

# 基于新型物理层技术的自组网MAC协议 研究

作者姓名\_\_\_\_\_郁威威\_\_\_\_\_

学校导师姓名、职称\_\_\_\_\_史 琰 副教授\_\_\_\_\_

企业导师姓名、职称\_\_\_\_\_苏 郁 高工\_\_\_\_\_

申请学位类别\_\_\_\_\_工程硕士\_\_\_\_\_



学校代码 10701  
分 类 号 TN82

学 号 1501120463  
密 级 公开

# 西安电子科技大学

## 硕士学位论文

### 基于新型物理层技术的自组网MAC协议 研究

作者姓名：郁威威

领 域：电子与通信工程

学位类别：工程硕士

学校导师姓名、职称：史 琰 副教授

企业导师姓名、职称：苏 郁 高工

学 院：通信工程学院

提交日期：2018 年 4 月



# **Study of a Distributed TDMA Protocol in Ad hoc Networks Based on a New Technique in PHY Layer**

A thesis submitted to  
XIDIAN UNIVERSITY  
in partial fulfillment of the requirements  
for the degree of Master  
in Electronics and Communications Engineering

By

Yu Wei-wei

Supervisor: Shi Yan

Title: Associate Professor

Supervisor: Su Yu

Title: Senior Engineer

April 2018



## 西安电子科技大学 学位论文独创性（或创新性）声明

秉承学校严谨的学风和优良的科学道德，本人声明所呈交的论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢中所罗列的内容以外，论文中不包含其他人已经发表或撰写过的研究成果；也不包含为获得西安电子科技大学或其它教育机构的学位或证书而使用过的材料。与我一同工作的同事对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

学位论文若有不实之处，本人承担一切法律责任。

本人签名：\_\_\_\_\_ 日 期：\_\_\_\_\_

## 西安电子科技大学 关于论文使用授权的说明

本人完全了解西安电子科技大学有关保留和使用学位论文的规定，即：研究生在校攻读学位期间论文工作的知识产权属于西安电子科技大学。学校有权保留送交论文的复印件，允许查阅、借阅论文；学校可以公布论文的全部或部分内容，允许采用影印、缩印或其它复制手段保存论文。同时本人保证，结合学位论文研究成果完成的论文、发明专利等成果，署名单位为西安电子科技大学。

保密的学位论文在\_\_\_\_年解密后适用本授权书。

本人签名：\_\_\_\_\_ 导师签名：\_\_\_\_\_

日 期：\_\_\_\_\_ 日 期：\_\_\_\_\_





## 摘要

移动自组织网络是一种无中心、自组织的无线网络，因其快速部署、抗毁性强等特点正在得到越来越广泛的应用，是未来无线通信网络的关键技术。定向天线是一种新型的物理层技术，具有传输距离远、传输速率高、抗干扰能力强的特点。为了提升移动自组织网络的性能，宽带自组网通信设备越来越多地采用定向天线。相对于全向传输，定向成功传输的条件更为严格，如何实现分布式的链路调度成为定向自组网的MAC协议的关键技术。同时，如何充分利用定向传输的空间复用增益来提高整个网络的吞吐能力，成为定向自组网性能优化的关键问题。

本文研究的自组网设备采用“全向+定向”的传输方式，在定向天线的基础上增加一部支持多方向同时接收的全向天线，用于节点快速入网和控制信息的接收。MAC协议的设计基于TDMA协议，重点进行了两个部分的研究，一是基于干扰抑制的分布式链路调度算法，二是链路共存测试算法。链路调度采用时隙预分配和动态预约相结合的方式，针对时隙动态预约必需的筛除干扰时隙过程，提出一种利用角度值来量化干扰波束覆盖范围的干扰时隙判定方法，有效地降低了预约链路引入的网络干扰。链路共存测试通过实际的链路传输来测试新建立的链路是否存在干扰，并对干扰链路进行释放。采用将共存测试过程“织入”链路调度过程的方式，极大地降低了共存测试所耗费的时间；针对参与测试链路的选择，提出了一种基于全向接收检测的选择方法，只将那些具有较大可能性产生干扰或被干扰的链路纳入测试的范围，既减少了共存测试耗费的开销也提高了测试的有效性；测试出干扰存在即对干扰链路进行释放，尽可能地减少时隙资源的浪费。

本文使用OPNET仿真工具对所设计的移动自组织网络MAC协议进行建模仿真，通过设置不同的网络拓扑、节点密度和业务负载等参数来检验链路调度算法和共存测试算法的性能。仿真结果表明，基于干扰抑制的链路调度算法可以有效地筛除干扰时隙，从而降低链路干扰，而在增加了链路共存测试算法后，网络中干扰链路的数量进一步地减少，网络的吞吐量等性能得到有效的提升。

**关键词：**自组织网络，定向天线，时分多址，分布式链路调度，链路共存测试



## ABSTRACT

A mobile ad hoc network features non-centralization and self-organization. It is being applied more and more widely because of its rapid deployment and strong resistance to damage, and it is a key technology for future wireless communications networks. Directional antenna is a new kind of technology, which has the characteristics of long transmission distance, high communication rate, and strong anti-interference ability. In order to improve the performance of mobile ad hoc networks, broadband ad hoc network communication devices increasingly use directional antennas. Compared with omnidirectional transmission, the conditions for successful directional transmission are more stringent. How to implement distributed link scheduling becomes a key technology for the MAC protocols in directional ad hoc networks. Moreover, how to make full use of the spatial multiplexing gain of directional transmission to improve the throughput of the entire network has become a key issue in the performance optimization of a directional ad hoc network.

The ad hoc networking equipment studied in this paper adopts both omni-directional and directional antennas. The omni-directional antenna supports multi-directional simultaneous reception for exchanging control information, and the directional one fulfills data transmission. The new MAC protocol is based on TDMA and optimized on two main parts. The one is the distributed link schedule algorithm based on interference suppression, and the other is the link coexistence test algorithm. The link schedule algorithm adopts a combination of slot pre-allocation and dynamic reservation. In order to find interfered slots, a method using angles to quantify the interference beam coverage is proposed, and the interferences introduced by existing links are effectively decreased. The link coexistence test algorithm tests the newly established links for interference through actual link transmission and releases the interfering links. By the method of "weaving" the coexistence testing into the link scheduling, the time spent in coexistence testing is greatly reduced. For the selection of test-participating links, a method based on omnidirectional receiving detection is proposed. In this method, only those links that have a high probability of interfering or being interfered will be selected, which not only decreases the cost of coexistence testing but also improves the effectiveness of the test. If there appears some interferences, the links with interference will be released as soon as possible to minimize the waste of time slot resources.

In this thesis, OPNET is used to model and simulate the new MAC protocol in mobile ad hoc networks. By setting different network topology, node density and service load parameters, the performance of link schedule algorithm and link coexistence test algorithm is verified. The simulation results show that the link schedule algorithm can effectively filter out the time slots with interference and reduce the link interference. After using the link coexistence test algorithm, the number of interfering links in the network is further decreased, and the performance of throughput is effectively improved.

**Keywords:** Ad hoc Networks, Directional Antenna, Time Division Multiple Access, Distributed Link Scheduling, Link Coexistence Testing

## 插图索引

图 1.1	全向和定向天线传输模式示意图 .....	3
图 1.2	定向天线空间复用示意图 .....	4
图 1.3	“耳聋”现象示意图 .....	4
图 2.1	邻居节点角度计算示意图 .....	10
图 2.2	同一方向及一跳关系示意图 .....	10
图 2.3	全向时分网络示意图 .....	11
图 2.4	定向时分网络示意图 .....	12
图 2.5	TDMA 时帧结构示意图 .....	13
图 2.6	BS 时隙划分时隙图 .....	13
图 2.7	直接分配时隙数量示意图 .....	15
图 2.8	间接分配时隙拓扑示意图 .....	15
图 2.9	调度分组格式示意图 .....	19
图 2.10	定向链路干扰示意图 .....	20
图 2.11	发送干扰时隙判定示意图 .....	20
图 2.12	接收干扰时隙判定示意图 .....	21
图 2.13	时隙动态预约过程示意图 .....	22
图 2.14	时隙动态释放过程示意图 .....	24
图 3.1	外部因素干扰示意图 .....	27
图 3.2	链路干扰模型示意图 .....	28
图 3.3	链路共存测试周期示意图 .....	30
图 3.4	门限抑制法原理示意图 .....	31
图 3.5	NLI 全向接收检测法原理示意图 .....	32
图 3.6	链路共存测试过程示意图 .....	33
图 3.7	干扰时隙释放过程示意图 .....	35
图 4.1	三维天线模型示意图 .....	37
图 4.2	定向天线模型示意图 .....	38
图 4.3	进程模型示意图 .....	38
图 4.4	实际仿真中的进程模型示意图 .....	39
图 4.5	节点模型示意图 .....	40
图 4.6	拓扑模型示意图 .....	41
图 4.7	链路调度仿真拓扑模型示意图 .....	43

图 4.8 较少重合链路拓扑吞吐量和丢包率随负载变化 .....	45
图 4.9 较少重合链路拓扑端到端时延随负载变化 .....	46
图 4.10 较多重合链路拓扑吞吐量和丢包率随负载变化 .....	46
图 4.11 较多重合链路拓扑端到端时延随负载变化 .....	47
图 4.12 链路共存测试仿真拓扑模型示意图 .....	48
图 4.13 较少邻近链路拓扑吞吐量和丢包率随负载变化 .....	49
图 4.14 较少邻近链路拓扑端到端时延随负载变化 .....	50
图 4.15 较多邻近链路拓扑吞吐量和丢包率随负载变化 .....	50
图 4.16 较多邻近链路拓扑端到端时延随负载变化 .....	51

## 表格索引

表 2.1 业务时隙状态 .....	18
表 4.1 链路调度过程仿真参数 .....	42
表 4.2 业务负载设置 .....	44





## 符号对照表

符号	符号名称	符号单位
$\alpha$	同一方向两个发送节点与接收节点的夹角	°
$M$	界定同一方向的门限值	°
$m$	一个超帧包含的复帧数	个
$n$	一个复帧包含的业务时隙数	个
$K$	定向天线的波束数	个
$N$	最大邻居节点数	个
$S_{avail}$	可以用于预分配的时隙集合	无
$S_{total}$	一个复帧包含的所有业务时隙集合	无
$S_{used}$	本节点用于发送或接收的预分配时隙集合	无
$S_{self}$	其他节点为本节点预分配的时隙集合	无
$S_{com}$	其他节点为公共邻居节点预分配的时隙集合	无
$Q_{avg}$	统计平均队列长度	无
$w_q$	队长反应因子	无
$q$	实时队列长度	无
$IS_{avg}$	统计平均空闲时隙数	个
$w_s$	空闲时隙数反应因子	无
$s$	实时空闲时隙数	个
$Q_{past}$	当前时刻之前的平均队列长度	无
$\beta$	预约弹性因子	无
$BS_i$	编号为 $i$ 的广播时隙	无
$TS_i$	编号为 $i$ 的业务时隙	无
$SS_i$	编号为 $i$ 的调度时隙	无
$\{N_A\}$	节点 A 的邻居节点集合	无
$l$	链路发送节点与另一条链路接收节点的连线	无
$w$	节点最大发送增益方向与 $l$ 的夹角	°
$w'$	节点最大接收增益方向与 $l$ 的夹角	°
$\Omega$	定向天线主瓣宽度	°
$D_{ij}$	节点 $i$ 记录的节点 $j$ 所在的角度	°
$I_T$	发送干扰时隙判定门限	°
$I_R$	接收干扰时隙判定门限	°

符号	符号名称	符号单位
$E$	一次可预约的最大时隙数	个
$\gamma$	链路开销	无
$D_{IT}$	共存测试周期	s
$INFO$	测试通知分组	无
$T_{ij}$	节点 $i$ 为节点 $j$ 预分配的时隙	无
$\varphi$	定向天线的球坐标系模型 phi 角	°
$\theta$	定向天线的球坐标系模型 theta 角	°
$C_r$	信道速率	Mbps
$L_b$	广播时隙长度	ms
$L_t$	业务时隙长度	ms
$L_m$	复帧长度	ms
$L_s$	超帧长度	ms
$G_m$	定向天线主瓣增益	dB
$G_s$	定向天线旁瓣增益	dB
$n_d$	一个复帧最多可以发送的业务分组数	个
$n_k$	网络拓扑中的节点数	个
$S$	网络最大吞吐量	packets/s

## 缩略语对照表

缩略语	英文全称	中文对照
BS	Broadcast Slot	广播时隙
DS	Direction Slot	方向时隙
IS-TDMA	Inteference Suppression TDMA	干扰抑制时分多址协议
MAC	Medium Access Control	多址接入控制
MANET	Mobile Ad-hoc Network	移动自组织网络
NL	New Link	新建链路
NLI	New Link Interference	新建链路干扰
OL	Old Link	已有链路
OLI	Old Link Interference	已有链路干扰
PAN	Personal Area Network	个人域网
RACK	Release Acknowledgement	时隙释放确认
RR	Release Receiver	释放接收节点
RREQ	Release Requeset	时隙释放请求
RT	Release Transmitter	释放发送节点
SACK	Subscribe Acknowledgement	时隙预约确认
SR	Subscribe Receiver	预约接收节点
SREP	Subscribe Response	时隙预约应答
SREQ	Subscribe Requeset	时隙预约请求
SS	Schedule Slot	调度时隙
ST	Subscribe Transmitter	预约发送节点
TDMA	Time Division Mutiple Access	时分多址协议
TS	Traffic Slot	业务时隙



# 目录

摘要 .....	I
ABSTRACT .....	III
插图索引 .....	V
表格索引 .....	VII
符号对照表 .....	IX
缩略语对照表 .....	XI
<b>第一章 绪论</b> .....	<b>1</b>
1.1 自组织网络概述 .....	1
1.1.1 自组织网络的特点 .....	1
1.1.2 自组织网络的应用领域 .....	2
1.2 定向天线概述 .....	3
1.2.1 定向天线的特点 .....	3
1.2.2 定向天线在自组织网络中的应用 .....	4
1.2.3 定向自组网 MAC 协议研究现状 .....	5
1.3 研究意义与目标 .....	5
1.4 章节内容安排 .....	6
<b>第二章 基于干扰抑制的链路调度过程设计</b> .....	<b>9</b>
2.1 算法的提出 .....	9
2.1.1 物理层能力 .....	9
2.1.2 定向时分复用 .....	11
2.1.3 TDMA 时帧结构 .....	12
2.1.4 业务时隙混合分配 .....	13
2.2 时隙预分配 .....	14
2.2.1 时隙快速分配 .....	14
2.2.2 时隙冲突分解 .....	16
2.3 分布式链路调度算法 .....	17
2.3.1 调度启动条件 .....	17
2.3.2 调度时隙 .....	18
2.3.3 业务时隙状态 .....	18
2.3.4 调度分组格式 .....	19
2.3.5 干扰时隙判定 .....	19

2.3.6	链路调度过程.....	22
2.4	复杂度与开销分析 .....	25
2.4.1	复杂度分析.....	25
2.4.2	开销分析.....	25
第三章	基于接收检测的链路共存测试过程设计 .....	27
3.1	算法的提出 .....	27
3.1.1	造成干扰的外部因素.....	27
3.1.2	共存测试的必要性.....	27
3.2	链路共存测试算法 .....	28
3.2.1	链路干扰模型.....	28
3.2.2	共存测试周期.....	29
3.2.3	测试链路选择.....	30
3.2.4	测试链路通知.....	32
3.2.5	共存测试过程.....	33
3.3	算法复杂度与开销分析 .....	35
3.3.1	复杂度分析.....	36
3.3.2	开销分析.....	36
第四章	计算机仿真与结果分析 .....	37
4.1	仿真平台搭建 .....	37
4.1.1	天线模型.....	37
4.1.2	进程模型.....	38
4.1.3	节点模型.....	40
4.1.4	拓扑模型.....	40
4.2	性能评价指标 .....	41
4.3	链路调度性能分析 .....	42
4.3.1	仿真参数与拓扑模型.....	42
4.3.2	不同业务负载下的性能分析.....	43
4.4	链路共存测试性能分析 .....	47
4.4.1	仿真参数与拓扑模型.....	48
4.4.2	不同业务负载下的性能分析.....	48
第五章	总结与展望 .....	53
5.1	研究总结 .....	53
5.2	研究展望 .....	53
参考文献	.....	55

## 目录

---

致谢 .....	57
作者简介 .....	59





## 第一章 绪论

### 1.1 自组织网络概述

移动自组织网络 (Mobile Ad-hoc Network, MANET)<sup>[1]</sup> 是一种结合了现代移动通信和计算机网络技术的新型无线网络, 网络中的终端节点可以自由地移动并彼此保持通信。MANET 是一种自治、多跳的分布式网络, 自治体现在不需要固定基础设施的支撑, 通过终端之间的协同工作即可完成组网, 弥补了在现有基础设施 (如基站、AP 等) 不能使用时终端设备之间无法通信的缺陷; 多跳是因为终端节点的发射功率及辐射距离均有限, 如果要和远距离的节点进行正常通信则需要通过处于中间位置的节点来进行分组转发。因此, 每一个网络终端节点既可以进行处理又可以进行数据的路由分发, 这些节点通过无线自组织的方式可以致使网络拓扑变成任意形状。MANET 不但能成为一个独立存在的无线网络, 而且能接入到蜂窝通信网络和 Internet 等现有网络的末端, 成为其中的一个子网。

#### 1.1.1 自组织网络的特点

相较于传统有基础设施或有移动能力的网络而言, MANET 具备许多可被利用的特点, 总结为以下几点<sup>[2,3]</sup>:

##### ① 无中心化

网络中分布的所有终端节点都是对等的, 不需要选取中央控制节点。终端节点可以在空间中自由地移动, 或退出或加入网络, 部分节点的突发故障对网络的整体运作也不会产生根本性的破坏。

##### ② 自组织

构建一个 MANET 网络不需要任何通信基础设施的支撑。网络节点开机后通过分布式的协商调度, 并基于一定的通信协议即可自主、快速地建立起一个网络。

##### ③ 多跳路由

相比一般的移动设备, MANET 设备节点的发射功率较小、信号辐射范围有限如果一个节点需要和辐射范围之外的另一节点建立链路, 只能通过处于链路中间的其他节点进行转发的方式, 这就使得移动自组网具备了多跳的特点。

##### ④ 动态拓扑

MANET 中的所有终端设备节点可以自由地移动并随时可能开/关机或产生故障, 节点的发射功率和覆盖范围也可能发生变化, 此外网络还存在信道间干扰和地理环境复杂的问题, 所有这些因素都可能会导致网络拓扑的变化, 并且发生变化的时机和频

率是无法进行预测的。

#### ⑤ 传输带宽限制

无线网络的信道资源很宝贵,这也直接导致了可用于 MANET 通信的信道带宽很有限,如果移动终端节点在信道接入过程中发生竞争、碰撞或相互干扰等情况,会进一步地加剧带宽限制的问题。

#### ⑥ 移动终端的局限性

MANET 中的设备节点通常具备体积小,质量轻,好携带等优点,但也有不少局限性,比如电源、计算性能有限等,给软件和硬件的设计和开发带来挑战。

### 1.1.2 自组织网络的应用领域

因其所具有的自组织、快速建立和抗毁性等诸多优点,MANET 的应用范围和应用前景不言而喻,主要集中于以下几个方面<sup>[3,4,5,6]</sup>:

#### ① 军事通信领域

在当代的军事战场上,通常无法建立基站等固定基础设施,携带通信设备的士兵、武器以及各种交通工具之间可以通过建立 MANET 网络来进行相互通信,从而实现协同作战;配备了各种传感装置的移动设备可以通过移动自组网将收集的有效战场信息及时地传回作战指挥中心。目前,移动自组织网络已经成为美国军队制造战术无线电台和无线互联网控制器等通信设备的重要技术。

#### ② 灾害恢复

当发生一些严重的自然灾害时,传统无线通信相关的基础设施容易损毁,而恢复通信对及时展开营救等措施至关重要,通过使用移动自组网技术,可以在灾区很快地构建起一个临时的无线通信网,将灾害带来的损失降至最低。

#### ③ 移动会议

当今时代,移动便携设备(如手机、笔记本电脑)已经成为了人们工作中不可缺少的一部分,移动自组织网络通过将各人所携带终端设备组成无线网络来提供移动会议的功能,使得可以在这些移动设备上进行提问、交流和资料共享等。

#### ④ 无线传感器网络

无线传感器网络由于能源、功率等十分有限,其单跳覆盖范围较小,为应用 MANET 技术提供了天然条件。移动自组织网络将所有传感器节点组织成无线网络,通过传感器节点的多跳转发可以实现远距离节点的相互通信,以应对各种不同的应用需求。

#### ⑤ 个域网络互联

个域网络(Personal Area Network, PAN)只由与个人密切相关的移动装置组成,且不能连接广域网。目前可用的个域网技术(如蓝牙技术)都受限于只能实现短距离

的室内通信，而 MANET 技术则可以实现各个个域网络之间的互联互通。

## 1.2 定向天线概述

近年来，随着对移动网络的能力需求不断增长以及可利用的频谱资源不断减少，人们越来越多地将注意力集中到对无线网络的空间复用问题上，于是，定向天线也得以在更多的无线网络场景中被应用。

### 1.2.1 定向天线的特点

定向天线只在指定的方向上进行传输，其他方向无增益，而传统的全向天线可以在所有方向上进行传输，因此，定向天线具有更窄的发射/接收波束。两种天线的传输模式及波束覆盖特点如图 1.1 所示，左图所示为全向天线，右图则为定向天线。

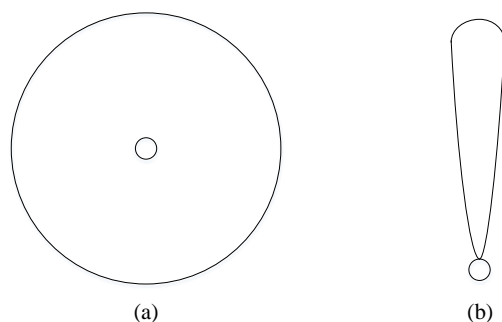


图 1.1 全向和定向天线传输模式示意图

结合上图，将定向天线的特点总结为以下四点<sup>[7]</sup>：

#### （1）节省能量

定向天线将所有的能量集中到一个方向进行发送和接收，为了覆盖到相同距离的节点，可以使用比全向天线更少的能量。

#### （2）空间复用

由于一条链路的传输只集中在一个方向上，因此该链路所在空间区域的其他方向上可以被其他节点用来建立通信链路，如图 1.2 所示，而这在使用全向天线的网络中是不可想象的，这也是定向天线相比全向天线最大的优势所在。

#### （3）更少跳数

当以一定的发射/接收功率通信时，定向天线因为有更高的发射/接收增益，所以其通信距离比全向天线更远，实现相距较远的两个节点之间的通信时，使用定向天线需要中间转发的跳数更少。

#### （4）更小端到端时延

这是由定向天线使用更少跳数的特点决定的，可以在一些对数据实时性要求较高

的网络场景中得以应用。

所有这些特点为使用定向天线的无线网络提供了减少链路干扰的能力，并且可以提高网络的效率和通信能力。

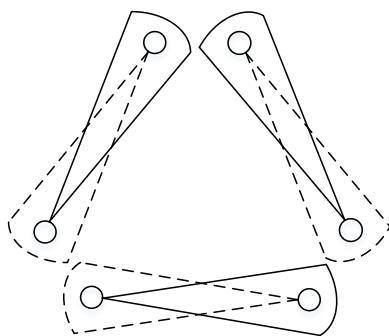


图 1.2 定向天线空间复用示意图

### 1.2.2 定向天线在自组织网络中的应用

空间复用能力是组建一个高性能分布式无线网络需要考虑的因素。为了和特定位置的另一节点进行通信，使用全向天线的节点需要向所有方向辐射信号，而在信号辐射范围内的其他节点不能同时使用无线信道，这无疑会造成信道资源的浪费。近年来，由于定向天线在空间复用上的优势，越来越多的研究人员将定向天线应用到 MANET 网络中<sup>[7,8,9,10]</sup>，以最大限度地发掘空间复用给网络性能带来的提升。

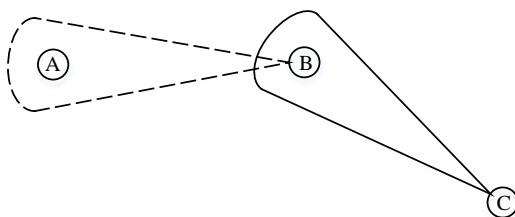


图 1.3 “耳聋”现象示意图

虽然定向天线能给 MANET 带来性能的提升，但是也给多址接入控制（Medium Access Control, MAC）协议的设计带来了挑战。一个节点的天线在某个时刻可能会指向任意方向，当它准备发送数据时不再像使用全向天线那样可以直接发送，而需要知道哪个方向是接收节点所在的方向，同样地，对于接收节点来说，它并不知道该指向何处来接收发送节点的数据，这种现象称为“耳聋”<sup>[11]</sup>，如图 1.3 所示。当节点 C 调整波束方向向 B 发送数据时，B 并不知道且正将其接收波束对准 A 进行接收，这就导致了节点 C 的传输失败。“耳聋”问题是应用定向天线的无线网络中最为常见的

问题，它增加了邻居节点发现和链路调度的难度。

### 1.2.3 定向自组网 MAC 协议研究现状

定向自组网的 MAC 协议设计一直都是定向自组网的一个重要研究方向，目前已有的定向自组网 MAC 协议大致可以分为两类：（1）基于随机接入的竞争型 MAC 协议；（2）基于时分多址（Time Division Multiple Access, TDMA）的无竞争的 MAC 协议<sup>[12,13]</sup>。

DMAC<sup>[14,15,16]</sup>（Directional MAC）是一种典型的基于竞争的定向自组网 MAC 协议，它在 IEEE 802.11 DCF 的基础上进行改进。在 DMAC 中，RTS、CTS、DATA 和 ACK 等帧的发送以及载波侦听机制采用定向的方式进行，并且使用 DNAV（Directional Network Allocation Vector）来预约信道在某一方向范围内进行传输。DMAC 未能给出定向传输的“耳聋”和隐藏终端问题的应对方案。MMAC（Multi-hop RTS MAC）<sup>[16]</sup> 是一种基于 DMAC 的 MAC 协议，它使用定向天线来交互控制帧，依据多跳 RTS/CTS/DATA/ACK 的顺序，并使用多跳 RTS 来模拟一个更远的通信覆盖距离。在 MMAC 中，节点位置信息的获取方法和多跳 RTS 的路径构造方法并没有明确的说明。

DTRA<sup>[17,18]</sup>（Directional Transmission and Reception Algorithms）是一种基于 TDMA 的 MAC 协议，它将一个帧划分成三个子帧。在邻居发现子帧中，节点始终进行发现邻居节点的尝试；在时隙预约子帧中，节点依据自身的分组传输需求来预约时隙；在业务传输子帧中，节点进行数据分组的发送。DTRA 只使用一部定向天线，但是在时隙预约时它没有考虑可能存在的干扰。PMAC<sup>[19]</sup>（Polling-based MAC）是一种更为复杂的基于 TDMA 的协议，它在传输数据分组之前还要进行 RTS 和 CTS 信息的交互。为了避免链路冲突，一个节点可能在被分配的时隙中保持静默状态，但在时隙预约时仍然没有考虑干扰。

相较于基于 TDMA，基于竞争的 MAC 协议很难知道邻居节点的 RTS 将从哪个方向发送而来，且需要上层协议来提供邻居节点的位置信息。在吞吐量表现上，基于 TDMA 的 MAC 协议也要远高于基于竞争的 MAC 协议。

## 1.3 研究意义与目标

### （1）研究意义

传统 MANET 的通信设备节点主要使用全向天线<sup>[20]</sup>，全向天线传输距离近、覆盖面积大的特点使得自组网存在端到端时延大、空间资源利用率低等问题，因此网络的性能仍然有很大的提升空间。定向天线的出现为自组织网络提供了一种新型的物理层

技术,使得网络节点间可以实现定向传输,改善网络的性能。与全向天线相比,首先,定向天线可以将发射增益集中到接收节点所在的方向,从而增大接收节点处的信噪比;其次,在同样的发射功率下定向天线的通信距离更远,从而减少网络的平均跳数,降低端到端时延;再者,定向天线的窄波束覆盖面积小,可以有效地降低链路之间的干扰,使得移动自组网可以充分地利用空间资源,从而提升网络容量和吞吐量。

移动自组网的 MAC 协议是无线通信领域的一个研究热点,目前,针对定向自组网设计的 MAC 协议大多基于 IEEE 802.11 DCF<sup>[21]</sup>,信道接入过程伴随竞争和碰撞。时分多址 (TDMA) 协议<sup>[22]</sup>是一种无冲突的多址接入协议,相比 CSMA/CA 这类竞争型的 MAC 协议,它在应对节点密度大、业务强度高场景时有着更好的性能表现。基于 TDMA 来设计定向自组网的 MAC 协议,一方面可以利用定向天线来提高网络的空间复用能力,另一方面可以利用 TDMA 来保证定向自组网信道接入的可靠性,综合提高移动自组网的网络性能。

## (2) 研究目标

本文的主要研究目标是将定向天线应用于移动自组织网络中,研究在新的物理层条件下基于时分多址协议的 MAC 协议设计,重点对分布式链路调度和链路共存测试两个部分展开研究。链路调度部分着重解决在分布式场景下业务时隙的动态预约和释放问题,在动态预约过程中使用一种新的干扰时隙判定方法,可以有效地抑制干扰时隙;链路共存测试部分从实际通信的角度出发,考虑了通信过程中由信道波动、不稳定等复杂的外部因素造成的链路干扰问题,提出一种高效的链路共存测试算法,对每一条完成调度的链路进行共存测试,及时发现干扰链路的存在并对其进行释放,减少链路干扰导致的丢包,进而提高网络的吞吐量性能。

## 1.4 章节内容安排

本文各章节的内容安排与概述如下:

第一章:绪论,首先对移动自组织网络和定向天线等通信技术的相关概念及发展情况等进行简要的介绍,然后介绍定向自组网 MAC 协议的研究现状,突出定向天线在移动自组网中的应用优势,最后介绍本文的研究意义与研究目标。

第二章:链路调度算法设计,主要介绍了时隙预分配与动态分配混合的分布式链路调度过程,基于链路之间的角度计算设计了一种干扰时隙的判定方法,并提出一种基于干扰时隙判定的链路调度算法,最后对算法的复杂度和开销展开分析。

第三章:链路共存测试算法设计,主要介绍了导致链路干扰的不可抗因素及进行链路共存测试的必要性,分析共存测试周期对测试算法的影响,设计了参与测试链路的选择和通知方法,并提出一种基于全向接收检测的链路共存测试算法,最后对算法

的复杂度和开销展开分析。

第四章：计算机仿真，使用 OPNET 仿真软件对本文所设计的 MAC 协议进行建模仿真，构建了仿真平台的各层模型，依据收集的仿真结果，重点分析了第二章和第三章所设计的算法对网络性能的影响。

第五章：总结与展望，主要对本文所提出的 MAC 协议和所做的与之相关的工作进行总结，概括其中的优点与不足，并对一些值得进一步地研究的方面进行展望。





## 第二章 基于干扰抑制的链路调度过程设计

在移动自组网中，TDMA 协议应用较为广泛。链路调度是 TDMA 协议的关键过程，调度机制的优劣直接决定着网络的吞吐量和时延等性能。在物理层具备了定向传输能力后，虽然网络的通信距离和空间复用能力得到提升，但是分布式链路调度的实现变得更加困难。如果没有一种合理的调度机制，那么相邻链路之间将出现严重的干扰，进而影响网络的吞吐量和时延等性能表现。本文基于 TDMA 提出一种具有干扰抑制功能的 MAC 协议（Inteference Suppression Time Division Mutiple Access, IS-TDMA），其针对定向传输所设计的分布式链路调度方法可以有效地降低链路干扰，充分发挥定向天线在提升自组织网络空间增益上的能力。

### 2.1 算法的提出

#### 2.1.1 物理层能力

##### （1）辅助全向天线

网络中的每一个节点都配备有一部定向天线和一部全向天线，定向天线为主传输天线，用于发送和接收与实际业务相关的数据，全向天线为辅助传输天线，它只具备接收的功能，用来全向接收与控制信息相关的数据（控制信息的发送由定向天线完成）。之所以进行这样的设计，是因为在只有一部定向天线的条件下，两个节点要想实现控制信息的交互，必须先将发送波束和接收波束的对准，对于较为复杂的交互过程可能会频繁进行操作，这无疑增加了 MAC 协议在邻居发现和链路调度等方面的设计难度，而如果增加一部全向天线来全向接收控制信息，那么将大大降低节点之间交互的难度。

##### （2）波达方向

本文所使用的天线具备记录波达方向的能力。所谓波达方向，即一个节点在接收到另一节点发送的数据时，可以获取信号的到达角，即其相对于标准参考方向的角度。由于节点在发送时是将定向天线的最大发送增益方向对准邻居节点的，因此波达方向可以反映邻居节点当前相对本节点所在的方向。若记节点的正北方向为标准 0 度方向，则可以用波达方向与正北方向的夹角度数来量化所有邻居节点所在的位置。在本文后面的内容中，很多都是基于邻居节点的角度值来进行设计。

以图 2.1 为例进行说明，节点 B 和 C 是节点 A 的两个邻居节点，当 A 接收 B 和 C 发送的包时，可以记录包的波达方向，并以正北为 0 度方向，则计算所得的波达方向与正北方向的夹角即为邻居节点所在的角度。节点 A 分别记录 B 和 C 在本节点的 45 度和 270 度方向。

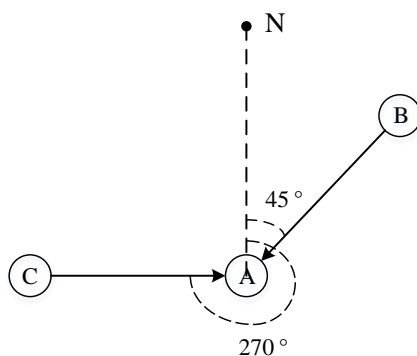


图 2.1 邻居节点角度计算示意图

### (3) 多方向接收

这里的方向是一个“方向范围”，不是具体的某一个角度所指的方向。本文使用定向天线的波束覆盖范围来表示一个方向，若能证明在节点同一个方向上的两个邻居节点必互为邻居节点，并确保一跳范围内的节点不使用相同的时隙来发送控制信息，即可在 MAC 层实现一个节点多方向接收控制信息。

传统的全向天线在同一时刻仅支持接收一个包，如果有多路信号同时到达，则接收机无法成功解调其中任何一路。而我们所使用的全向天线支持多个方向同时接收，即只要同时到达的多个包来自节点的不同方向，接收机就有能力解调出每一路信号。反之，若两个节点在另一节点的同方向上向其发送，该节点将接收不到任何一个包。这里的同一方向是指两个发送节点与同一接收节点连线的夹角  $\alpha$  小于一定的门限值  $M$ （本文中取  $M = 18^\circ$ ）。

为了实现全向天线可以同时多方向地接收控制帧，需要保证节点在同一方向上的多个邻居节点在不同的时刻进行发送，即在节点同一方向上的多个邻居节点不能复用相同的时隙。

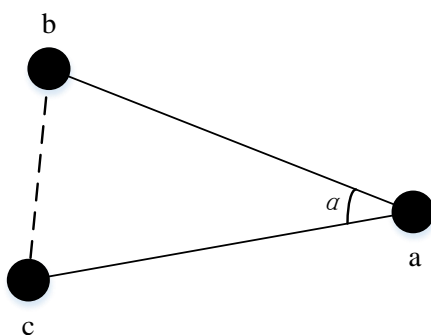


图 2.2 同一方向及一跳关系示意图

**定理 1** 如果两个节点在另一个节点的同方向，则这两个节点之间一定满足一

跳的关系。

证明：如图 2.2 所示，b、c 节点在 a 节点的同一方向上，定义节点的最大通信距离为  $L$ ， $ab \leq L$ ， $ac \leq L$ 。假定门限值  $M = 45^\circ$ ，则  $\alpha < 45^\circ$ 。根据式 2-1 所示的余弦定理：

$$bc^2 = ab^2 + ac^2 - 2ab \cdot ac \cdot \cos A \quad (2-1)$$

可知，边  $bc$  的长度随着  $\alpha$  的增大而增大，考虑极端情况令  $\alpha = 45^\circ$ ，得  $bc^2 = ab^2 + ac^2 - \sqrt{2}ab \cdot ac$ 。由图 2.2 可知，当  $\alpha$  确定时，边  $bc$  的长度完全由边  $ab$  和边  $ac$  的长度决定，b、c 两点则可以在固定轨道上来回滑动，由于  $\alpha$  小于  $60^\circ$ （等边三角形），所以极端情况为一条边长等于 0，另一条边长等于  $L$ ，此时边  $bc$  取最大值，为  $L$ 。而实际的  $M$  值是肯定小于  $45^\circ$  的，所以  $bc$  的长度最大不会超过  $L$ ，也即 b、c 一定满足一跳关系。证毕。

因此，由定理 1 可得，只要用于发送/接收控制帧的时隙的复用准则满足一跳范围不冲突，即可实现全向天线的多方向同时接收。

### 2.1.2 定向时分复用

在基于全向天线的分组时分网络中，由于天线全向覆盖的特点，相同的时隙中，一个接收节点的一跳范围之内只能存在一个发送节点，否则接收节点处将接收不到任何分组，如图 2.3 所示。

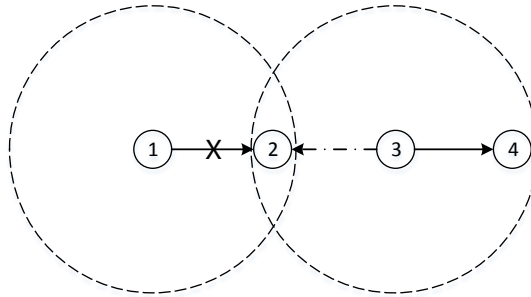


图 2.3 全向时分网络示意图

1 节点和 3 节点复用了同一时隙，在该时隙中，1 节点向 2 节点发送业务分组，3 节点向 4 节点发送业务分组，由于 2 节点在 1 和 3 节点的重合发送范围内，因此 1-2 链路将传输失败。在全向时分复用网络中，要实现无冲突建立链路，时隙需要满足两跳范围内唯一的复用准则。

而在定向分组时分网络中，由于定向天线的波束只覆盖一个很窄的方向，因此在

其他方向的邻居接收节点不会被干扰，如图 2.4 (a) 所示；虽然不需要再约束于全向场景下的时隙复用准则，但是在很多情况下，邻近的定向链路之间还是可能存在干扰的，如图 2.4 (b) 所示，3 节点同时处于 1 节点和 2 节点的发送主瓣覆盖范围内，使得 1-3 链路的传输失败。

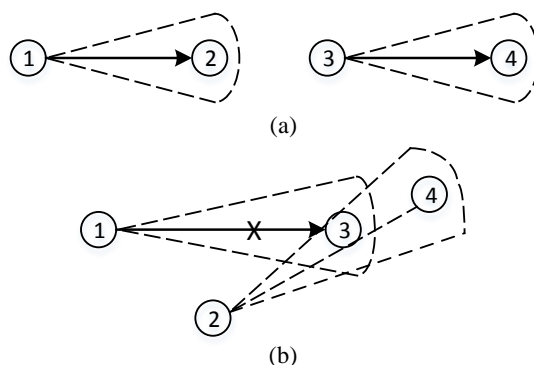


图 2.4 定向时分网络示意图

虽然应用定向天线可以提高链路的空间复用能力，但定向 TDMA 不像全向 TDMA 一样具有时隙复用准则来保证链路之间无冲突，如果不对定向链路进行合理的调度，网络中可能会出现大面积的链路冲突，达不到使用定向天线来提高网络吞吐量的初衷。本章及后面章节所涉及的内容均是致力于减少定向链路之间的冲突以获得一个较高的时隙复用率和空间复用率。

### 2.1.3 TDMA 时帧结构

本文所设计的 IS-TDMA 协议将时间信道划分成超帧、复帧和时隙三个不同长度的层级结构<sup>[23,24]</sup>，具体的划分方法如图 2.5 所示。

如图 2.5，超帧是最大的时间划分单位，一个超帧由  $m$  个连续的复帧组合而成， $m$  的大小由时隙复用准则和网络节点密度决定；一个复帧又可划分成多个具体的时隙，包括一个广播时隙（Broadcast Slot, BS）和  $n$  个业务时隙（Traffic Slot, TS）， $n$  的大小可以根据网络中每个节点上的平均业务强度来调整，以确保不会出现分组堆积导致的时延过大现象。

之所以将一个复帧划分出广播时隙和业务时隙，是因为网络中的节点不仅需要传输业务分组，在节点入网、出网和链路调度等过程中还要进行一些控制信息的交互。因此，BS 时隙主要用于周期性地与邻居节点进行同步和邻居表等信息的交互，每个节点只能占用一个；业务时隙用于向邻居节点发送业务分组，每个节点可以根据自身业务强度占用多个。

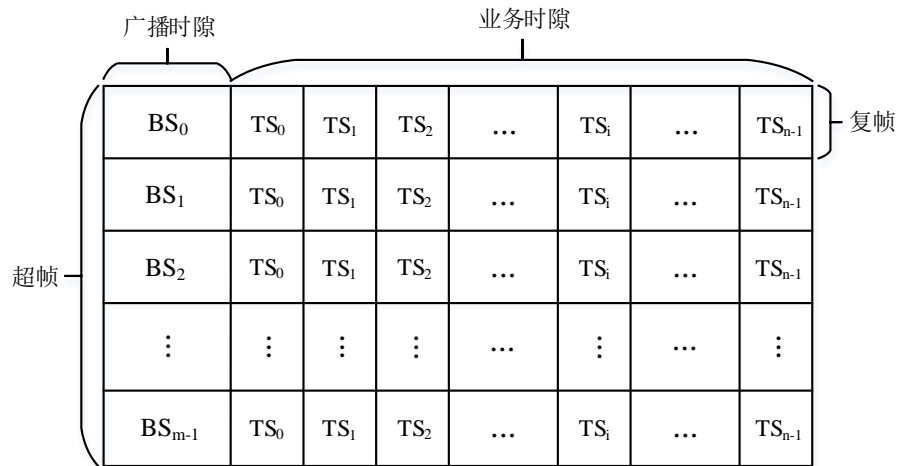


图 2.5 TDMA 时帧结构示意图

考虑到使用定向天线发送一次控制信息只能覆盖一个方向的邻居节点，而要实现控制信息的广播，则需要旋转定向天线在所有方向上均发送一次，因此我们将 BS 时隙等分成如图 2.6 所示的编号为  $0, 1, 2, \dots, K-1$  的  $K$  个微时隙，定义此微时隙为方向时隙（Direction Slot, DS）。其中， $K$  等于定向天线的波束个数，本文所使用的定向天线的波束个数为 20。

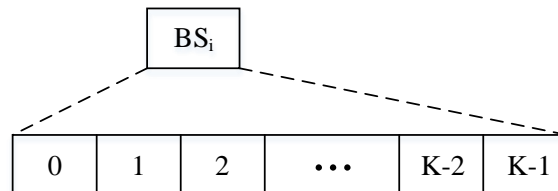


图 2.6 BS 时隙划分时隙图

当一个节点作为全网的建网节点开启建网时，它会选取某一时刻作为网络的起始时刻，也即超帧的 0 号复帧的起始时刻，后面的时间按照超帧、复帧和时隙长度进行划分即可；而当一个普通节点进入网络中，它先要完成入网同步过程，随后将选取的基准节点的下一个即将到来的超帧起始时刻作为本节点的超帧起始时刻，后面的时间同样按照设计的帧结构进行划分。这里提到的节点的入网同步有多种方法，由于节点的入网设计并不是本文的主要内容，因此之后章节的设计都是默认节点在已完成入网同步的基础上进行。

#### 2.1.4 业务时隙混合分配

业务时隙分配策略的优劣决定着 TDMA 协议对网络性能的影响，常见的时隙分

配方式有静态分配和动态分配。静态分配是指网络按照某种预定规则给在网节点分配固定数量的时隙，这种方式缺乏灵活性，容易出现时隙浪费的现象；动态分配是指节点按照自身需要的时隙数量进行动态预约，这种方式对时隙的利用率更高。IS-TDMA协议使用了一种业务时隙的“预分配+动态分配”的混合分配方式，时隙预分配在动态分配开启之前为节点与所有邻居节点之间快速分配一个可靠的通信时隙，这样可以有效地应对在动态预约到时隙前突然注入的业务导致的队列分组堆积现象。

## 2.2 时隙预分配

### 2.2.1 时隙快速分配

时隙预分配的主要目的是在节点完成入网但尚未开启时隙动态预约之前，为其与每个邻居节点之间分配一个单向发送时隙，该时隙一方面用于维持节点在预约到动态时隙之前的基本业务通信需求，另一方面则可为之后的动态预约和共存测试过程提供可靠的通信信道。从所分配时隙的复用性角度考虑，存在两种预分配方式：

#### (1) 直接分配

直接分配是指节点入网后即可从网络处获得无冲突的时隙，这种分配方式很简单，不需要考虑时隙冲突问题，但是用于分配的时隙数量较大，时隙的复用率较低。假设一个节点  $P$  的最大邻居节点数为  $N$ ，则  $P$  节点需要为邻居节点分配  $N$  个不同的发送时隙，记其集合为  $A_p$ ，假设这  $N$  个邻居节点为自身的  $N$  个邻居节点分配的时隙的总集合为  $A_T$ ，被除本节点外的  $N-1$  个邻居节点分配的接收时隙的总集合为  $A_R$ ，则对于任一时隙  $S_i \in A_p$ ，需要满足式 2-2。

$$S_i \notin A_T \cup A_R \quad (2-2)$$

如图 2.7 所示，节点 0 有  $N$  个邻居节点，每个邻居节点有除节点 0 外的  $N-1$  个邻居节点，且已与这  $N-1$  个邻居节点完成了时隙预分配，考虑极端情况，即图中 1~ $N$  号节点用于为其邻居节点分配的时隙号相同，例如 1 节点和 2 节点与其  $N-1$  个邻居节点之间的预分配时隙均为 1, 2, 3, ...,  $2(N-1)$ ，那么这  $N$  个邻居节点最少已占用  $2(N-1)$  个时隙，此时 0 节点与这  $N$  个邻居节点之间分配的  $2N$  个时隙不能与前  $2(N-1)$  个有重合。因此在最理想的情况下，一个节点要完成与所有邻居节点的时隙预分配需要至少  $4N-2$  个时隙，当  $N$  的值较大时，例如  $N$  等于 20，则需要 78 个时隙用于预分配，而本文所使用的一个复帧中业务时隙总数只有 80，这显然是不合理的。因此，直接分配的方式并不适用于 IS-TDMA 的时隙预分配。

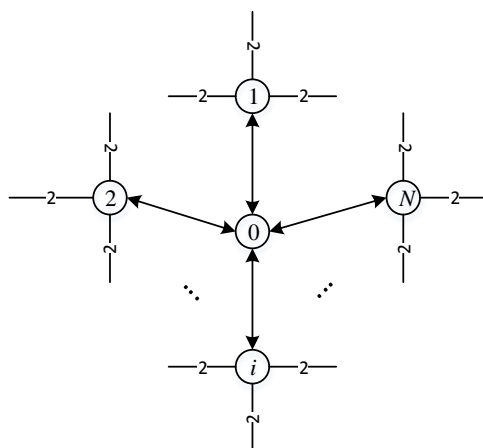


图 2.7 直接分配时隙数量示意图

## (2) 间接分配

相比直接分配，间接分配需要辅以节点的交互，但由于预分配过程追求快速分配的特点，因而交互的过程不能过于复杂。为此，我们设计了一种基于 BS 时隙占用位置的顺序分配方法，其大体思路为：占用 BS 时隙靠后的节点根据接收的前面已完成预分配的节点广播的 HELLO 包进行时隙分配的决策，HELLO 包是一种在 BS 时隙中广播的用于交互同步、时隙状态等控制信息的包格式，注意节点只为已完成预分配且在其广播的 HELLO 包中包含预分配结果的那些邻居节点分配时隙（单向时隙）。以图 2.8 所示的拓扑为例，具体实现步骤如下：

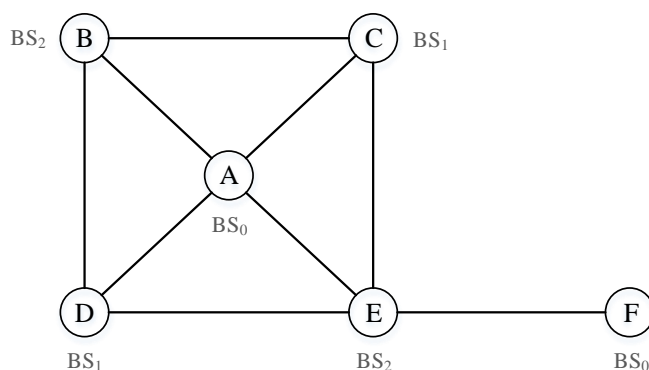


图 2.8 间接分配时隙拓扑示意图

① 占用  $BS_0$  的节点（简称为  $BS_0$  节点）首先发起时隙预分配，此时网络中还没有节点开启预分配，所以节点可以直接分配时隙，无需与其他节点进行交互；但是由于 BS 采用一跳范围不冲突的复用准则，可能会出现占用同一 BS 的多个节点（如图中的 A 和 F）给同一个节点分配时隙的情况，针对此问题，可以采取一种角度映射的策略，即将本节点所有可分配时隙的集合  $S_{avail}$  映射到一个圆上，根据邻居节点所在角度即可得到所分配的时隙，有效地降低了多个节点同时给一个节点分配时隙的冲突概率。

② 当  $BS_0$  节点完成时隙预分配后，其分配的结果要放在 HELLO 包中广播出去，此时周围占用其他 BS 的邻居节点可以接收 HELLO 包并获取  $BS_0$  节点的预分配结果。以  $BS_1$  节点（节点 C 和 D）为例，可以从 A 节点广播的预分配信息中获取分配给本节点的时隙集合  $S_{self}$  和分配给  $BS_1$  节点准备要分配时隙的公共邻居节点的时隙集合  $S_{com}$ ，于是由式 2-3 计算可得剩余可分配时隙集合  $S_{avail}$  为：

$$S_{avail} = S_{total} - S_{used} - S_{self} - S_{com} \quad (2-3)$$

上式中， $S_{total}$  为一个复帧中的所有业务时隙集合， $S_{used}$  为  $BS_1$  节点在接收到  $BS_0$  节点的广播信息之前已分配用于本节点发送或接收的时隙集合。 $BS_1$  节点为广播了包含预分配结果的 HELLO 包的邻居节点进行时隙预分配，分配的时隙从  $S_{avail}$  中选取，分配方法与步骤①中的角度映射法相同。

③ 同样地，当  $BS_2$  节点（节点 B 和 E）进行时隙分配时，节点已经获取了周围占用  $BS_0$  和  $BS_1$  的节点的时隙分配结果，在此基础上按照步骤②中的方法计算  $S_{avail}$ ，从而为那些已接收其广播包含预分配结果的 HELLO 包的邻居节点（即  $BS_0$  节点和  $BS_1$  节点）进行时隙预分配。

④ 占用后续 BS 的节点则依此类推。

节点间通过这样的简单交互过程，不但可以较快速度地完成时隙的预分配（通过实际测试发现，一般的拓扑结构在四个超帧之内即可完成），而且可以有效地降低时隙分配过程中的时隙冲突问题。

### 2.2.2 时隙冲突分解

在上一小节介绍的时隙预分配方法中，虽然采用了角度映射的方法来降低冲突的概率，但两个节点同时给公共邻居节点分配时隙时发生冲突的可能还是存在的；此外，在实际通信环境中，由于天线辐射角度的精确度及信道的复杂性等原因，相近的同时通信的定向链路之间还是存在相互干扰的可能。针对以上两种可能的时隙冲突，存在以下两种解决方法。

#### （1）时隙分配导致的冲突

针对两个节点同时给另一节点分配时隙产生的冲突，冲突节点自身通过扫描时隙状态表可以检测出多个邻居节点分配相同时隙的情况，从而可以通过在 BS 时隙中广播冲突时隙信息来分解冲突。

#### （2）信道环境导致的冲突

针对实际通信环境引起的链路冲突，我们通过监测一段时间内链路接收节点的收包情况来判断是否存在时隙冲突，比如一段时间内接收节点在所有业务时隙上都能正



常接收到业务包，则判定为无时隙冲突，否则，将不能正常收包的时隙判定为冲突时隙，并同样通过 BS 时隙广播冲突信息的方式来分解时隙冲突。

## 2.3 分布式链路调度算法

### 2.3.1 调度启动条件

在完成入网和时隙预分配等一系列操作后，节点已经具备了与邻居节点间的低业务强度通信能力，当节点上业务强度逐渐增长，则需要通过链路调度来获得更多的可用时隙链路。链路调度包含时隙动态预约和动态释放两个部分，每个部分有着各自的判定指标和启动条件。

#### (1) 判定指标

对于动态预约，考虑到链路需求与节点上的业务强度有关，我们通过计算平均队列长度  $Q_{avg}$  来作为判定指标<sup>[25,26,27]</sup>。由于自组织网络的业务具有突发性，某些情况下节点的队列处于空状态，突然到来的业务将队列迅速填满，又很快地被取空，因此在计算  $Q_{avg}$  时为了过滤掉队列长度的短期突发性变化，而体现出长期的平均变化，通过式 2-4 来统计  $Q_{avg}$ ：

$$Q_{avg} = (1 - w_q) * Q_{avg} + w_q * q \quad (2-4)$$

上式采用了类似于低通滤波器的原理，定义  $w_q$  为反应因子， $q$  表示当前队列的实时长度。 $w_q$  决定了对  $q$  的反应灵敏度，如果  $w_q$  过大，则无法过滤突发的变化，如果  $w_q$  过小，则对队列的实时长度反应太慢。因此，选择合适的权值  $w_q$  可以最大程度地过滤掉短期的抖动，反映队列长度的长期变化。在本文的设计中，我们设定  $w_q$  的值为 0.05。

对于动态释放，采用平均空闲时隙数  $IS_{avg}$  作为判定指标。 $IS_{avg}$  的统计方法与原理和上面所述的  $Q_{avg}$  相似，计算公式如式 2-5 所示：

$$IS_{avg} = (1 - w_s) * IS_{avg} + w_s * s \quad (2-5)$$

式中  $w_s$  为反应因子， $s$  表示节点当前的空闲时隙数量。同样地，对  $w_s$  值大小的选择时分重要，本文中设定  $w_s$  的值为 0.5。

#### (2) 启动条件

对于动态预约，在每个超帧的开始时刻计算一次  $Q_{avg}$ ，记该时刻之前的平均队列

长度为 $Q_{past}$ ，当满足下式的条件时，即启动预约过程。

$$\frac{Q_{avg} - Q_{past}}{m} * \beta \geq 1 \quad (2-6)$$

式 2-6 中， $m$  等于一个超帧中的复帧数量， $\beta$  是预约弹性因子，用于适当增加预约的时隙数量，以确保即使有些时隙预约失败也能尽量达到预约的需求数量，本文中设定  $\beta$  的值为 1.5。上面的式子表达的含义是当过去一个超帧中节点队列的分组堆积长度超过了一个时隙在一个超帧中可以发送的分组数量时，则开启动态预约时隙。

对于动态释放，在每个复帧的开始时刻计算一次  $IS_{avg}$ ，但为了避免频繁地释放时隙导致出现时隙不停地预约-释放的震荡现象，一般以 1~2 个超帧为单位进行启动条件的判断。若  $IS_{avg} > 1$ ，则开启动态释放，释放的时隙数量为  $IS_{avg} - 1$ ；当  $IS_{avg} = 1$  时不开启时隙释放，目的是为了保留预分配时隙以维持节点间的基本通信能力。

### 2.3.2 调度时隙

通过上面的分析可知，链路调度的发起者是节点自身，因此需要为每个节点预留用于链路调度的时隙，该时隙主要用于节点向邻居节点发送链路调度分组。由于实际通信过程中动态预约和释放的操作不会太过频繁，因此节点在一个超帧存在一个时隙即可满足所需。观察 2.1.2 节的时帧结构可以发现，BS 时隙是最符合要求的，但 BS 时隙已被分配用于周期广播同步和邻居信息，为了降低功能模块之间的耦合度，需要另外寻找一个时隙。从实现简单的角度考虑，我们将每个复帧的第一个业务时隙  $TS_0$  与 BS 时隙进行绑定，分配给节点专门用于动态预约和释放时隙，称之为调度时隙(SS, Schedule Slot)，同 BS 时隙一样按照 0,1,2,..., $m-1$  的顺序进行编号，即占用  $BS_i$  的节点也会占用  $SS_i$ 。

### 2.3.3 业务时隙状态

时隙的动态预约和释放都会改变节点上业务时隙的状态，而要能正常实现动态预约和释放，记录节点的业务时隙状态是必不可少的步骤。在每个节点的本地都存储了一张如表 2.1 所示的时隙状态表。

表 2.1 业务时隙状态

id	state	neighbor_id	point_angle	lock
----	-------	-------------	-------------	------

表中 *id* 字段表示业务时隙号，*state* 字段表示节点在该业务时隙上的射频状态，有发送（+1）、空闲（0）和接收（-1）三种状态，*neighbor\_id* 字段表示在该时隙中与本节点建立链路的邻居节点号，*point\_angle* 字段表示在该业务时隙节点的定向天线所指向的角度，也即该时隙链路对端的邻居节点所在的角度，若 *state* = 0，则记 *point\_angle* = 360°。

### 2.3.4 调度分组格式

调度分组用于在节点之间传递链路调度所需的控制信息，只在调度时隙上被发送或接收。将在调度时隙上传递的调度分组归类为控制帧，因此节点采用全向天线进行调度分组的接收。调度分组的具体格式如图 2.9 所示。

TYPE	STATE	SRC_ID	DEST_ID
SLOT_TABLE			

图 2.9 调度分组格式示意图

下面是对各个字段的具体解释：

**TYPE:** 指示该包的类型是一个链路调度分组；

**STATE:** 指示该次调度的调度状态，有整数 0~4 五种可能的值，分别对应预约请求、预约应答、预约确认、释放请求和释放应答这五种状态；

**SRC\_ID:** 指示该调度分组的发送节点编号；

**DEST\_ID:** 指示该调度分组的接收节点编号；

**SLOT\_TABLE:** 这不是单个字段，而是多个字段组成的字段域，存放的是该次调度所要预约/释放的多个时隙的编号，存放的个数可以根据需要进行修改。

### 2.3.5 干扰时隙判定

节点发起时隙动态预约时，只进行单向时隙的预约，即只预约向邻居节点发送的时隙，接收时隙由邻居节点预约的发送时隙决定。在选择要预约的时隙时，除了需要判断时隙是否处于空闲状态，还要判断时隙是否是干扰时隙，干扰时隙包括发送干扰时隙和接收干扰时隙。发送干扰时隙是指在选定的发送时隙上发送数据会干扰在相同时隙上接收数据的邻居节点的正常接收，如图 2.10（a）所示，CD 是原先存在的占用时隙  $TS_i$  的链路，若节点 A 向 B 预约  $TS_i$  作为发送时隙，则在该时隙中 AB 链路将干扰 CD 链路的正常接收；接收干扰时隙是指在选定的接收时隙上接收数据会被相同时隙上发送数据的邻居节点的发送所干扰，如图 2.10（b）所示，节点 B 如果同意在  $TS_i$

时隙接收 A 的发送，则同样在  $TS_i$  时隙发送的 C 节点将干扰 B 的正常接收。

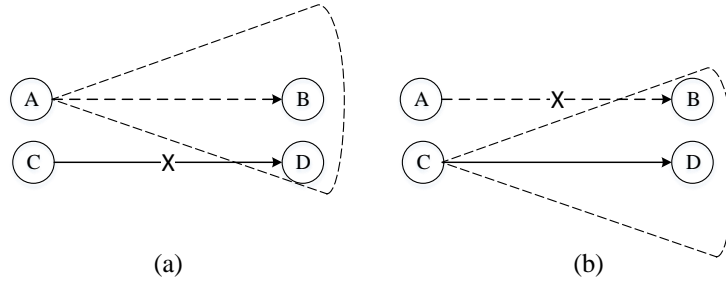


图 2.10 定向链路干扰示意图

通过以上分析，虽然定向天线因其波束窄的特点而容易实现空分复用，但是窄波束还是具备一定的覆盖范围，导致了图 2.10 中的链路干扰问题。因此，为了确保通过时隙动态预约建立的链路之间不存在干扰，最大程度地发挥定向天线的优势，首要的问题是建立一种有效的干扰时隙判定准则，以便在选择预约时隙时即可将干扰时隙进行排除。

针对上述问题，现有的一些解决方法<sup>[24]</sup>通过记录节点所有波束方向上的节点和这些节点的时隙状态，并判断在要预约时隙的发送/接收节点的发送/接收波束上的邻居节点的时隙状态来排除可能干扰的时隙。采用波束进行测量的方法虽然能排除大多数的干扰时隙，但对干扰时隙的抑制仍然不够精确。在我们的设计中，由于物理层具备记录邻居节点所在角度的能力，且节点将最大发送/接收增益方向对准邻居节点的角度进行发送和接收，因此提出一种基于角度精确计算的干扰时隙判定准则。

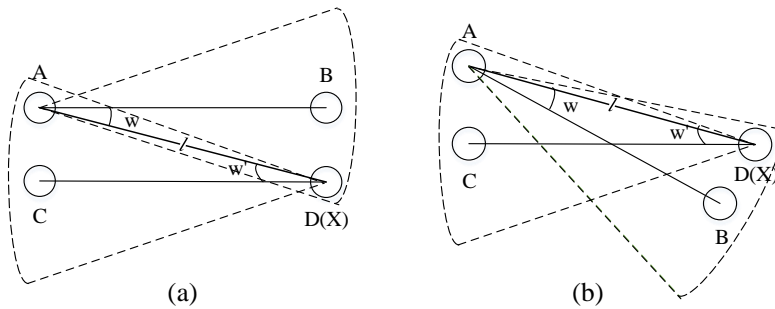


图 2.11 发送干扰时隙判定示意图

### (1) 发送干扰时隙判定

假设节点 A 从所有空闲的业务时隙中选择  $TS_i$  时隙作为要预约向节点 B 发送的时隙，记 A 的邻居节点集合为  $\{N_A\}$ ，则对于任一节点  $X \in \{N_A\} - B$ ，若同时满足下面的三个条件：

- ① 节点 X 在  $TS_i$  时隙上处于接收状态；
- ② 节点 X 在节点 A 的天线发送主瓣覆盖范围内；

③ 节点 A 在节点 X 的天线接收主瓣覆盖范围内。

则可以判定时隙  $TS_i$  是一个发送干扰时隙。如图 2.11 (a) 和 (b) 所示的两种情形，节点 A 向 B 预约时隙  $TS_i$ ，两种拓扑下的节点  $D \in \{N_A\} - B$  均满足以上三个条件，因此要预约的时隙被判定为干扰时隙。

对于条件①，通过读取节点本地存储的时隙状态表即可完成判断；而对于条件②和③，虽然从图 2.11 中可以很直观看出判定的大致方法，但是为了进行计算机的仿真计算，需要将其进行精确地量化。为此，我们将条件②和③量化成了对图中所示的  $w$  和  $w'$  两个标量的大小的判断。将节点 A 和 X 之间的连线标记为  $l$ ，则  $w$  表示发送节点 A 的最大发送增益方向与  $l$  形成的夹角， $w'$  表示接收节点 X 的最大接收增益方向与  $l$  形成的夹角。假设定向天线的主瓣宽度为  $\Omega$ ，对照图 2.11，可得  $w$  和  $w'$  的计算方法如式 2-7 所示。

$$\begin{aligned} w &= |D_{AD} - D_{AB}|, \\ w' &= |D_{AD} - (D_{DC} + 180) \% 360| \end{aligned} \quad (2-7)$$

在式 2-7 中， $D_{ij}$  变量表示节点  $i$  上记录的节点  $j$  所在角度。定义  $I_T$  为发送干扰时隙判定门限，从图 2.11 中可以看出  $I_T = \Omega/2$ 。当  $w < I_T$  时，则满足条件②，当  $w' < I_T$  时，则满足条件③。

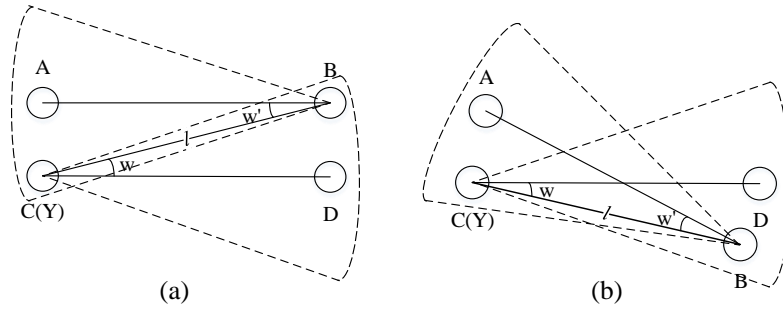


图 2.12 接收干扰时隙判定示意图

## (2) 接收干扰时隙判定

承接 (1) 中的假设，若时隙  $TS_i$  未被判定为干扰时隙，则节点 B 将接收到 A 要预约时隙  $TS_i$  的请求，此时，B 需要判定  $TS_i$  是否为接收干扰时隙。记 B 的邻居节点集合为  $\{N_B\}$ ，则对于任一节点  $Y \in \{N_B\} - A$ ，若同时满足下面的三个条件：

- ① 节点 Y 在  $TS_i$  时隙上处于发送状态；
- ② 节点 Y 在节点 B 的天线接收主瓣覆盖范围内；
- ③ 节点 B 在节点 Y 的天线发送主瓣覆盖范围内。

则可以判定时隙  $TS_i$  是一个接收干扰时隙。如图 2.12 (a) 和 (b) 所示的两种情形, 节点 B 判断时隙  $TS_i$  是否可被预约用于接收 A 的发送, 两种拓扑下的节点  $C \in \{N_B\} - A$  均满足以上三个条件, 因此要预约的时隙被判定为干扰时隙。

同样地, 将条件②和③量化成了对图中所示的  $w$  和  $w'$  两个标量的大小的判断。首先将节点 B 和 Y 之间的连线标记为  $l$ , 则  $w$  表示发送节点 Y 的最大发送增益方向与  $l$  形成的夹角,  $w'$  表示接收节点 B 的最大接收增益方向与  $l$  形成的夹角。根据图 2.12, 可得  $w$  和  $w'$  的计算方法如式 2-8 所示。

$$\begin{aligned} w &= |D_{CD} - (D_{BC} + 180) \% 360|, \\ w' &= |D_{BC} - D_{BA}| \end{aligned} \quad (2-8)$$

类比发送干扰时隙判定, 定义  $I_R$  为接收时隙干扰判定门限, 由图 2.12 可以看出  $I_R = \Omega/2$ 。当  $w' < I_R$  时, 则满足条件②, 当  $w < I_R$  时, 则满足条件③。

### 2.3.6 链路调度过程

IS-TDMA 协议的分布式链路调度包括时隙动态预约和动态释放两个过程。定义发起动态预约时隙的节点为预约发送节点 (Subscribe Transmitter, ST), 接收预约通知的节点为预约接收节点 (Subscribe Receiver, SR); 定义发起动态释放时隙的节点为释放发送节点 (Release Transmitter, RT), 接收释放通知的节点为释放接收节点 (Release Receiver, RR)。动态预约过程通过 ST 和 SR 节点之间的三次握手完成, 而动态释放过程通过 RT 和 RR 节点之间的两次握手完成。具体的调度步骤如下:

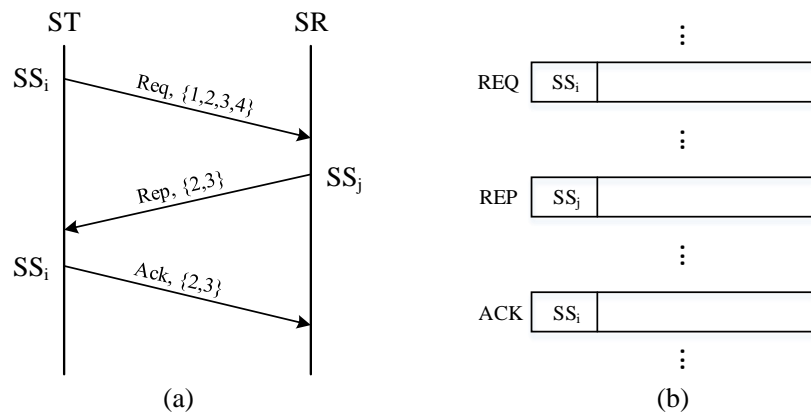


图 2.13 时隙动态预约过程示意图

#### (1) 动态预约

时隙的动态预约由预约请求 (SREQ)、预约应答 (SREP) 和预约确认 (SACK)

三个部分组成，对应了 ST 和 SR 节点之间的三次握手，如图 2.13 (a) 所示。节点通过时隙调度分组交互调度相关信息，包中携带有当前的预约阶段类型和要进行预约的时隙序列。ST 和 SR 节点的调度分组均在调度时隙中进行发送，假设 ST 和 SR 节点分别占用编号为  $i$  和  $j$  的 SS 时隙，则一次完整的动态预约过程的持续周期为一个超帧，如图 2.13 (b) 所示。

下面结合图 2.13 (a)，对预约过程的三次握手进行详细说明。

#### ① 预约请求

a. ST 节点占用的调度时隙  $SS_i$  到达，通过计算式 2-4 和式 2-6 判断启动时隙动态预约。

b. ST 节点从业务时隙中选择要预约数量的时隙，对于任一业务时隙  $TS_i$ ，若同时满足以下三个条件：1) 处于空闲状态；2) 不处于锁定状态；3) 通过了发送干扰时隙判定，则该时隙可以被预约，选定该时隙并对该时隙加锁。

c. ST 节点选择的时隙达到了预定数量，则将选择的时隙序列（假设为编号 1,2,3,4 的时隙）写入调度分组，并写入调度类型为预约请求（SREQ）。

d. ST 节点向 SR 节点发送调度分组。

#### ② 预约应答

a. SR 节点接收 ST 节点发送的预约请求分组（SREQ），提取包中携带的预约发送时隙序列。

b. SR 节点对提取的时隙进行一一检查，对于任一时隙，若同时满足以下三个条件：1) 处于空闲状态；2) 不处于锁定状态；3) 通过了接收干扰时隙判定，则判定该时隙通过检查，可被预约用作本节点的接收时隙，对该时隙加锁。

c. SR 节点将通过检查的时隙序列（假设为编号 3,4 的时隙）写入调度分组，并写入调度类型为预约应答（SREP）。

d. SR 节点向 ST 节点回复调度分组。

#### ③ 预约确认

a. ST 节点接收 SR 节点发送的预约应答分组（SREP），提取最终可预约的时隙序列，将未预约成功时隙进行解锁。

b. ST 节点修改最终可预约时隙的时隙状态为发送状态，指向节点为 SR 节点，指向角度为 SR 节点所在角度，并解除锁定状态。

c. ST 节点将最终可预约时隙序列写入调度分组，并写入调度类型为预约确认（SACK），向 SR 节点发送调度分组。

d. SR 节点接收预约确认分组（SACK），提取其中的预约时隙序列。

e. SR 节点修改提取时隙的时隙状态为接收状态，指向节点为 ST 节点，指向角度为 ST 节点所在角度，并解除锁定状态，预约过程完成。

## (2) 动态释放

时隙的动态释放由释放请求 (RREQ) 和释放确认 (RACK) 两个部分组成, 对应 RT 和 RR 节点之间的两次握手, 如图 2.14 所示。RT 和 RR 节点同样在调度时隙中通过调度分组进行交互, 但是当一个调度时隙中同时存在动态预约和动态释放两种操作时, 优先处理动态释放。显然, 一次完整的时隙动态释放过程在最多不超过一个超帧的时间内即可完成。

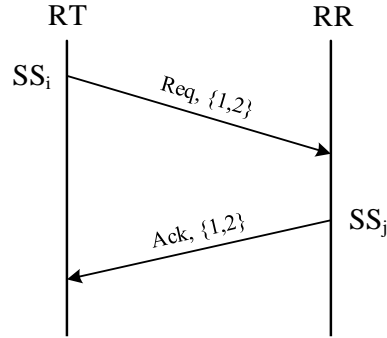


图 2.14 时隙动态释放过程示意图

下面结合图 2.14, 对时隙释放过程进行详细的说明。

### ① 释放请求

- a. RT 节点的调度时隙  $SS_i$  到达, 通过式 2-5 计算的值判断启动时隙动态释放。
- b. RT 节点从已预约的时隙中选择  $IS_{avg} - 1$  个时隙用于释放, 标记这些时隙将不能再用于发送业务分组。
- c. RT 节点将选择的要释放的时隙序列写入调度分组, 并修改调度类型为释放请求 (RREQ)。

- d. RT 节点向 RR 节点发送调度分组。

### ② 释放确认

- a. RR 节点接收 RT 节点发送的释放请求分组 (RREQ), 提取包中携带的要释放的时隙。
- b. RR 节点逐一修改这些释放时隙的时隙状态为空闲状态, 指向节点和指向角度均为空值。
- c. RR 节点将释放时隙序列写入调度分组, 并修改调度类型为释放确认 (RACK), 向 RT 节点发送调度分组。
- d. RT 节点接收 RR 节点发送的释放确认分组 (RACK), 确认 RR 节点端已完成释放, 提取其中的释放时隙。
- e. RT 节点逐一修改释放时隙的时隙状态为空闲状态, 指向节点和指向角度均为



空值，释放过程完成。

## 2.4 复杂度与开销分析

通过观察上述的调度过程可以发现，时隙动态释放相比时隙动态预约要简单很多，因此，我们重点对时隙动态预约的复杂度和开销进行分析。

### 2.4.1 复杂度分析

假设节点一次动态预约的最大时隙数量为  $E$ ，则为了寻找  $E$  个符合条件的时隙，最坏情况下可能要遍历所有的业务时隙，其复杂度为  $O(n)$ ；通过 2.3.5 节的分析可知，在对每一个时隙进行发送/接收干扰时隙判定时，需要对该时隙中所有邻居节点的时隙状态进行检查，也即要遍历邻居节点列表，我们定义的最大邻居节点数量为  $N$ ，因此遍历的复杂度为  $O(N)$ 。综合以上分析可得，动态预约一个业务时隙的时间复杂度为  $O(n*N)$ ，本文中  $E \leq 10$ ，因此时隙动态预约的时间复杂度为  $O(n*N)$ 。

### 2.4.2 开销分析

该调度算法的开销主要由存储开销和链路开销组成。

#### (1) 存储开销

在上述调度过程中，节点本地主要存储的数据有邻居节点列表和时隙状态表。在本文的设计中，单条邻居节点记录的大小不超过  $500B$ ，单条时隙状态信息记录的大小不超过  $100B$ ，假设最大邻居数量  $N$  等于 20，业务时隙总量  $n$  等于 100，则单个节点上总共需要存储的数据量不会超过  $20kB$ 。而现代网络节点的存储容量一般都以  $mB$  为计量单位，因此节点上的存储开销几乎可以忽略不计。

#### (2) 链路开销

为了方便调度过程的实现，我们划分了调度时隙专用于链路调度分组的发送与接收，而调度时隙实际占用的是编号为 0 的业务时隙，且为每个节点都分配了一个，因此我们可以定义链路开销为一个调度时隙的长度与复帧长度的比值，用  $\gamma$  表示。以实际的仿真为例，设定业务时隙的长度为  $0.455ms$ ， $BS$  时隙的长度为  $6.4ms$ ，一个复帧包含 80 个业务时隙，复帧长度为  $42.8ms$ ，则计算得  $\gamma$  约等于 1.063%，可见算法耗费的链路开销很小。



## 第三章 基于接收检测的链路共存测试过程设计

在很多的 TDMA 链路调度设计方案中, 时隙的动态预约都是通过发送节点与接收节点之间的交互完成, 对于可预约时隙的选择有各自的筛选规则, IS-TDMA 协议亦是如此。但是, 由于真实的无线网络环境的复杂性, 一些设计时无干扰的链路在实际通信时可能存在干扰, 这将直接导致网络的吞吐量下降。为了进一步抑制干扰, 本文为 IS-TDMA 协议设计了一种“织入”链路调度过程的链路共存测试方法, 实现链路调度完成即开启测试, 通过实际的链路传输来测试链路的质量, 并及时释放干扰链路。

### 3.1 算法的提出

#### 3.1.1 造成干扰的外部因素

通过使用我们在上一章所设计的分布式链路调度方法, 可以在算法层面避免大多数的链路干扰, 以上的算法设计均是基于理想的无线信道和定向天线。然而, 真实的网络环境中信道由于噪声和阻挡物等因素存在波动和不稳定性, 实际使用的定向天线的覆盖范围也不可能像设计时所划分的那样精确, 且不可避免地存在旁瓣干扰<sup>[28]</sup>的问题。显然, 这些客观存在的外部因素都有可能造成链路干扰, 而链路调度过程是无法发现并避免这些干扰的。

#### 3.1.2 共存测试的必要性

从网络中所有链路的角度来看, 虽然外部因素可能只对小部分的链路造成干扰, 但还是会导致这些链路上传输的业务分组丢失, 如果这些链路上有持续不断的业务分组需要传输, 那么这样的丢包现象是难以接受的。

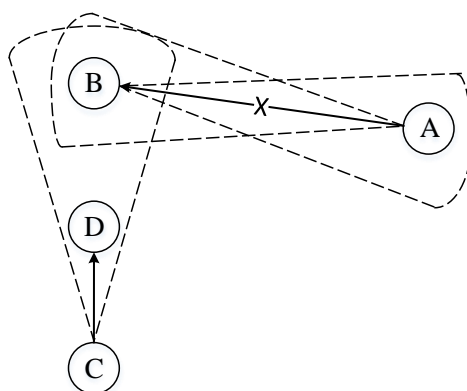


图 3.1 外部因素干扰示意图

图 3.1 所示的是网络中可能出现的一种链路情形，其中，CD 链路是一条占用时隙  $TS_i$  传输的无干扰链路，而 AB 链路正在预约时隙  $TS_i$ ，由 2.3.5 节分析的干扰时隙判定方法可知，时隙  $TS_i$  在节点 A 处顺利通过发送干扰时隙判定，在节点 B 处判断接收干扰时隙时，虽然计算得  $w$  等于 0，满足条件③，但显然  $w'$  是不满足条件②的，因而节点 B 未把时隙  $TS_i$  判定为接收干扰时隙，AB 链路成功预约到了时隙  $TS_i$ 。从算法设计的层面来看，在时隙  $TS_i$  中 AB 应该是一条无干扰的链路，但通过我们实际的仿真发现，CD 链路的发送是会干扰 AB 链路的接收的，而这种干扰在链路调度过程的设计中是无法被避免的。因此，在调度完成的链路正式开启业务传输之前，对其进行链路共存测试是十分必要的。

## 3.2 链路共存测试算法

### 3.2.1 链路干扰模型

在一次链路调度的过程中，定义当前网络中已经存在的可用于正常传输业务的链路为已有链路（Old Link, OL），通过本次调度添加的链路为新增链路（New Link, NL）。由于 NL 链路的发送可能干扰 OL 链路的接收，NL 链路的接收也可能被 OL 链路的发送所干扰，因此网络中存在两种可能的链路干扰模型，记前者为已有链路干扰（Old Link Interference, OLI），后者为新增链路干扰（New Link Interference, NLI）。图 3.2（a）和（b）分别对应了这两种链路干扰模型。

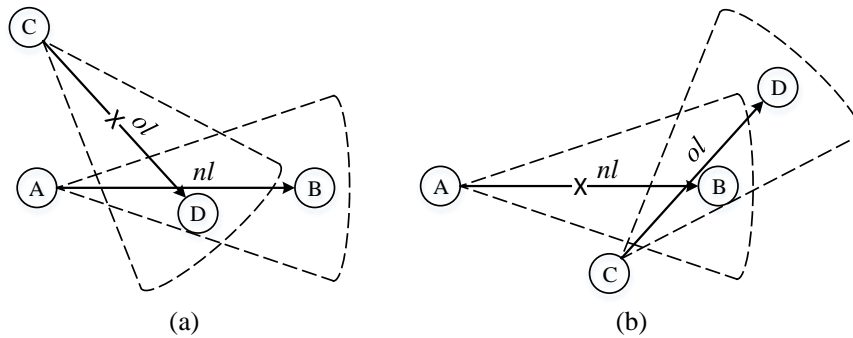


图 3.2 链路干扰模型示意图

在图 3.2 中，CD 链路是一条 OL 链路，AB 是一条与 CD 占用相同业务时隙的 NL 链路。在 OLI 干扰模型中，节点 A 计算的  $w$  和  $w'$  的值不符合发送干扰时隙的条件，未将预约时隙判定为干扰发送时隙；在 NLI 干扰模型中，节点 B 计算的  $w$  和  $w'$  的值不符合接收干扰时隙的条件，未将预约时隙判定为接收干扰时隙，这使得在整个调度

的过程中都未能检测出这两种链路干扰，继而在实际的通信中导致 OL 和 NL 链路的失效。

通过以上对链路干扰模型的分析可知，实现链路共存测试的关键在于找出 NL 链路周围所有可能会导致 OLI 和 NLI 干扰的 OL 链路，并在 NL 链路调度完成之后，让 NL 链路和这些选定的 OL 链路协商在同一个或多个测试时隙中一同发送测试分组，这里所指的测试时隙应该具体到一个超帧中的某一个复帧上的某一个或多个被预约的时隙，然后通过各条链路上测试分组的实际接收情况来判断是否存在链路干扰。若判断存在干扰，则启动释放 NL 链路，否则，将 NL 链路标记为 OL 链路。

### 3.2.2 共存测试周期

在一条 NL 链路调度完成之后到通过链路共存测试之前的这段时间内，NL 链路上是不能进行业务传输的，我们将这段时间定义为 NL 链路的共存测试周期，记为  $D_{IT}$ 。共存测试过程直接决定了时隙预约的成功与否，若测试所耗费的时间  $D_{IT}$  过大，将对网络造成以下影响：

(1) 假设 ST 节点向 SR 节点一次预约多个时隙，那么在整个  $D_{IT}$  时间内，这些完成调度未测试的时隙由于尚不确定是否存在链路干扰，因此都将处于空闲状态，这无疑造成了时隙资源的浪费；

(2) 若这些时隙中有多个干扰时隙，但因为测试周期过长而迟迟没有释放，而这些干扰时隙又减少了节点上的剩余可预约时隙，这对于有时隙预约需求却又找不到可预约时隙的节点来说，会造成节点队列中业务分组的堆积现象，从而降低网络的吞吐量和时延性能。

可见，共存测试周期是衡量链路共存测试过程优劣的一个重要指标，设计一个具有较小  $D_{IT}$  值的共存测试算法对于提高网络的性能而言很有必要。考虑链路共存测试中选择参与测试的 OL 链路和通知这些链路参与测试等过程都需要节点之间进行交互，且这些交互都与 ST 和 SR 节点有关，而我们设计的链路调度的三步握手过程正是在 ST 和 SR 节点之间进行，因此可以利用链路调度的三步握手，将链路共存测试过程巧妙地织入其中，如图 3.3 所示。这样做的好处是测试链路的选择和通知操作不再耗费额外的时间，在链路调度完成之后的下一个复帧就直接对其进行共存测试，争取以最少的测试开销来换取 NL 链路的可靠性。

从图 3.3 中可以发现，虽然整个测试过程将持续超过一个超帧的时间，但当剔除与调度过程的重合部分时，实际用于测试的时间仅为一个复帧，定义该复帧为测试复帧，需要在测试复帧中进行测试的时隙为测试时隙。这意味着完成预约的时隙仅在测试复帧中处于空闲，这样的测试速度和时隙浪费对于使用一整个超帧完成时隙预约的链路来说是完全可以接受的。

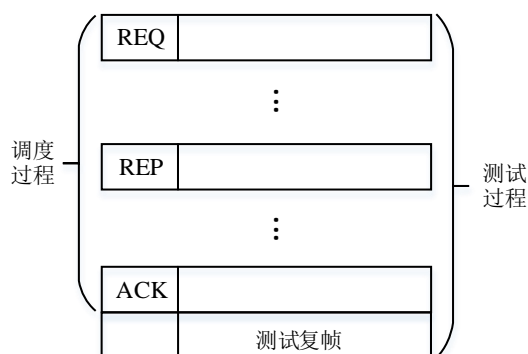


图 3.3 链路共存测试周期示意图

### 3.2.3 测试链路选择

从前面的分析可知，网络环境的复杂性和定向天线的非理想化等外部因素使得链路干扰在调度阶段具有不确定性。这种不确定性使得预约得到的 NL 链路在未经过测试的情况下不能开启业务传输，因为无法判断该链路可能和哪些 OL 链路之间存在相互的干扰。而对 NL 链路进行测试的关键第一步就是要找出所有这些潜在的可能干扰或被干扰的 OL 链路，针对两种不同的链路干扰类型，需要设计两种不同的测试链路选择方法。

#### (1) 已有链路干扰

已有链路干扰（OLI），即已经存在的 OL 链路的正常接收被 NL 链路的发送所干扰，干扰模型如图 3.2（a）所示。对于 NL 链路 AB 而言，即需要确定出所有可能被干扰的 OL 链路 CD，有如下两种可用的确定方法：

##### ① 门限抑制法

在 2.3.5 节中，我们通过对定向天线覆盖范围的分析，将干扰时隙的判定门限设置成了定向天线主瓣宽度的一半。这是严格基于理想化定向天线的窄波束来设计的，但实际使用的定向天线的主瓣还无法做到设计所要求的覆盖精度，因此干扰时隙的判定过程中对符合条件的邻居节点的筛选条件（即条件②和③）变得过于苛刻，导致一些可能被干扰的接收节点被限制在了条件之外。由此，在选择哪些链路参与测试时，我们想到可以增大干扰发送时隙的判定门限  $I_T$ ，比如令  $I_T = \Omega$  或者  $I_T = 2\Omega$ ，具体的大小可以根据测试的效果进行动态调整，然后将修改后的门限值作为参与测试链路的选择门限（注意：干扰时隙的判定依旧使用原  $I_T$  值进行），以图 3.4 为例进行说明。

在图 3.4 中，AB 链路是一条新建链路，CD 链路是一条已有链路，两条链路占用相同的时隙。在节点 A 发起时隙预约时，对接收节点 D 进行了一次干扰发送时隙判定，但由于  $w'$  不满足小于  $I_T$  的条件，节点 A 并未将时隙判定为发送干扰时隙。当开

启测试过程并进行测试链路选择时，节点 A 再次对  $w$  和  $w'$  进行判断，此时使用的判断依据是增大后的  $I_T$ ，则  $w$  和  $w'$  均小于  $I_T$ ，因此节点 A 将选择 CD 作为参与测试的链路。对于 AB 链路邻近的其他链路，采用同样的方法进行筛选。

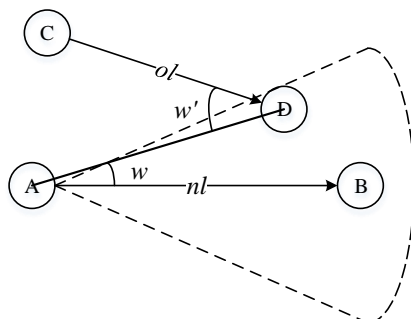


图 3.4 门限抑制法原理示意图

该方法的原理实际上是选择一部分干扰时隙判定时遗漏的链路，对实际定向天线的缺陷进行了一定程度的弥补。它的优点是实现简单、筛选过程短，但其缺点则更为突出：1) 门限值的大小很难确定，如果太小，则起不到筛选 OL 链路的效果，如果太大，则会有太多的链路符合条件，增加测试的开销；2) 没有从真实网络环境的复杂性角度考虑，在门限值约束范围内的链路无干扰可能在实际通信时又出现干扰。

## ② 全向接收检测法

门限抑制法最大的不足是没有基于实际通信的情况进行测试链路的选择，这也导致了即使选择再多的 OL 链路参与测试也不能确保测试过的链路一定无干扰，这样的测试也就不具备多大的价值。为了兼顾定向天线的主瓣覆盖问题和真实网络环境的复杂性，我们设计了一种基于全向接收检测的参与测试链路选择方法。

该方法利用了链路调度过程的三步握手，以图 3.4 为例进行说明。当节点 A 向 B 发送调度分组时，A 的邻居节点在该调度时隙处于全向接收状态，而调度分组 SACK 中包含了节点 A 要预约的时隙，假设该时隙为  $T$ （为了叙述方便这里只讨论单个时隙，多个时隙的过程与单个时隙类似），因此可以根据接收节点 A 所发送调度分组的情况来确定接收节点所在的链路是否需要参与测试。对于任一节点  $D \in \{N_A\} - B$ ，若同时满足以下两个条件：1) 成功接收到 A 发送的调度分组；2) 在时隙  $T$  上处于接收状态，则节点 D 在时隙  $T$  上的链路 CD 将被选为参与测试链路。

该方法的设计主要基于以下假设：如果一个节点可以定向接收到另一个节点定向发送的包，那么它也一定可以全向接收到该节点定向发送的包。显然，该假设是成立的。全向接收检测法实际上是用可能被干扰节点的全向接收代替定向接收进行判断，这样做虽然会不可避免地增加一些不需要参与测试的链路，但是它基于实际通信接收进行选择的方法有效地应对了门限抑制法没有考虑的复杂网络环境问题，这也是本文



最终选择使用全向接收法的最主要原因。

## (2) 新建链路干扰

新建链路干扰 (NLI)，即刚刚预约完成的 NL 链路的正常接收被 OL 链路的发送所干扰，干扰模型如图 3.2 (b) 所示。对于 NL 链路 AB 而言，即需要确定出所有可能会干扰节点 B 的接收的 OL 链路 CD。这里同样有门限抑制和全向接收检测两种方法，门限抑制法的设计与上面分析的已有链路干扰基本没有差别，此处不再赘述。下面将重点对全向接收检测法的设计进行介绍。

全向接收检测的核心思想是：让可能被干扰的节点在链路调度的过程中全向接收可能的干扰链路定向发送的控制帧。在 NLI 干扰模型中，可能被干扰的接收节点是 NL 链路的接收节点 B，可能的干扰链路则为邻近的 OL 链路，只要让节点 B 全向接收这些 OL 链路发送的控制帧即可达到目的。但是，在节点 A 和 B 的链路调度过程中 OL 链路的发送节点除了广播 HELLO 包外不会发送任何控制信息，而全向发送的 HELLO 包显然不能满足需求。考虑到在调度过程中 OL 链路只有在业务时隙上进行定向传输，而每个节点都配备有一部全向天线，我们调整节点 B 在接收到 A 的预约请求 SREQ 之后到向 A 发送预约应答 SREP 之前的预约业务时隙上用全向天线进行接收，并设置 OL 链路在没有业务分组发送时发送空分组，确保 OL 链路始终繁忙以便于节点 B 的全向接收检测。假设 A 向 B 预约业务时隙  $T$ ，则该方法的实现如图 3.5 所示。

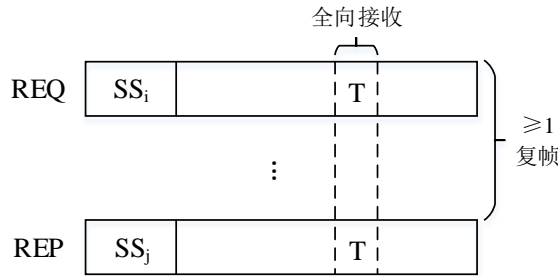


图 3.5 NLI 全向接收检测法原理示意图

在图 3.5 中，节点 B 在  $SS_i$  时隙接收到调度分组 SREQ，在  $SS_j$  时隙到来之前的所有复帧的  $T$  时隙上全向接收，即使  $SS_i$  和  $SS_j$  是相邻时隙，也能保证节点 B 至少全向接收一个复帧。结合图 3.2 (b)，对于任一节点  $C \in \{N_B\} - A$ ，若同时满足以下两个条件：1) 节点 B 成功接收其发送的业务分组；2) 在  $T$  时隙处于发送状态，则 C 在时隙  $T$  上的链路 CD 将被选为参与测试链路。

### 3.2.4 测试链路通知

在完成参与测试 OL 链路的选择之后，另一个关键问题是如何通知这些链路的发



送节点在测试时隙中参与测试，即通过哪个节点来发送通知分组，记为 *INFO*。由于对两种干扰的测试链路的选择发起节点不同，因此对应的通知发送节点也不同。

### (1) 已有链路干扰

在图 3.2 (a) 中，AB 链路确定 CD 链路需要参与共存测试后，需要通知节点 C 在测试复帧的测试时隙中发送测试分组，但是节点 A 和 B 都不一定是节点 C 的邻居节点，而 D 是真正确定 CD 链路要参与测试的节点且是 C 的邻居节点，因此选择节点 D 来发送 *INFO*。节点 D 占用 DC 链路上的一个业务时隙向 C 发送 *INFO*，在 2.2 节中讨论的时隙预分配方案保证了节点 D 和 C 之间始终存在至少一个可靠的业务时隙。

### (2) 新建链路干扰

在图 3.2 (b) 中，由于参与测试的链路 CD 是由节点 B 进行选择的，且节点 B 一定是节点 C 的邻居节点，因此由 B 向 C 发送 *INFO*。考虑节点 B 在该次调度中需要发送调度分组 *SREP*，且邻居节点在调度时隙中全向接收，可以将 *INFO* 放在调度时隙中进行发送，从而减少对业务时隙的占用。

## 3.2.5 共存测试过程

共存测试过程包含对预约时隙的干扰检测过程和对测试未通过的干扰时隙的释放过程两部分。为了叙述方便，结合图 3.6 以单个时隙 *T* 上的测试过程为例进行说明。

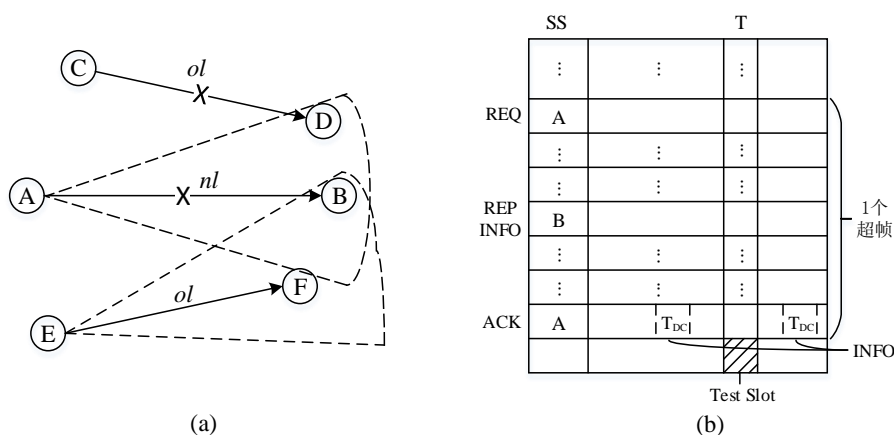


图 3.6 链路共存测试过程示意图

图 3.6 (a) 将两种类型的干扰模型放在一起表示，CD 链路是被 AB 链路干扰的 OL 链路，EF 链路是干扰 AB 链路的 OL 链路，注意 AB 链路周围可能存在多条类似 CD 和 EF 的链路，图中的 CD 和 EF 可以理解为这两种链路的统称；图 3.6 (b) 是调度过程交织测试过程的帧结构图。

### (1) 链路共存测试

- a. 节点 A 的调度时隙  $SS_A$  到达, A 向 B 发送时隙预约请求调度分组 SREQ, 其中携带了 A 向 B 预约的发送时隙  $T$ ;
- b. 节点 B 接收到 A 发送的调度分组 SREQ, 对其中要预约的时隙  $T$  进行接收干扰时隙判定 (假设时隙  $T$  可以通过判定), 并设置本节点在从当前复帧开始到调度时隙  $SS_B$  所在复帧之间所有的  $T$  时隙上使用全向天线进行接收;
- c. 节点 D 接收到 A 发送的调度分组 SREQ, 提取预约时隙  $T$ , 通过查看本地存储的时隙状态表判断本节点在时隙  $T$  上是否处于接收状态, 若是, 则等待接收 A 节点发送的调度分组 SACK (确定时隙  $T$  是否是最终要预约的时隙, 节点 B 处可能通不过干扰时隙判定), 否则, 可以确定本节点不会被干扰;
- d. 节点 B 的全向天线在时隙  $T$  中成功接收到节点 E 向 F 发送的业务分组, 将节点 E 和时隙  $T$  添加到 NLI 干扰测试通知表  $n\_info\_table$  中;
- e. 节点 B 的调度时隙  $SS_B$  到达, B 向 A 发送时隙预约应答调度分组 SREP, 其中携带了最终可以预约的时隙  $T$ ; 在 SREP 发送结束后, 节点 B 检查测试通知表  $n\_info\_table$  是否非空, 若是, 则向其中记录的节点逐一发送测试通知  $INFO$ , 此处即向节点 E 发送  $INFO$ , 其中包含测试时隙  $T$  和该时隙具体所在的位置 (下一个到来的  $SS_A$  时隙所在复帧的下一复帧);
- f. 节点 A 接收到 B 发送的调度分组 SREP, 确定预约时隙  $T$ , 设置在测试时隙向 B 发送测试分组的事件中断, 并进行时隙状态修改的操作;
- g. 节点 E 接收到 B 发送的测试通知, 提取时隙号  $T$  和时隙位置, 设置在测试时隙  $T$  中向 F 发送测试分组的事件中断;
- h. 节点 A 的调度时隙  $SS_A$  再次到达, A 向 B 发送时隙预约确认调度分组 SACK, 其中携带了最终要预约的发送时隙  $T$ ;
- i. 节点 B 接收到 A 发送的调度分组 SACK, 设置在测试时隙等待接收 A 发送测试分组的事件中断, 并修改本地的时隙状态表;
- j. 节点 D 接收到 A 发送的调度分组 SACK, 确定了 A 最终预约时隙  $T$ , 则将节点 C 和时隙  $T$  添加到 OLI 干扰测试通知表  $o\_info\_table$  中, 并设置在测试时隙等待接收 C 发送测试分组的事件中断;
- k. 节点 D 向 C 发送的业务时隙  $T_{DC}$  到达, D 检查测试通知表  $o\_info\_table$  是否非空, 若是, 则向其中记录的节点逐一发送测试通知  $INFO$ , 此处即向节点 C 发送  $INFO$ , 其中包含测试时隙  $T$  和该时隙具体所在的位置 (即下一复帧);
- l. 节点 C 接收到 D 发送的测试通知, 提取时隙号  $T$  和时隙位置, 设置在测试时隙  $T$  中向 D 发送测试分组的事件中断;
- m. 测试时隙  $T$  到达, 节点 A 向 B、C 向 D、E 向 F 发送测试分组;
- n. 节点 B 和 D 检查是否成功接收到测试分组, 若两个节点中的任何一个未能成

功接收，则判定时隙  $T$  存在链路干扰，AB 链路不能建立；否则，将该 NL 链路标记为 OL 链路，测试完成。

随着链路共存测试过程的结束，一条 NL 链路最终是否是无干扰的链路也得以呈现，若链路上无干扰，则从下一复帧开始即可利用该链路进行业务传输，且该链路将成为一条 OL 链路。通过以上分析并结合图 3.6 可以看出，链路共存测试在完成链路调度的下一复帧就能完成，可见所设计测试过程的高效性。

## （2）干扰链路释放

如图 3.7 所示，承接链路共存测试的结束时刻，干扰链路释放过程如下：

- a. 若节点 B 或 D 中任一节点未接收到测试分组，则设置在下一复帧的预分配时隙  $T_{BA}$  或  $T_{DC}$  中向节点 A 发送时隙释放通知  $rls\_info\_1$  的事件中断；
- b.  $T_{BA}$  或  $T_{DA}$  时隙到达，节点 B 或 D 向 A 发送释放时隙  $T$  的通知；
- c. 节点 A 接收到 B 或 D 节点发送的时隙释放通知，修改本地的时隙状态表，并设置在下一复帧的预分配时隙  $T_{AB}$  中向节点 B 发送时隙释放通知  $rls\_info\_2$  的事件中断；
- d.  $T_{AB}$  时隙到达，节点 A 向 B 发送释放时隙  $T$  的通知；
- e. 节点 B 接收到 A 发送的时隙释放通知，修改本地的时隙状态表，干扰时隙释放结束。

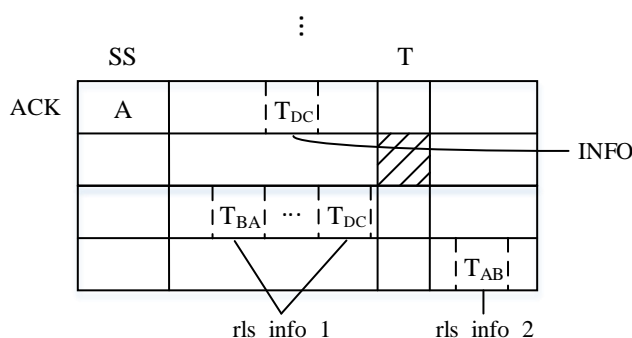


图 3.7 干扰时隙释放过程示意图

通过上述分析，如果在完成链路共存测试之后，还要考虑干扰链路的释放过程的话，整个测试的周期需要在原来的基础上再增加两个复帧，即一个超帧加上三个复帧的时间，但因为测试过程充分利用了链路调度过程的三次握手来进行前期的测试链路的选择和通知，单独被用于进行共存测试和释放的时间至多为三个复帧。

## 3.3 算法复杂度与开销分析

由 3.2.5 节的完整共存测试过程可知，测试过程是“织入”在调度过程中的，两

个部分在运行时是不可分割的，但我们在 2.4 节中已经对调度过程的复杂度和开销进行了分析，此处只对与链路共存测试相关操作的复杂度和开销进行分析（不考虑干扰时隙的释放过程）。

### 3.3.1 复杂度分析

假设一次链路调度过程最多预约的时隙数为  $E$ ，则最多需要对  $E$  个时隙进行共存测试，还是以以一个时隙为例进行分析。在共存测试过程中，只有第  $e$  和第  $k$  步需要遍历 `n_info_table` 和 `o_info_table`，这两张表中各自最多可以存储  $N$  条记录，则遍历的时间复杂度为  $O(N)$ ，而剩余的其他操作均能在常数时间内完成，即复杂度为  $O(1)$ ，因此对一个时隙进行共存测试的时间复杂度为  $O(N)$ 。又  $E \leq 10$ ，则一次共存测试过程的时间复杂度为  $O(N)$ 。

### 3.3.2 开销分析

共存测试过程的开销同样包括存储开销和链路开销。

#### （1）存储开销

由于测试过程中使用的邻居信息表和时隙状态表与调度过程中存储的相同，因此不将该部分的存储开销计入测试过程。测试过程主要使用到了 `n_info_table` 和 `o_info_table` 这两张测试通知表，而表中只记录邻居节点编号、时隙编号和时隙位置这三个字段，总计 12 个字节，假设最大存储记录数量为 100，则测试过程共消耗 2.4 *kB*，相对现代节点的存储能力，这些存储开销可以忽略不计。

#### （2）链路开销

从图 3.7 中可以发现，测试过程的链路开销主要来自用于发送测试通知的  $T_{DC}$  时隙和最后进行测试的时隙。首先计算测试一个时隙的链路开销，一般情况下，参与 OLI 干扰测试的 CD 链路数量不大于 5，则测试一个时隙需要的业务时隙数量不大于 6，假设一次最多预约的时隙数量为 10，则一次完整的测试过程耗费的时间最多为 60 个业务时隙，而测试过程持续的时间为一个超帧加一个复帧，令链路开销  $\gamma$  为耗费时间与持续时间的比值。以实际的仿真为例，设定业务时隙的长度为 0.455 *ms*，BS 时隙的长度为 6.4 *ms*，一个复帧包含 80 个业务时隙，复帧长度为 42.8 *ms*，则计算得  $\gamma$  等于 5.8%，而实际通信过程中的链路开销是一定远小于该值的，因此共存测试过程只耗费很少的链路开销。

## 第四章 计算机仿真与结果分析

本文使用 OPNET 软件对所设计 IS-TDMA 协议进行仿真，OPNET 是一个为用户提供一系列仿真模型的网络仿真技术软件包，包含三个主要模块。本文所做的仿真工作都是基于 OPNET Modeler 模块，通过网络模型中插入自定义的探针模型来收集数据并进行统计，从而对 MAC 协议的性能进行分析。

### 4.1 仿真平台搭建

OPNET 采用了分层建模的思想，最底层为进程（Process）模型，用状态机来描述要仿真的协议，进程模型的上层为节点（Node）模型，该层将进程模型封装成模块，各个模块之间通过状态线来连接，最上层为网络模型，反映真实的拓扑结构<sup>[29,30]</sup>。三层模型对应了要仿真的协议、运行协议的节点设备和网络，全面地反映网络的特性。

#### 4.1.1 天线模型

OPNET 使用天线模式编辑器来创建天线模型<sup>[31]</sup>，其原理是用若干个离散的二维的圆锥形表面来模拟三维的天线增益，并为这些圆锥形表面设置增益值，如图 4.1 所示。将二维圆锥面统称为分层（plane），每个分层与  $z$  轴的夹角为  $\varphi$ ，圆锥表面的点投影到水平面与  $x$  轴的夹角为  $\theta$ ，OPNET 实际上就是利用球坐标系的  $\theta$  和  $\varphi$  来创建三维天线模型， $\varphi$  的范围是  $0\sim180^\circ$ ， $\theta$  的范围是  $0\sim360^\circ$ 。

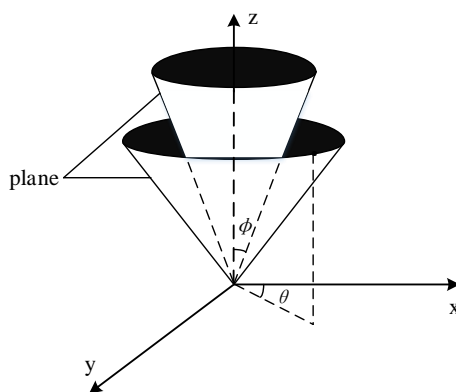


图 4.1 三维天线模型示意图

模拟一个天线模型一般需要很多个分层，且分层数量越多越能接近连续的增益，通过给这些分层指定增益值可以得到整个天线的增益值。以一个分层为例， $\varphi$  值决定了天线的波束范围，对于定向天线模型，通常先将  $\varphi$  值固定，然后设置该  $\varphi$  值在  $\theta$  范

围 ( $0\sim 360^\circ$ ) 内的增益值。图 4.2 为本文的仿真中所使用的定向天线模型, 设置天线主瓣宽度为  $18^\circ$ , 设置最大增益值为  $20\text{ dB}$ 。

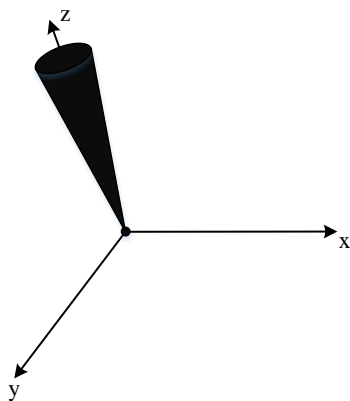


图 4.2 定向天线模型示意图

### 4.1.2 进程模型

进程模型用状态机之间的转移来进行协议的描述, 是整个网络仿真模型层次中最核心的一层。图 4.3 是本文所设计的 MAC 协议的简化进程模型, 图中主要对处于网络状态的节点的状态转移进行了描绘, 而省去了节点建网、入网和同步部分的状态转移, 因为这部分并不是我们设计的重点。

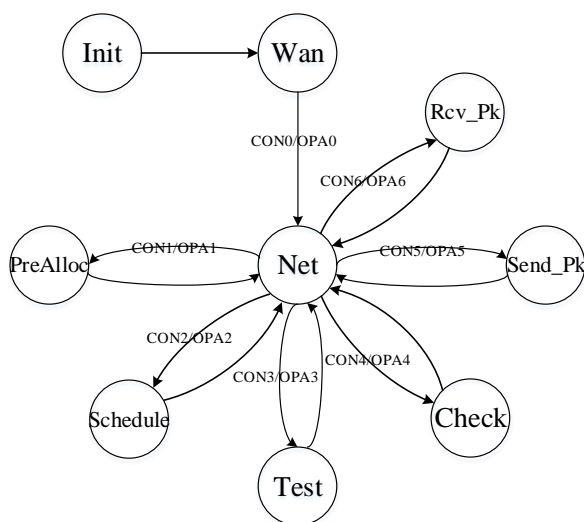


图 4.3 进程模型示意图

图 4.3 中带有条件的转移线表示的是一次有条件的状态转移, 而不带有条件的转移线表示的是一次强制状态转移。下面对进程模型各部分的功能进行简要介绍。

**Init:** 初始化, 节点开机之后便进入初始化状态, 对节点上的资源分配和定向天

线指向等进行初始化。

**Wan:** 等待入网，用这一个状态机简化了节点的入网过程，节点在此处等待接收邻居节点发送的确认本节点可以入网的信息。

**Net:** 在网并空闲，节点成功进入了网络状态，并处于空闲。

**PreAlloc:** 时隙预分配，节点完成入网之后，首先检查时隙预分配标志位，若还未进行预分配则进入时隙预分配，预分配过程结束后，节点重新回到空闲状态。

**Schedule:** 链路调度及共存测试，节点计算平均队列长度和空闲时隙数量，满足预约/释放条件时进入此状态机，进行时隙的动态预约和释放，其中动态预约过程交织了链路共存测试过程，并添加节点在测试时隙发送/接收测试分组的中断事件，结束回到空闲状态。

**Test:** 启动测试，发送测试分组事件中中断被调度，进入测试状态机，各参与测试链路进行测试分组的传输，测试时隙结束回到空闲状态。

**Check:** 测试反馈，接收测试分组事件中中断被调度，进入测试结果检查状态机，测试反馈结果决定了链路最终能否建立，结束回到空闲状态。

**Send\_Pk:** 发送分组，节点上各种类型的分组都进入此状态机中发送，发送完成后回到空闲状态。

**Rcv\_Pk:** 接收分组，底层有分组到达的流中断触发此状态机进行分组的接收，并对接收的不同类型的分组进行相应的处理，接收完成后回到空闲状态。

在实际的 OPNET 进程模型编辑器中，关于 IS-TDMA 协议的进程模型编写如图 4.4 所示，图中在进入 INIT\_NET 状态机之前的部分，全部用于节点的建网和入网过程，而之后的 NET 状态的各种状态转移过程用于链路调度和链路共存测试。

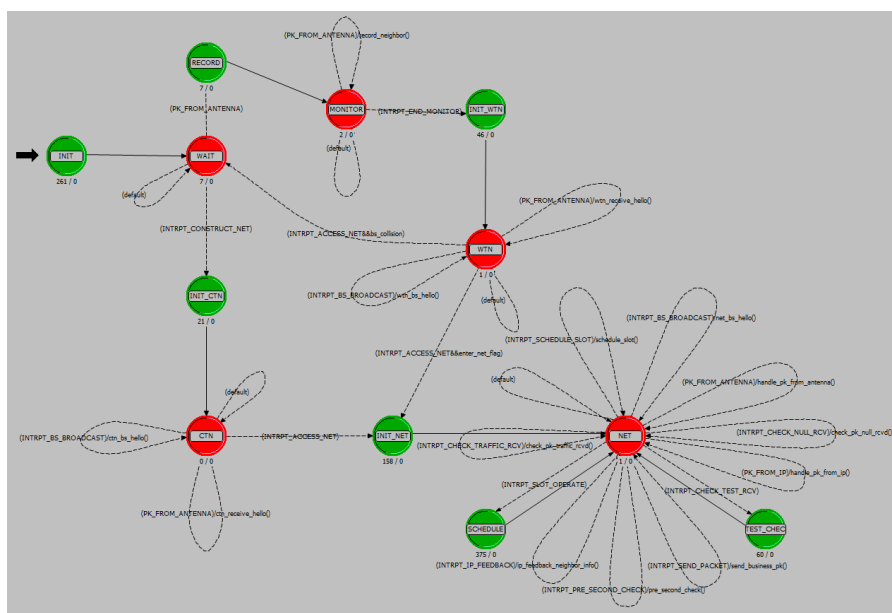


图 4.4 实际仿真中的进程模型示意图

### 4.1.3 节点模型

节点模型是对真实网络中的设备节点的模拟，它反映了真实网络进行通信时的协议栈特性，并将各层协议封装成单独的模块，各个模块之间通过状态线进行连接，模块间可进行数据交互。本文所使用的节点模型如图 4.5 所示。

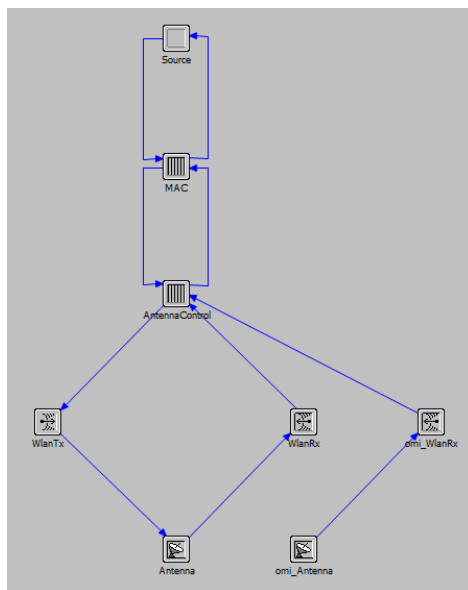


图 4.5 节点模型示意图

**Source:** 业务源模块，用于产生向网络中其他节点发送的业务分组，并将其发往下一层模块，可以对业务产生的速率进行控制；负责接收下层模块提交的其他节点发送的业务分组，并进行业务接收统计。

**MAC:** MAC 协议模块，调用 4.1.2 节编写的进程模型，是对 MAC 协议的封装。

**AntennaControl:** 天线控制模块，用于控制定向天线调整指向时波束的旋转，以及节点的接收在定向和全向模式之间的切换。

**WlanTx/WlanRx:** 无线发/收信机模块，用于将来自上层模块的分组发送至天线和将来自天线的分组发送至上层模块。

**Antenna/Omni\_Antenna:** 定向/全向天线模块，定向天线用于业务数据的收发和控制数据的发送，全向天线用于控制数据的辅助接收，**Antenna** 模块将调用 4.2.1 节中设计的天线模型。

### 4.1.4 拓扑模型

拓扑模型，也可称为网络模型，是对真实网络场景的模拟，拓扑模型中包含若干



个封装有节点模型的设备节点，并且对网络场景的区域大小和节点的通信距离等进行设置。本文所使用的仿真场景为一块  $20\text{km} \times 20\text{km}$  大小的区域，单个节点的最大通信距离为  $15\text{km}$ ，场景中节点的数量以及所有节点构成的拓扑形状可以根据不同的仿真需求进行修改。如图 4.6a 和图 4.6b 即为常见的全连通网络和多跳网络的拓扑模型。

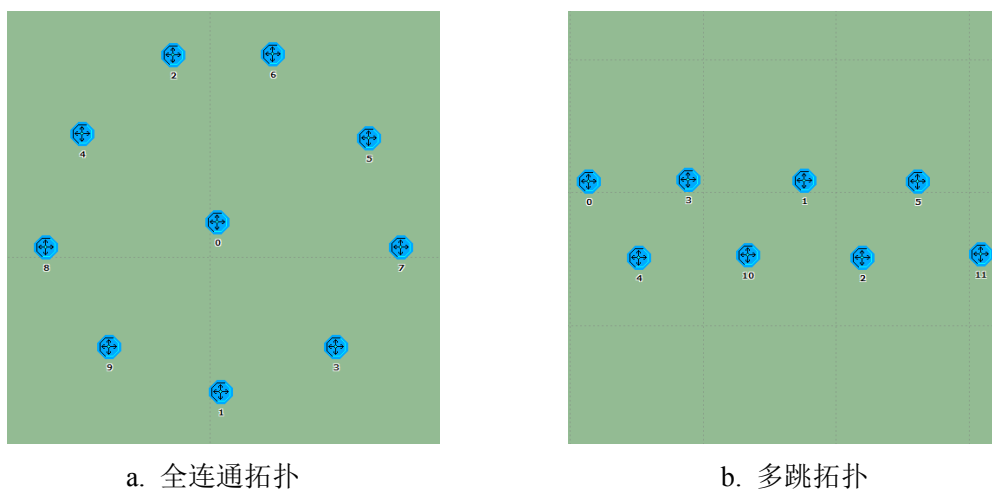


图 4.6 拓扑模型示意图

## 4.2 性能评价指标

本文主要从网络吞吐量、网络时延和丢包率三个方面来对所提出的 IS-TDMA 协议的性能进行评估，通过在 OPNET 仿真中自定义这些指标的探针来对仿真结果进行收集和统计。

### (1) 网络吞吐量

网络吞吐量是指网络在单位时间内最多可以传输的数据量，以比特、字节和分组等进行计量，大多数 MAC 协议都将吞吐量作为衡量协议性能的最重要的指标之一。本文使用分组作为传输单位，并统计单位时间内整个网络可以接收的分组总数作为网络的吞吐量。

### (2) 网络时延

网络时延是指网络中所有分组从源节点处产生到被目的节点接收所持续的平均时间，反映了网络的拥堵程度，对于本文设计的 MAC 协议而言，网络时延可以间接地反映链路调度的效率。

### (3) 丢包率

丢包率是指整个传输过程中丢失的分组数占发送的分组总数的比重，可以很好地反映链路的质量，本文对丢包率的统计采用注入负载减去该负载下的最大吞吐量再除以注入负载的方式。

## 4.3 链路调度性能分析

### 4.3.1 仿真参数与拓扑模型

#### (1) 仿真参数

由于本文所提出的 MAC 协议基于 TDMA 协议，因此需要对时帧结构中各划分单位的长度设置，其他需要设置的物理参数还包括信道速率、天线主瓣宽度和主/旁瓣增益等，具体的参数值如表 4.1 所示。

表 4.1 链路调度过程仿真参数

仿真参数	参数值
信道速率 $C_r$	20Mbps
广播时隙长度 $L_b$	6.4ms
业务时隙长度 $L_t$	0.455ms
广播时隙个数 $m$	10
业务时隙个数 $n$	80
复帧长度 $L_m$	42.8ms
超帧长度 $L_s$	428ms
定向天线主瓣宽度 $\Omega$	$18^\circ$
定向天线主瓣增益 $G_m$	20dB
定向天线旁瓣增益 $G_s$	-20dB
干扰时隙判定门限 $I_T/I_R$	$9^\circ$
仿真时间	2min

#### (2) 拓扑模型

针对链路调度算法的仿真，我们主要采用了如图 4.7 所示的两种典型的多跳拓扑结构。它们都存在链路重合的情况，如图 4.7a 中的 0-9、6-7 链路以及图 4.7b 中的 8-2、4-6 链路，将这种距离很近、一条链路的发送节点和另一条链路的接收节点的辐射范围有重合的链路称为重合链路，当两条重合链路预约了相同的时隙进行通信时将出现链路干扰。

由于本文使用的时隙动态预约算法的三次握手与 DTRA 协议<sup>[18]</sup>类似，因此在本文搭建的 OPNET 仿真平台上模拟运行 DTRA 协议的时隙预约部分，用来比较普通的无干扰抑制的 DTRA 协议和本文所设计的有干扰抑制的 IS-TDMA 协议在网络性能上的表现差异。

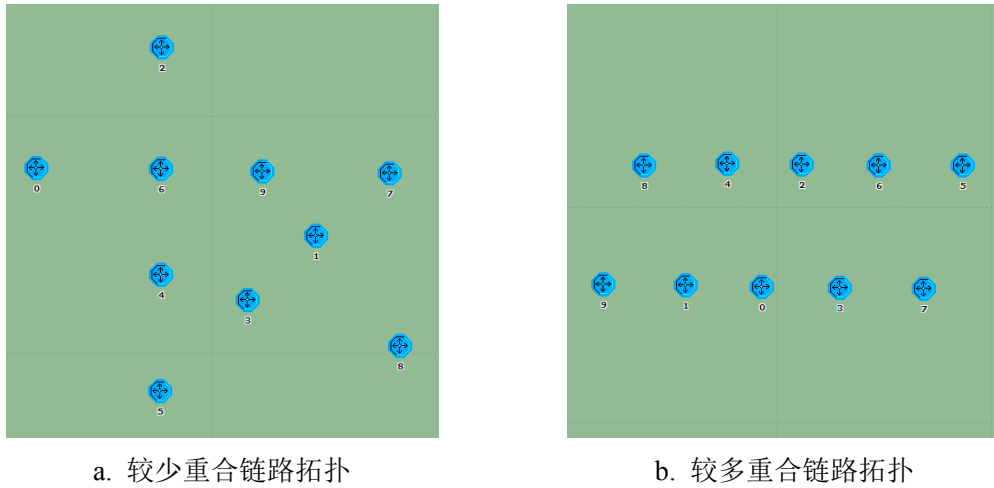


图 4.7 链路调度仿真拓扑模型示意图

### 4.3.2 不同业务负载下的性能分析

#### (1) 业务负载设置

为了对图 4.7 所示的两个仿真拓扑的业务负载进行设置，首先需要计算网络的理论最大吞吐量。假设 TDMA 帧结构中一个复帧的长度为  $L_m$ ，单位为秒，一个复帧中最多可以发送的业务分组数量为  $n_d$ ，则网络的最大吞吐量  $S$  可以由下式计算而得：

$$S = \frac{n_d}{L_m} \quad (4-1)$$

本文中所使用的复帧由一个 BS 时隙和  $n$  个 TS 时隙构成，除去用作调度时隙的 0 号时隙，一个复帧有  $n-1$  个时隙用于发送业务分组。由于定向传输可以实现空间复用，同一时刻网络中最多可能存在  $n_k/2$  条链路， $n_k$  表示节点总数，这里存在最多链路的情况只出现在像图 4.6a 所示的全连通且任意两条链路之间都不相互干扰的拓扑，而网络中的大部分拓扑在同一时刻是无法真正实现  $n_k/2$  条链路的。因此，理想情况下，一个复帧总共可以发送的业务分组数量  $n_d$  如式 4-2 所示：

$$n_d = (n-1) * \frac{n_k}{2} \quad (4-2)$$

将  $n_d$  和  $L_m$  的具体表达式代入式 4-1，最大吞吐量  $S$  可以表示为如式 4-3：

$$S = \frac{(n-1) * n_k}{2 * (L_b + n * L_t)} \quad (4-3)$$

对于图 4.7 的仿真拓扑，将  $n_k = 10$ 、 $n = 80$  代入式 4-3，可得网络的最大吞吐量为  $9228 \text{ packets/s}$ 。为了避免过载，实际通信时业务负载应小于节点服务速率，则平分到每个节点上的业务负载应小于  $922 \text{ packets/s}$ ，也即分组到达间隔不应小于  $0.00108\text{s}$ 。而在网络仿真中，为了验证网络最大吞吐量，需要考虑过载的情况，因此，业务负载的设置如表 4.2 所示。

表 4.2 业务负载设置

到达间隔 (s)	负载 (packets/s)
0.005	2000
0.003	3333
0.00175	5714
0.0015	6666
0.00125	8000
0.0012	8333
0.00115	8695
0.0011	9090
0.001	10000

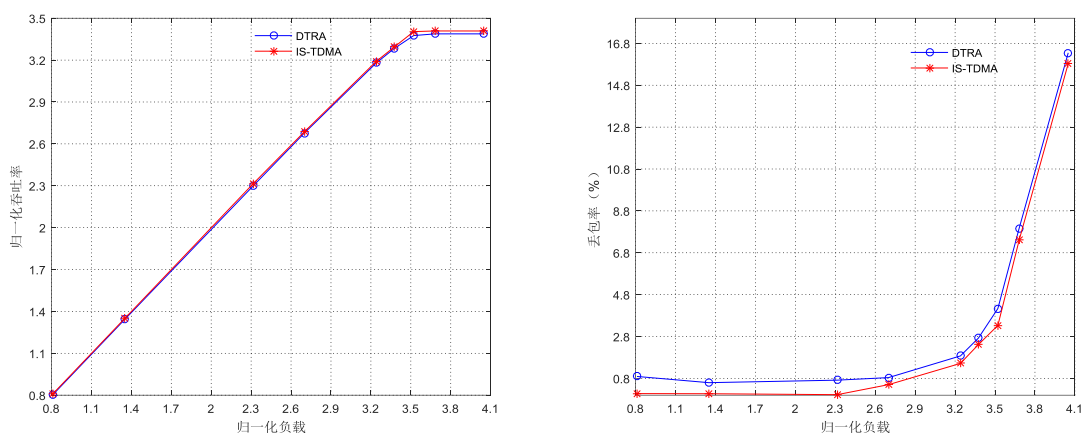
## (2) 仿真结果分析

本文所使用的每一个仿真结果均取自 5 次相同参数仿真的平均值，且在仿真图中对负载和吞吐量进行了归一化处理。

### ① 拓扑模型一

图 4.8a 是对图 4.7a 所示的拓扑进行仿真得到的网络吞吐量和丢包率结果。该拓扑结构的特点是多跳、但节点分布较为紧密，存在链路重合的情况，但是相比一般的链路，重合链路只占总链路数的很少部分。由图 4.8a 可以发现，当有干扰抑制过程，即在时隙动态预约时进行干扰时隙判定时，IS-TDMA 的网络吞吐量要高于 DTRA，但优势并不明显，这是因为网络中的重合链路数量较少，导致干扰抑制产生作用的机会较少。当负载较小时，预约相同时隙的可能性较小，因此有、无干扰抑制对吞吐量的影响很小；而当负载逐渐增大至最大服务速率时，节点需要预约更多的时隙，这时若没有干扰抑制过程，则预约相同时隙的可能性将增大，如图 4.8a 中归一化负载大

于 3.24 时，吞吐量也随之明显降低。



a. 吞吐量随负载变化

b. 丢包率随负载变化

图 4.8 较少重合链路拓扑吞吐量和丢包率随负载变化

从图 4.8a 中可以看出，在开启干扰抑制功能后，该拓扑模型在归一化负载达到 3.52 时吞吐量基本已达到最大值，有干扰抑制的 IS-TDMA 的最大归一化吞吐量约为 3.41，而无干扰抑制时的最大归一化吞吐量约为 3.39，当负载继续增大时网络将处于过载状态。

图 4.8b 是和图 4.8a 的吞吐量所对应了丢包率曲线，从图中可以看出，当业务负载较小时，无干扰抑制的 DTRA 的丢包率要明显高于有干扰抑制的 IS-TDMA；而当业务负载逐渐增大（未过载），即使网络有干扰抑制过程，也出现了少量的丢包现象，这是由实际信道的波动、不稳定等因素导致的占用相同时隙的相邻链路之间的干扰，在 4.4 节的仿真中的共存测试过程将对该种干扰进行抑制；当网络处于过载状态时，显然丢包率将呈现陡增趋势。

网络的平均端到端时延结果如图 4.9 所示，从总体上看，网络在有、无干扰抑制过程时的端到端时延相差并不大，有干扰抑制过程的 IS-TDMA 的时延略大一些。当注入的负载较小时，由于节点需要进行的时隙预约次数不多，所以调用干扰时隙判定的次数不多，图中显示在归一化负载小于 2.9 时，有干扰抑制与没有干扰抑制的时延都处于较低水平；而当负载逐渐增大，IS-TDMA 在预约每一个时隙时都要进行干扰时隙判定，这将耗费一定的时间，且当负载足够大时，由于时隙数量有限，可能会出现无时隙可预约的情形，这时必须等待时隙释放，而等待的时间会导致分组在队列中的堆积，因此时延略高于无干扰抑制的协议；当网络处于过载状态时，两种协议的端到端时延都将趋于无穷大。虽然 DTRA 的时延比 IS-TDMA 有轻微的优势，但是这是以吞吐量的下降为代价的。

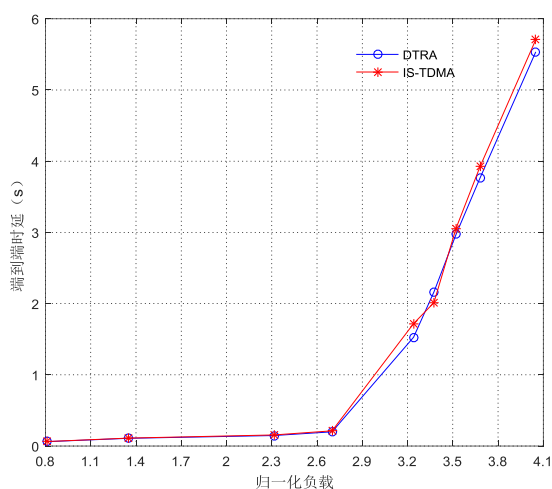
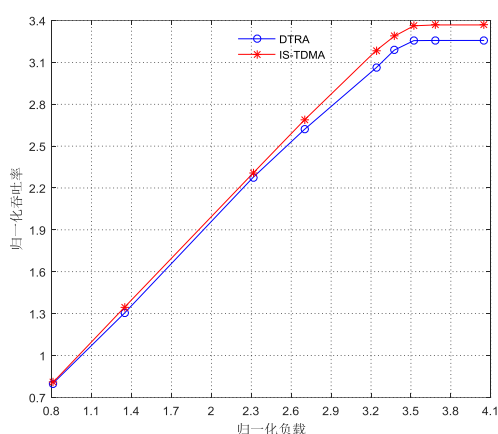


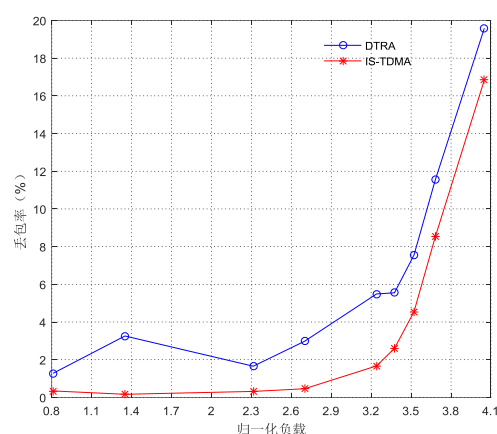
图 4.9 较少重合链路拓扑端到端时延随负载变化

## ② 拓扑模型二

图 4.10a 所示的吞吐量曲线图对应了图 4.7b 中的拓扑模型，该拓扑的特点是多跳且存在较多的重合链路。在对上一个拓扑的吞吐量进行分析时，解释了重合链路对吞吐量的影响，此处，由于拓扑中有较多的重合链路，因此加入干扰抑制对网络吞吐量的提升效果将更为明显。当注入的业务负载逐渐增大，节点需要更多的发送时隙，重合链路预约相同时隙的可能性也随之增大，若不进行干扰抑制，则重合链路的的同时传输必然使得其中一条或两条链路传输失败，降低网络的吞吐量；反之，若进行干扰抑制，则可以有效地避免重合链路预约相同的时隙，从图 4.10a 中可以看出当负载较大时使用干扰抑制给吞吐量带来的提升效果更加突出。



a. 吞吐量随负载变化



b. 丢包率随负载变化

图 4.10 较多重合链路拓扑吞吐量和丢包率随负载变化

综合以上两种不同拓扑的吞吐量表现,本文提出的干扰抑制方法对吞吐量的提升效果与具体的网络拓扑结构相关,也即对于有较多重合链路的拓扑能取得更好的效果。该拓扑在使用了干扰抑制过程后,当归一化负载达到 3.52 时,归一化吞吐量可以达到最大值 3.37。同样地,由于信道不稳定等外部因素的影响,网络仍然存在丢包现象,丢包率的曲线如图 4.10b 所示。

从图 4.10b 中可以发现,当网络拓扑中存在较多的重合链路时,如果在预约时隙时不使用干扰抑制,那么将出现很明显的丢包,且随着注入负载的增大,丢包问题会愈发严重。在网络进入过载状态之前,无干扰抑制的 DTRA 的丢包率最大达到了 8% 左右,这对正常的无线通信来说是无法接受的。相比前一个有较少重合链路的拓扑,该拓扑在开启干扰抑制后的丢包率更高,这是因为不理想的信道导致的链路干扰问题对于该拓扑的影响更大。

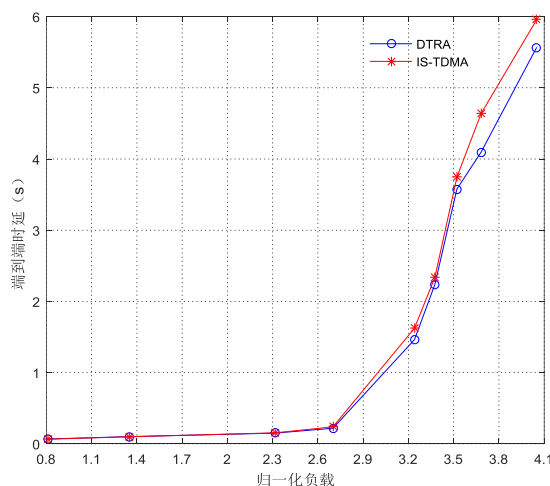


图 4.11 较多重合链路拓扑端到端时延随负载变化

图 4.11 是对第二个拓扑模型的端到端时延仿真曲线,该拓扑下的端到端时延表现与前一个拓扑的基本类似。在归一化负载达到 3.24 时,使用干扰抑制的 IS-TDMA 的时延相比 DTRA 有一个明显的增长,这是因为在时隙预约时由于时隙数量有限和时隙干扰判定的限制,没有找到可以预约的时隙,于是等待可预约时隙的时间导致了端到端时延的增加。通过以上分析可知,调用干扰抑制过程将不可避免地增加端到端时延,但这较少的时延代价却可以换来较大的网络吞吐量提升。

## 4.4 链路共存测试性能分析

链路共存测试是对链路调度部分预约完成的链路进行通信测试,我们将重点对 IS-TDMA 协议使用链路共存测试算法前后的网络性能进行比较。

### 4.4.1 仿真参数与拓扑模型

#### (1) 仿真参数

共存测试过程的仿真基于前面的链路调度过程，即保持干扰抑制始终开启，基本的仿真参数与表 4.1 所示的相同。需要额外添加的一个参数是完整的共存测试周期  $D_{IT}$ ，设置其值为 1.3 个超帧，即  $0.5564\text{ms}$ 。

#### (2) 拓扑模型

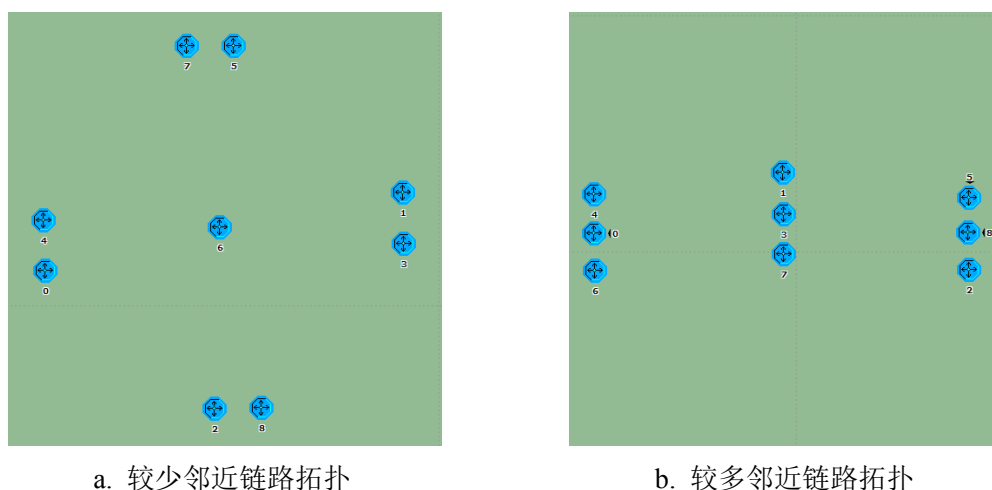


图 4.12 链路共存测试仿真拓扑模型示意图

由于一般的拓扑结构(如图 4.7 所示的拓扑)在经过链路调度的干扰时隙判定后，链路之间存在干扰的情况很少，因此无法满足我们对共存测试的仿真需求。为了构造出相邻链路之间的干扰，需要使用那些有多个相距很近的节点的拓扑模型，如图 4.12 所示。图 4.12a 的拓扑中有多个两两成簇的节点，每两对这样的节点形成的邻近链路可以通过干扰时隙判定，但由于距离太近依然可能存在干扰；图 4.12b 的拓扑中节点更加密集，相比图 4.12a 的拓扑有更多的邻近链路，链路干扰现象也将更加明显。

### 4.4.2 不同业务负载下的性能分析

#### (1) 业务负载设置

对于链路共存测试算法的仿真，节点上使用与链路调度算法相同的分组到达间隔，但由于选取的两个拓扑模型中的节点数量为 9，因此每种分组到达间隔下的网络总的业务负载将有所变化。

#### (2) 仿真结果分析

##### ① 拓扑模型一

图 4.13a 所示的吞吐量随负载变化的曲线对应了图 4.12a 的拓扑模型，我们对网络开启、关闭共存测试过程分别进行仿真。从图中可以看出，是否开启共存测试对网



络的吞吐量有比较大的影响。

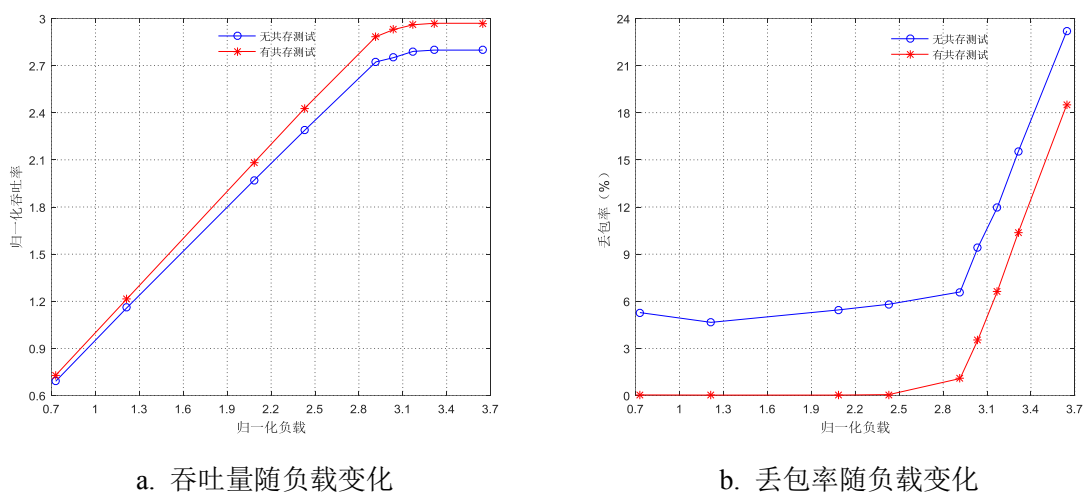


图 4.13 较少邻近链路拓扑吞吐量和丢包率随负载变化

当关闭链路共存测试时，以拓扑图中的 7-1 和 5-3 链路为例，若 7-1 链路已成功预约时隙 1 并开启了业务传输，此时 5-3 也要预约时隙 1，在节点 5 和节点 3 端时隙 1 均能通过干扰时隙判定，最终预约成功并开启传输，但是实际通信中 5-3 链路的建立导致了 7-1 链路和自身的传输失败。而当开启链路共存测试时，5-3 链路完成对时隙 1 的预约后需要先对链路传输进行测试，测试出链路冲突后 5-3 链路的 1 时隙将被释放，并选择预约其他业务时隙，有效地降低了网络中的链路干扰，提高网络吞吐量，如图 4.13a 中所示，开启共存测试时，吞吐量相比不开启时有明显提升。

从图 4.13a 中可以发现，有链路共存测试的网络吞吐量相比无链路共存测试的提升量随着负载的增大而明显增大。当归一化负载增大至 3.17 时，网络吞吐量基本已到达最大值，关闭共存测试时网络的最大归一化吞吐量约为 2.8，而开启共存测试后网络的最大归一化吞吐量可以达到约 2.97，且网络中邻近干扰链路越多，吞吐量的提升就越大。

图 4.13b 是该拓扑下开启、关闭共存测试过程对应的丢包率曲线。当注入的负载未达到网络最大吞吐量时，开启共享测试的仿真丢包率处于很低的水平，基本无丢包；反观关闭共存测试时的仿真曲线，可以发现自负载注入开始，丢包率就已接近 6%，且随着负载的增大呈现出逐渐增长的趋势，丢包现象很严重。

端到端时延的仿真曲线如图 4.14 所示，图中有链路共存测试的端到端时延始终要略大于无链路共存测试，原因在于无共存测试时时隙预约的三步握手完成后即可开启业务传输，而开启共存测试后，在业务传输开始前，测试过程需要耗费一个复帧，当测试出干扰时，干扰时隙释放过程又额外需要两个复帧，因此，网络的端到端时延比无共存测试时更大一些。然而，无链路共存测试的时延优势是以吞吐量的下降为代

价的，且吞吐量下降的幅度是很明显的，而时延上的优势并不是很突出，因此添加链路共存测试对于提升网络性能很有必要。

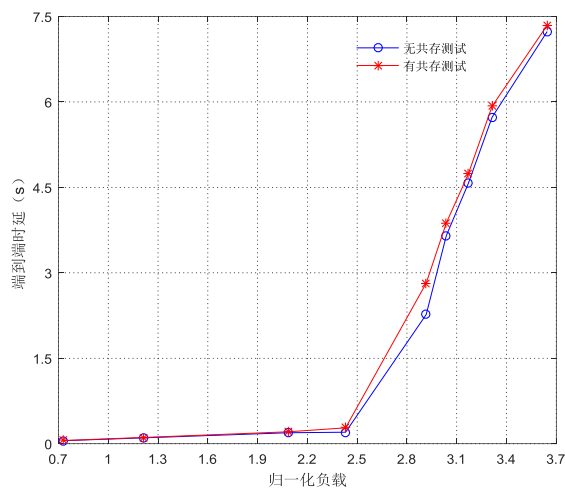
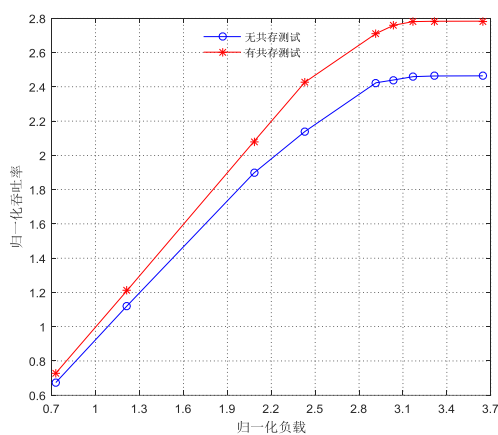


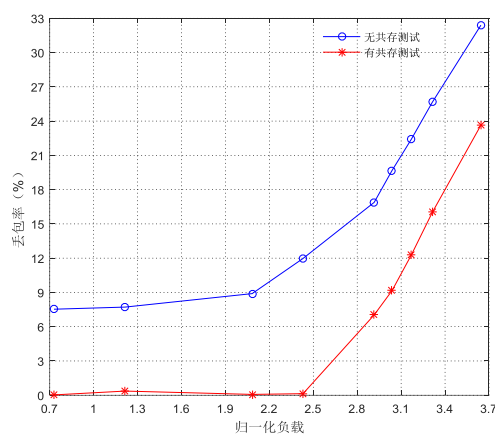
图 4.14 较少邻近链路拓扑端到端时延随负载变化

## ② 拓扑模型二

图 4.15a 所示的曲线是对图 4.12b 中的拓扑模型开启、关闭链路共存测试时的网络吞吐量仿真结果。相比上一个拓扑模型，该拓扑模型中存在更多的邻近链路，且这些链路之间有较大可能性存在干扰。观察拓扑中的 0-1 和 7-3 链路，根据 2.3.5 节的干扰时隙判定方法，0-1 链路可以预约和 7-3 链路相同的时隙进行传输，而实际测试发现节点 7 的发现将会干扰 1 节点的接收，导致 0-1 链路始终传输失败。从下图的吞吐量曲线可以发现，当关闭共存测试时，网络吞吐量出现较大幅度的下降，这正是由上面分析的链路干扰导致的。



a. 吞吐量随负载变化



b. 丢包率随负载变化

图 4.15 较多邻近链路拓扑吞吐量和丢包率随负载变化

相比前一个拓扑模型的吞吐量曲线,该拓扑模型下的吞吐量在开启共存测试前后的提升量更加明显,可见,在应对网络节点密度大、邻近链路多的拓扑情形时,本文所提出的链路共存测试算法可以发挥更好的效果。

图 4.15b 显示了网络中的丢包率随负载变化的曲线,从图中可以发现,关闭共存测试时网络的丢包率始终处于一个较高水平,在过载之前最高甚至达到了 22%左右,这是由网络中存在较多的邻近链路导致的链路干扰造成的;当开启共存测试后,丢包现象得到了很明显的改善,在负载较小时几乎可以控制在无丢包,而当负载很大时,还是出现了丢包的情况,这说明本文提出的链路共存测试算法在应对如图 4.12b 所示的复杂拓扑情形时仍然存在一些不足。

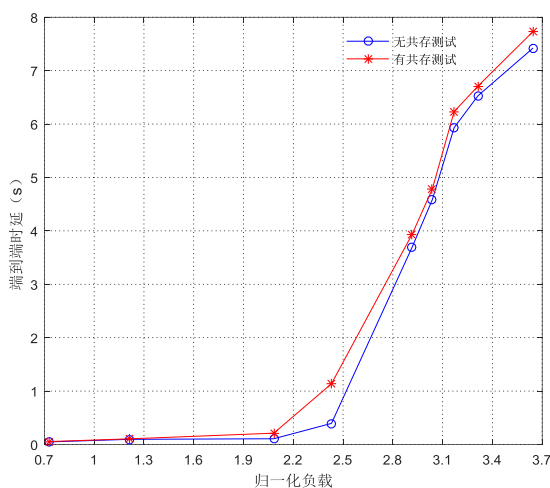


图 4.16 较多邻近链路拓扑端到端时延随负载变化

网络的端到端时延性能曲线如图 4.16 所示,从图中可以发现,当归一化负载增大至 2.43 时,开启共存测试的端到端时延相比未开启时有一个很明显的增长,这种现象可能由两方面的原因造成:一是时隙预约数量达到饱和,需要等待可预约的时隙;二是很多预约的时隙未通过共存测试,需要进行时隙释放。总体上,加入链路共存测试确实会增大网络的平均端到端时延,但相比不加入链路共存测试时非常大的吞吐量下降幅度,小幅度的时延增长代价是完全值得的。



## 第五章 总结与展望

### 5.1 研究总结

随着天线和无线网络技术的不断发展,定向天线已越来越多地应用到移动自组织网络中。定向天线凭借其传输距离远、抗干扰能力强等特点给移动自组网的网络性能带来了提升,包括通信范围、网络时延和网络吞吐量等性能;同时,由于定向天线的成功传输需要收发双方在空间上实现对准,因此给定向自组网的 MAC 协议设计提出了更高的要求。本文主要围绕定向自组网的链路调度和链路共存测试两个方面进行 IS-TDMA 协议的设计,具体研究内容可以总结为以下几个方面:

(1) 采用“定向+全向”的物理层通信能力,定向天线用于业务分组的收发和控制分组的发送,而全向天线仅用于接收控制分组,这样可以有效地提高节点入网和链路调度等过程的效率。

(2) 针对时隙动态预约过程中对可预约时隙的选择过程,根据定向天线的主瓣覆盖范围和收发天线的对准特点,设计了  $w$  和  $w'$  两个角度值来更加精确地量化链路之间的干扰,分别从发送和接收节点的角度判定要预约的时隙是否存在干扰,从而实现对于干扰时隙的抑制,提高链路调度的成功率。

(3) 考虑实际通信信道的波动、定向天线覆盖不精确等现实因素的影响,完成链路调度的链路和网络中的已有链路仍然存在相互干扰的可能性,为此,本文设计了一种新建链路和已有链路的共存测试方法,通过将测试准备过程与链路调度过程交织,可以实现调度完成即开启测试,对链路干扰进行测试的同时降低测试耗费的开销,该方法可以有效地保证链路的可靠性并提升网络的吞吐量。

(4) 使用 OPNET 软件对 IS-TDMA 协议进行仿真,仿真结果显示基于角度量化的干扰抑制方法和链路共存测试方法在网络性能提升方面有着不错的表现。

### 5.2 研究展望

虽然本文所设计的 MAC 协议能够有效地提升定向自组网的网络性能,但是由于本文作者的能力和 research 时间有限,该研究课题中仍有一些内容可以进行更加深入的研究,主要包括:

(1) 在第二章所设计的时隙预分配方法中,划分了足够多的时隙用于应对不同拓扑场景的预分配时隙数量需求,这样做其实存在着一定的时隙浪费,可以进一步地研究节点密度、拓扑形状等与预分配时隙数量的关系,最大化时隙复用率。

(2) 链路共存测试算法在应对较为复杂的、节点密度大的拓扑场景时会出现一

定的丢包现象，可以对测试链路的选择、通知以及测试结果的判定等方面进行更深入的优化，提高链路共存测试算法的普适性。

（3）时隙的动态预约和动态释放的启动判断是同步进行的，仿真中节点出现轻微的在预约和释放之间来回震荡的现象，需要进一步地研究预约和释放的频率控制，使得时隙的预约和释放更平稳地进行。

## 参考文献

- [1] 于宏毅. 无线移动自组织网络[M]. 北京: 人民邮电出版社, 2005.4.
- [2] 何立阳, 蒋国华, 何军. 移动自组织网络的研究应用[J]. 网络安全技术与应用, 2009, 9:40-43.
- [3] 刘晓红. 移动 Ad hoc 网络的特点及应用[J]. 现代通信, 2007(Z3):96-100.
- [4] 莫代会, 印敏, 王宏达. 一种新的军用移动 Ad Hoc 网络及 DSR 路由算法[J]. 电信快报, 2005, 12:15-17.
- [5] 霍永华, 方子希, 于建, 齐忠杰. Ad Hoc 网络在应急应用领域的效能评估研究[J]. 无线电通信技术, 2015, 41(2):12-14,18.
- [6] 杨卫东, 张光昭. Ad Hoc 网络在紧急救援中的应用[J]. 电信科学, 2008(4):101-105.
- [7] Tu Y, Zhang Y, Zhang H. A novel MAC protocol for wireless Ad hoc networks with directional antennas[C] IEEE International Conference on Communication Technology. IEEE, 2013:494-499.
- [8] 王杉, 张旭东, 魏急波, 王建新. 定向天线在 Ad Hoc 网络中的设计与应用[J]. 计算机工程与应用, 2006, 21(4): 99-102,110.
- [9] Chi-Kin Chau, Richard J. Gibbens, Don Towsley. Impact of directional transmission in large-scale multi-hop wireless ad hoc networks[C] Proceedings IEEE INFOCOM. IEEE, 2012:522-530.
- [10] Rabin Bhusal, Sunil K. Timalsina, Sangman Moh. Impact of directional antennas on contention based MAC protocols in ad hoc networks[C] International Conference on Computing and Networking Technology (INC, ICCIS and ICMIC). IEEE, 2012:405-408.
- [11] Jianyu Niu, Ruonan Zhang, Lin Cai, Jianping Yuan. A fully-distributed directional-to-directional MAC protocol for mobile ad hoc networks[C] International Conference on computing, Networking and Communications(ICNC), 2015:766-770.
- [12] Das A, Zhu T. A Reservation-Based TDMA MAC Protocol Using Directional Antennas (RTDMA-DA) For Wireless Mesh Networks[C] Global Communications Conference, 2007. GLOBECOM '07, Washington, Dc, Usa, 26-30 November. DBLP, 2007:5102-5106.
- [13] Jing Wang, Yu Zhang, Long Jiang. A Novel Time-Slot Allocation Scheme for Ad Hoc Networks with Single-Beam[C] IEEE International Conference on Communication Software and Networks (ICCSN). IEEE, 2015:227-231.
- [14] Mineo Takai, Jay Martin, Rajive Bagrodia, and Aifeng Ren. Directional virtual carrier sensing for directional antennas in mobile ad hoc networks[C] ACM MobiHoc, 2002.
- [15] Ram Ramanathan, J.Reddi, Cesar Santivanez, D.Wiggins, S.Polit. Ad hoc networking with directional antennas: a complete system solution[J]. IEEE Journal on Selected Areas in Communications, 2005, 23(3):496-506.

- [16] R. R. Choudhury, S. Yang, R. Ramanathan and N. H. Vaidya. Using Directional Antennas for Medium Access Control in Ad Hoc Networks[C] Proc. ACM Mobile Computing and Networking (MobiCom), 2002:59-70.
- [17] Zhensheng Zhang. Pure directional transmission and reception algorithms in wireless ad hoc networks with directional antennas[C] IEEE ICC, 2005:3386-3390.
- [18] Zhensheng Zhang. DTRA: directional transmission and reception algorithms in WLANs with directional antennas for QoS support[J]. Network, IEEE , 2005, 19(3):27-32.
- [19] G.Jakllari, W.Luo, S.V.Krishnamurthy. An Integrated Neighbor Discovery and MAC Protocol for Ad Hoc Networks Using Directional Antennas[C] IEEE Transactions on Wireless Communications. IEEE, 2007:1114-1024.
- [20] Raja Jurdak, Cristina Videira Lopes, Pierre Baldi. A survey, classification and comparative analysis of medium access control protocol for ad hoc networks[C] IEEE Communications Surveys & Tutorials. IEEE, 2004:2-16.
- [21] IEEE Standards Department. Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications[J]. IEEE std 802.11-2007, 2007.
- [22] 李建东, 盛敏, 李红艳. 通信网络基础.第 2 版[M]. 北京: 高等教育出版社, 2011.
- [23] 王竞. 移动自组织网络时分多址协议的设计与实现[D]. 西安: 西安电子科技大学, 2005.
- [24] 王金杰. 纯定向自组织网络中时分多址协议的研究[D]. 西安: 西安电子科技大学, 2017.
- [25] 封宁, 白光伟. RED 算法的数学模型研究[J]. 计算机工程与设计, 2008, 29(9):2179-2180.
- [26] 熊沐阳. 基于主动队列管理的 RED 算法及改进[D]. 华中师范大学, 2014.
- [27] 朱洁静. 分布式时隙冲突检测和分解算法研究[D]. 西安: 西安电子科技大学, 2015.
- [28] Jin-Jia Chang, Wanjiun Liao, Ting-Chao Hou. Reservation-Based Directional Medium Access Control (RDMAC) Protocol for Multi-Hop Wireless Networks with Directional Antennas[C] IEEE International Conference on Communications. IEEE, 2009:1-5.
- [29] 陈敏. OPNET 网络仿真[M]. 北京: 清华大学出版社, 2004.
- [30] 李馨, 叶明. OPNET Modeler 网络建模与仿真[M]. 西安: 西安电子科技大学出版社, 2006.
- [31] 李默, 孙岷, 赵亮. OPNET 天线建模及无线管道阶段仿真研究[J]. 综合电子信息技术, 2015, 41(1):60-63.



## 致谢

在这篇硕士论文即将完成之际，回望过去三年的研究生生涯，不禁让我心生感触。这三年，是匆匆的三年，是忙碌的三年，更是不断成长的三年；这三年，有过遭遇挫折的低落，也有过收获成功的喜悦。在这里，我要真诚地感谢每一位一直以来关心、支持和帮助我的人。

首先，我要感谢我的导师史琰教授。在这三年的时间里，老师在我的学习和生活上都给予了很多的帮助，让我受益良多。本论文正是在老师的悉心指导之下完成的，从课题的选择、算法的设计、仿真工作的开展以及论文的撰写定稿，老师都付出了大量的精力和心血。老师深厚的知识积累、开阔的科研视野深深地影响着我，让我在科研的道路上勇敢前进。老师严谨的治学态度、平和的生活态度以及不断攀登的科研态度潜移默化地感染了我，让我无畏将要到来的生活和工作中的挑战。能够师从史琰老师，我感到万分荣幸。在此，再一次对老师说一声感谢，您的教诲我毕生难忘，祝福您在以后的工作和生活上更加幸福和如意。

感谢王金杰、李乐和赵永智三位师兄给予我学习和生活上的帮助和关心，尤其感谢王金杰师兄在我科研道路上的指导和帮助，让我找到了正确的方向，祝师兄们生活幸福、事业有成。

感谢实验室一起奋斗的何雨航、刘汉涛、弭宝辰和陶伟伟等小伙伴，有你们的陪伴，实验室多了很多的欢笑，感谢你们在我遇到困难时给予的无私帮助，在即将分别之际，衷心地祝你们天天快乐、前程似锦。

感谢实验室的师弟、师妹们，你们为实验室注入了很多的活力，你们刻苦的学习态度让实验室的科研氛围更加浓郁，祝你们能够学有所成。

感谢我的室友兰冲、罗渠元和张尧，你们陪我一起走过这三年的时光，我不会忘记和你们在一起收获的欢笑与幸福，祝罗、张两位博士在科研的道路上收获硕果。

特别感谢我的父母，是你们不辞辛劳、无微不至地将我养育成人、培养成才，是你们每一次在我遭遇挫折时鼓励我、陪伴我、给予我温暖，并支持帮助我找回自信，没有你们就没有今天的我，祝你们身体健康，越来越幸福。

感谢我的女朋友，不论顺境逆境始终陪伴在我身边，给予我直面困难的勇气和拼搏未来的信心。

感谢严于律己、永不言弃的自己，有志者，事竟成！

最后，再一次真心地感谢你们！我将带着你们的关心和支持继续扬帆起航！



## 作者简介

### 1. 基本情况

郁威威，男，江苏盐城人，1992 年 8 月出生，西安电子科技大学通信工程学院电子与通信工程专业 2015 级硕士研究生。

### 2. 教育背景

2011.09～2015.06 中国矿业大学，本科，专业：信息工程

2015.09～西安电子科技大学，硕士研究生，专业：电子与通信工程

### 3. 攻读硕士学位期间的研究成果

#### 3.1 发表学术论文

[1] 史琰, 郁威威. 基于 TDMA 的分布式全双工链路调度算法研究[J]. 电子学报, 2018, 46(2): 507-512.(EI)

#### 3.2 申请（授权）专利

[1] 史琰, 郁威威, 盛敏等. 基于 TDMA 的同时同频全双工无线链路调度方法: 中国, 201610344863.X[P]. 申请日期 2016-05-24.

