调度链路中不能有非法链路；之后再对算法的性能进行分析。

### 1.2.2 全双工调度算法

当前，由于物理干扰模型更加接近实际情况的诸多特性，使得对物理干扰模型的研究越来越多，也使得现在对MLS问题的处理有更多的方法。然而对于基于定向SINR模型的研究相对不多，增加了一定的复杂度。本文为了进一步扩展对MLS问题的研究，我们将问题从半双工延伸至全双工。当链路的每个节点都可以作为发送方或者接受方时，对链路的调度控制需要更加精细的处理。

这里给出对模型上的一些修改。首先由于链路的双向传输，我们需要对链路距离进行重新定义，这里我们假设链路的两个节点分别是和，根据之前的定义我们知道到的实际距离为，然后我们给出两条链路和的实际距离为：。这里我们对链路的实际距离的定义是采取的两条链路节点之间的最短距离，以保证链路的所有节点都可以成功传输。根据链路距离我们重新定义总相关干扰：

在我们给出算法的具体流程之前，我们首先考虑一下全双工传输对链路调度问题带来的挑战。由于链路的两端节点都可以作为发送或者接收方，我们需要保证链路满足条件：，来使得两端节点都可以正常通行。这里给出“边界距离”的定义，它表示两个链路之间需要保持的最小距离。

定义：边界距离（BD）：对于链路，在边界距离以内不应该有任何其他链路存在。

其中表示链路的实际距离，下面我们给出针对链路调度问题的全双工算法（FLSDA），该算法基于之前的半双工算法机制。

1）首先我们用候选序列表示算法的输入，这是一组以链路长度作为排序标准的升序序列。算法选择该序列中的第一个链路，也就是最短的链路作为最优解序列中的第一个元素，并从候选序列中删除。

2）对于而言，为了保证满足。算法选择机制排除所有与的距离小于边界距离的链路。即：。

3）然后算法选择受到链路干扰的所有链路，并组成一个新的被干扰集合。该集合中的所有链路满足：。

4）选择最短的链路并删除所有与距离小于边界距离的链路，并重复该步骤知道所有链路相互之间的最短距离都不小于。

5）最后根据当前最优解序列，删除所有属于候选序列并满足的链路。

6）重复上述选择策略直到候选序列为空。

由于该全双工算法是LSDA的拓展，本文将对全双工算法FLSDA的算法细节进行数学理论分析，分析算法的正确性及算法性能。

### 1.2.3 全双工调度算法正确性分析

上文已给出本课题研究的两大核心算法，基于定向天线的半双工链路调度策略LSDA以及扩展的全双工链路调度策略FLSDA，在这一部分本文将证明两大核心算法的算法正确性问题，由于LSDA和FLSDA的核心算法思想基本一致，本章节主要针对分析FLSDA的正确性问题，并且可以将其扩展到LSDA。

定理1.1：基于定向天线的全双工调度算法提供合法解。

证明：为了方便证明过程的描述，首先给出几个重要定义。全双工调度算法输出的最优解组为，其中有链路，由于算法的基本特性，首先被选择到最优解组中的链路长度较短，用来代表链路集合中链路长度比短的情况，用表示链路长度比长的链路集合，换句话说由于最优解的链路以链路长度升序排列，用链路将该最优解组分成两部分表示。从FLSDA算法细节可以得出，调度集合中的所有链路在该链路被置入调度集合时，其所受到的链路影响值小于，即：对于集合中的链路，当该链路加入到调度集合中时，其所受到的干扰影响满足：，由对链路干扰影响的定义情况知，对于一条链路而言，想要在单时隙调度时保持自身的正常无碰撞通信，该链路所受到的总干扰影响成立，回到算法，由上可知对链路的证明只需要针对链路加入调度集合后的干扰影响进行分析便可。所以由分析得之证明成立，可以简化为证明。

下面本文将做进一步的数学分析，首先根据算法中的安全距离筛选步骤来做一个几何上的分析，对于调度集合中的任意链路而言，该链路周围以接收节点为圆心，半径为的圆形区域是没有可以对链路产生任何干扰的链路的发送节点的存在，这里所言的没有干扰链路的存在并不表示在该圆形区域中没有链路，本次研究针对的是定向天线的节点，故而该区域内可以有链路的存在，但是这些链路的发送节点的定向传输范围不能覆盖链路的接收节点，所以这些链路对不能造成任何干扰，不影响的发送。对于任意的两个对链路都有干扰的发送节点并且成立，由三角定理及算法可知，由此可知集合中对有干扰的每条链路周围以发送节点为圆心，半径为的圆形区域相互之间没有任何交叉，可以从中得出，与链路相距一段距离范围内并且对有干扰的链路发送节点的数量是有限的，并且随着距离的增加中链路发送节点的距离就会越来越远，总之，如此看来对链路的干扰可能是有一个上限存在的，下面我们继续进行算法正确性的证明。

此时将关注点放在中可以干扰到的发送点集合对链路的影响上，首先对整个集合上的链路进行划分，以链路的接收节点为中心做一系列的圆环划分。用来表示第圆环内所包含的链路的发送节点，有满足并且成立。我们知道成立，故而中不包含任何链路的发送节点。对于除了以外的所有圆环，有在该圆环内的满足的发送节点，拥有一个半径为的圆形区域，在这里本文再定义一个扩展圆环，包含所有原来圆环中的所有发送节点和该节点所拥有的圆形区域，我们计算该扩展圆环的最小面积：

由于每个满足的发送节点周围面积为的圆形区域之间没有任何相交，并且在圆环中的发送节点到链路的最短距离为，我们使用面积的方式来讨论每个圆环区域中的发送节点的上限数量。对于中的发送节点对链路所产生的总干扰的最大值：

上述公式推导过程中，其中的值始终不变，为固定的传输功率，并且成立，表示在圆环中存在的最多的对链路有干扰的发送节点的数量，计算中本文取最短距离保证得到干扰的最大值。由于，得到成立，又。我们可以得到上述推导所得的链路所受到的来自的最大干扰值，下面我们推导链路所受到的来自中节点的总干扰值：

该最后一步的不等式推导在参数满足时成立，相对的链路受到的来自中节点的干扰影响值A满足：

在这里我们为了算法的简单化，首先令外界环境噪声干扰，同时对于所有的发送节点的定向天线进行统一，每个天线的天线增益、定向角度都保持一致，故而可以推出中的最后推导的成立，为了保证调度集合的正确性，我们需要有成立，即：

综上本文证明中得知，也就是表示对于调度集合中的链路其信号干扰噪声比满足成立，证明了算法结果的正确性。

### 1.2.4 全双工调度算法性能分析

在这里我们将对单时隙调度算法的性能进行简单的分析，首先我们用表示调度算法所得到的最优调度集合，用表示给定调度集合的最优调度解。我们比较单时隙调度算法所得到的解与最优解来分析算法的性能，通过计算被算法排除的合法链路的数量进行衡量。首先我们定义表示被算法排除的合法链路的集合，即：。我们将该集合分成两个部分，在调度算法中我们通过了两个步骤来筛选合法的调度链路，在分析时我们也将其分为两个部分有：表示算法在第一步通过距离选择过程中所排除的合法链路的集合，表示算法第二部分通过干扰影响值排除的合法链路的集合。下面我们将对这两部分集合分别分析。

定理4.2：假设集合当前的一个合法调度集合，是该集合中的一条链路，则与该链路接收节点距离在以内的集合中的链路的发送节点数量并且会对产生干扰的链路最多只有条。并且与发送节点距离在的链路最多为。

证明：对于发送节点，并且满足，对链路产生的相关干扰

有链路所受到的干扰影响值，因此最多的发送节点为。由于节点与接收节点的距离最大为，则与发送节点的距离最多，同样的方法我们可以得到距离发送节点距离在并且有的发送节点最多有，定理成立。

定理4.3：当存在的情况下，有成立。

证明：对于链路集合，该集合是在链路加入到调度集合是所删除的所有链路。有表示与接收节点的距离满足的链路。因此，对于集合中所有的发送节点都在节点的距离范围内。由定理4.2可得到集合中最多有条链路，故而该定理成立。同时可知道在定向的情况下，在此步骤删除的链路数相比于全向天线，因链路的方向以及发送节点的定向发送方向都保持随机性，有成立。

下面我们分析被排除链路中的另一种情况，给出集合的比例情况。首先我们先给出一些定义和定理来为的分析提供一些便利和理论依据。

定义4.1：我们用集合和表示度量空间中的互不相交的点的集合。有对于正整数，存在节点，当节点周围满足的节点集合中，包含的集合中的节点数比中的节点数多倍比例时，称该节点为-主导节点。

定义4.2：在上述定义的和集合中，节点和集合，我们定义表示对于所的节点，满足的集合。

定理4.4：对于二维欧式空间中定义的上述和集合，以及正整数，当满足时，在集合中存在至少一个-主导节点。

下面我们通过上述定理来给出的比例上限分析。

定理4.5：当存在的情况下，有成立。

证明：在这里我们通过反证法对该情况进行分析，首先假设的情况发生，我们考虑集合中的发送节点，定义和。有定理4.4可以知道集合中存在一个-主导节点，并且。下面我们讨论链路在算法调度集合中，以此来推翻假设并证明定理正确性。

对于任意，定义表示节点在集合中并且满足的集合。这里我们有定理知道：，。由三角关系知道有，，并且对于任意成立。因此发送节点对接收节点所产生的干扰影响

我们比较集合中节点对的总干扰影响：

从上述推导我们可以得出所受到的干扰影响比较，由于链路为合法链路，我们知道其所受到的总链路干扰影响值，因而我们得到

因此，我们知道链路所受到的来自调度集合中的干扰值不足以将其排除，也就是我们推出了之前假设的范例，假设不成立，定理正确。

综上所述，调度算法误删的合法节点的保证在调度解集合的常数比例中，很好的控制了误删比例。同时本算法我们使用了定向天线，有效的降低了节点间的干扰，干扰比例降低为，表示定向天线的定向角度。可以得之该单时隙调度算法相比较全向天线的算法有很明显的性能提高。

## 1.3 实验与结果分析

本部分通过仿真实验的方式分析所提出算法性能表现情况，为了提高实验的说服力，本文总共进行了两部分实验分析，分别对半双工算法和全双工算法进行性能分析。对实验结果进行分析，并与基于全向天线中的调度算法（我们称之为OSSA，由Goussevskaia O等人[9]提出）进行了比较。我们从链路的数量和定向天线的角度等方面对算法进行了验证。

在这里我们称本文的算法为LSDA，在本实验中，为了保证仿真实验的正确性，为实验结果分析提供便利，同时尽可能的简化实验的难度，我们定义了以下的几个前提条件：

（1）所有链路分布在二维空间中，二维空间设定为，并且链路不动；

（2）所有链路的传输功率保持一致，并且固定不变；

（3）所有链路的发送节点使用的定向天线的定向角度保持一致并且固定；

（4）实验中，外界环境噪声所造成的影响我们暂时不做考虑，定义；

在上述条件满足的情况下，我们研究在随机分布的网络拓扑结构中的调度情况，在实验中，我们做多组实验以保证结构的正确性，下面我们先分析本文中算法的性能情况，在进行比较，最后对实验进行总结。

本实验的场景图，简单表示如下所示：

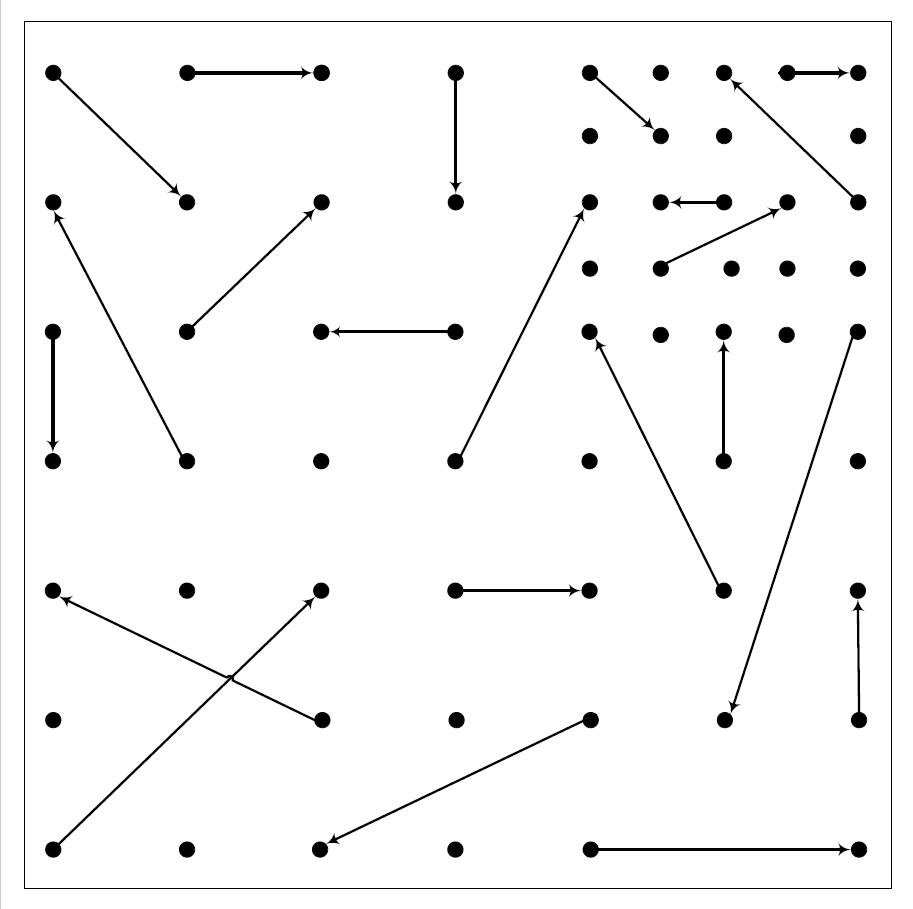


图1.4 随机分布场景图

### 1.3.1 性能衡量标准及参数定义

在单时隙调度算法中，性能的衡量标准比较单一，算法得到的调度集合的大小和算法调度集合的正确性在一般情况决定了算法的好坏性。对于给定的待调度链路集合，我们通过算法所得的调度集合中合法链路数量评判算法性能，对此也分成多种情况：（1）二维空间链路分布相对密度较小时的算法性能情况；（2）链路分布相对密度较大时的算法性能情况；（3）随机分布拓扑结构的网络中的算法性能情况；（4）集群分布的网络拓扑下的算法性能；（5）定向天线定向角度不同时的算法情况。这些特殊情况在判断一个算法的性能时都是需要进行考虑的，可能不同的算法设计就是为了某些特殊需要，这类算法在那些特殊情况下的性能表现的比较好。综上，我们也不能随意的对某个算法的性能好坏进行评价。

在这里我们给出实验中所用的部分参数的定义。路径衰落指数，硬件决定的信号干扰噪声比的阀值，外界干扰噪声。实验中链路的距离有，在不做特殊说明的情况下，我们一般使用所有链路的距离为，并且固定不变。定向天线一般情况下，天线角度。在角度为时，天线的增益一般满足；对于角度为，此时天线增益。实验中可能为了特殊分析，使用了一些不同的数据。

### 1.3.2 半双工算法性能分析

前文中已经通过数学理论分析的方式给出了本文算法的性能证明，在任意场景下可以实现输出结果的常数项近似。本部分将利用仿真实验来验证算法性能，通过与经典算法的对比提供实验说服力，并且通过控制变量法分析外界环境对算法的影响情况。

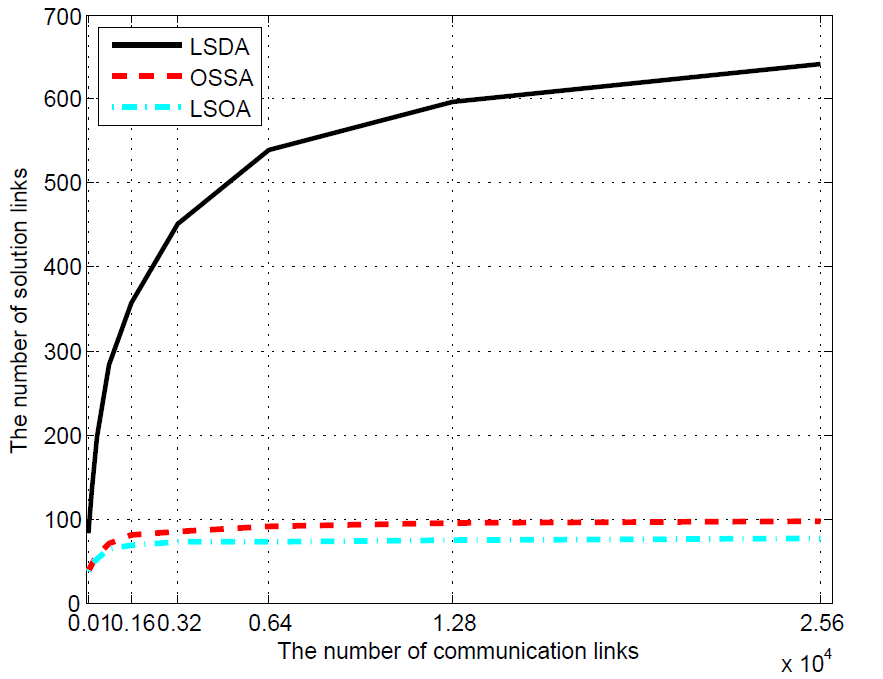


图1.5 单时隙调度算法

本文首先通过一个对比实验给出算法收到链路密度的影响并对LSDA算法的吞吐量性能表现。实验结果如图1.5所示，该实验中，本文通过调整固定区域中的通信链路数量来控制链路密度，并且通过算法输出最优解组中的链路数量作为算法性能的评判标准，单时隙中可以并行传输的通信链路数量多表示当前时隙的网络吞吐量较大。实验参数设置如下，首先通信链路在当前区域呈现随机分布的特性，定向环境中所有信号发送节点采用的天线定向角度为，并且有通信链路数量满足。图1.5中，三种算法在不同链路密度下的性能对比，其中LSOA算法是本文调度算法在全向无线网络环境中的算法表现。

通过实验图曲线可以得出，在不同链路密度下本文调度算法LSDA相比于OSSA算法都呈现较好的吞吐量表现，图中可以看出在吞吐量层面，LSDA实现了多倍的性能增益。两种算法随着链路密度的增大可达到的吞吐量逐渐趋于稳定，由于固定区域的网络容量恒定，导致调度算法所带来的吞吐量增益降低。从两种算法的最大网络容量可以说明定向天线对局部区域的网络容量有很大的提升作用，有效的降低通信链路间干扰，使得固定区域中的通信链路数量增多。换个角度分析LSDA算法与LSOA算法的性能比较，可以看出LSDA调度算法在全向网络环境中的性能表现略有不足，LSOA的各方面都稍弱于OSSA算法，这方面的原因主要来源于定向天线的复杂干扰因素，使得LSDA算法设计较为复杂，考虑因素过多。同时另一方面可以说明定向天线的天线角度对本文调度算法的影响较大，下面本文将分析天线角度的影响情况。

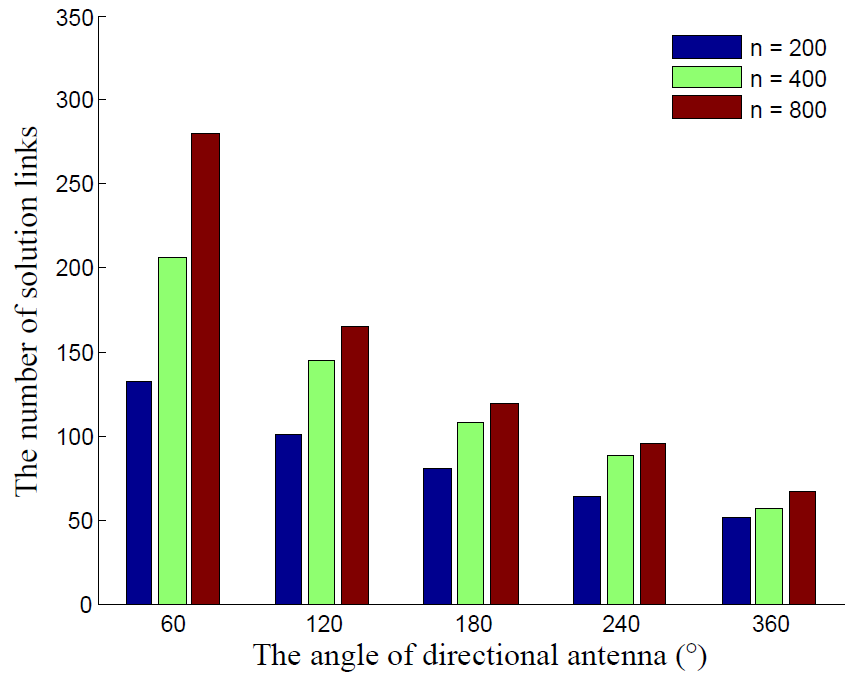


图1.6 不同角度下单时隙调度算法的性能比较

为了分析无线网络中发送节点的定向天线角度对LSDA调度算法的影响情况，本文在不同的链路密度下对算法进行分析。实验参数设置如下，首先本文对LSDA算法在三种通信链路密度下进行对比，分别设置的通信链路数量为200、400以及800，定向天线的天线角度设置为。其他参数设置如默认情况，其中天线功率恒定不变，从而有天线增益随着定向角度的变化而变化，并且满足增益。

实验结果如图1.6所示，本文采用柱状图的形式展示算法性能。从图中可以得出，在定向天线角度对算法性能影响方面，无论实验场景中的通信链路密度如何，网络吞吐量都呈现一致的变化趋势，当天线角度较小是，算法性能最佳，但是网络容量最大，随着天线角度的增大，网络吞吐量增益逐渐降低。这种变化趋势得益于定向天线对信号干扰范围的减小，提高了固定区域中的网络整体容量。从另一层面分析，通信链路数量增加提高单时隙网络吞吐量，而定向天线角度的增大削弱了网络容量，综合图1.6中的柱状图可以发现，定向天线角度的增大对区域网络容量的降低是不可弥补的，即使增加当前网络中通信链路的数量也不足以使得算法性能有较大的提升。说明了定向天线角度对算法性能有极其重要的影响。

上述实验中对定向天线角度的分析，由于在改变天线角度的同时一定程度上影响了天线的增益，为了提高实验的说服力，本文将单独对天线增益对算法影响进行实验对比。

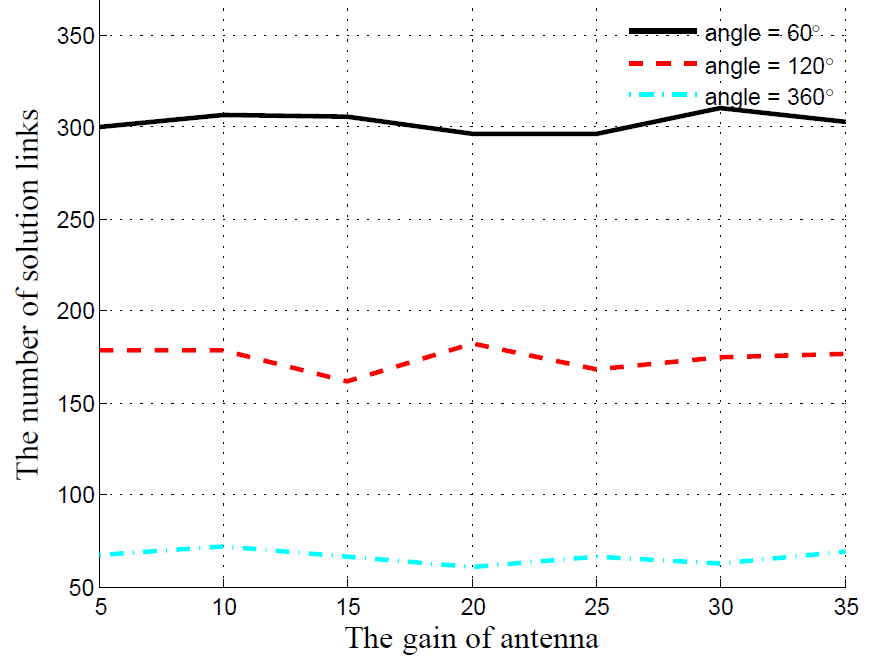


图1.7 天线增益对单时隙调度算法的性能影响

上图1.7中给出了本文提高实验说服力的一组对照实验，实验为了说明天线增益对算法性能的影响情况。实验中，本文采用LSDA算法在不同定向天线角度环境下的吞吐量性能比较，从天线角度为由于上述结果中只体现了定向角度在、以及三种条件进行分析，并且天线增益的变化情况为。实验中其他影响参数皆为恒定值，其中通信链路密度。

通过实验图1.7中的曲线可以看出，在同一定向天线干扰角度的情况下，本文调度算法LSDA的吞吐量增益几乎一致，随着天线增益的不同，吞吐量始终保持在差比可接受范围内的恒定。综合三条实验曲线可以得出相比于定向天线增益而言，天线干扰角度是决定LSDA算法吞吐量性能的关键。由此可以证实实验过程中天线增益单时隙调度算法性能影响几乎忽略不计。

接下来本文将从其他角度来分析算法的性能表现情况，分别对网络环境的路径衰落指数以及信号接收SINR阈值进行分析。

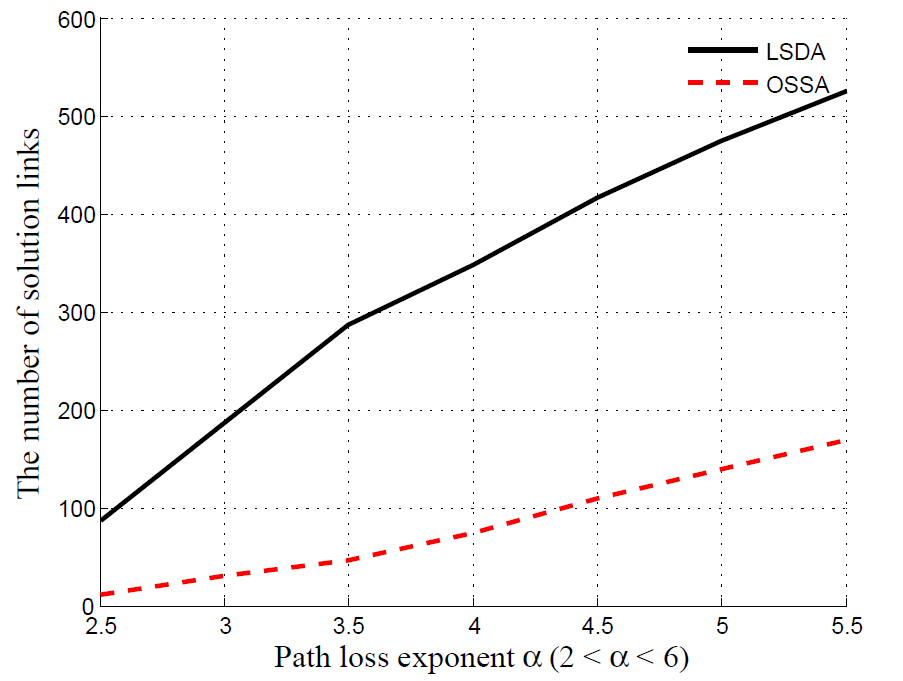


图1.8 路径衰落指数对调度算法的影响

为了方便分析路径衰落指数对调度算法的影响情况，本文采用本文调度算法与OSSA算法的性能比较进行分析，验证算法受到的来自路径衰落指数的影响。实验中本文主要设置如下，首先对网络区域中的链路密度保持恒定，通信链路数量为，无线网络环境中定向天线的干扰角度为，天线增益保持恒定值。与前文实验一致，采用单时隙调度算法给出的并行通信链路的数量作为算法性能的衡量参数，实验中自变量为路径衰落指数并且满足。

图1.8是本才实验结果，通过实验图中曲线可以简单的得出，调度算法随着路径衰落指数的增加表现出较好的吞吐量性能，这方面主要得益于调度路径衰落指数降低了链路的信号传输距离，从而通信链路之间的相互干扰影响降低，单时隙并传输的链路数量增加。从理论上来说，图中的吞吐量提高与算法本身关系较小，但是吞吐量提高的比例或者说增长速度与调度算法有密切关系，从中也证明了本文调度算法LSDA在不同路径衰落指数下的性能表现都较为理想。

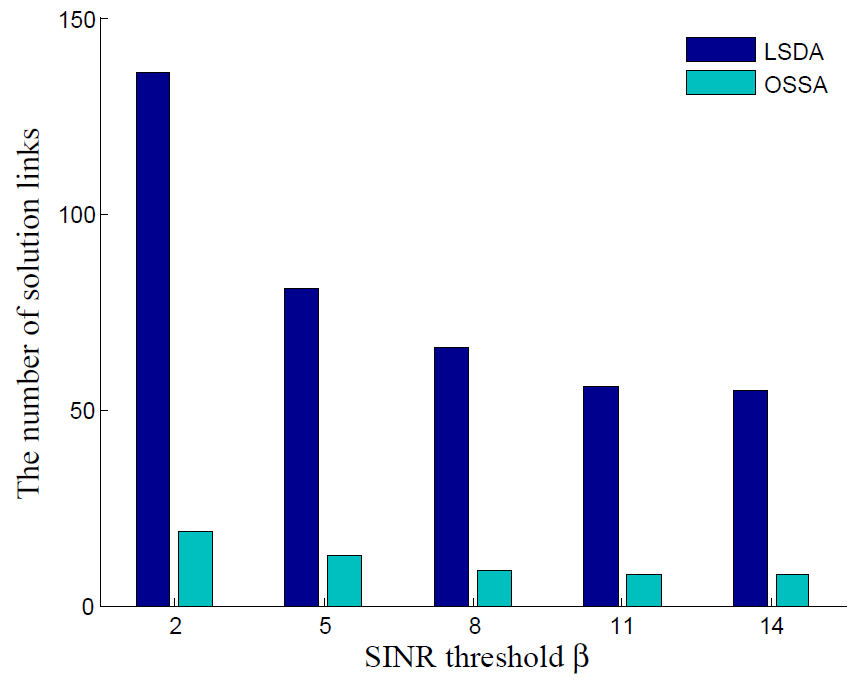


图1.9 SINR阈值对调度算法的影响

对本文半双工算法的实验分析最后一部分是研究SINR阈值对调度算法性能的影响情况，并对比不同调度算法的影响变化情况。实验中本文同样采用的是控制变量法，保持其他影响因素不变，单时隙信道竞争通信链路数量，路径衰落指数满足，定向天线干扰角度满足，其他参数详见默认参数设置，SINR阈值变化情况为。

图1.9的柱状图为本次实验结果，通过实验图可以得出SINR阈值对调度算法的吞吐量影响情况基本一致，随着SINR阈值的增大降低了网络单时隙的并行通信链路数量，从而降低当前网络吞吐量。不同调度算法随SINR阈值变化的主要不同在于增长幅度，本文调度算法随着阈值的变化而呈现大幅度的性能降低，但是在算法性能趋于稳定的恒定值上，LSDA拥有较高的吞吐量。SINR阈值的变化影响了信息接收成功率，接收节点对信号强度的要求提高，从而容忍外界干扰能力降低。实验可以证明随着SINR阈值的变化，优势的调度算法可以呈现较高的网络容量。

### 1.3.3 全双工算法性能分析

在分析完LSDA算法的性能情况后，本文继续考虑全双工算法FLSDA的性能情况，实验过程中默认参数都已在之前列出。

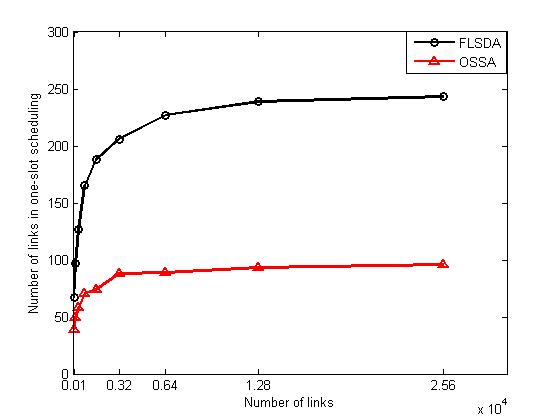
****

图4.13 ，时FLSDA算法与OOSA算法的比较

图4.13和图4.14给出了算法的部分性能情况。首先我们在自变量逐渐变化的条件下，比较两个算法FLSDA和OSSA（全向半双工算法）的网络容量，这里的网络容量是指单个时隙同时传输的链路数量。通过图4.13可以很显然的看出在定向天线角度时，两个算法的去曲线变化情况比较相似，由开始的快速增长逐渐达到最大值后停止。同时我们可以得到在整体算法表现上FLSDA优于OSSA，可以说明利用定向天线的情况下，即使在全双工这种较为劣势的条件，FLSDA算法也有较好的表现。

除此之外，我们给出了路径衰落指数对FLSDA算法的影响（图4.14）。实验中我们依然按照默认参数进行处理，定向角度，链路数量。自变量路径衰落指数变化范围为：。通过实验结果可以看出，路径衰落指数对两个算法的影响情况基本一致，路径衰落指数对算法也起到决定性的影响，同时当指数达到之后，对算法的影响开始变小，趋于稳定。

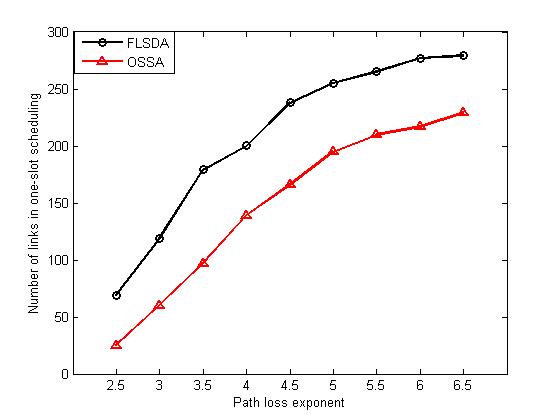


图4.14 ，时路径衰落指数对FLSDA算法和OOSA算法的影响

在实验中还有一些不足的地方，如我们基本忽视了天线定向角度和天线增益的对应关系，没有在这个方面进行详细的实验分析。链路的距离我们简单的默认了所有链路长度一致等，这些都可能影响实验的结果。同时我们还没有对网络拓扑结构进行分类分析，仅验证了在随机拓扑网络中的网络性能，实验中我们还没有验证外界噪声环境对网络的影响，这个影响也是可以对网络性能起决定性作用的因素。

### 1.3.4 实验结果总结

综合实验中所得到的结果，我们可以看出相较于全向天线的调度算法，定向天线对降低链路间干扰有明显的效果。无论是在链路密度较大的拓扑结构中还是在密度较小的网络中，算法均表现出了优于全向天线调度算法。在对算法的其他方面进行测试中，我们也得出了天线的定向角度对调度算法的影响，天线角度小，相对的链路间的干扰也变低，调度链路的数量会有增加。同时相对于定向角度，天线增益对调度算法的影响稍显微弱。总而言之，由于定向天线的引入单时隙链路调度的调度算法效果有所提高，并且该算法有足够的稳定性，对于链路密度不同的网络情况都表现了良好的性能。

在接下来的工作中，我们会针对本次实验中所出现的一些缺陷或不足来完善对定向天线调度的研究，通过对定向天线知识的累积，逐步增加天线的复杂度，优化调度算法，将定向天线拓展到智能天线等更加复杂的天线下。并且增加对比实验的数量，提高仿真实验的说服力，为算法性能分析提供有力的证明。

# 2 水下MAC协议的设计与实现

## 2.1 水下MAC协议原理

### 2.1.1 水下协议概述

普遍意义下，水下MAC协议的设计目标是为了提高水下网络容量并降低水下数据传输的冲突率，同时由于水下网络环境的传播延迟大、流动性高以及协同工作复杂等特性，使得水下MAC协议的设计中需要考虑较多影响因素。本文除了MAC协议的基础目的以外，关键的是解决水下动态网络的适应性，由于海洋水流变换不定，使得网络节点的位置始终处于变动状态，这样会导致局部区域的网络节点密度的变化以及网络的负载变化情况。从网络传输角度考虑，节点的动态使得同一链路在不同时刻的延迟都有所变化，因而水下MAC协议的设计同样需要考虑节点的延迟问题。

本文研究以提高网络吞吐量为最终目的，其中经典协议ALOHA实现减少通信过程中的控制包开销来提供有效的数据传输速率。相关研究表明基于ALOHA的MAC协议设计在网络低负载情况下相较于其他协议有较好的表现。本文期望通过对ALOHA协议的深入研究，在此基础上，数据汇集节点通过对所有数据发送节点进行集中式调度，通过合理的机制来控制发送节点数据发送方式，以此来设计并研究水下MAC协议。

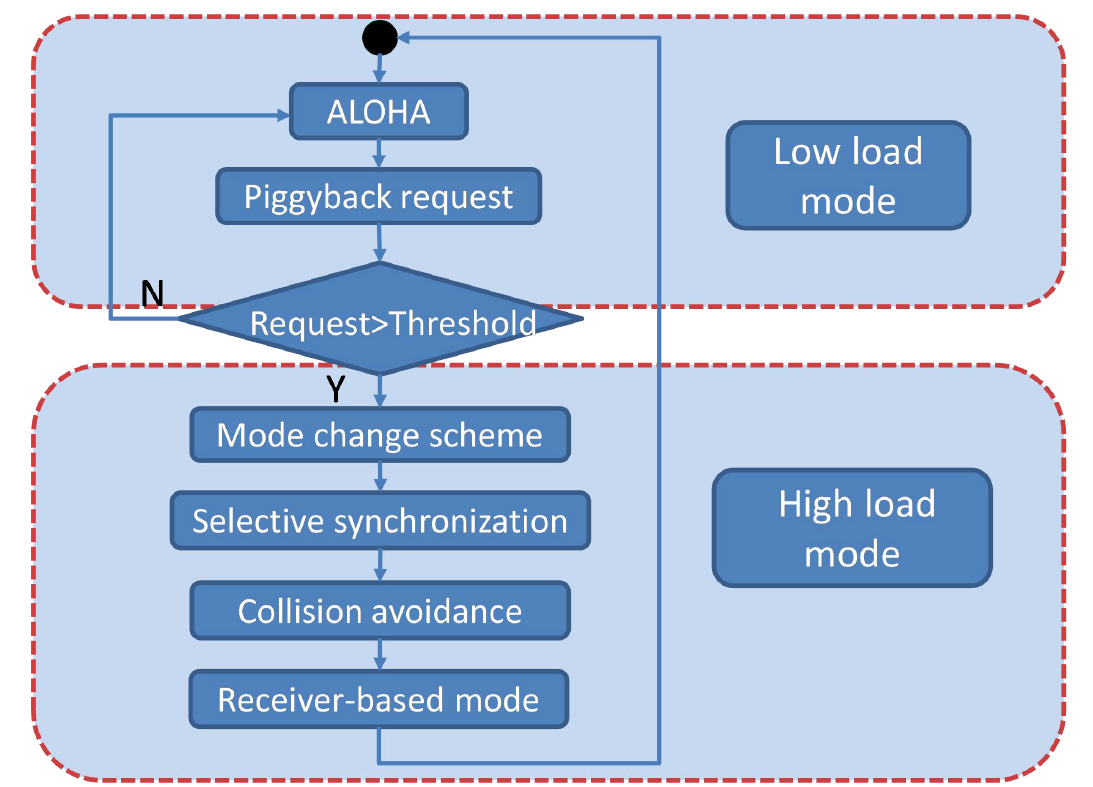


图2.1 水下MAC状态转换图

为了实现水下网络通信效率，本文希望通过提高每次成功传输的数据量达到设计目的，接下来本文将给出水下MAC协议中的其他设计思路。本文水下MAC协议将引入一下两个模式：低负载与高负载模式，如图2.1所示，两种负载模式可以进行互相转换，在低负载模式下，对于任意数据传输都可以采用ALOHA协议，区别在于本文采用了夹带技术（piggyback）用以通知数据接收节点下个传输时隙将会有数据发送节点对其进行数据传输。例如在图2.2中，当前时刻节点2正在先节点1进行数据传输，此时将数据包包头的预定接收节点（Intended Receiver）和预定数据包大小修改用以表示当前节点2有需要发送给节点3的缓存数据包，并且节点2不会进行数据发送除非节点3有数据接收需求或者当前时间门限耗尽。当节点3侦听到那些预传输控制包后会评估其是否繁忙，这里本文采用有效值来作为判断节点是否方面的度量标准，当一个节点的请求超过有效性值，该节点会进入高负载传输模式。

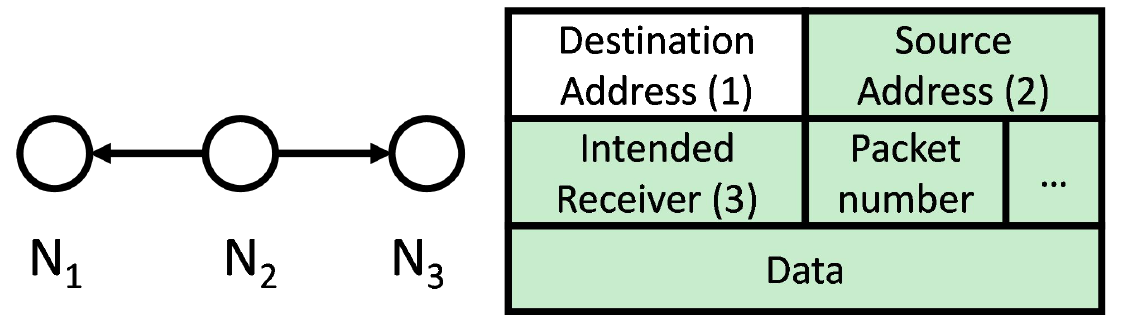


图2.2 夹带技术举例

在高负载传输模式中，所有的传输过程都是基于接收者来工作，接收节点首先需要计算所有与其直接通信的邻居节点传输时延，随后综合接收节点侦听到的发送请求来调度发送节点的数据传输顺序。对于调度机制的设计本文主要希望通过调度节点数据传输时间来使得所有的发送节点可以依次进行数据传输而不产生冲突。对于发送节点而言，当其侦听到该调度策略后，会将该数据发送安排加入自身的时间表中，并按照特定的调度策略来准时发送数据，当这些阶段都进行完成后，每个节点再重新回到低负载模式。

### 2.1.2 协议工作原理

在前文提到，本文水下MAC协议根据网络负载的不同分为两种模式：低负载模式和高负载模式。此外当节点从低负载模式切换至高负载模式的时候，预定接收者与其邻居节点的内部传输时延将会在很短的时间内计算完成，从而本文水下MAC协议不仅仅适用于不同传输流量的网络环境下，同时在节点拥有移动特性的动态网络中同样有效。

接下来本文将解释水下MAC协议的工作原理，如图2.3所示，这是解释水下MAC协议工作的经典拓扑结构。拓扑图中实线的尾部节点表示数据发送节点，箭头所指节点为数据包目的节点，虚线表示接收节点的串听过程（例如虚线箭头所指节点并不是数据传输的目的节点，但是却可以侦听并接收到该数据）。

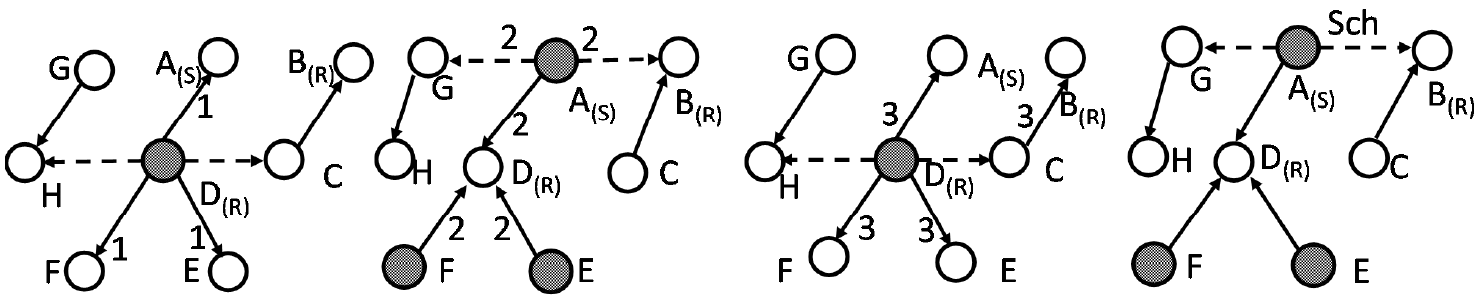


图2.3 节点局部时延评估的同步性

（1） 低负载模式（基于ALOHA协议）：