

分类号\_\_\_\_\_

密级\_\_\_\_\_

UDC \_\_\_\_\_

编号\_\_\_\_\_

# 中国科学院研究生院 博士学位论文

低轨卫星通信网络的星上交换和路由技术研究

曾媛

指导老师 梁旭文 研究员 刘会杰 研究员

中国科学院上海微系统与信息技术研究所

申请学位级别 博士学位 学科专业名称 通信与信息系统

论文提交日期 2009 年 5 月 论文答辩日期 2009 年 6 月 6 号

培养单位 中国科学院上海微系统与信息技术研究所

学位授予单位 中国科学院研究生院

答辩委员会主席 甘仲民 教授

Design and analysis of the satellite onboard switch and routing protocol  
in LEO Satellite communication networks

DISSERTATION

Presented in Partial Fulfillment of Requirements for the Degree of Doctor of  
Philosophy in the Graduate School of Chinese Academy of Sciences

By

May 2008

# 低轨卫星通信网络的星上交换和路由技术研究

曾媛 (通信与信息系统)

指导老师: 梁旭文 研究员 刘会杰 研究员

## 摘 要

本文针对低轨 (LEO) 卫星通信网络提出了 DRAST (Distributed Routing Algorithm Based-on Simplified Topology) 路由协议、SDRR (Subdivision Deficit Round Robin) 分组调度算法以及权值优化切换策略。在此基础上, 路由协议、调度算法和切换策略的性能得到了详细分析和仿真验证。研究的主要内容包括:

(1) 提出一种基于拓扑简化的分布式路由协议 (DRAST)。针对卫星网络拓扑进行简化, 采用不同方式计算路径, 保证了端到端时延性能, 有效降低了星间链路切换次数, 减少了路由开销, 屏蔽了卫星网络的动态性; 系统研究了路由协议中的时延参数, 建立了求解端到端时延参数的优先级排队模型, 解决了时延参数的求解问题。

(2) 设计了一种新的适用于星载交换机的 SDRR 调度算法, 复杂度低, 时延性和公平性较好; 调度时结合路由策略分配带宽, 有效降低队列和处理时延。

(3) 分析了低轨卫星网络中的切换测量; 设计了权值优化的切换方法, 有效降低了切换频率; 通过马尔可夫链建立数学模型, 推导出卫星网络中阻塞率和掉话率的估算。

(4) 提出了基于 OPNET 环境的低轨卫星网络软件仿真平台结构, 全面研究了网络拓扑结构、网络接入策略、邻居卫星节点发现方式、卫星运行轨道自主维护和星际链路模拟的实现方式; 通过此平台仿真验证了本文提出的各种算法的可行性以及良好的性能。

**关键词:** 低轨卫星网络; 路由协议; 调度算法; 切换策略; OPNET 仿真

# Design and analysis of the satellite onboard switch and routing protocol in LEO satellite communication networks

Zeng Yuan (Communication and information system)

Directed by: Liang XuWen, Liu HuiJie

## Abstract

The dissertation presented a distributed routing algorithm based on simplified topology (DRAST), a subdivision deficit round robin scheduling algorithm (SDRR) and a handoff strategy by optimizing the weights of key parameters. The performances of the proposed routing protocol, scheduling algorithm and handoff strategy are evaluated efficiently through simulations and analysis. Specifically, following topics are studied carefully:

(1) The framework of DRAST and its optimization method are introduced. The satellite networks topologies are simplified, which decreases efficiently ISL handoff numbers. An optimal method for collecting satellite state information is designed to reduce the communication overheads of routing protocol and provide the end-to-end delay bound guarantee. The end-to-end delay is systematically studied by building a priority-based queuing model. The model can be employed to solve the result of end-to-end delay.

(2) A low complexity SDRR scheduling method is proposed. The scheduling algorithm with good delay and fairness performance is suitable for onboard satellite switch. In the SDRR, the bandwidth allocation of traffic transmission depends on the selected path in a routing algorithm. As a result, the SDRR scheduling strategy efficiently reduces the queuing delay as well as processing delay.

(3) The measurement rule for handoff is analyzed. A handoff strategy by optimizing the weights of key parameters was proposed to significantly decrease the link handoff numbers. Based on a continuous-time Markov chain, the mathematic model is built with which forced terminated probability and blocking probability can be evaluated.

(4) Within the OPNET network simulation environment, we construct the simulation platform, and study the issues regarding network topology, access policy, neighboring satellite discovery, autonomic orbit maintenance, and satellite link simulation. The performance of the proposed satellite networks routing protocol and scheduling algorithm are evaluated thoroughly. The simulation results demonstrate that proposed DRAST routing method and SDRR scheduling strategy in this dissertation have outstanding performance.

As a summary, the research work presented in this dissertation build a foundation for the implementation of routing and scheduling in the LEO satellite networks.

**Key words:** LEO satellite networks; routing protocol; scheduling algorithm; handoff strategy; OPNET simulation

## 目 录

摘 要 .....	I
Abstract .....	II
目 录 .....	III
第一章 绪论 .....	1
1.1 研究背景及意义 .....	1
1.2 低轨卫星通信网络概述 .....	3
1.2.1 低轨卫星通信网络架构 .....	3
1.2.2 低轨卫星轨道模型 .....	4
1.3 国内外研究现状 .....	5
1.3.1 路由算法国内外研究现状 .....	5
1.3.2 调度算法国内外研究现状 .....	8
1.3.3 切换策略国内外研究现状 .....	10
1.4 研究内容 .....	13
1.4.1 路由算法设计与优化 .....	13
1.4.2 交换调度算法设计与优化 .....	13
1.4.3 切换策略设计与优化 .....	13
1.4.4 仿真验证 .....	14
1.5 论文结构安排 .....	14
第二章 低轨卫星通信网络通信协议及仿真平台设计 .....	16
2.1 低轨卫星通信网络通信协议架构 .....	16
2.1.1 透明转发模式通信体制 .....	17
2.1.2 星上处理和交换下通信体制 .....	18
2.1.3 星上处理模式通信流程 .....	22
2.2 仿真平台体系结构 .....	24
2.2.1 基于数据包的通信过程 .....	25
2.2.2 事件驱动机制 .....	26
2.2.3 有限状态机机制 .....	26
2.2.4 仿真平台中的网络模型 .....	28
2.2.5 仿真平台中的节点模型 .....	29
2.2.6 仿真平台中的进程模型 .....	31
2.3 仿真平台和通信链路的实现 .....	31
2.3.1 网络接入协议的实现 .....	31
2.3.2 卫星星际链路的模拟实现 .....	32
2.3.3 多波束天线的仿真模型 .....	34
2.4 本章小结 .....	35
第三章 低轨卫星通信网络中的路由协议 .....	36
3.1 低轨卫星星座模型 .....	36
3.2 低轨卫星通信网络路由的独特性 .....	37
3.3 近年来有效的路由策略 .....	38
3.4 DRAST 路由算法描述 .....	39
3.4.1 低轨卫星通信网络拓扑特征 .....	39
3.4.2 低轨卫星通信网络拓扑的简化 .....	41

3.4.3 路由策略参数分析.....	42
3.4.4 DRAST 路由策略的执行.....	50
3.5 仿真设计和结果分析.....	52
3.5.1 仿真模型设计.....	53
3.5.2 仿真结果.....	58
3.6 本章小结.....	62
第四章 低轨卫星通信网络中的分组调度算法.....	63
4.1 星载交换机结构.....	63
4.1.1 交换机结构概述.....	63
4.1.2 星载交换机模型.....	65
4.2 典型的星载交换机调度算法.....	66
4.3 SDRR 调度算法.....	68
4.3.1 DRR 算法描述.....	68
4.3.2 SDRR 算法描述.....	69
4.3.3 调度算法仿真模型.....	71
4.3.4 SDRR 算法仿真结果及分析.....	73
4.3.5 SDRR 在星载 CICQ 交换机中的执行.....	76
4.4 星载交换机 SDRR 调度算法仿真.....	77
4.4.1 星载交换机仿真模型.....	77
4.4.2 调度算法与路由策略的结合.....	79
4.4.2 仿真结果分析.....	79
4.5 本章小结.....	81
第五章 低轨卫星通信网络中的切换策略.....	83
5.1 切换执行.....	83
5.2 切换测量.....	85
5.3 权值优化切换策略.....	90
5.4 马尔可夫链分析.....	91
5.5 仿真结果.....	96
5.6 本章小结.....	98
第六章 全文总结.....	99
参考文献.....	101
作者攻读博士学位期间发表的论文.....	108
致 谢.....	109
作者简历.....	110

## 第一章 绪论

卫星通信, 简单地说, 就是指地球上(包括地面、水面和低空域)的无线电通信站(包括固定台站、车载、舰载、机载、手持终端等)之间利用人造地球卫星作中继站而进行的通信。数十年来, 随着卫星在通信、Internet、导航、广播、电视等方面的发展, 尤其是蜂窝通信技术、Internet 技术、多媒体技术的飞速发展, 卫星通信也取得了长足的进步, 涉及的业务领域和频段不断拓宽。由于具有通信距离远、费用与通信距离无关、覆盖面积大、不受地理条件限制等一系列特点, 卫星通信已经成为重要的现代通信手段之一, 在全球通信中大展身手, 特别是军事领域, 卫星以独一无二的优势, 得到世界各国的高度重视。正在蓬勃发展的低轨道(LEO, Low Earth Orbit)卫星通信网络系统即是其中一个重要的组成部分<sup>[1-14]</sup>。

### 1.1 研究背景及意义

卫星网络系统是由不同层次、不同轨道上多种类型的卫星系统按照空间信息资源的最大有效综合利用原则有机构成的智能化体系。卫星从轨道类型可以分为 3 种: 低轨道(LEO)(距地面数百至 1500km, 运行周期 2~4 小时); 中轨道 (MEO) (距地面 5000~20000km, 运行周期 4~12 小时); 静止轨道 (GEO) (距地面 35800km, 运行周期为一个恒星日, 即 23 小时 56 分 04 秒)<sup>[2]</sup>。

静止轨道 (GEO) 卫星移动通信系统技术成熟, 具有卫星数量少、全球覆盖、24 小时通信不必切换卫星、卫星跟踪控制简单等优点, 但也存在轨道高、传播路径远、衰减大、延时长等不足。中、低轨道卫星属于非静止卫星, 对实现全球个人移动通信显示出极大的优越性。LEO 的传播损耗通常比 GEO 少几十 dB, 具有传播延时小, 发射功率小等优点, 但需要部署大量的卫星以实现全球覆盖, 组网技术和控制切换等比较复杂。

表 1-1 现有的 LEO 卫星系统

组 织	卫 星	类 型	转 发 器	轨 道 位 置
Globalstar, San Jose, CA, USA	48颗LEO卫星, 其中4颗在轨道中闲置	移动通信, 卫星电话, 全数字	投影范围内16个点波束。每个波束包括1~13个1.25MHz多信道。采用CDMA多址接入。移动终端采用L和S频段链路	6个52度倾斜轨道平面, 每个高度为1413km的轨道平面含8颗卫星
Iridium LLC, Washington, DC, USA	66颗LEO卫星, 其中7颗在归到中闲置	移动通信, 卫星电话, 全数字	48点波束。8MHz频带内7个RF信道。移动终端采用L频段链路。22GHz卫星交互链路。采用TDMA/FDMA多址接入。	6个84.6度倾斜轨道平面, 每个高度为898km的轨道平面含11颗卫星
Orbcomm Global L.P., Dillies, VA, USA	28颗LEO卫星	面向手机和移动终端的数据传输	带地面覆盖波束弯管转发器。0.1秒脉冲VHF链路中数据传输速率最高达2400bps (上行链路148MHz, 下行链路137MHz)	45度倾斜轨道平面含24口卫星。2颗位于70度倾斜轨道, 2颗位于108度倾斜轨道

LEO 卫星通信网络是研究和发展的热点，这是因为：首先，空间的星座网络可以减少对地面的依赖型，并实现全球覆盖；其次，它们的时延小，往返时延为几十毫秒以下量级；最后，低功率用户可以直接接入系统，更易向用户提供全球漫游业务<sup>[15]</sup>；此外，多颗卫星按一定规则组成的星座可实现真正的全球覆盖，而且具有更高的频率复用率；同时，星上处理、转发、蜂窝通信、多波束、频率复用等技术的日渐成熟也为低轨道卫星移动通信提供了必须的技术保障。典型的 LEO 系统有 Iridium、Globalstar 和 Teledesic 系统，描述如表 1-1 所示。

低轨卫星系统中卫星之间既可以通过星际链路互连在一起，也可以通过地面中继实现相互之间的互连。Globalstar 系统中无星间链路，卫星对信号不做任何处理。Iridium 系统中卫星采用星上处理和交换技术，通过星际链路（ISL）提供通信路径，用于传送信号、提供路由和转发数据包<sup>[16]</sup>。星上处理和交换技术作为通信星座系统与卫星有效载荷的核心技术，用于完成星座系统空间段的信息交换和组网功能，对它展开研究，可以提高系统容量、通信质量、网络连通性和抗干扰能力等，适应业务量增长的需求，保证服务质量。

星上处理主要实现路径计算和根据带宽分配及复用传输分组的功能。星上处理虽然带来卫星的复杂性，但系统能更加自由的进行通信，能自主传递数据或语音业务，真正实现任何地点、任何时间以及任何人的无缝通信，其优越性不言而喻。星上路由系统和交换系统是星上处理和有效载荷的核心，担负着星间数据链路的传输、分发和各种业务的 QoS 保证等功能，是卫星网络技术研究的关键环节。有效的路由策略可以降低分组传输时延、丢包率、避免网络拥塞以及降低卫星的复杂度；合理的分组调度策略能降低分组在卫星节点的处理时延、降低对卫星缓存区容量的要求以及有效利用卫星无线带宽资源。因此，开展卫星网络路由协议和调度算法及其优化理论的研究非常必要。

卫星通信网络与地面网络在特征上有着明显的区别，这是导致能很好运行于地面网络路由协议、分组调度算法以及切换策略不能适用于卫星网络的主要原因。卫星网络和地面网络的主要区别有：

- 相对地面无线网络，卫星网的传播时延较大。
- 卫星网络的拓扑结构无时无刻不在发生变化，而地面网络一旦建成其拓扑结构基本上是静态不变的。星座运行有规律性，卫星网络的拓扑结构呈周期变化，具有可预测性。卫星网络节点的数目基本固定。
- 低轨卫星移动很快，因此，通信用户经常在同一颗卫星不同波束覆盖区范围内，导致用户链路波束切换；如果卫星的移动导致正在通话用户位于不同卫星覆盖区域，则产生卫星切换。同理，星间链路的时变性也会导致星间链路发生切换。
- 星地链路信道衰落大，且具有多径效应以及多普勒频移；信道误码率高，且



卫星链路是典型的非对称网络。

针对这些区别,本课题的研究意义在于解决路由协议、分组调度以及切换策略中的技术难点,在实时动态变化的卫星网络拓扑前提下,实现较少的拓扑信息交互和较低的路由计算量,保证路由协议可靠有效的运行;在星上条件(处理能力、存储量和带宽资源)受限以及星上交换机端口繁多且不对称的情况下,保证及时高效地调度各业务分组;在卫星网络拓扑频繁的变化中提供尽可能好的链路切换。

## 1.2 低轨卫星通信网络概述

### 1.2.1 低轨卫星通信网络架构

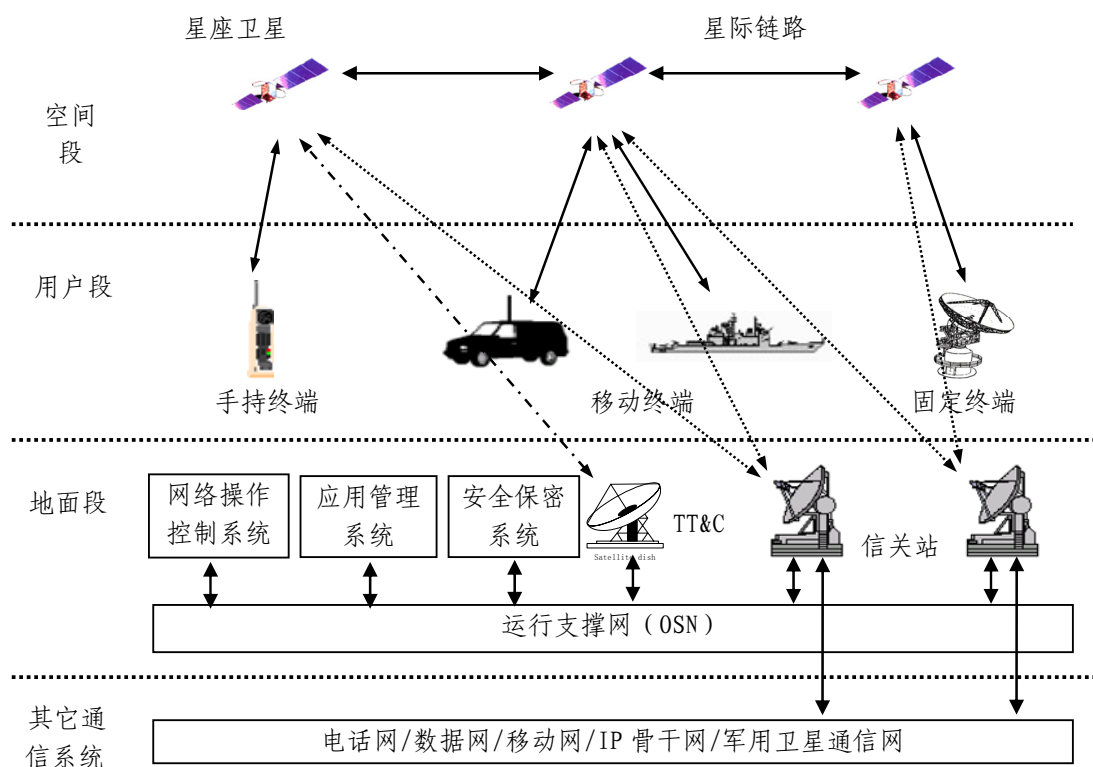


图 1-1 低轨卫星网络的组成

图 1-1 给出了低轨卫星网络的基本组成示意图。从图中可以看到,低轨卫星网络由地面段、用户段和空间段组成<sup>[4]</sup>。空间段由多颗星座卫星组成,实现对全球或部分区域的连续或间断覆盖;用户段是指各类用户终端,包括手持机、车(机、船)载站、航天载体终端、便携站和固定站等;地面段负责对空间的卫星网和用户段的控制,包括信关站、网络操作控制系统、应用管理系统、安全保密系统、遥测遥控站(TT&C)以及连接它们的运行支持网络等。信关站完成信息路由分配、用户管理、移动管理、接入有关数据库和与地面网的接口,一般都接有访问位置登记器和住地位置登记器。

地面段其余系统控制卫星星座的正常运行，管理各个信关站的资源，管理所有用户并为用户提供各种接入服务以及各种形式的应用。

### 1.2.2 低轨卫星轨道模型

所谓星座是指相互之间具有特定工作关系的一组（群）卫星，它们为了完成同一个目的而协同工作。通常，一个卫星星座包含多颗处于相同或不同轨道的卫星，这些具有类似功能的卫星在统一控制下完成单颗卫星难以完成的任务。低轨卫星星座的轨道模型分为极轨道星座和 Walker 倾斜轨道星座<sup>[17]</sup>。

极轨道星座的特征：

- 卫星轨道平面倾斜角度接近 90 度，铱星系统是极轨道星座的典型实例，其轨道倾角为 84.6 度；
- 轨道平面内卫星等间隔分布，相邻卫星之间的角度差为  $2\pi/S$ （假设  $S$  是每个轨道内卫星的个数）；
- 相邻轨道平面之间的夹角为  $\pi/S$ ；
- 相邻轨道中的卫星相互偏移，以提供无缝的地面覆盖，相同旋转方向平面上相位偏移为  $\pi/S$ ；
- 为了连续覆盖相反旋转方向的轨道，轨道平面之间的角度间隔较小。

极轨道星座中，每颗卫星与同轨道面前后两颗卫星形成两条轨内的链路（同轨星间链路），与左右两个相邻轨道面的两颗卫星形成两条异轨星间链路。极轨道星座各卫星轨道面以相同的方向围绕地球旋转，最后一个轨道平面与第一个轨道平面相邻区域的卫星运行方向相反（称作缝）。接缝处之间无异轨星间链路，因此，极轨道星座对于地面站而言不是均匀分布。

Walker 星座是由 J.G.Walker 研发的一种星座，由  $N$  颗卫星在  $P$  个相同倾斜卫星轨道平面组成、能够全球覆盖的卫星星座。Walker 星座不存在像极轨道星座那样的缝。Walker 星座由以下参数描述：

- $S=N/P$  表示每个轨道面卫星的个数， $F$  表示 Walker 星座的相位因子 ( $F=1,2,\dots,P-1$ )， $F$  确定了相邻轨道平面上卫星之间的偏离角度  $\Delta w_f = 2\pi F/N$ ；
- 倾斜轨道平面的上升节点沿着赤道等间隔排列；
- 倾斜轨道平面之间具有固定的平面偏移  $\Delta\Omega = 2\pi/P$ ；
- 每个倾斜轨道平面内卫星以  $\Delta w = 2\pi/S$  间隔均匀分布。

Walker 星座通常采用 Walker 符号  $N/P/F$  表示。同样，Walker 星座每颗卫星一般维持两条同轨星间链路和两条异轨星间链路，卫星对于地面站而言完全均匀分布，而且仅仅通过异轨星间链路可以遍历所有的卫星。这个特性是极轨道星座不具有的。

现有的文献主要以极轨星座为模型研究卫星网络。本文结合课题，以 24/3/1

Walker 星座模型为平台研究低轨卫星通信网络的路由策略和交换调度算法。

### 1.3 国内外研究现状

#### 1.3.1 路由算法国内外研究现状

路由策略需要实现三部分功能：使用网络状态信息、计算路径和传递用户信息。网络状态信息包括链路的传播时延、星载交换机缓存区占用状态、各星间链路信道占用情况以及异轨星间链路的连接状态等。

已提出的 LEO 卫星网络星间路由策略主要是基于卫星网络的系统周期分割机制和覆盖区域分割机制设计。

##### 1. 系统周期分割机制

根据卫星网络运行的周期性，把卫星轨道周期  $T$  划分为  $n$  个时间间隔，即  $[t_0, t_1), [t_1, t_2), \dots, [t_{n-1}, t_n] = T$ ，并具有以下特点：

- A) 异轨星间链路的连通和断开仅发生在离散时间点  $t_0, t_1, \dots, t_n$ ，在时间间隔  $[t_{i-1}, t_i)$  内，认为卫星网络拓扑结构固定不变；
- B) 时间间隔  $[t_{i-1}, t_i)$  较小，在时间间隔周期内认为所有星间链路都是固定不变（即链路长度不变）。

Werner 提出的基于 ATM 的路由<sup>[18]~[19]</sup>，Chang 提出的基于 FSA 的路由<sup>[20]~[21]</sup>，Gounder 提出的基于快照的路由<sup>[22]</sup>都属于基于系统周期分割机制的路由协议，这些算法主要考虑时延约束选择最佳路径。

文献<sup>[23]</sup>提出概率路由协议（Probabilistic routing protocol），以最小化重路由和切换概率为目标计算路径。文献<sup>[24]</sup>提出一种分布式路由策略，采用面向连接的交换方式，为降低通信过程中链路切换次数，以星间链路的生存时间以及总的时延作为选择路径的衡量标准。路由中需要实时传递链路状态信息到邻近卫星，建立连接时的路由过程类似于泛洪路由，容易带来环路和路由死循环。

文献<sup>[25]</sup>选择流量较低的链路作为最佳路径，此算法能有效避免拥塞，并最大化利用剩余的带宽资源。文献<sup>[26]</sup>和<sup>[27]</sup>采用集中式路由策略，在分割好的静态拓扑下采取 Dijkstra 算法计算路径，算法中以传输时延和队列时延作为衡量链路的标准，因此，网络需要周期性传递各星间链路的状态信息。频繁的网络状态变化导致引入不可接受的花费，同时较高的传播时延带来最佳路径的计算误差。

文献<sup>[28]</sup>提出在路由中考虑分组的调度，并比较采用不同调度策略的性能，结果显示如果卫星有一定的缓存空间，不同调度策略带来相似的系统吞吐量及时延性能。

文献<sup>[29]</sup>中 Traffic Class Dependent (TCD) 路由策略被提出，为满足不同类型业务的 QoS 要求，选择路径时以不同的参数为优化准则：譬如星上队列和处理时延、星间链路长度、卫星业务的负载或者可用带宽资源等。TCD 路由策略能满足业务 QoS 要求，但路由开销较高。

文献<sup>[30]</sup>提出 SRED (Satellite Routing for end-to-end delay), 路由时采用 WFQ (Weighted Fair Queuing) 调度策略, 其带宽分配取决于业务要求和网络可用资源。此算法牺牲系统的吞吐量保证业务的时延性能。文献<sup>[31]</sup>提出 HPSR (High Performance Satellite Routing) 办法, 与 SRED 不同在于分配的带宽低于数据的平均速率, 能获得较高的吞吐量。上述两种路由策略计算复杂, 星上需要较强的处理能力。

文献<sup>[32]</sup>提出修正业务偏离算法, 针对不同业务选择不同的路径, 避免链路拥塞, 由于没有根据全网络状态计算路径, 其选择的路径不是最优路径。文献<sup>[33]~[34]</sup>提出的路由策略类似于<sup>[32]</sup>, 选择偏离源卫星节点或最短路径中的节点的路径, 抑制了高纬度异轨链路拥塞状态。文献<sup>[35]~[37]</sup>中在源和目的卫星之间计算得到多条路径, 各业务选择不同的路径传递, 此策略导致分组的无序接收。

文献<sup>[38]</sup>在路由时设置时间门限, 初始化选择的路径低于此门限值, 后续路由时根据业务请求采用逐步更改链路方法优化路径, 但终保持所选路径的时延低于门限值。文献<sup>[39]</sup>和<sup>[40]</sup>应用距离向量路由算法于卫星网络, 一种为 Darting 算法, 另一种为 Bellman-Ford 算法, Darting 算法性能较好, 但数据分组头部需要储存路由信息。

基于系统周期分割机制的路由协议优点是:

- 把动态拓扑结构规约成按时间段分隔的一系列连续的静态拓扑结构, 从而在静态拓扑结构的基础上利用现有技术计算路由;
- 路由计算大多由地面预先集中处理, 卫星不需要实时计算, 仅需在时间分隔点更新路由表;
- 路由协议的开销小。

缺点是:

- 由于系统周期可能分隔成大量的时间段, 因而需要大量的存储空间存放路由信息, 需要考虑路由信息的存储问题;
- 对通信流量变化、链路拥塞以及故障等实时情况的适应性较差。

该类算法的关键是: 适当划分系统周期和抽取网络拓扑结构, 同时在大运算量、精确性和存储空间大小等因素之间存在折衷。

## 2. 覆盖区域分割机制

将地球表面覆盖域划分成不同的区域, 并给各个区域赋予不同的固定的逻辑地址。其特点是: 在给定时刻, 对于最靠近区域中心的卫星, 其逻辑地址就是该区域的逻辑地址; 卫星在运行过程中根据覆盖区域的变化, 其逻辑地址也动态变化; 逻辑地址带有地面节点的地理位置信息。

而这些区域的固定点即为虚拟节点, 这些虚拟节点所构成的虚拟网络就嵌入在卫星星座上。也就是说, 在任意时刻, 任何一个虚拟节点都有一个卫星来体现。当卫星移动时或地面终端用户进行通信切换时, 虚拟节点上的状态如路由表、信道分配等信息在物理节点(卫星)之间进行移交, 这些虚拟节点就在星座之间交换, 从而完成了

路由表的交换。

Hashimoto 提出了基于 IP 的路由<sup>[41]</sup>；Ekici<sup>[42][43]</sup> 提出了分布式路由，执行无状态的分配式路由协议，此策略不会带来额外的处理，基于传播时延选择路径，仅仅当发生链路拥塞后把分组转移到备份的路径上，此协议适用于倾角为90度的卫星系统，不能满足业务的QOS需求。Mauger 提出的基于 TDMA 的路由<sup>[44]</sup>，Berndl 提出的虚拟拓扑路由<sup>[45]</sup>，都属于覆盖区域分割机制。文献<sup>[46]</sup>提出把卫星网络融入地面Internet网络中，通过卫星IP地址进行路由。文献<sup>[47]</sup>提出的基于地理位置进行分组路由，是一种选择最短路径的分布式路由策略。

文献<sup>[48][49]</sup>提出三种静态路由策略：Dogleg、Parallel Highways和Polar Hop，路由中不考虑各链路状态以及业务的分布情况，在固定的星间链路拓扑图中计算路径。上述三种算法仅仅使用地理位置计算路径，容易导致拥塞，且传播时延较大。

文献[50]和[51]应用虚拟节点提出类IP路由策略，克服了卫星网络高度动态性。文献[52]把动态的卫星网络定义为静态的逻辑拓扑表，与分割地理区域不同的是此算法根据卫星在空间的经纬度位置划分逻辑节点，每一个固定经纬度对应具体的一颗卫星（称逻辑节点）。当卫星移动时，在同一轨道面的后继卫星替代前一颗卫星，每颗卫星在某时间段对应某一逻辑节点。根据星间链路长度计算最短路径，并对长度进行误差校正。

基于覆盖区域分割机制的路由算法的优点是：

- 利用网络拓扑结构的规则性和周期性，隐藏卫星的移动性；
- 源/目的地址信息中包含有地理位置信息；
- 无需预先进行路由计算，根据路由策略以及流量负载、故障等情况实时选择路径，自适应性强；
- 需要的存储空间比较小。

缺点是：

- 要求网络拓扑结构非常规则；
- 路由计算利用局部状态信息，路由不一定最优化；
- 路由计算利用全局状态信息，需要大量带宽资源传递状态信息，路由协议开销大。
- 对星上处理能力要求较高。

另外，基于拓扑依赖策略的路由算法是一种针对具有特定拓扑特征的算法。基于这种策略的路由算法对于卫星网络的拓扑结构以及卫星的移动特性具有认知能力，通过不断分析当前网络的拓扑结构特点采用不同的路由策略。这种算法是严格针对某一特定类型的卫星星座网，所以一般对于不同的卫星网络要分别设计不同的算法。覆盖域切换重路由协议（Footprint Handover Rerouting Protocol, FHRP）<sup>[53]</sup>（应用于 Walker Star Constellations）就是基于拓扑依赖策略的一种路由协议。这种算法减少了通信和

处理开销，但并不能保证路径的最优性。文献[54]提出了卫星切换动态概率路由优化策略，为了达到对网络资源的有效使用，当卫星切换时考虑带宽、业务分布等参数优化并重建原路径。

文献[55]假设单层 LEO 星座为双层星座，根据星际链路动态特征和流量分布，在各轨道面选择某卫星节点作为此轨道面的管理中心，在低轨卫星网络中采用双层星座的路由策略。文献[56]基于地理地址提出分布式路由策略，选择最少跳数的路径，路由中没有考虑星间链路状态信息。

### 1.3.2 调度算法国内外研究现状

分组调度算法在分组数据传输的基础上提供带宽分配和复用，决定交换机性能，其主要通过时延、公平性、算法复杂度、吞吐量、各个业务流的隔离度和带宽利用率来衡量。

1) 时延：即分组在星载交换机中等待的时间，为保证端到端的时延必须降低分组队列时延，并保证其不受其余业务的影响。为避免分组丢失，较高的队列时延需要较大的空间缓存分组，因此，好的时延性能不仅影响业务 QoS 的要求，也意味着对缓存区容量的要求降低，这对资源受限的星载交换机来说非常重要。

2) 公平性：保证各业务流根据分配的带宽资源获得相应的服务，避免某些业务一直处于饥饿状态（分组不能得到服务），而某些业务一直占用信道获得服务。譬如，两个业务流  $i, j$  在时间段  $(t_1, t_2)$  总共接收的服务分别为： $W_i(t_1, t_2), W_j(t_1, t_2)$ ， $r_i, r_j$  分别表示分配服务流  $i, j$  的最小服务速率，公平指数  $F_{i,j}$  表示业务流之间的公平性：

$$\left| \frac{W_i(t_1, t_2)}{r_i} - \frac{W_j(t_1, t_2)}{r_j} \right| \leq F_{i,j} \quad (1.1)$$

$F_{i,j}$  是常量，根据不同的调度算法有不同的表示。

3) 算法复杂度：需要执行的操作的数目以及如何选择分组进行传输的处理。星载交换机处理能力有限，需要采用复杂度低的调度算法。

4) 吞吐量：单位时间内在星载交换机中传输的分组数目，好的分组调度算法能保证队列的分组及时传输、减少拥塞、降低时延、增大吞吐量。

5) 隔离度：各业务流以及新到达的业务分组不会影响正在传输的分组和正在等待传输的业务。

6) 带宽利用率：保证星上有限的带宽资源被充分利用，如果某链路信道出现错误，其带宽可以被其余业务所用。

调度算法大大影响星载交换机的性能。由于卫星传播时延大，现有的星载交换机很少考虑到交换和处理时延，大多采用地面已有的分组调度策略，很少考虑到卫星的实际情况，不能充分利用星上有限资源，时延和公平性能不高。星上应用的调度算法

广泛分为以下几类：类 GPS（Generalized Processor Sharing）调度算法、最大匹配（极大数目或权重）调度算法和轮询调度算法。

### 1. 类 GPS 算法

类 GPS（Generalized Processor Sharing）调度算法也称为分类优先级调度，维持一个全局变量：系统虚拟时间，分组到达时基于系统虚拟时间计算其虚拟完成时间，按照完成时间的递增顺序被服务。

最有名的这类算法有 WFQ<sup>[57]</sup>、WF<sup>2</sup>Q<sup>[58]</sup>、VC<sup>[59]</sup>、SCFQ<sup>[60]</sup>、LFVC<sup>[61]</sup>和 SPFQ<sup>[62]</sup>。它们的区别在于采用不同的方法计算、更新系统虚拟时间。类 GPS 算法的时延性能、公平性较好，但算法复杂度较高。其复杂度由两个因素组成：系统虚拟时间（ $VT$ ）的计算和选择下一个被服务队列的分组。后者的复杂度固定为  $O(\log N)$ 。WFQ 和 WF<sup>2</sup>Q 中计算  $VT$  复杂度都为  $O(N)$ ，SCFQ、VC 和 LFVC 的复杂度降低，但算法性能变差。

### 2. 极大匹配算法

极大匹配算法的思想在于分配各输入端口的分组前往不同的输出端口，匹配的端口对数目越多，通过率越高，分组平均时延越小。对于均匀业务，即每一输入端口到达的分组去往不同的输出端口的概率相等，通过增大匹配的数量可以提高通过率，降低时延。对于非均匀业务，必须考虑分组到达及去向等特性，通过对端口赋予不同权重来体现，端口匹配追求权重最大。

实际中，通常采用极大数目匹配，通过依次建立局部最大逼近全局最大，而后续匹配不能拆除前面建立的匹配，所以建立局部最大匹配的顺序决定了算法的性能。为充分利用带宽，算法尽力在每个时隙段最大化输入与输出连接的数目。为了保持交换机中高的吞吐量、避免非空队列长时间不能接收服务而导致的饥饿现象、以及保持快速交换和简单执行，陆续提出了 PIM<sup>[63]</sup>、RRM<sup>[64]</sup>、iSLP<sup>[65]</sup>和 FIRM<sup>[66]</sup>等不同算法。

另一种为权重匹配法，设置端口不同权重调整其在竞争中的地位，使某些端口获得更多的服务机会，适应非均匀到达的业务。权重匹配法的性能取决于权重和端口匹配算法两方面。权重的构造影响和支配着端口匹配过程中端口的选择及顺序，而匹配算法又决定最后得到的匹配权重，两者互相影响，互相作用。

目前权重匹配多采用迭代法，对应选择权重最大的端口，称为加权迭代法，例如 LQF<sup>[67]</sup>、OCF<sup>[68]</sup>、iLPF<sup>[69]</sup>和 LIPF-TM<sup>[70]</sup>等算法，复杂度均为  $O(N^2)$ 。Tabatabase 等人提出了复杂度为  $O(N)$  的 MSM<sup>[71]</sup>（maximal sorted matching）算法，先对所有端口的权重排序，再用简单的排序进行极大匹配。

由于存在输入输出两类竞争，权重的构造应该同时考虑到输入端口和输出端口两者的状态和要求。现有算法多数只采用反映输入端口状态的权重，称之为单边权重。Tabatabase 等人以输入端口中等待传输的分组数和输出端口需要接收的分组数为权重，称之为双边对称权重。为到达最佳吞吐量以及降低时延，一般采用对称权重，但

算法复杂。星载交换机中,对应输入输出端口有不同类型的多条链路,各链路对应的带宽以及传输速率不同,因此,端口对应的参数变化较大,权重匹配调度算法不适用于星载交换机。

### 3. 轮询算法

基于轮询的调度算法由于计算复杂度低而备受重视,尤其对于高速、宽带的卫星网络交换机。轮询的调度算法属于维持运作(work-conserving)的算法,work-conserving表示一旦有分组服务器就会传输此分组。维持运作的轮询调度不仅执行简单,且能有效利用卫星网络带宽资源;但没有考虑交换机输入输出端口状态,其延时和公平性能不如类GPS,因此,为了提高时延性能许多研究者提出了不同的轮询调度算法,较经典的有WRR<sup>[72]</sup>、ERR<sup>[73]</sup>、JVCRR<sup>[74]</sup>、RR-AF和RR-FQ<sup>[75]</sup>、CRR<sup>[76]</sup>、DRR<sup>[77]</sup>、Aliquem(active list queue method) DRR<sup>[78]</sup>、MCRR<sup>[79]</sup>和HMCRR<sup>[79]</sup>(hierarchical MCRR)等。轮询调度策略以固定的顺序按照预分配的带宽循环服务各业务流,各算法的区别在于调度不同业务流时如何分配其可用的带宽资源。表2-1比较几种经典调度算法的性能。

表 1-2 各种算法性能和实现的比较

算法类型	虚拟时间 复杂度	选下个分组 复杂度	全局复杂度	时延 界限	公平性	最坏情况 公平性
GPS	理想(不实际)	理想(不实际)	理想(不实际)	0	公平	很好
WFQ	$O(N)$	$O(\log N)$	$O(N)$	小	公平	差
WF <sup>2</sup> Q	$O(N)$	$O(\log N)$	$O(N)$	小	公平	好
VC	$O(1)$	$O(\log N)$	$O(\log N)$	小	不公平	差
LFVC	$O(1)$	$O(\log \log N)$	$O(\log \log N)$	小	公平	好
SCFQ	$O(1)$	$O(\log N)$	$O(\log N)$	大	公平	差
DRR	NULL	$O(1)$	$O(1)$	大	公平	差
WRR	Null	$O(1)$	$O(1)$	大	公平	差
JVCRR	$O(1)$	$O(1)$	$O(1)$	小	一般	差

### 1.3.3 切换策略国内外研究现状

在低轨卫星网络中切换分为两大类:链路层切换和网络层切换。链路层切换包括卫星切换、波束切换和星间链路切换。

波束切换指在用户通信过程中,由于卫星的移动造成用户从卫星的一个覆盖区切换到另一个临近的覆盖区。这种切换由于不涉及卫星话音通道路由的变化,相对比较简单,但由于卫星波束覆盖范围小,发生频率高,大概1~2分钟会发生一次波束切换。当通信用户从一颗卫星移动到另外一颗卫星需要发起卫星切换。由于卫星的改变会影响正在进行会话的路由以及卫星资源分配,因此为保证QOS要求,在卫星切换中还需考虑连接的动态路由算法以及切换呼叫的接入控制。异轨星间链路在某些时间段处于关闭状态,利用这些星间链路正在进行的通信需要转移到其余的星间链路,由此造成星间链路切换。这种切换如果处理不当,会引起会话阻塞以及信号中断。



当通信端（卫星或者用户）因为卫星覆盖区域的移动或用户终端移动，改变其网络地址（在 LEO 系统中存在不同网络）时，需要发起网络层切换使高层协议的连接转移到新的网络地址。

本文主要研究与路由密切相关的链路层切换策略。

成功的切换策略能最小化或者有可能消除终止正在进行的通话，通常以强迫中断概率  $P_f$  来衡量。虽然从用户的角度来看，最小化  $P_f$  很重要，而新呼叫的阻塞概率  $P_b$  也是衡量网络性能的重要参数。合理的切换算法应该在  $P_f$  和  $P_b$  两者之间取得均衡。低轨卫星网络中的几种典型的切换策略如下所述。

### 1. 改进型移动 IP 切换算法

文献[80]中借鉴地面移动 IP 于 LEO 卫星网络，并在地面建立数据库保存并更新卫星与用户节点的移动模型，以及记录用户的通信状态。当某用户发起切换时根据记录执行卫星或波束的切换。虽然切换执行简单，但需要庞大的数据库记录不同的通话模式以及移动模型。

### 2. 保证切换策略

Maral et al<sup>[81]</sup>提出一种保证切换策略，只有当前波束，以及随后需要切换到的邻近波束同时有空闲信道时，才接入新呼叫，且当第一次切换发生时，同时对随后要切换的波束发起信道预留要求，假如波束中信道都繁忙，切换请求在 FIFO 队列中排队等候。过早发起请求会占用信道资源，导致新呼叫阻塞率很高。针对此缺点，几种改进的保证切换策略提出。

#### 1) 弹性信道锁定切换 (ECL Elastic Channel Locking)<sup>[82]</sup>

在此算法中，对于正在进入的呼叫并不立刻对下一个波束发起信道锁定要求，延迟一段时间  $T_a$  后才发起请求。而延迟时间的确定根据切换失败概率确定，并结合考虑系统模型、业务模型以及新呼叫阻塞概率和呼叫中断概率的平衡。

#### 2) 基于时间信道预留算法<sup>[83]</sup> (TCRA Timed based Channel Reservation Algorithm)

基于时间预留信道是动态变化的信道锁定切换。为了提高保证切换性能和有效利用资源，其发送信道预留的请求时刻  $T_a$  根据估计在当前波束用户的位置信息计算，此算法计算量较大。

#### 3) 动态多普勒测量切换优先算法<sup>[84]</sup> (DDBHP Dynamic Doppler Based Handover Prioritization)

DDBHP 算法定义一个时间门限  $t_{TH}$ ，当用户在波束中逗留的时间到达门限时，发起切换。为了完成切换，必须准确地知道用户的位置。在 DDBHP 中根据多普勒频偏找到用户在波束中的具体位置，从而能准确地发起切换请求。时间门限值的确定取决于系统中波束尺寸、每个波束的信道数目、业务强度以及各种业务类型等可变因素。此算法无法根据系统变更时间门限值。

以上几种算法的比较如表 1-3 所示。

表 1-3 保证切换算法的比较

标准	ECL	TCRA	DDBHP
保证度	时间变量 $T_a$	时间变量 $T_a$	时间变量 $t_m$
$P_b$	随着 $T_a$ 减少而增加	依赖于某个区域用户的数目	依赖于时间变量 $t_m$
$P_f$	随着 $T_a$ 增加而减少	无效值	实际可以为 0
时间变量选择标准	对切换的 QoS 要求	用户在随户波束中期望的通过时间	多普勒效应

### 3. 优先切换策略

优先切换策略主要是结合信道分配策略提高切换性能，降低掉话概率。分为以下几种策略：

#### 1) 保护信道切换策略<sup>[85]</sup>

策略中专门为切换呼叫预留信道，如果某波束中占用的信道数大于某个门限，原始呼叫请求就被拒绝。

#### 2) 排队切换策略<sup>[86]</sup>

当用户进入重叠区域时，发起对后续波束切换请求，如果后续波束内所有信道均处于忙状态，切换请求则进入队列排队。

根据切换请求在队列中的排序策略分为：先进先出策略（FIFO）<sup>[86]</sup>、最后有用时刻（LUI, last useful instant）策略<sup>[86]</sup>和基于优先度量的策略（MBPS, measurement-based prioritization scheme）<sup>[87]</sup>，后两种策略的执行复杂度远远高于前者，因此，在卫星切换排队策略中，一般采用 FIFO 的规则。

#### 3) 动态信道预留策略<sup>[87]</sup>

此策略对切换请求的管理与新呼叫都采用动态分配信道方案，在传统保护信道策略的基础上，系统为每个波束设置一个动态变化的信道阈值，根据切换呼叫和新增呼叫的到达率动态调整预留的信道数目。动态信道预留方案具有自调节功能，保证了切换呼叫的优先级和新增呼叫的公平性。而且，此算法采用波束局部的信息进行预测，动态调整信道阈值，比起采用波束之间交换信息进行预测的信道预留策略，实现上更为简单，减小了信息交换带来的不确定性。

#### 4) 保护信道+排队切换策略

结合保护信道和排队切换两种策略，既配置一定数目信道保护切换顺利进行，也对切换请求进行排队等候空闲信道，降低呼叫中断率。

### 4. RADAR（Refined Admission Detecting Absence Region）切换策略<sup>[88]</sup>

此切换策略结合呼叫接入控制策略，对新呼叫所在波束及邻近切换转移波束中所占用的信道带宽进行估计，如果满足呼叫带宽要求，则接入新呼叫，否则拒绝接纳。算法中对两个邻近波束采用时间窗的方法估计带宽占用情况。算法中当时间窗分割越细对各波束中带宽资源的占用率估计越精确，但带来大量信息的更新、保持和交互，

增加系统的复杂度。

## 1.4 研究内容

低轨卫星网络具有通信距离远、覆盖范围广、机动灵活、通信线路稳定可靠、传输质量高等特点，在信息基础设施中具有重要地位。由于卫星的移动速度快、网络拓扑周期变化，因此现有的无线网络和组网技术不适用于卫星网络，通信协议、路由协议<sup>[89]</sup>、分组调度、无线资源管理、多播协议<sup>[89]</sup>和链路切换<sup>[90]</sup>等都要针对卫星网络的特点进行设计。以下简单介绍低轨通信卫星网络在组网方面需要解决一些关键技术。

### 1.4.1 路由算法设计与优化

确定低轨卫星网络的通信协议和交换体制，针对不同类型业务设计通信流程，在此模型基础上应用分布式路由思想和区分服务理论建立多约束 QoS 路由模型。采用排队理论进行优先级排队策略的分析，对于较为复杂的端到端时延参数，采用排队理论建立多优先级模型，给出求解方法，分析其性能。

在路由协议设计中，利用星座拓扑变化的周期性、规律性和可预测性，采用时隙划分优化方法，屏蔽卫星的移动特性。根据本文研究的星座特征，对卫星网络拓扑进行简化，每个卫星节点根据简化的拓扑结构和目的地址独立进行路由计算。系统的开销得到有效的控制，使得路由策略更加灵活，更具扩展性、鲁棒性以及更好的 QoS 保证，提高了星上能源和资源的利用率。

### 1.4.2 交换调度算法设计与优化

根据卫星链路特征、信道分配、业务类型确定星载交换机结构，在此结构上结合星载交换机的硬件设备、存储设备以及处理能力设计合适的分开式调度结构和策略，有效降低调度算法的复杂度，并保持较好的时延性能和各业务的 QoS 要求。理论上，根据极限和边界情况分析调度算法的性能。在调度算法设计中，可以根据分组所选的路径分配各分组的传输带宽，有效利用网络中的剩余带宽。

### 1.4.3 切换策略设计与优化

在低轨道卫星移动通信系统中，由于卫星相对于地面的高速移动和用户终端的移动性，需要进行频繁的切换。切换成为影响系统性能的关键技术，不同的切换策略将直接影响到切换延时、切换频率、频率利用率、QoS、呼叫阻塞率以及端到端时延抖动等。在设计切换算法时，首先根据低轨卫星网络特征确定进行切换测量的准则，基于精确的测量和对性能的要求选择多个参数确定可进行切换的卫星或波束，建立多权值的函数，通过平衡各权值优化函数，选取合适的卫星或波束，结合合理的执行流程进行切换。通过理论分析权值优化切换策略带来系统中强迫中断概率和新呼叫阻塞概率的结果。同时，为降低星间链路切换次数，本文通过优化网络拓扑结构设计路由

协议, 综合考虑路由协议和切换策略的设计。

#### 1.4.4 仿真验证

结合路由协议设计、星载交换机结构和调度算法的设计和仿真技术的发展现状, 利用数学建模、统计分析的方法, 在 OPNET 仿真平台基础上, 建立网络仿真的体系框架。针对卫星网络的具体特征, 从网络拓扑结构、网络接入策略的设计、邻居卫星节点的发现方式、卫星运行轨道的自主维护、STK 导入的轨道模型、星际链路模拟、星地链路模拟和无线管道建模等方面进行研究, 实现 OPNET 环境下的路由协议仿真、星载交换机调度算法仿真以及低轨卫星网络通信流程和协议的仿真。

采用系统分析、理论计算和软件仿真相结合的方法, 对路由协议体系的收敛性、协议开销, 以及算法的 QoS 保证能力等性能进行分析和验证; 对星载交换中调度算法的吞吐量、时延、复杂度以及带宽利用率等性能进行分析和验证, 并与现有的路由协议和调度算法进行比较。同时, 对系统的通信协议、层次功能划分、通信流程、信令和业务帧等的设计的结果进行了验证。

#### 1.5 论文结构安排

本论文在十一五型号背景预研所确定的 24/3/1 构型低轨卫星网络中, 提出了一种 DRAST 路由协议和 SDRR 调度算法。主要思想是: 分割周期内卫星网络动态拓扑结构为一系列静态拓扑, 为降低星间链路的切换次数简化拓扑结构; 基于简化的拓扑设计一种简洁的分布式路由协议, 此协议能保证端到端的时延性能, 复杂度低, 路由协议开销小, 适用于资源受限的卫星网络。同时, 针对星载交换机的分组转发问题提出一种新的调度策略, 以低的处理能力高效转发各业务分组, 保证星载交换机掉包率低、无拥塞。

论文后面的结构安排如下:

第 2 章全面研究了低轨卫星网络体系和通信结构, 重点研究了卫星网络的通信协议和交换方式; 深入研究了基于 OPNET 的软件仿真平台, 实现了 OPNET 环境下 24/3/1 低轨卫星网络的通信协议和流程的仿真, 为后文研究路由协议、调度算法和切换策略建立了仿真验证平台;

第 3 章首先分析并简化卫星网络拓扑结构, 给出了基于优先级的时延参数排队模型和求解方法, 然后, 详细描述本文提出的 DRAST 路由算法, 最后在建立的 OPNET 软件仿真平台中进行算法仿真并分析结果;

第 4 章首先分析了星载交换机的结构, 基于交换机结构详细描述了本文提出的 SDRR 调度算法, 通过数学方法分析了其性能, 在 OPNET 中建立验证调度算法模型, 分别执行了 SDRR 与 DRR 两种调度策略, 比较仿真并分析结果, 最后在卫星网络仿真平台中执行 SDRR 调度算法, 验证其性能;

第 5 章首先简单描述了在低轨卫星网络中执行切换的流程, 基于位置测量提出

了适用于低轨卫星通信网络的权值优化切换策略, 并采用马尔可夫链分析了切换策略的性能, 最后在卫星网络仿真平台中简单验证切换策略;

第 6 章总结全文并提出了今后的研究工作。

## 第二章 低轨卫星通信网络通信协议及仿真平台设计

具有星上处理和交换能力的卫星就象悬挂在太空中的交换机,它可为来自不同波束的信息提供路由和交换,实现任一输入波束与任一输出波束间的通信连接,使位于多个波束覆盖区内的所有地面的用户能够相互通信,从而组成可进行星上交换的卫星通信网。在这种网络结构中,卫星是网络的交换结点,是地面信关站的空间延伸体。考虑到在卫星中使用信道需要采用时分多址和码分多址接入的方式,因而卫星网络的通信协议不同于地面网络。

本章描述了低轨卫星网络的通信协议和交换体制的设计;提出了验证通信协议、路由协议以及调度算法等性能的仿真平台。

### 2.1 低轨卫星通信网络通信协议架构

在低轨卫星通信网络中卫星担当的角色分为两种,一种为透明转发器,也称弯管式转发器,卫星对传输的信号进行简单的转发和中继,当信息由地面传输到卫星后不要恢复为基带信号,只需简单变频再转发至地面。在这种模式下,卫星处理简单,性能可靠。另一种即为本文主要研究的星上处理和交换模式。

本文研究 24/3/1 的 Walker 星座,星上具备处理和交换功能,空间段通过星间链路组成卫星通信网。系统采用典型的星间链路网络拓扑结构,每颗卫星有 4 条星间链路,其中两条为同轨星间链路,另外两条为异轨星间链路。卫星组网结构如图 2-1 所示。每一列为一个卫星轨道。在 24/3/1 的星座配置中,三个轨道的卫星形成对中国大陆地区的全方位覆盖。轨道内的星间链路相对稳定,轨道间的星间链路将随着卫星的运动而动态发生改变。

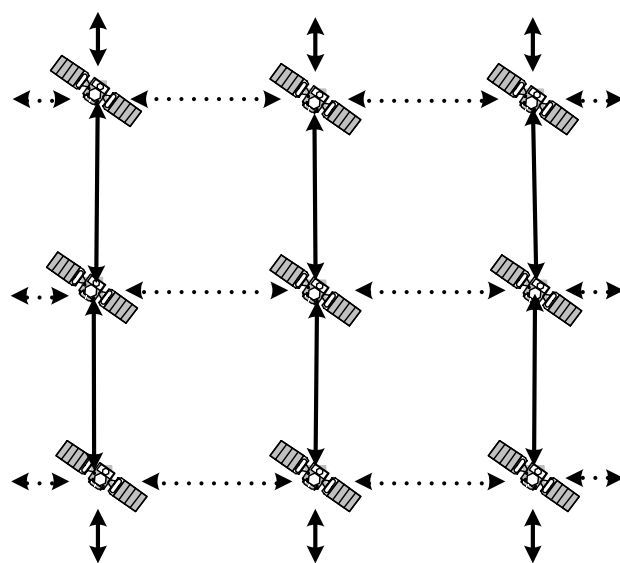


图 2-1 卫星的空中组网结构

### 2.1.1 透明转发模式通信体制

为了降低卫星的复杂度，当用户位于地面信关站覆盖区时，卫星处于透明转发工作模式，系统中用户之间不能直接通过卫星通信，所有通信都经过信关站转发，信关站在整个通信过程中起一个中央枢纽站的作用。系统中根据需要设置 5~8 个信关站，各信关站可以独立工作。信关站之间通过运行支撑网络实现相互之间的通信。其工作模式如图 2-2 所示，a 图表示用户位于同一颗卫星的覆盖范围，b 图表示用户位于不同卫星覆盖范围。

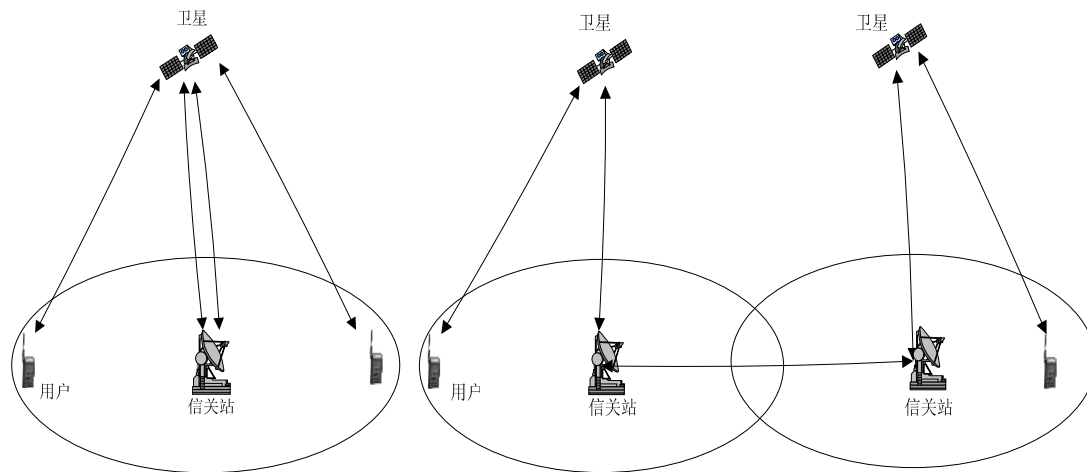


图 2-2 (a) 同一覆盖区终端之间通信

(b) 不同覆盖区终端之间通信链路

当卫星为透明转发器时，通信协议分层如图 2-3 所示。卫星上只有变频设备，数据分组未经任何处理转发到地面，由地面网络查找目的用户位置，再经过卫星转发到目的地。本文主要研究卫星网络路由算法和星载交换机调度策略，下文重点描述卫星采用星上处理和交换时通信协议的分层设计、通信流程。

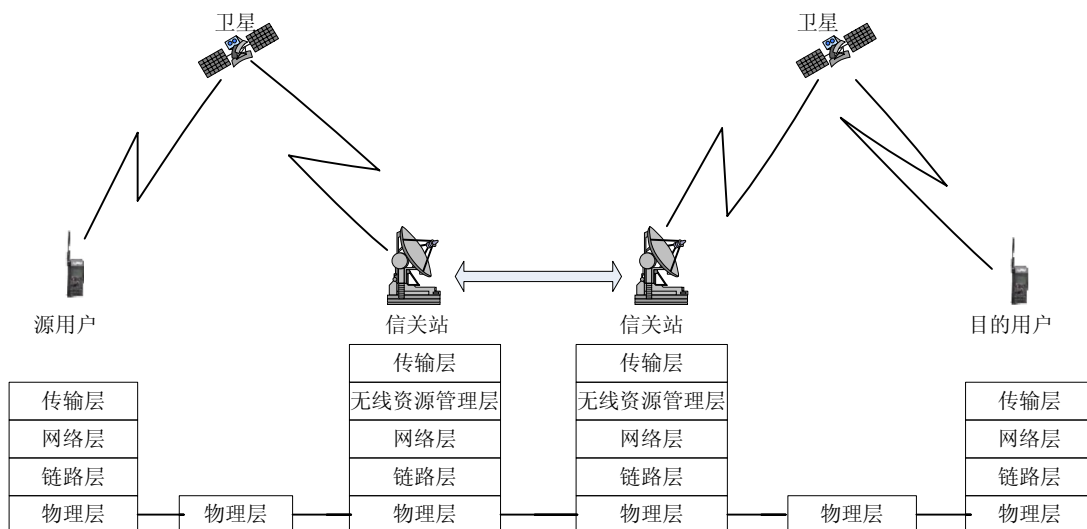


图 2-3 透明转发模式系统通信协议分层框图

### 2.1.2 星上处理和交换下通信体制

当用户位于无信关站覆盖区时，此刻与其通信的卫星也无信关站覆盖，星上启动处理和交换功能，承担部分的呼叫控制、资源分配和路由工作，通过星间链路在卫星之间传递话音和数据，并在信关站的辅助下完成用户端到端之间的通信。通信示意图如 2-4 所示，a 图表示用户位于同一颗卫星的覆盖范围，b 图表示用户位于不同卫星覆盖范围。

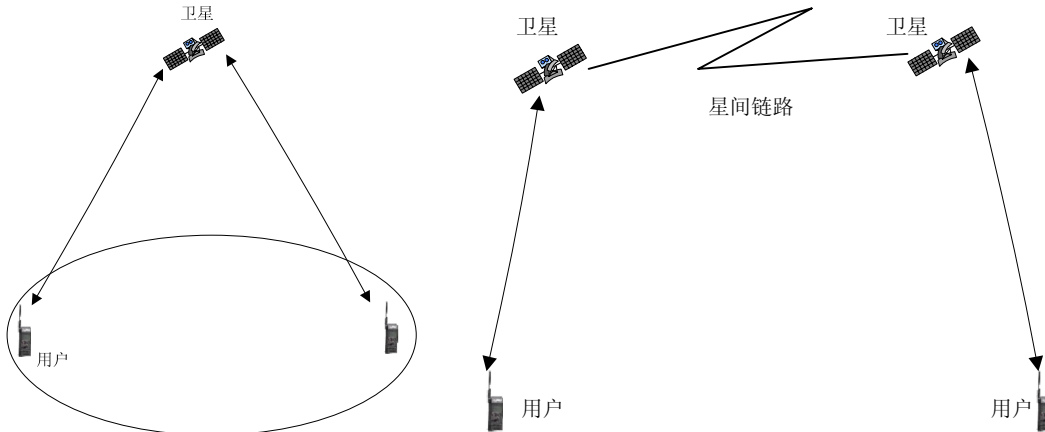


图 2-4 (a) 同一覆盖区终端之间通信

(b) 不同覆盖区终端之间通信

根据星上处理和交换技术的不同，可以把采用各种星上处理和交换技术的转发器分为三类<sup>[4]</sup>：载波处理转发器、比特流处理转发器和全基带处理转发器。图 2-5 给出这三种处理转发器与透明转发器结构框图的对比。

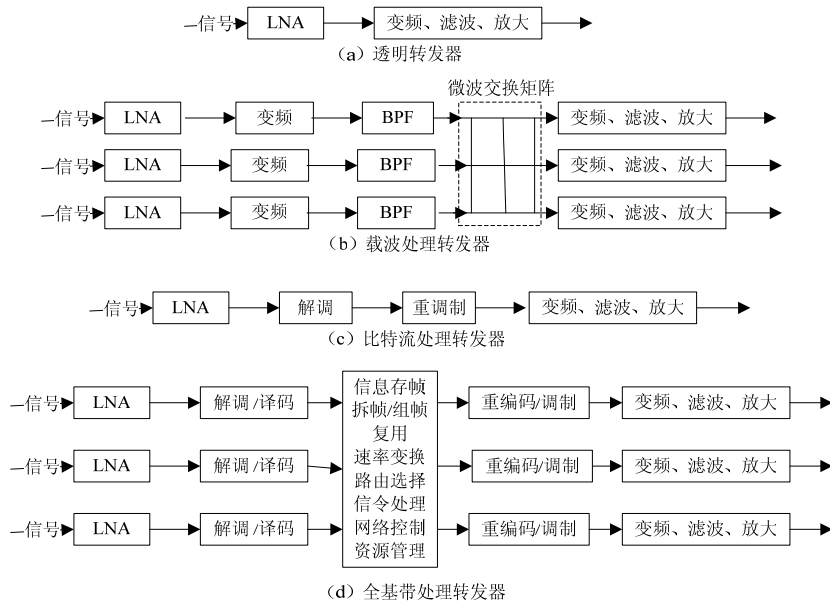


图 2-5 四类转发器结构方框图的对比

载波处理转发器最简单，以载波为单位直接对射频信号处理，不对信号进行解调和重调制，相对于透明转发器而言，差别在于星上具有载波交换能力；比特流处理转



发器是把射频信号变为中频,对信号解调,因此需要重调制,没有交换能力,但因为信号再生,能对误比特性能改善,能对卫星系统进行优化设计。星上处理转发器最复杂,此方式不仅可以在星上进行动态选路或指定处理方式,还具有信令处理能力,可以有效利用卫星资源,减少用户通信时间。本文研究的卫星通信网络中应用全基带处理模式。

考虑到低轨卫星通信网络的带宽资源以及星上处理、存储能力有限,星地距离较远,卫星网络的动态变化大,星座系统中的各种业务和信令都通过分组的形式来传送,即采用分组交换体制。

分组交换有两种方式:一种是虚电路方式,采用面向连接的工作方式,具有连接建立、数据传送和连接拆除三个阶段:在用户数据传送前先建立端到端的虚连接;一旦虚连接建立后,属于同一呼叫的数据分组均沿着这一虚连接传送;通信结束时拆除该虚连接。虚连接即逻辑连接,通过通信连接上的所有交换节点保存选路结果和路由连接关系来实现连接。另一种是数据报方式,采用无连接工作方式,在呼叫前不需要事先建立连接,而是边传送信息边选路,并且各个分组依据分组头中的目的地址独立进行选路<sup>[91]</sup>。

当低轨卫星网络备受重视之时,许多人展开了对网络中采用何种分组交换方式的研究:有的文献采用类似 ATM 的面向连接交换方式,而有的则采用类似 IP 的无连接交换方式<sup>[92~97]</sup>。卫星网络中 ATM 网络技术特点是:可适应灵活的路由和业务需求特征,实现资源高效利用;利用卫星的点对多点和多点对多点连接能力,可为大量用户提供有效的接续;可适应不同比特率用户和系统的接入。但是,这种连接必须在变化性很强的卫星网络环境中建立和维护,实现过程非常复杂,要求很高,可扩展性较差,其面临的问题有三点:

- 1) 初始建立路径时,需要大量信令信息支持,从源用户选取初始入口源卫星,目的用户选取出口目的卫星的上下行链路,再从源卫星到目的卫星的星间链路的连接,需要分配好各链路的带宽。
- 2) 初始建立的连接会由于卫星的快速移动引起连接的切换。
- 3) 一旦连接发生切换,需要信令支持重新建立连接和优化路径。

随着 Internet 的广泛使用导致 IP 网络无所不在,IP 网络中无连接的体制在卫星网络中越来越受到重视。目前对卫星网络中实现 IP 路由选择已经提出了一些解决方法,它们的主要思想是:利用 LEO 星座网络所表现出的特殊性质,对星座网络拓扑变化进行某些规定或抽象,使得路由协议可以屏蔽星座网络的变化,实现一种透明性。

本文研究的低轨卫星网络支持数据和语音两种类型业务,针对实时性要求较高的语音业务采用面向连接的分组交换体制,在用户通信之前建立一条端到端的连接,预留各链路带宽资源,既保证端到端的时延性能,保证分组的有序性,同时避免由于卫星网络高度动态带来的链路的中断,但此方式需要耗费带宽资源支持大量信令信息;

针对数据业务则采用无连接分组交换体制,采用随机接入进入网络,根据分组中头部信息选择路径并传递分组,虽然分组的时延性能有所降低,但无需信令信息支持分组的传递,节省了带宽资源并简化了系统。

基于上述通信体制和交换模式,要求卫星配置交换设备、路由器以及无线资源管理模块,相当于一个小型的地面信关站,其通信协议分层如 2-6 所示。下面具体描述各节点协议层次的功能。

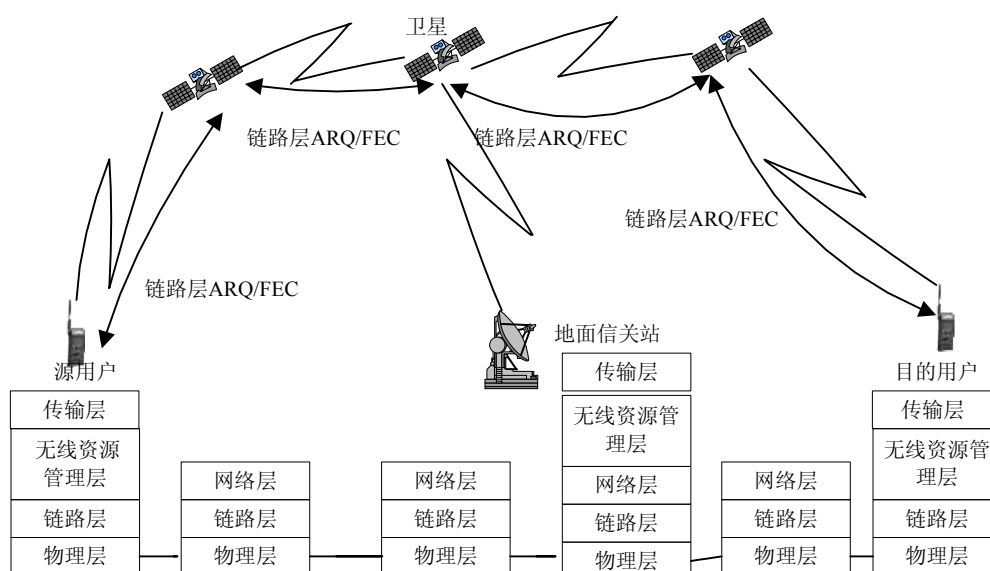


图 2-6 星上处理模式系统通信协议分层框图

## 用户端

传输层：产生建立连接所需的各种信令信息分组，产生业务分组；

无线资源管理层：辅助系统中信道的建立、释放以及用户链路的切换；

链路层：对信令和业务分组封装（设置分组的头部、尾部、内容以及分组的总长）。

## 卫星

网络层：运行路由算法以及分组调度算法；根据分组类型正确传递到正确输出端口；实现通信链路的建立和拆除等；具有无线资源管理的功能，辅助信道的建立、释放以及各链路的切换，记录并更新资源管理信息列表（表中记录各种连接信息、各信道的占用信息等）；

链路层：对数据包解帧、分析以及封装；

物理层：用于用户节点、信关站节点和卫星节点收发信息。

## 信关站

地面多个信关站通过有线进行传输信息，实现对卫星网络的控制、网络资源的管理以及用户呼叫过程的建立、各种信令的传递、各种切换的执行等，同时承担与地面其它系统进行通信的接口任务。

在低轨卫星网络中，为避免网络的拥塞需要执行流量控制、拥塞避免以及差错控制。现有的研究主要借鉴地面的 TCP 协议，TCP 机制的有效运行依赖于下面 3 个假

设:

- (1)数据包的丢失一定是由网络拥塞所引起的;
- (2)网络的拓扑结构是稳定的;
- (3)通信路径上的传播时延也相对稳定。

由于卫星网具有传播时延长、链路传输较易出错等特点,使得原本为有线网络设计的 TCP 协议用于卫星网时效率很低。卫星 TCP 网络面临的各种问题源于卫星信道和卫星网络的各种固有特性,主要有 3 个方面。

#### (1) 信道差错率

卫星信道的比特差错率(BER)大约为  $10^{-6}$  数量级,这远远高于高速有线媒质(如光纤)。另外空间信道的各种随机因素(如雨衰等)使得信道出现突发错误。噪声相对高的卫星链路大大地降低了 TCP 的性能,因为 TCP 是一个使用分组丢失来控制传输行为的丢失敏感协议,它无法区分拥塞丢失和链路恶化丢失。较大的 BER 过早地触发了窗口减小机制,虽然这时网络并没有拥塞。此外,ACK 分组的丢失使吞吐量进一步恶化。

#### (2) 传播延迟

低轨卫星网络的单向传播延迟一般是 20~25ms,系统延迟还受星间路由选择、星上处理以及缓存等因素的影响。一般而言,延迟对 TCP 的影响体现在:降低了 TCP 对分组丢失的响应,特别对于仅想临界发送超过缺省启动窗口大小(仅超过一个 TCP 数据段)的连接更是如此。此时用户必须在慢启动状态下,在第一个 ACK 分组收到前,等待一个完全的往返延迟;卫星延迟和不断增加的信道速度(10Mbit/S 或更高)还要求有效的缓存;增加的延迟偏差(variance)反过来也会通过在估算中加入噪声影响 TCP 定时器机制,这一偏差会过早产生超时或重传,出现不正常的窗口大小,降低了总的带宽效率。简单地增加 TCP 定时器粒度(tranularity)在此没有多大帮助,因为尽管较大的值可以降低错误超时,但带宽利用不足也将因较长的延迟而增加。

#### (3) 信道不对称

许多卫星系统在前向和反向数据信道间有较大的带宽不对称性,采用速度较慢的反向信道可使接收机设计更经济且节省了宝贵的卫星带宽。考虑到大量 TCP 传输的较大单方向性特性(如从 Web 服务器到远端主机),慢速反向信道在一定程度上是可以接受的。但非对称配置对 TCP 仍有显著的影响。例如,由于 ACK 分组会丢失或在较大数据分组后排队,较慢的反向信道会引起像 ACK 丢失和压缩(compression)的有害影响,从而大大减小吞吐量,有资料显示吞吐量随不对称的增加呈指数减小。此外,前向和反向信道速率的较大不对称会由于线速率突发错误较大而明显加重前向缓存拥塞。

此外,星地之间的频繁切换、卫星网络拓扑结构快速变化、终端之间通信路径改变都会导致延时变化。这些都会对 TCP 协议造成不利影响。

卫星网中 TCP 业务的性能与服务质量成为近年关注的问题,为提高 TCP 在卫星网络中的性能,国内外已展开了不少研究工作<sup>[98~102]</sup>,优化措施包括:调整传输控制协议在卫星链路应用中的参数;减轻卫星链路上 TCP 的负荷;使用协议转换网关。除此之外,链路层采用 ARQ(Automatic Retransmission Request 自动请求重传)和 FEC(Forward Error Correction 前向纠错)技术,可以减少因链路传输出错丢包而导致 TCP 发送窗口缩减、吞吐量降低,并通过逐跳链路层中继机制,在低层提供可靠的数据传输。因此,本文中卫星节点的链路层采用 ARQ 和 FEC 机制,空间段应用空间通信协议 CCSDS,实现 TCP over LEO 卫星通信网络,保证低轨卫星通信网络在传输层中端到端的可靠数据传递。

### 2.1.3 星上处理模式通信流程

当网络的通信协议明确,各种业务的通信流程可以确定。当卫星处于星上处理和交换模式时,采用无连接交换方式的数据业务的通信流程如图 2-7 所示。图中接入卫星表示与源用户通信的卫星,出口卫星表示与目的用户通信的卫星,地面段每个信关站有一个 VLR (visit location register) 拜访位置寄存器和一个 HLR (home location register) 归属位置寄存器,记录并更新用户的位置以及移动等信息。下面具体描述通信流程。

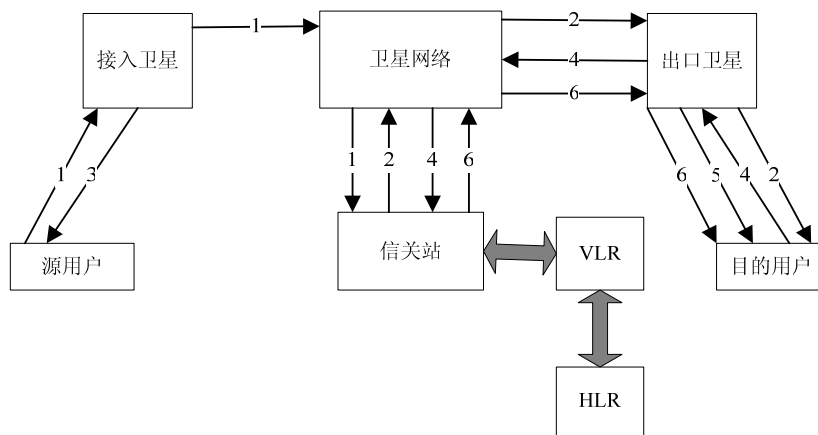


图 2-7 数据业务通信流程

1. 源用户产生业务分组,其格式如下所示:

业务类型	分组类型	源用户编号	目的用户编号	数据
------	------	-------	--------	----

其中分组类型用来区别分组为业务信息还是信令信息。用户通过随机接入的方式占用接入卫星的信道(接入卫星不在地面任何信关站覆盖范围),利用星上处理和路由计算功能,经由星间链路传递分组到某卫星(位于信关站覆盖区域)并下发到信关站。

2. 信关站根据分组头部信息在地面网络中通过VLR和HLR获得目的用户的位置信息,形成寻呼信息分组,其格式如下所示:

业务类型	分组类型	呼叫编号	目的用户编号	接入卫星编号	出口卫星编号
------	------	------	--------	--------	--------

分组中的呼叫编号表示不同用户对之间的通信。信关站传递寻呼信息到卫星，并通过星间链路传递到出口卫星，最后下载到目的用户。

3. 接收到源用户的业务分组后，接入卫星产生确认信息发送到源用户。
4. 接收到寻呼消息后，目的用户产生应答信息，并沿着传递寻呼信息的路径传输此应答信息到信关站。
5. 接收到寻呼应答信息后，出口卫星对目的用户发出确认消息。
6. 接收到寻呼应答消息后，更改暂存信关站的业务分组头部信息，其格式如下：

业务类型	分组类型	接入卫星编号	出口卫星和信道编号	数据
------	------	--------	-----------	----

经过卫星网络传递此数据分组到出口卫星，并通过指定的信道下载到目的用户。

语音业务采用面向连接的交换体制，源和目的用户通信之前，首先通过一系列信令消息建立端到端的逻辑连接，当卫星采用星上处理和交换模式时，其通信流程如图 2-8 所示。

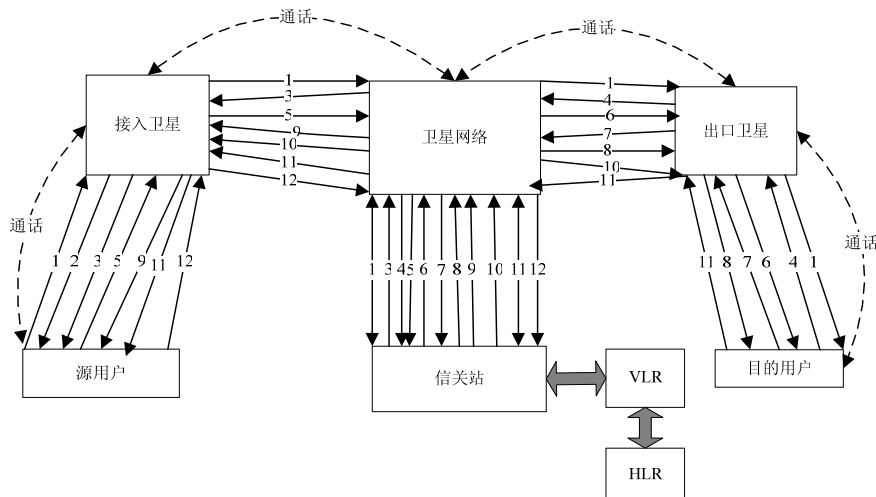


图 2-8 语音业务通信流程

1. 源用户发起呼叫请求信息，其格式如下所示：

业务类型	分组类型	源用户编号	目的用户编号
------	------	-------	--------

分组类型表示此分组为请求信息包。通过星上路由计算传递请求信息分组到最近信关站覆盖的卫星，并下载到信关站；信关站根据呼叫请求信息通过地面的 VLR 和 HLR 找到目的用户位于哪颗卫星覆盖区（即寻找出口卫星），形成寻呼信息，其格式如下所示：

业务类型	分组类型	呼叫编号	目的用户编号	接入卫星编号	出口卫星编号
------	------	------	--------	--------	--------

同样，经由星间链路传递此寻呼信息到达出口卫星，并下载到目的用户。

2. 接收到呼叫请求信息, 接入卫星发送确认信息给源用户。
3. 接收到呼叫请求信息, 信关站分配接入卫星的业务信道给源用户。
4. 接收到寻呼信息后, 目的用户发出应答信息到信关站。
5. 接收到信道分配消息后, 源用户发出确认信息到达信关站。
6. 信关站分配出口卫星中的业务信道给目的用户。
7. 接收到信道分配消息后, 目的用户发出确认消息给信关站。
8. 接收到目的用户的确认消息后, 信关站发起振铃消息到目的用户。
9. 同时, 信关站发起回铃音消息给源用户。
10. 建立连接过程中, 由于卫星的快速移动带来接入卫星或出口卫星的改变, 信关站需要重新配置接入卫星、出口卫星以及对应的信道等资源的分配。
11. 接收到振铃消息后, 目的用户发起摘机应答信息到达源用户。
12. 接收到摘机应答消息后, 源用户发送确认信息到达信关站。

此刻, 逻辑连接完成建立, 源用户和目的用户直接通过星间链路进行端到端通信, 当通信结束时, 用户发起结束呼叫信息经由建立的连接到达目的用户, 释放卫星中占用的信道资源。卫星中建立资源管理信息列表记录并更新各通信中的逻辑连接状态, 当通话结束时, 及时删除各卫星、信关站中的记录。

无论数据业务或是语音业务, 当无信关站覆盖时用户业务的顺利传递取决于星上的路由计算、各分组在星上的快速交换调度以及各卫星带宽资源的合理分配, 本文基于上述的通信协议重点研究了星上处理的关键技术: 路由协议和分组调度策略。

## 2.2 仿真平台体系结构

卫星网络具有一些特有的性质, 例如星座设计特性、用户链路和星间链路特性、用户链路及星间链路的切换及拓扑的动态变化等。这些都对系统的路由、交换调度策略以及传输层协议和上层应用的性能产生很大影响。但是, 为了研究这些问题而搭建实验平台在时间上和经济上往往都是不允许的, 因此需要有一个切实可行的网络模拟方法。

目前常用的网络模拟方法有: 测试法、数学分析法和软件仿真。

测试法主要是对现有网络进行采样与检测, 通过统计等有关理论评价网络的性能。建一个实际的卫星网进行测试是不现实的。

数学分析法主要采用数学的理论和方法, 对网络系统进行某些假设, 得出某些相关理论, 进而做出网络性能的评价。由于卫星网络拓扑十分复杂并且具有时变特性, 因此, 对通信协议的性能进行理论分析非常困难。所以这种方法也不适合于模拟卫星网络。

在这种情况下, 借助软件仿真, 建立合适的仿真模型, 对设计的应用于卫星网的通信协议及其性能进行仿真, 可以获得较为准确的仿真结果以及不同协议的性能比

较，是可行的性能验证及仿真方案。因此，如何建立一个基于软件的模拟仿真平台就成了评价路由协议性能、交换调度策略性能、通信体制合理性等上层功能的一个关键问题。

路由协议、调度策略、交换体制以及通信协议等性能的分析与评估，主要是网络中的性能指标。路由协议和交换调度性能和指标的评估基于一个重要的假设，即所设计的路由协议以及交换调度策略确实能够在特定的网络上稳定可靠地运行。

本章基于 OPNET 仿真环境，提出了卫星网络的协议体系结构和软件仿真平台结构。针对卫星网络的具体特征，从网络拓扑结构、网络接入策略的设计、邻居卫星节点的发现方式、卫星运行轨道的自主维护和星际链路模拟等方面进行研究，在 OPNET 环境中实现了卫星网络路由协议、通信协议、通信流程、切换策略和交换调度等的模拟。在软件仿真平台基础上，通过在不同场景下设置不同的外部输入条件和参数，可以对各种路由算法以及调度策略的性能进行比较和分析，比较的性能指标主要包括端到端时延、时延抖动、带宽利用率、丢包率、网络负载和吞吐量等。

### 2.2.1 基于数据包的通信过程

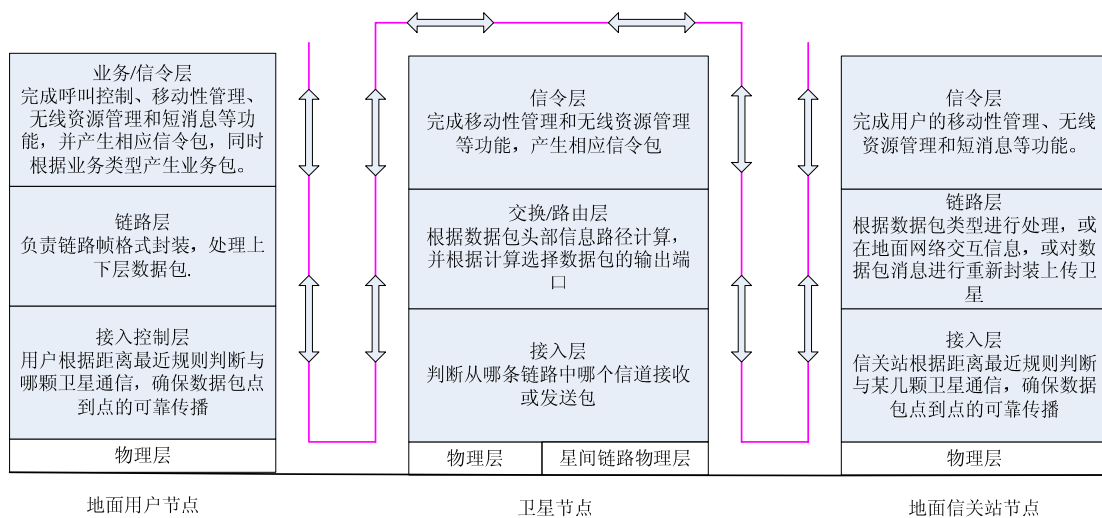


图 2-9 仿真平台节点协议体系结构

仿真中采用基于数据包的通信机制完成以下功能：

- (1) 模拟实际物理网络中包的流动，包括在网络设备节点间的流动和节点内部的处理过程；
- (2) 模拟实际网络协议中的封包和拆包的过程，可以生成、编辑任何标准和自定义的包格式；
- (3) 模拟通信过程中看到的任何特定包的包头（header）和净荷（payload）等内容。

仿真中，信令包和业务包根据 2.1 节中描述的通信协议和流程在低轨卫星网络中传递，当分组在两个用户之间进行传输时，经历用户节点、信关站节点和卫星节点各协议层的过程如图 2-9 所示。各节点的协议层次划分根据既定的通信协议架构和标

准，所有业务分组和大部分信令分组由用户节点产生，经过封帧后传递到卫星节点，卫星节点根据分组内容处理并转发数据，针对某些分组需要形成新的分组包，信关站节点协助网络完成对包的路由和传输。

### 2.2.2 事件驱动机制

在 OPNET 仿真平台中采用基于事件驱动的仿真，首先要确定仿真事件。仿真事件是指引起网络状态发生变化的事件，包括数据包的发送和到达、节点或链路的失效和恢复，以及组播引起的中断。除了这些和网络直接相关的状态变化，还有其他类型的中断事件也会间接改变网络状态，如进程自中断事件、程序调度中断事件、统计量引起的中断事件、仿真开始的中断和仿真结束的中断、外部系统引起的中断事件等。正是这些不同类型的中断事件，在仿真中可以按照需求设置和处理不同的中断事件，从而完成协议和算法仿真及分析。

事件驱动仿真需要一个仿真核心来调度排队的事件，即事件调度器（Event Scheduler）。它对所有进程模块希望完成的事件和计划该事件发生的时刻进行管理和维护。事件调度器主要维护一个具有优先级的队列，它按照事件将要发生的时间对其进行排序，并遵循先进先出（FIFO）的原则来执行事件。

事件驱动机制决定了其事件推进机制：仿真核心处理完当前事件 A 后，把它从事件列表（Event List）中删除，并获得下一事件 B，这时事件 B 会变成中断 B，并且被执行。如果事件 B 发生的时间 T2 大于当前仿真时间 T1 时，仿真核心将仿真时间推进到 T2，并触发执行中断 B；如果 T1 等于 T2 时，仿真时间不会推进，直接触发执行中断 B。总之，执行事件不需要仿真时间，而事件和事件之间可能跨越仿真时间。

由于执行事件不需要仿真时间，就可能存在同一仿真时间点上有多多个事件。

在仿真核心的事件列表中，那么它们将按照先进先出 FIFO 的顺序被仿真核心处理。当时间上不能区分事件优先级时，可以通过手动设定事件优先级来区分同一时间刻多个事件执行的顺序，或者增加冗余状态来界定同一时刻事件的发生顺序。

### 2.2.3 有限状态机机制

对于事件驱动的仿真来说，仿真的控制权不断的在进程模型和仿真核心（Simulation Kernel）之间转换。事件到达时，仿真核心确定该事件应该交给哪一个进程进行处理，随后将控制权交给该进程模型，进程模型进入激活状态，进程模型根据到达的事件种类进行相应的处理。

对于进程模型主体之一的状态转移图而言，两个最主要的组成元素就是“状态”和“状态转移”。

“状态”的含义就是进程在仿真过程中所处的众多模式中的一个，状态之间是互



斥和互补的；进程不可能同一时刻处于两个状态中，因此说状态之间是互斥的。而所有的进程状态构成了进程状态空间的全集，所有状态之间又是互补的。进程在收到中断之后，将会从一个状态转移到另一个状态。

状态又可以分为两种：强制状态和非强制状态。两种的区别在于：状态执行时是否能够停留在本状态等待下一次中断。对于强制状态来说，仿真核心不允许其停留在这个状态，强制其立刻转移到下一个状态。而非强制状态恰恰相反，进入非强制状态后的仿真将停留在那个状态等待中断触发唤起。每个进程都至少有一个状态，那就是初始状态，初始状态可以是强制状态，也可以是非强制状态，它是进程被访问的第一个状态。

每个状态都有相应的操作与其对应，我们称这些操作为执行代码。每个状态的执行代码分为上下两部分，上面部分称为进入执行代码，下面称为离开执行代码。进入执行代码是进程进入该状态时进行的操作，而离开执行代码是进程离开该状态时进行的操作，如图 2-10 所示。

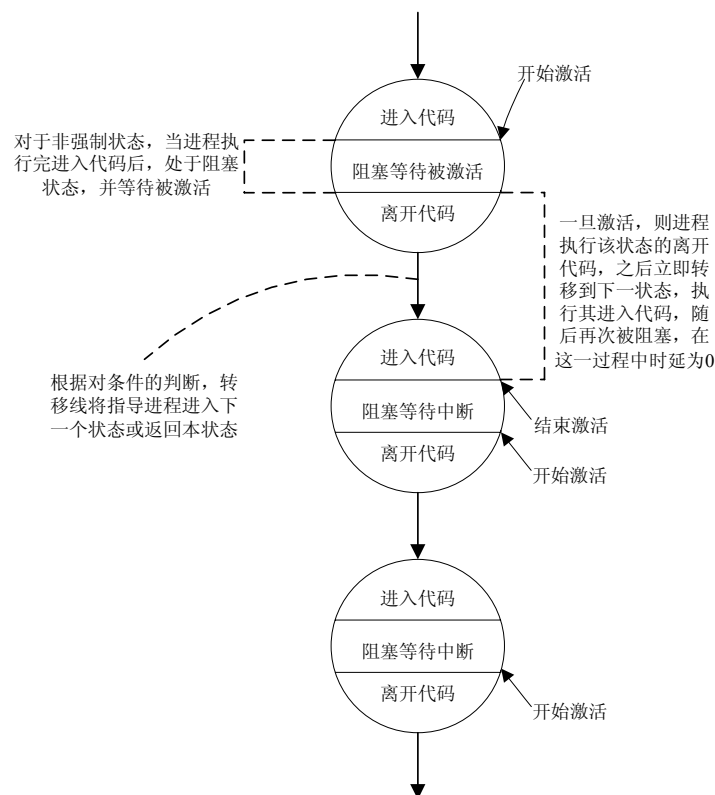


图 2-10 非强制状态的执行流程

对于进程建模来说，首先区分强制状态和非强制状态之间的区别是极为重要的。非强制状态可以认为是仿真系统中真正的状态。当进程执行完非强制状态的进入代码时，进程进被阻塞在这里，等待新的中断处理，此时控制权已经转交给仿真核心，进程的相关信息都存储起来。图 2-10 详细显示了非强制状态的执行流程。该图详细的显示了进程唤起的开始点和结束点。进程在执行完进入代码后被阻塞，等待被激活。

进程在准备执行离开代码前又被重新唤起。假如只有非强制状态，那么执行的顺序如下所述：

- 1) 执行当前状态的离开代码；
- 2) 转移到下一个状态；
- 3) 执行下一个状态的进入代码。

强制状态就其本质来说不能称为状态，并且也不能代表仿真系统的实际状态，因为仿真系统不会停留在强制状态上。强制状态只是为了便于编程而提出来的状态。仿真系统执行完强制状态的进入代码后，并不停留，而是直接执行离开代码。所以对强制状态来说，其进入代码和离开代码的功能是一样的。而在实际编程中，强制状态的离开代码一般是空的，只在进入代码部分进行代码编写。

“状态转移”描述了进程模型中从一个状态向另一个状态转移的过程和条件。转移包含四个部分：源状态、目的状态、转移条件和转移执行代码。一般来说，转移条件多是布尔表达式，用来判断是否应该转移到目的状态。进程只要一完成源状态的离开代码，就立即对转移条件进行判断。如果源状态为强制状态，转移代码和源状态的进入和离开代码就可以视为一体；如果源状态是非强制状态，则只有源状态的离开代码才可以和转移条件代码视为一体。

## 2.2.4 仿真平台中的网络模型

在仿真中采用了三层建模机制，最底层为进程模型，以状态机来描述；其次为节点模型，由相应的协议模型构成，以反映设备特性；最上层的是网络模型。三层模型和实际的协议、设备、网络基本对应，比较全面的反映了网络的相关特性。下面从网络模型、节点模型、协议模型三个角度分别说明卫星网络仿真平台设计。

网络模型主要反应网络中节点间的拓扑结构关系、地理布局以及节点的移动性。卫星网络模型包括卫星节点以及地面节点（用户节点和信关站节点），如图 2-11 所示。网络中每个节点完全采用分布式处理数据包，即采用相同节点模型的节点，通过配置节点的属性值，每个节点可以获得不同的处理能力和移动轨迹。

网络中通过软件 STK 导入各卫星的轨道模型，卫星轨道根据 24/3/1 的 Walker 星座模型设置。OPNET 提供的与 STK 之间的文件接口能够较为方便的将 STK 产生的星座卫星轨道导入，作为软件仿真时星座卫星的运行的基础。星座卫星运行的规律性以及先验性，是进行用户位置确定、接入、切换、拓扑生成、路由产生的基础，在很大程度上保证了系统的可靠运行。在 OPNET 仿真中，仿真系统运行的规律性和先验性，依赖于导入的 STK 卫星轨道位置。

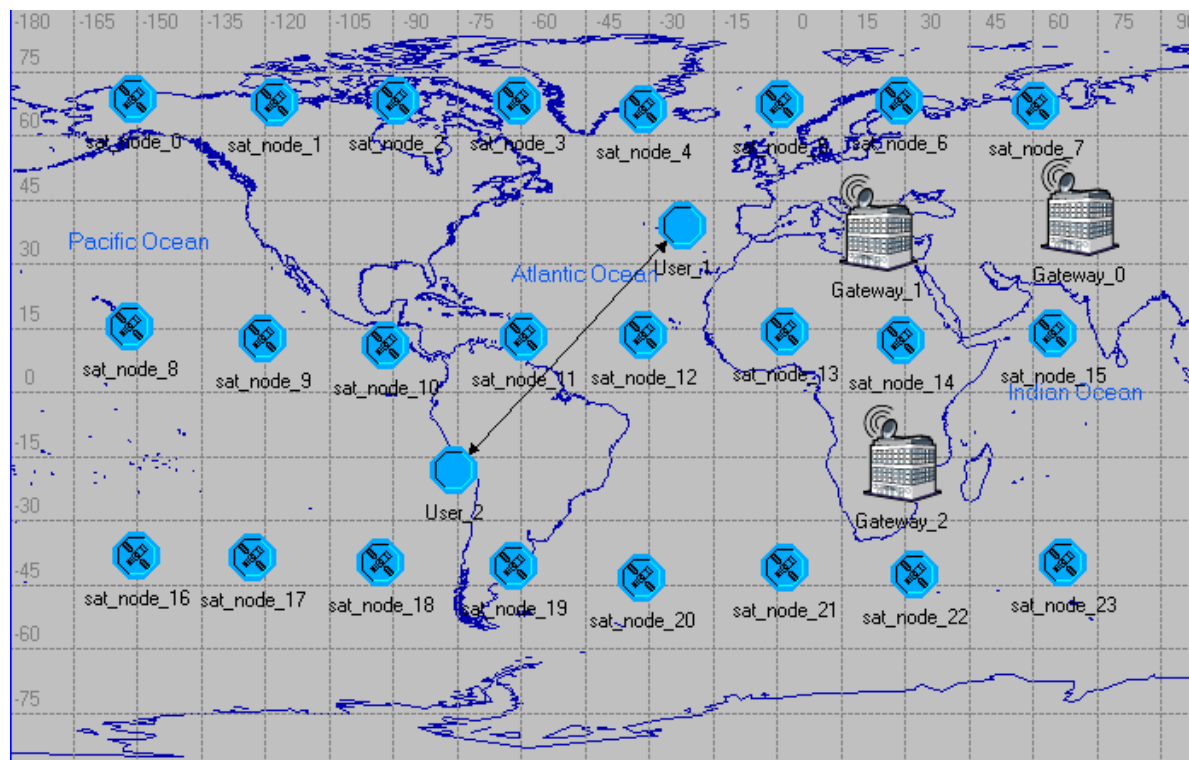


图 2-11 卫星网络的网络模型

### 2.2.5 仿真平台中的节点模型

节点模型在三层建模机制中对应于具体网络设备特性。每个节点模型需要确定所包含的协议模块，并且定义不同协议模块间数据包的流向等。图 2-12 和图 2-13 分别给出了卫星节点和地面用户节点的模型设计。

卫星节点模型根据其协议层设计：

物理层包括用于用户链路通信的收发信机以及接入模块（此为软件仿真特设的模块，保证物理层的可靠性连接），在用户侧实行多波束天线，因此有多对收发信机和定向天线；卫星与卫星之间的 4 条星间链路共用一对收发信机、接入模块和一全向天线；卫星与信关站的馈电链路通信的一对收发信机、接入模块和一全向天线。针对不同链路设置了多个链路层，同时为了完成网络层的功能，卫星节点布置了交换模块、路由模块、无线资源管理模块以及信息列表模块，这样设置主要是为了方便程序标准化，易于阅读以及修改。

用户节点在软件中包括有接入控制模块、封包的链路模块以及产生业务和信令的模块。卫星节点和地面节点各模块功能描述如表 2-1 所示。

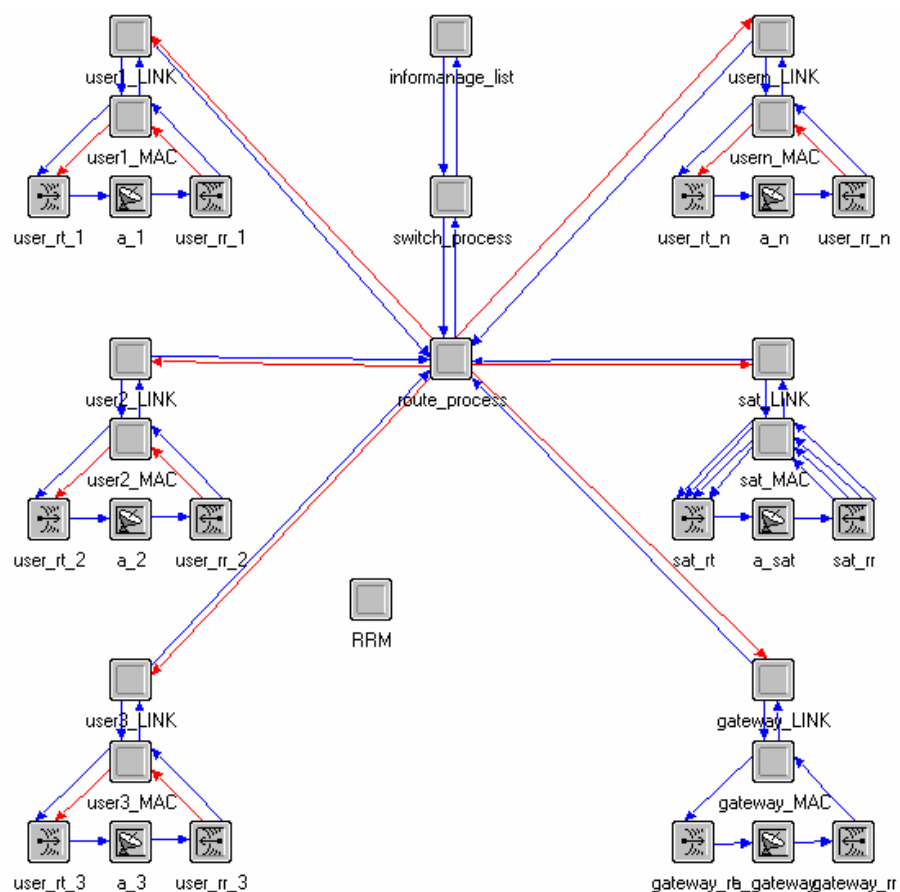


图 2-12 卫星节点模型

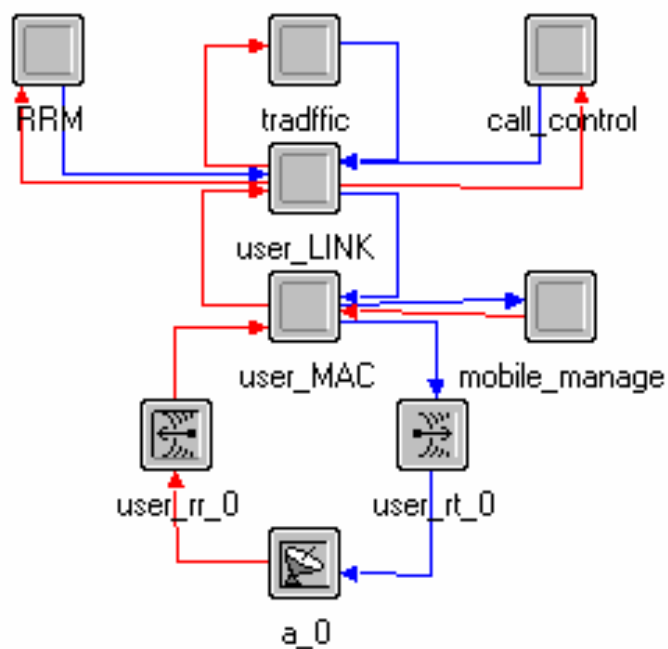


图 2-13 用户节点模型

表 2-1 卫星节点和地面用户节点模块简要描述

模块名称	功能简要描述
user_MAC,sat_MAC,gateway_MAC	计算哪些节点对可以通信
user_LINK,sat_LINK,gateway_LINK	根据包类型处理, 或转发或重新封包
route_process	卫星存储拓扑表计算路由
swich_process	对已经封好的包进行交换调度
RRM	无线资源管理
informanage_list	卫星上建立资源信息列表
call_control,mobile_manage	用户端需要对呼叫建立发起信令信息以及对用户移动的管理
traffic	产生各种类型的业务

### 2.2.6 仿真平台中的进程模型

进程模型是三层建模机制的最底层, 是实施各种算法和协议的载体, 因此大部分仿真工作量都集中在进程建模过程中, 节点中各模块的功能通过编写代码在进程中实现。对于事件驱动的仿真, 进程模型是产生事件并处理事件的主体。仿真中进程模型采用的是有限状态机 (Finite State Machine) 来描述事件处理过程。当状态发生变化的时候, 状态机才工作, 状态不发生变化的时间段不执行任何模拟计算, 即这个时间段被跳过。因此, 与时间驱动的仿真相比, 事件驱动的仿真计算效率得到很大的提高。

## 2.3 仿真平台和通信链路的实现

### 2.3.1 网络接入协议的实现

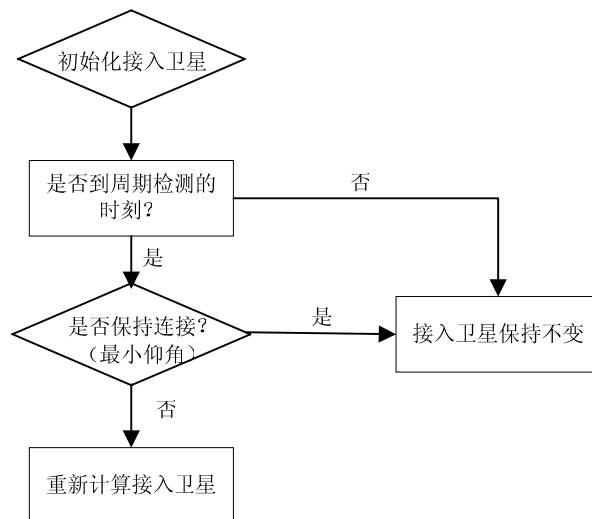


图 2-14 地面节点与覆盖卫星接入策略流程图

仿真中, 为了完成端到端的通信过程, 必须实现地面节点与覆盖卫星的接入策略。目前接入算法采用的是最短距离接入, 算法过程如下图 2-14 所示。首先, 初始化接入卫星, 地面节点在接入层模块计算并比较卫星网络中哪颗卫星距离最短, 并接入此

颗卫星。判断地面节点是否还在接入卫星范围内,是根据卫星的仰角范围是否大于最小仰角来确定。如果卫星仰角大于最小仰角,表明地面节点还在接入卫星覆盖范围内,否则不在该接入卫星覆盖范围。如果不在此卫星接入范围内,地面节点需要重新计算最短距离的卫星进行接入。

### 2.3.2 卫星星际链路的模拟实现

在卫星网络的路由仿真中,尽管网络层协议、链路层算法和协议、物理层协议以及传播媒质都会对网络性能产生很大的影响,但是由于主要研究的对象是网络层协议对网络性能的影响,因此仿真平台中对物理层做了以下合理假设,而将注意力主要集中在其对网络层协议的影响:

- 不考虑信道资源的管理和分配,以及信道复用等技术,只考虑信道的误码率、丢包率对网络性能的影响。
- 不考虑信号调制、检测、编码等技术,只考虑无线传输环境对网络拓扑的影响;

根据这些假设,在卫星网络的路由仿真中简化了链路层、物理层和传输媒质的建模,只对网络层产生直接影响的因素进行了建模和仿真。路由仿真中,主要通过以下 14 个阶段来模拟无线链路对数据包的影响,如图 2-15 所示。

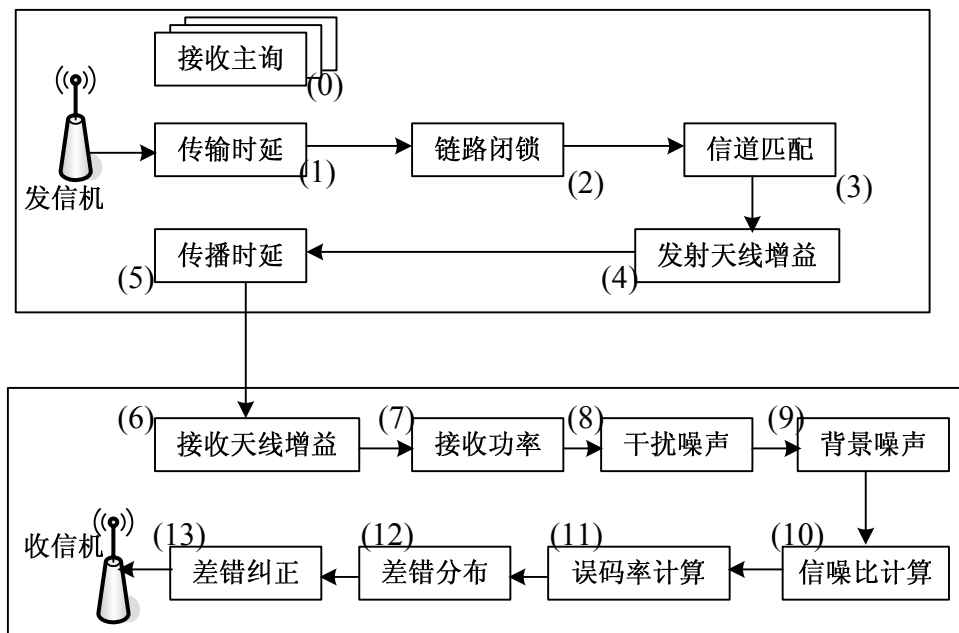


图 2-15 系统级仿真的无线管道模拟

第 0 个阶段确定接收组,这个阶段只在仿真编译的时候执行一次,发送包的时候不再执行;第 1 个阶段计算每个数据包传输时延,传输时延主要与链路带宽和包的大小有关;第 2 个阶段计算链路是否闭合,即计算收发机之间是否被障碍物遮挡,如卫星链路是否被地球遮挡;第 3 个阶段计算信道是否匹配,一般是看接收频段和

发送频段是否一致,收发信道是否满足一定的匹配关系;第 4 个阶段设置发送天线增益;第 5 个阶段计算传播时延,传播时延与收发机之间的距离以及无线电报传播速率有关;第 6 个阶段设置接收天线增益;第 7 个阶段计算数据包到接收机端的接收功率;接收功率与发送功率大小以及路径损耗有关;第 8 个阶段计算背景噪声;第 9 个阶段计算干扰噪声;第 10 个阶段计算数据包的信噪比;第 11 个阶段根据信噪比计算误比特率;第 12 个阶段计算误比特数;第 13 个阶段进行误比特纠错。

数据包经过这 14 个阶段,如果满足误比特数的要求,数据包就被接收端正确接收,否则数据包被丢弃。当然也可以根据信噪比等参数来确定数据包是否能正确接收,这时就要将后面几个阶段进行简化处理。

卫星网络中的链路主要有以下特点:

- 网络中的卫星节点虽然使用无线链路进行通信,但链路大部分是点对点或者点对多点,潜在广播式的链路很少,这一点与地面移动无线网络存在较大区别(地面移动无线一般是对传播范围内的节点潜在广播);
- 卫星之间的链路会因为地球的遮挡,周期性引起链路的通断;
- 卫星之间的链路大多采用点波束激光通信,星间链路的相互干扰很小并且链路在太空中的传播损耗也较小;
- 卫星链路的传播时延比较大。
- 在卫星网络中有与地面用户节点通信的用户链路、与信关站通信的馈电链路以及卫星与卫星之间通信的星间链路。
- 用户链路中针对上下行信道分为不同的逻辑信道:下行链路中分为寻呼信道、业务信道;上行链路中分为接入信道、业务信道。

根据卫星链路的上述特点,设置上面的 14 个无线链路阶段从而合理模拟卫星链路。仿真中着重对以下几个阶段进行了设计:

- 第 0 个阶段收发组的设置,主要通过函数 `op_radio_txch_rxgroup_set()` 设置每一次数据包发送的接收机组;
- 第 2 个阶段链路闭合的计算,主要考虑地球对卫星链路的遮挡;
- 第 3 个阶段中信道匹配主要是信道频点、带宽的完全匹配。卫星节点各链路信道频率和带宽设置为定值,用户根据卫星的接入信道更改信道参数;信关站节点与用户节点相同;同轨星间链路信道完全匹配,在仿真任意时刻都可以传输数据包;异轨星间链路则根据卫星网络拓扑确定其是否可以通信,如果可以通信信道完全匹配,第三章将详细叙述。
- 第 7 个阶段接收功率的计算,在自由空间传播的路径损耗基础上再加上一些其他损耗,如雨衰等损耗;
- 第 9 个阶段干扰噪声的计算,信道间的干扰噪声主要存在于卫星与地面节点的上下行链路,卫星与卫星之间的链路信道间干扰很小;

其他无线阶段都采用缺省的函数。在信道配置中,需要合理考虑各信道发射功率,使物理信道能理想通信。

### 2.3.3 多波束天线的仿真模型

系统设计中,用户链路采用多波束天线。因此,无线管道建模中涉及到多波束天线的建模,多波束天线的存在使得用户在通信过程中会产生波束切换,对于通信流程有极大的影响,因此必须对多波束天线进行建模。OPNET 提供了 Antenna Pattern 对天线进行建模,但没有提供多波束的实现方法。本文采用多个 Antenna 模型,根据系统提供的收发天线方向图,分别实现收发各 16 个用户链路波束天线的方向图,结合多个收发信机,共同构成多波束天线。

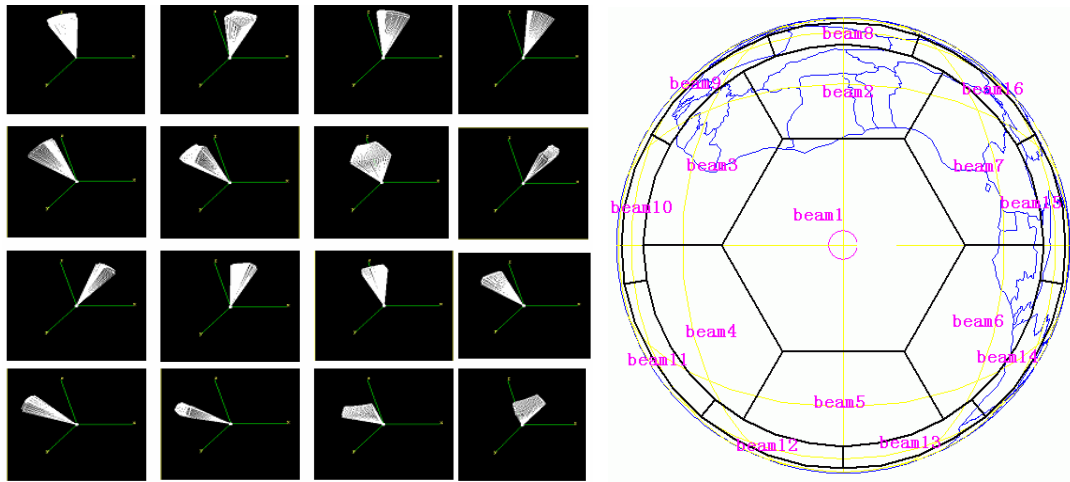


图 2-16 发射天线

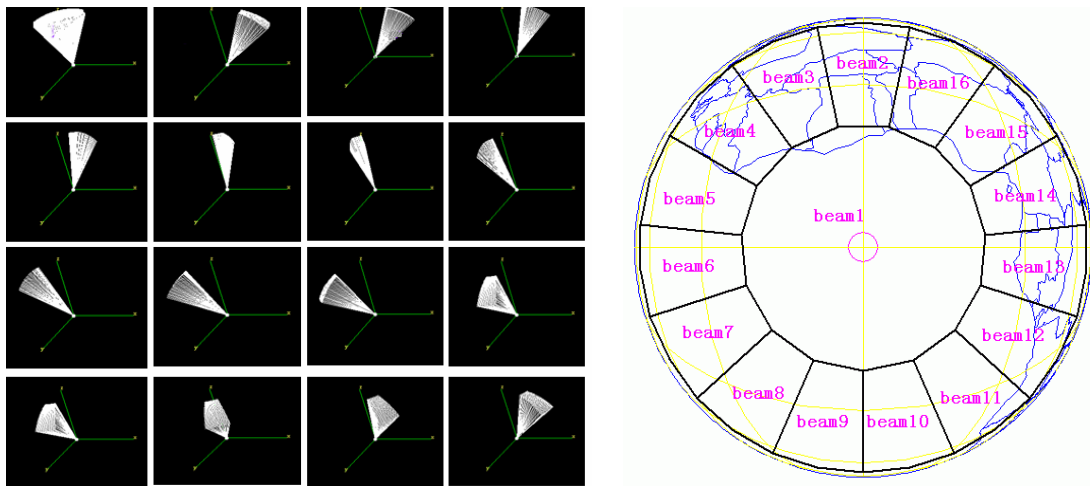


图 2-17 接收天线

图 2-16 和图 2-17 分别为星上发射和接收多波束天线的建模情况,左图为 OPNET 中根据右图天线的方向增益图设置的天线模型三维图。其中,发射天线中心波束增益 10db,第二圈波束增益 13db,外圈波束增益 16db。接收天线中心波束增益 10db,外圈波束增益 13db。



为了实现多波束天线模型,首先,在 OPNET 的天线本体坐标中根据各波束在各方向的增益值设置各波束为一副定向天线,如上左图所示;根据当前时刻卫星位置和天线的增益方向图计算各波束投影在地面的位置(位置的计算需要周期更新),设置各定向天线中的属性:“target latitude”,“target longitude”,“target altitude”为计算的各波束投影在地面的位置。可以看出,上述方法是通过多幅定向天线模拟多波束天线,只有在天线覆盖区域范围内地面节点才可以与卫星进行无线传输。

在整个仿真平台中,卫星网络拓扑实现处于非常重要的地位。由于卫星网络的快速变化,将导致节点之间链路的切换问题。对于星地链路来说,主要通过地面节点在链路层执行接入协议,通过过滤和匹配,建立点到点的无线链路。对于星间链路来说,首先需要根据星座运行特征以及星间链路实际可通信规则确定轨道周期各时刻的网络拓扑结构表,结合路由协议准则,每颗卫星存储轨道周期各时刻的拓扑表,根据拓扑表通过卫星节点的接入层和路由层联合过滤和匹配建立星间链路。卫星星间链路的拓扑表由 STK 分析获得。

根据上文描述的通信协议、语音通信流程以及数据通信流程在 OPNET 的网络模型中运行。用户节点、卫星节点以及地面信关站节点联合实现端到端的通信。在模拟实际的通信过程中,可以把路由协议以及调度策略在星载交换机以及相应的模块中运行,并在不同的仿真场景中验证算法的性能。

## 2.4 本章小结

本章主要简单描述了低轨卫星网络的通信协议架构,针对不同类型的业务提出交换体制,描述了其通信流程;简要描述了基于 OPNET 仿真环境设计和实现的卫星网络软件仿真平台。基于 OPNET 软件环境的卫星网络仿真平台架构的研究,为其他非标准网络协议尤其是卫星无线网络协议的仿真、路由算法、调度算法和切换策略等提供了基础,是网络协议研究中理论与实践的结合。

### 第三章 低轨卫星通信网络中的路由协议

低轨卫星通信网络中路由分为用户—卫星上下行链路路由和星间路由。上下行链路路由主要考虑用户选取哪颗卫星的链路接入,由于卫星的快速移动会引起用户链路切换。星间路由只考虑空间段从源卫星到达目的卫星的路径选择,由于星间链路的动态变化引起卫星网络拓扑结构的变化,导致星间链路发生切换。

本章主要研究在特定的低轨卫星通信网络中星间路由问题。首先,简要介绍本文研究的低轨星座模型;分析低轨卫星通信网络中路由协议的重要性,概述最近几年提出的低轨卫星星间路由策略;基于已有的研究,提出基于拓扑简化的分布式路由策略(Distributed Routing Algorithm Based-on Simplified Topology, DRAST),详细描述拓扑的简化、排队时延的推导和路由算法的实现;最后,基于 OPNET 卫星网络仿真平台建立侧重仿真路由协议的各种仿真场景,并在不同的参数设置下进行仿真,分析结果。

#### 3.1 低轨卫星星座模型

本文采用的低轨星座为 Walker 星座,系统结构在第一章已经描述。参数为 24/3/1,该星座由 24 颗卫星组成,均匀分布在 3 个轨道面上,相邻轨道相位因子为 1,轨道倾角为 30 度。相邻轨道的相邻卫星的相位差为:

$$360^{\circ} \times F/N = 15^{\circ}$$

此星座主要覆盖中低纬度地区,其相位关系直接影响到各个卫星之间的位置关系,以及连通关系。下面表 3-1 中给出了 24 星 Walker 星座的相位关系。我们对星座中轨道面及卫星编号,三个轨道面分别称为轨道面 1,轨道面 2 和轨道面 3,同一轨道面里的卫星用 1~8 的子编号区分,例如, Sat26 代指轨道面 2 中的 6 号卫星。

表 3-1 24 星各星之间相位关系

卫星编号	纬度幅角 (°)	卫星编号	纬度幅角 (°)	卫星编号	纬度幅角 (°)
Sat11	0	Sat21	15	Sat31	30
Sat12	45	Sat22	60	Sat32	75
Sat13	90	Sat23	105	Sat33	120
Sat14	135	Sat24	150	Sat34	165
Sat15	180	Sat25	195	Sat35	210
Sat16	225	Sat26	240	Sat36	255
Sat17	270	Sat27	285	Sat37	300
Sat18	325	Sat28	330	Sat38	345

相对于极地轨道星座倾斜轨道星座的优点是避免了相对运行轨道之间的缝。卫星网络结构如图 3-1 所示。在卫星网络中存在两种类型的星间链路(inter-satellite links ISLs): 同轨星间链路(intra-plane ISLs)和异轨星间链路(inter-plane ISLs)。每颗卫

星可与同轨道前后卫星建立两条稳固的同轨星间链路,与相邻轨道面的可见卫星最多可达 6 颗,用于接入异轨星间链路的天线只有两幅,可建立两条异轨星间链路。考虑星座网络的拓扑结构时,同轨邻星的链路非常稳定,而异轨的邻星链路是动态变化的。根据两星之间距离、方位角、俯仰角的变化率以及天线的跟踪扫描特征确定异轨星间链路的建立(链路建立的时间段以及哪对卫星可以建立链路),对于采用何种方式的异轨建链方案能够达到最好的网络性能本文不予详细描述,直接采用在 STK 中分析得出的方案。

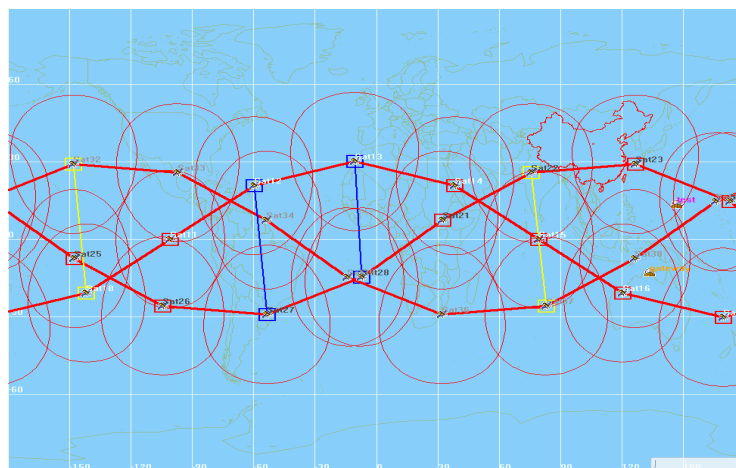


图 3-1 带星间链路的 Walker 星座结构图

### 3.2 低轨卫星通信网络路由的独特性

相对地面移动用户或固定用户来说,低轨卫星网络中的节点移动迅速,引起星间链路以及用户链路的频繁切换,是一个令人感兴趣的快速移动网络类型。尽管网络拓扑结构具有较高的时变特征,但可以简化其动态的特征。首先,卫星网络拓扑的变化可以预先确定(不考虑卫星失效的状态);其次,拓扑结构有一定的规律和对称性,能导致对于到达大多数目的地的路径选择具有多样性和相似性。上述的这些特征可以在设计路由策略时用到。

然而,与地面无线网络分组路由协议设计比较,低轨卫星网络几个基本不同的设计使得卫星网络路由技术面临着特别的挑战,主要体现在以下几点:

1. 稳定可靠、快速收敛的要求。网络拓扑迅速变化和链路状态信息实时变化,要求路由系统具有高稳定性和快速收敛的能力。
2. 低功耗、高可靠性要求。空间系统的能量供给是有限的,星上设备维护不便,因此要求路由系统必须具备低功耗和高可靠性运行的能力,以便提高系统的生命周期。
3. 有限的物理指标和强大的处理性能需求之间的矛盾。空间硬件系统的质量、体积等都受到限制,这对于提高路由系统性能是不利的,而对路由系统的性能预期、处理需求却要求很高,特别是处理需求不是静态停止的,而是动态增长的,因此

要求路由系统能够平衡两者的关系。

4. 星际链路的实现策略选择。同一轨道上的卫星间链接是静态的，而不同轨道上卫星间的链接则是动态的。卫星网络组成中，既包括动态链接又含有静态链接，而且链路切换相当频繁，路由选择算法的综合设计是比较困难的。
5. 链路带宽的限制，以及经济的限制不能允许在星座中布置过多的卫星都会对卫星网络带来复杂性，也给星间路由带来诸多限制。
6. 实际系统的性能和用户服务质量(QoS)需求的差异。卫星链路中存在着高误码率和时延过大等问题，这些同用户所要求的及时、准确地完成信息传输是矛盾的，因此要求路由系统采用一系列方法和策略缩小传输时延、降低差错率、防止拥塞。

针对这些问题，本课题研究意义在于解决路由协议的技术难点，在实时动态变化的卫星网络拓扑前提下，实现较少的拓扑信息交互和较低的路由计算量，保证路由协议可靠有效的运行；同时，在星上条件受限情况下，对路由协议进行优化，以提供尽可能好的 QoS 保证。

### 3.3 近年来有效的路由策略

近几年许多文献提出的星间路由策略不仅保证时延性能，更针对各业务的不同需求进行设计，譬如针对网络业务拥塞问题对业务进行分流，或为避免系统的复杂设计简洁的路由协议等，下面简要描述近几年提出的路由策略。

文献[103]提出 Distributed Load-Aware Routing(DLAR) 分布式路由算法。DLAR 分两步：第一步计算比当前卫星更接近目的卫星节点的卫星，标记这些卫星为下一跳节点；第二步基于传播时延和队列时延选择下一跳节点。路由中，通过水平方向和垂直方向的卫星标记避免路径死锁。此路由策略时延较低。下一跳节点的计算依赖各卫星的当前位置以及系统确定的动态特征，因此，星上需要大量空间存储卫星各时刻位置，且路由时趋于选择高纬度异轨链路导致部分链路拥塞。

文献[104]提出 Destruction-resistant 路由算法，划分星座为簇群，每一簇分配一颗卫星为管理节点（称为簇头）。各卫星节点仅仅了解本簇内的各星间链路的状态，全网链路状态信息通过簇头传递。此算法通过分簇管理使路由高效化，但带来系统复杂化，并增加了信令信息维持簇群的管理，算法执行的复杂度依赖于卫星网络的规模。文献[105]在低轨卫星网络中应用改进的遗传算法，建立优化函数选择路径：

$F_{Y_i} = D_{Y_i} * (1 - P_{Y_i})$ ， $D_{Y_i}$  表示所选路径  $Y_i$  的时延参数， $P_{Y_i}$  表示呼叫阻塞概率函数。此算法复杂度大。

文献[106]提出 Control Route Transmission (CRT) 路由策略，策略中应用区域分割概念，假设卫星为虚拟节点。CRT 算法采用面向连接分组交换机制，每颗卫星在卫星网络中定时广播其业务量，各卫星根据接收的信息建立最短路径路由表。CRT 算法需要占用大量带宽资源定期广播星间链路状态，网络越大需要花费越多的时间交

流信息，容易带来掉包现象；此策略还需要严格的网络同步。

文献[107]中提出基于 antnet 的路由策略，收集各链路状态信息，在建立路由表时根据链路的状态计算在路径中被选链路的概率，当建立路径后，利用 Forward-Ant 和 Back-Ant 更新链路状态。此算法需要经过多次迭代才能建立稳定的路由表。

文献[108]中提出备份路径的策略保证在链路意外失效情况下路由的有效性。文献[109]提出位置辅助按需路由策略，可以看作 ad hoc 网络中 AODV(ad-hoc On-demand distance vector)路由协议针对低轨卫星系统的特性的变型，根据卫星的位置接近目的地，采用序列编号避免环路，此算法星上需要时刻更新和储存大量信息。

上述研究虽然能保证路由的时延性能和健壮性，但增加了系统的复杂性。本文提出一种基于拓扑简化的分布式路由策略（DRAST），为降低星间链路切换频率，根据卫星运行周期和规律简化拓扑，卫星只需要存储拓扑表，结合拓扑特征在星上根据源卫星和目的卫星号动态寻找最佳路径。在保证时延性和有效性基础上，重点解决了如何降低路由协议的复杂度，以及降低对卫星存储空间的要求。算法执行简单，能保证端到端的时延。

### 3.4 DRAST 路由算法描述

#### 3.4.1 低轨卫星通信网络拓扑特征

同一轨道面卫星位置相对固定，因此同轨星间链路是永久性链路。而异轨面的卫星由于相对位置发生变化，不仅链路的长度会时刻变化，且会关闭。例如铱星系统，当卫星进入极地地区时会暂时关闭异轨星间链路；相反方向运动轨道面的卫星由于相对运动速度太快，也会关闭链路。因此异轨链路的变化会带来卫星网络拓扑结构发生变化，且这种变化是由卫星轨道模型决定的，因此，在设计路由算法时，首先需要分析并规划出卫星网络的拓扑特征。

在 3.1 中已经提出 24/3/1 低轨卫星可建立两条同轨星间链路和两条异轨星间链路，在 STK 中分析各种建立异轨链路的方案，根据卫星的运行轨道以及天线的跟踪、扫描采用如表 3-2 所示的建链方案。

表 3-2 异轨卫星建立链路方案

轨道面 1——轨道面 2	轨道面 1——轨道面 3	轨道面 2——轨道面 3
Sat 11——Sat 27	Sat 11——Sat 32	Sat 21——Sat 37
Sat 12——Sat 28	Sat 12——Sat 33	Sat 22——Sat 38
Sat 13——Sat 21	Sat 13——Sat 34	Sat 23——Sat 31
Sat 14——Sat 22	Sat 14——Sat 35	Sat 24——Sat 32
Sat 15——Sat 23	Sat 15——Sat 36	Sat 25——Sat 33
Sat 16——Sat 24	Sat 16——Sat 37	Sat 26——Sat 34
Sat 17——Sat 25	Sat 17——Sat 38	Sat 27——Sat 35
Sat 18——Sat 26	Sat 18——Sat 31	Sat 28——Sat 36

低轨卫星轨道周期约为 114 分钟，此卫星不会经过极地地区，也不存在“Seam”，

因此异轨链路是否可以通信主要参考卫星之间距离及方位角等相对运动参数。表 3-3 和表 3-4 给出了一个轨道周期内卫星 Sat11 与 2、3 轨道面异轨邻星 Sat27、Sat32 之间的链路特性统计数据。

表 3-3 一个轨道周期内 Sat11 和 Sat27 的相对运动参数

邻星 参数	Sat11-Sat27	Sat27-Sat11
星间距 $R/\text{km}$	4546.2~8157.5	4546.2~8157.5
方位角 $A/(\circ)$	-66.9~66.9	113.1~180, -180~-113.1
俯仰角 $Z/(\circ)$	-16.9~-31.4	-16.9~-31.4
星间距变化速率/ $(\text{km/s})$	-3.3~3.3	-3.3~3.3
方位角变化速率/ $(\circ/\text{s})$	-0.078~0.078	-0.078~0.078
俯仰角变化速率/ $(\circ/\text{s})$	-0.013~0.013	-0.013~0.013

表 3-4 一个轨道周期内 Sat11 和 Sat32 的相对运动参数

邻星 参数	Sat11-Sat32	Sat32-Sat11
星间距 $R/\text{km}$	4546.0~8157.3	4546.0~8157.3
方位角 $A/(\circ)$	113.1~180, -180~-113.1	-66.9~66.9
俯仰角 $Z/(\circ)$	-16.9~-31.4	-16.9~-31.4
星间距变化速率/ $(\text{km/s})$	-3.3~3.3	-3.3~3.3
方位角变化速率/ $(\circ/\text{s})$	-0.078~0.078	-0.078~0.078
俯仰角变化速率/ $(\circ/\text{s})$	-0.013~0.013	-0.013~0.013

参考表中卫星对之间相对运动参数，考虑星间距离不能超过 6800km，且两星之间相对运动速度不能太大情况下才能建立链路，分析得到 Sat11 在一个轨道周期内与 Sat27、Sat32 可分别在两个时间段建立链路，如图 3-2 所示。每个时段持续约 32 分钟。Sat11 与 Sat27 可以建立链路的第一个时间段为（以一个轨道周期 1 小时 54 分钟为参照）00:25~00:57（单位：小时:分钟），第二个时间段为 01:22~01:54；Sat11 与 Sat32 可以建立链路的时间段为 00:01~00:33 和 00:58~01:30。依此标准类推，得出如表 3-5 所示的全部异轨链路在一个轨道周期内建立的时间段。由于卫星轨道具有对称性，从表中可以看出某些星间链路建链时间段是重合的。譬如，Sat11-Sat27 和 Sat15-Sat23 之间的建链时间段相同，Sat12-Sat28 和 Sat16-Sat24 相同。一个轨道周期中，表 3-5 所示的 24 条异轨星间链路建立的时间段为 24 段。

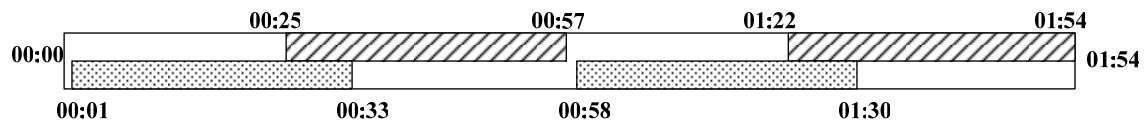


图 3-2 Sat11 与 Sat27、Sat32 的建链时段示意图

根据建立链路的时间段设置网络拓扑图以及更新时刻，总共需要 48 张拓扑表，称之为全拓扑。当任何一条异轨链路关闭或建立时即为拓扑更新时刻。上述的拓扑结构没有考虑卫星相对于地球的位置变化，仅仅考虑的是空间段卫星网络各节点之间的

连通和关闭。

表 3-5 一个轨道周期内各异轨星间链路建立时间段

1-2 轨道面	建链时间段	1-3 轨道面	建链时间段	2-3 轨道面	建链时间段
Sat11-Sat27	00:25~00:57	Sat11-Sat32	00:01~00:33	Sat21-Sat37	00:20~00:52
	01:22~01:54		00:58~01:30		01:17~01:49
Sat12-Sat28	00:10~00:42	Sat14-Sat35	00:15~00:47	Sat22-Sat38	00:06~00:37
	01:08~01:40		01:12~01:44		01:03~01:35
Sat13-Sat21	00:00~00:28	Sat13-Sat34	00:00~00:04	Sat23-Sat31	00:00~00:23
	00:53~01:25		00:29~01:01		00:49~01:20
	01:51~01:54		01:27~01:54		01:46~01:54
Sat14-Sat22	00:00~00:14	Sat12-Sat33	00:00~00:18	Sat24-Sat32	00:00~00:09
	00:39~01:11		00:44~01:16		00:34~01:06
	01:36~01:54		01:41~01:54		01:32~01:54
Sat15-Sat23	00:25~00:57	Sat15-Sat36	00:01~00:33	Sat25-Sat33	00:20~00:52
	01:22~01:54		00:58~01:30		01:17~01:49
Sat16-Sat24	00:10~00:42	Sat18-Sat31	00:15~00:47	Sat26-Sat34	00:06~00:37
	01:08~01:40		01:12~01:44		01:03~01:35
Sat17-Sat25	00:00~00:28	Sat17-Sat38	00:00~00:04	Sat27-Sat35	00:00~00:23
	00:53~01:25		00:29~01:01		00:49~01:20
	01:51~01:54		01:27~01:54		01:46~01:54
Sat18-Sat26	00:00~00:14	Sat16-Sat37	00:00~00:18	Sat28-Sat36	00:00~00:09
	00:39~01:11		00:44~01:16		00:34~01:06
	01:36~01:54		01:41~01:54		01:32~01:54

### 3.4.2 低轨卫星通信网络拓扑的简化

分布式路由机制主要用于 IP 路由，其原理是每个网络节点进行独立的本地路由计算，路由决策仅仅依赖数据包的目的地址和节点自身所了解的网络拓扑。针对星上资源有限的特点，结合分布式路由本地路由计算的特性，可采用逐跳路由的方法减少星上路由信息的存储量。但为了降低路由计算量，需要设计复杂度低的路由方案，卫星网络拓扑的复杂度以及快速的变化决定了路由算法的复杂度，因此本文从简化卫星网络拓扑着手设计路由策略。

譬如，图 3-3 选取部分节点说明如何简化网络拓扑，图中双向箭头的实线表示同轨星间链路，双向箭头的虚线表示异轨星间链路。当源卫星为 Sat15，目的卫星为 Sat22 时，最短路径为 Sat15—Sat14—Sat22 或 Sat15—Sat23—Sat22，但为了降低星间链路切换次数，选择 Sat15—Sat14—Sat13—Sat21—Sat22 路径，经过的异轨链路为

Sat13-Sat21, 此链路持续经历了 12 个连续的拓扑。对于面向连接的交换体制来说, 用户双方建立通信之前需要预先建立一条路径, 如果通话过程中链路频繁切换将会需要大量信令来支持切换, 引入时延抖动, 因此, 选择跳数较多但切换次数降低的路径。同理, 图中卫星 Sat11 到达卫星 Sat27 选取经过 Sat17-Sat25 为最佳路径。

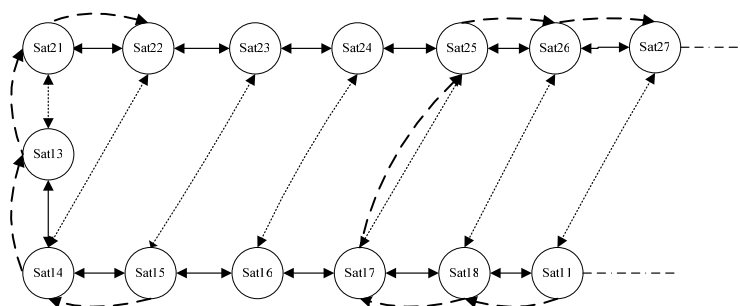


图 3-3 部分拓扑路径选择示意图

从对上图举例分析来看,为避免星间链路的频繁切换,路由时,总是选择保持建立时间最长的异轨链路,因而业务主要集中在这部分异轨星间链路上。基于此,在卫星网络拓扑中只选择建立时间集中且建立时间段具有最大重叠性的异轨链路,达到既减少轨道周期内拓扑表的数目,又简化拓扑结构的目的。路由的复杂度取决于拓扑的复杂性,在简化的拓扑中能设计简洁的路由算法。

分析 3.4.1 中建立的拓扑表, 建立如表 3-6 所示卫星对之间的异轨链路。根据对称性, 表 3-6 方格中符号标记相同的卫星对之间建立链路的时间段相同, 譬如 Sat12—Sat28 与 Sat16—Sat24 建立链路的时间段相同。统计各异轨链路建立时间段为 12 段, 一个轨道周期内仅需设置 24 张卫星网络拓扑图。拓扑持续时间最长为 16 分钟, 最短为 1 分钟。轨道周期内拓扑数目减少 50%, 拓扑结构大大被简化, 卫星根据分组头部路由信息逐跳计算路径, 由于拓扑更新平缓, 结构简单, 卫星处理简单, 符合卫星硬件受限的要求。

表 3-6 简化拓扑下建立异轨星间链路的卫星对

轨道面 1——轨道面 2	轨道面 1——轨道面 3	轨道面 2——轨道面 3
◎ Sat12——Sat 28	□ Sat 11——Sat 32	○ Sat 21——Sat 37
◇ Sat14——Sat22	☆ Sat13——Sat34	△ Sat23——Sat 31
◎ Sat16——Sat 24	□ Sat15——Sat 36	○ Sat25——Sat33
◇ Sat 18——Sat 26	☆ Sat17——Sat38	△ Sat 27——Sat 35

### 3.4.3 路由策略参数分析

路由算法性能的评价在于两个方面:

- (1) 为每个接收的业务流提供服务质量保证, 保证端到端的时延、无掉包率。
- (2) 网络全局资源的最佳利用, 合理利用各链路的带宽资源, 且保证路由算法执行简单, 星上无需过大处理能力。

因此路由参数包括端到端时延参数、掉包率、复杂度和可用带宽参数。可用带宽



参数是指分组到达时，链路上剩余的实际带宽。当可用带宽大于分组大小时分组可以通过，否则，需要为该分组重新分配路由。

端到端时延：指从源用户到目的用户的传播时延、传输时延、排队时延和处理时延的总和。

传播时延：分组在卫星节点间的传播时间（距离/光速）。在上述星座中同轨星间链路距离约为 6000 km，异轨星间链路距离在 4900~6800 km 变化。近似认为各星间链路传播时延相等，约为 20 ms。

传输时延：业务分组大小和链路带宽的比值。

传播时延和传输时延的计算方法简单，不做具体讨论。下面分析的主要时延参数是节点上的排队时延和处理时延，分析的方法采用排队理论构建模型。

本文提出的路由策略中，卫星根据拓扑表和分组头部信息查找路由的处理时延较低，可以忽略不计。在低轨星座系统中具体到某次通话的端到端的时延如下公式所表示，公式中各符号表示意义如表 3-7 所示：

$$D = t_p + t_q \quad (3.1)$$

$$t_p = Q \times t_{UDL} + W \times t_{ISL} \quad (3.2)$$

$$t_q = \sum_{j=1}^H \left( \frac{L_{i,j}}{\phi_{i,j}} + \sum_{k=1}^K \frac{L_{k,j}}{\phi_{k,j}} \right) \quad (3.3)$$

表 3-7 本章中公式表征意义

符号	表示意义	符号	表示意义
$D$	端到端时延	$N$	星座轨道面数目
$t_p$	路径中各链路传播时延之和	$lat$	卫星纬度位置
$t_q$	路径中各链路队列时延之和	$D_{inter}$	异轨星间链路距离
$Q$	路径中经历的上下行链路总数	$H$	路径中经历各卫星节点总数
$W$	路径中星间链路的跳数	$j$	路径中各卫星节点编号
$D_{intra}$	同轨星间链路距离	$L$	分组包的长度（单位：bits）
$R$	卫星轨道面半径	$\phi$	信道中可用的带宽（单位：bits）
$M$	每个轨道面的卫星数目	$k$	路径中各卫星节点缓存区中其余各会话编号
$i$	此次会话编号	$K$	路径中各卫星节点缓存区中各会话总数

端到端时延包括总的传播时延  $t_p$  和总的队列时延  $t_q$ 。其中传播时延与星地距离以及卫星间距离相关，可通过距离和光速的比值得到。传播时延包括上下行链路和星间链路的传播时延，卫星轨道高度估计约为 1400km，上下行链路时延相同，约为 4.67ms。同轨和异轨星间链路的距离计算分别如下：

$$D_{intra} = \sqrt{2R} \sqrt{1 - \cos\left(\frac{360^\circ}{M}\right)} \quad (3.4)$$

$$D_{inter} = \sqrt{2R} \sqrt{1 - \cos(\frac{360^\circ}{2 \times N})} \times \cos(lat) \quad (3.5)$$

上式 (3.1) 中总时延包括传播时延  $t_p$  和队列时延  $t_q$ ，传播时延较容易估计，上面已经陈述。而总的队列时延与路径经历的卫星节点个数以及在各节点中的排队和处理时间相关，通过分析出此时延的值，可以得出其在端到端时延  $D$  中所占的比例和影响，是衡量路由策略性能的参数，也能指导调度算法和路由算法的设计。卫星节点中的排队时延和处理时延，分别对应上面公式 (3.3) 中的第二项和第一项，与各卫星节点缓存区队列长度、分配的带宽以及分组长度相关，而总的队列时延  $t_q$  可以通过路径中各节点累加得到，因此首先采用排队理论构建模型分析单个卫星节点中排队和处理时延。

本文研究的卫星网络中存在两种业务：话音业务和数据业务，除了业务还有针对语音业务建立连接需要的信令信息，定义信令消息和语音业务的优先级为一级，数据业务优先级为二级，设定其到达都为 Poisson 过程，建立马尔可夫链，认为卫星节点输出端口排队符合 M/M/1/m 模型。模型中设定所有业务服从 DRR 调度规则，根据优先级分配业务可用带宽，通过指针轮询调度队列中各到达的业务包。

卫星节点输入输出端口包括 1 个用户链路端口、1 个馈电链路端口、2 个同轨星间链路端口、2 个异轨星间链路端口。每个端口分配固定容量的缓存区，卫星节点排队模型结构如图 3-4 所示。当卫星从各条链路接收到业务后，进入接入、路由处理模块，进行路由和交换处理，并根据业务类型分配其在信道上可用带宽，然后根据路由信息确定下一跳进入某条输出链路端口，并在对应链路端口的缓存区排队等候被服务。

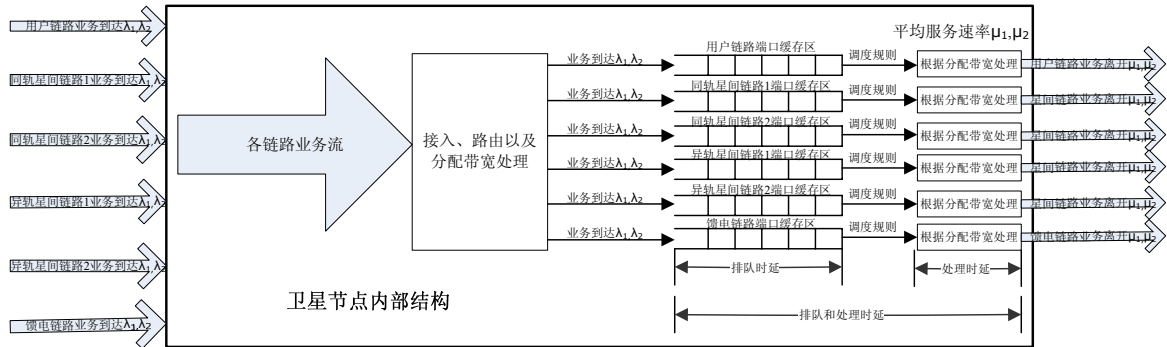


图 3-4 卫星节点排队时延和处理时延的队列模型

分析队列和处理时延，需要对实际系统做一定的简化，设定语音和信令业务到达率为  $\lambda_1$ ，数据业务到达率为  $\lambda_2$ ，其到达为泊松过程，具有独立性和线性的特征，经由接入处理模块后进入各链路缓存区时，其到达仍旧为泊松分布且到达率不变。在星上交换机中分配语音和信令业务传输带宽为  $\phi_1$ ，设置分组长度为  $L_1$ ，其处理时延如公式 (3.6) 所示， $\mu_1$  表示语音服务速率；分配数据业务的带宽为  $\phi_2$ ，设置其分组包长度为  $L_2$ ，其处理时延如公式 (3.7) 所示， $\mu_2$  表示数据服务速率。节点中排队时延和处理

时延的排队模型可以用图 3-5 表示。

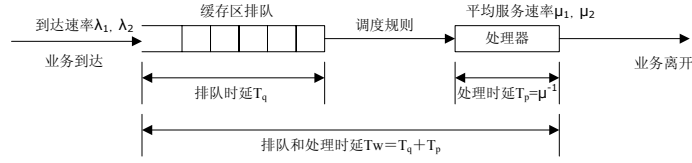


图 3-5 排队时延和处理时延的排队模型

$$\mu_1 = 1 / \frac{L_1}{\phi_1} \quad (3.6)$$

$$\mu_2 = 1 / \frac{L_2}{\phi_2} \quad (3.7)$$

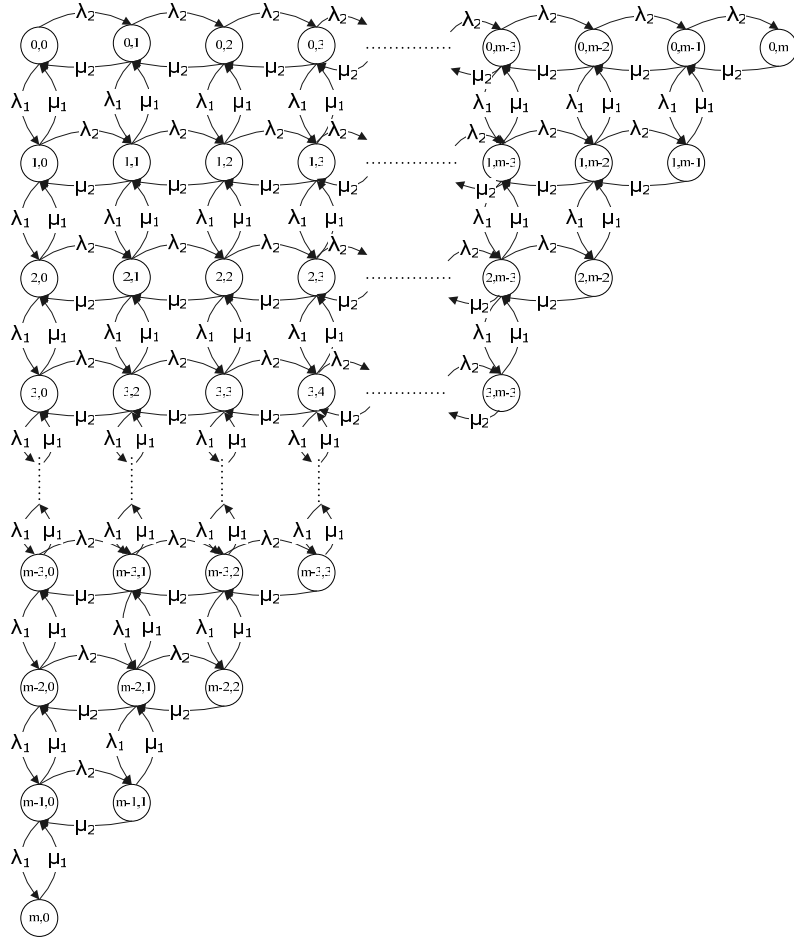


图 3-6 排队系统状态转移图

排队系统采用连续时间的马尔可夫链建模，其状态转移如图 3-6 所示，假设卫星节点某链路对应缓存区状态为  $N(t)$ ，时刻  $t$  的业务数量表示为  $N(t) = (i, j)$ ，其中  $i$  表示网络中语音和信令业务的数量， $j$  表示网络中数据业务的数量，设定缓存区容量为  $m$ 。

显而易见， $\{N(t), t \geq 0\}$  是一个生灭过程，上图遍历了队列中各种状态，以及各

状态转移的情况。设系统处于某状态的概率为  $p(i, j; t)$ ，则  $p(i, j; t) = P\{N(t) = (i, j)\}$ ，令  $\rho_1 = \lambda_1/\mu_1$ ， $\rho_2 = \lambda_2/\mu_2$ ，只有当  $\rho_1 + \rho_2 < 1$  时，系统才会到达平衡状态，平衡方程可表示为：

$$(\lambda_1 + \lambda_2) p(0, 0) = \mu_1 p(1, 0) + \mu_2 p(0, 1) \quad (3.8)$$

$$(\lambda_1 + \lambda_2 + \mu_2) p(0, j) = \lambda_2 p(0, j-1) + \mu_1 p(1, j) + \mu_2 p(0, j+1) \quad (1 \leq j \leq m-1) \quad (3.9)$$

$$(\lambda_1 + \lambda_2 + \mu_1) p(i, 0) = \lambda_1 p(i-1, 0) + \mu_1 p(i+1, 0) + \mu_2 p(i, 1) \quad (1 \leq i \leq m-1) \quad (3.10)$$

$$\mu_2 p(0, m) = \lambda_2 p(0, m-1) \quad (3.11)$$

$$\mu_1 p(m, 0) = \lambda_1 p(m-1, 0) \quad (3.12)$$

$$(\mu_1 + \mu_2) p(i, m-i) = \lambda_1 p(i-1, m-i) + \lambda_2 p(i, m-i-1)$$

$$(1 \leq i \leq m-1; 1 \leq j \leq m-1; i+j=m) \quad (3.13)$$

$$(\lambda_1 + \lambda_2 + \mu_1 + \mu_2) p(i, j) = \lambda_1 p(i-1, j) + \lambda_2 p(i, j-1) + \mu_1 p(i+1, j) + \mu_2 p(i, j+1)$$

$$(1 \leq i \leq m-2; 1 \leq j \leq m-2; i+j \neq m) \quad (3.14)$$

图 3-6 显示两种业务的到达是相互独立的，求解公式 (3.8) ~ (3.14)，各状态的概率可以用 (3.15) 式表达：

$$p(i, j) = \left( \frac{\lambda_1}{\mu_1} \right)^i \left( \frac{\lambda_2}{\mu_2} \right)^j p(0, 0) = \rho_1^i \rho_2^j p(0, 0) \quad (3.15)$$

$$\text{又有 } \sum_{j=0}^m \sum_{i=0}^m p(i, j) = 1 \quad (i+j \leq m) \quad (3.16)$$

联合公式 (3.8) ~ (3.16)，得到：

$$p(0, 0) = 1 / \sum_{l=0}^m \left[ \rho_1^l \sum_{k=0}^{(m-l)} \rho_2^k \right] \quad (3.17)$$

则，

$$p(i, j) = \rho_1^i \rho_2^j / \sum_{l=0}^m \left[ \rho_1^l \sum_{k=0}^{(m-l)} \rho_2^k \right] \quad (3.18)$$

表 3-8 公式中参数的意义

符号	意义
$\lambda_1$	语音业务平均到达率
$\lambda_2$	数据业务平均到达率
$\mu_1$	语音业务平均处理速率
$\mu_2$	数据业务平均处理速率
$T_{p1}$	每个语音业务的平均处理时间
$T_{p2}$	每个数据业务的平均处理时间
$V_1$	处理语音业务时处理器的利用率
$V_2$	处理数据业务时处理器的利用率
$q_1$	系统中的平均语音业务数量（包括等待和正在处理的业务）
$q_2$	系统中的平均数据业务数量（包括等待和正在处理的业务）
$T_p$	每个业务的平均处理时间
$T_q$	一个业务在系统中的平均时间（排队时间和处理时间）
$T_{q1}$	一个语音业务在系统中的平均时间（排队时间和处理时间）
$T_{q2}$	一个数据业务在系统中的平均时间（排队时间和处理时间）
$W$	一个业务的平均等待时间
$W_1$	一个语音业务的平均等待时间
$W_2$	一个数据业务的平均等待时间
$\bar{N}$	节点中所有业务的平均队长
$N_Q$	节点中正在排队的业务数量（不包括正在传输的业务分组）
$N_{Q1}$	节点中正在排队的语音业务数量
$N_{Q2}$	节点中正在排队的数据业务数量

缓存区平均队长为，

$$\bar{N} = E[N] = \sum_{i=0}^m \sum_{j=0}^m (i+j)p(i,j) \quad (i+j \leq m) \quad (3.19)$$

把公式 (3.18) 代入公式 (3.19)，得到：

$$\bar{N} = E[N] = \left( \sum_{t=0}^m \rho_1^t \sum_{s=0}^{m-t} (s+t) \rho_2^s \right) \frac{1}{\sum_{l=0}^m \left[ \rho_1^{(m-l)} \sum_{k=0}^l \rho_2^k \right]} \quad (3.20)$$

对 (3.20) 进行计算后可以得到如下值:

$$\bar{N} = \frac{\rho_1 \rho_2}{1 - \rho_1 \rho_2} - \frac{(m+1) \rho_1^{m+1} \rho_2^{m+1}}{1 - \rho_1^{m+1} \rho_2^{m+1}} \quad (3.21)$$

根据 Little<sup>[110]</sup>定理:

$$\lambda = \lambda_1 + \lambda_2 \quad (3.22)$$

$$T_q = \frac{\bar{N}}{\lambda} \quad (3.23)$$

$T_q$  表示队列时延和处理时延,  $\lambda$  表示系统中各类业务的平均到达率。因此, 在已知  $\lambda$  和  $\bar{N}$  的情况下, 可以求解队列时延和处理时延。

把等式 (3.21) 代入 (3.23), 得到:

$$T_q = \frac{1}{\lambda} \frac{\rho_1 \rho_2}{1 - \rho_1 \rho_2} - \frac{(m+1) \rho_1^{m+1} \rho_2^{m+1}}{1 - \rho_1^{m+1} \rho_2^{m+1}} \quad (3.24)$$

此式算出的结果即是在某个卫星节点中总的队列时延和处理时延, 从式可以看出, 在已知  $\lambda_1, \lambda_2, \mu_1, \mu_2$  和  $m$  的情况下, 根据 (3.24) 求解得出队列时延和处理时延。则平均的排队时延如下式:

$$W = T_q - \frac{1}{\mu_1 + \mu_2} = \frac{1}{\lambda} \frac{\rho_1 \rho_2}{1 - \rho_1 \rho_2} - \frac{(m+1) \rho_1^{m+1} \rho_2^{m+1}}{1 - \rho_1^{m+1} \rho_2^{m+1}} - \frac{1}{\mu_1 + \mu_2} \quad (3.25)$$

根据 Little<sup>[110]</sup>定理:

$$N_Q = \lambda W \quad (3.26)$$

把公式 (3.25) 代入 (3.26) 得到:

$$N_Q = \lambda W = \lambda \left( \frac{1}{\lambda} \frac{\rho_1 \rho_2}{1 - \rho_1 \rho_2} - \frac{(m+1) \rho_1^{m+1} \rho_2^{m+1}}{1 - \rho_1^{m+1} \rho_2^{m+1}} - \frac{1}{\mu_1 + \mu_2} \right) \quad (3.27)$$

对上式简化后得到下式:

$$N_Q = \frac{\rho_1^2 \rho_2^2}{1 - \rho_1 \rho_2} - \frac{(m + \rho_1 \rho_2) \rho_1^{m+1} \rho_2^{m+1}}{1 - \rho_1^{m+1} \rho_2^{m+1}} \quad (3.28)$$

根据排队理论,符合泊松到达多优先级排队中存在如下关系,可以得出队列中各业务所占的比例及消耗的时间:

$$\rho = \rho_1 + \rho_2 \quad (3.29)$$

$$T_{p1} = 1/\mu_1 \quad (3.30)$$

$$T_{p2} = 1/\mu_2 \quad (3.31)$$

$$V_1 = \lambda_1 T_{p1} \quad (3.32)$$

$$V_2 = \lambda_2 T_{p2} \quad (3.33)$$

$$T_p = \frac{\lambda_1}{\lambda} T_{p1} + \frac{\lambda_2}{\lambda} T_{p2} \quad (3.34)$$

$$T_q = \frac{\lambda_1}{\lambda} T_{q1} + \frac{\lambda_2}{\lambda} T_{q2} \quad (3.35)$$

$$q_1 = \lambda_1 T_{q1} \quad (3.36)$$

$$q_2 = \lambda_2 T_{q2} \quad (3.37)$$

$$N_{Q1} = \lambda_1 W_1 \quad (3.38)$$

$$N_{Q2} = \lambda_2 W_2 \quad (3.39)$$

求解等式 (3.8) ~ (3.39), 得到,

$$N_{Q1} = \frac{\rho_1 (\rho_1 T_{p1} + \rho_2 T_{p2})}{T_{p1} (1 - \rho_1)} \quad (3.40)$$

$$N_{Q2} = N_{Q1} \frac{\lambda_2}{\lambda_1 (1 - \rho)} \quad (3.41)$$

$$T_{q1} = T_{p1} + \frac{\rho_1 T_{p1} + \rho_2 T_{p2}}{1 - \rho_1} \quad (3.42)$$

$$T_{q2} = T_{p2} + \frac{T_{q1} - T_{p1}}{1 - \rho} \quad (3.43)$$

根据上述的计算,粗略估计在某卫星节点中最大的等待时延约为 25ms 左右,相当于在卫星与卫星之间经历一跳的传播时间。根据卫星运行周期、业务分组的大小和数量,可以估算出用户通信时端到端的时延。上述理论推导可以估算端到端时延值以及不同业务各时延值,能有效分析网络中各业务服务质量。

### 3.4.4 DRAST 路由策略的执行

DRAST 策略设计的目标是根据简化的动态拓扑模型选择最小耗费路径转发数据包。衡量最小耗费路径由下式来描述：

$$\text{LinkCost} = \text{PWF} \times \text{PropagationDelay} + \text{QWF} \times \text{QueuingDelay}$$

PWF 表征传播时延的权重，QWF 表征队列时延的权重。当分组经过路径每个节点时，记录并累加在各节点中花费的队列时延和处理时延，最后在目的卫星节点统计出此路径的路径花费，传递路径耗费信息到地面信关站。信关站记录并更新接收到的各路径耗费信息。不管是面向连接的语音通信还是无连接的数据业务，路径的选择参考信关站记录的路径耗费信息列表。

分布式路由是指每个网络节点独立进行路由计算，路由决策由数据的目的地址和节点自身得知的网络拓扑决定。针对卫星网络拓扑快速变化和星上资源有限的特点，结合分布式路由本地路由计算的特性，可采用分布式路由的方法减少路由计算量和路由信息的存储量。

语音业务通过信令建立连接后，后续业务分组传输到源卫星（接入卫星），根据卫星记录此会话的连接，在分组头部加入路由信息，形成路由信息帧，经由星间路由到达目的卫星，业务流程如图 3-8 中虚线箭头所示；数据业务分组每次通过源卫星送达地面信关站，通过地面网络查找到目的用户所在目的卫星的覆盖区域，分配数据业务分组头部路由信息，同样形成路由信息帧，再上传到卫星，经由星间路由到达目的卫星。流程如图 3-8 中实现箭头所示。

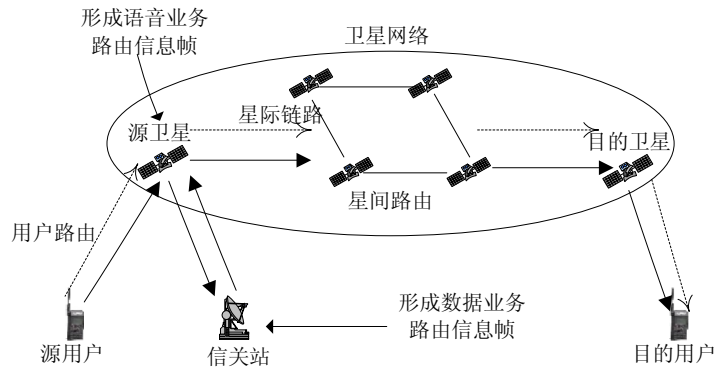


图 3-8 业务分组流程示意图

本文提出的路由算法针对形成路由信息帧后进行，帧格式如表 3-9 所示。表中业务类型表示数据或语音业务，卫星缓存区中根据业务类型进行分组调度；会话序号主要用于语音业务在星上记录的路径信息标号，在一颗卫星上其序号可以循环使用；包序号用于目的用户对接收的包排序，以及对掉包的统计；路径方向下文将描述。

表 3-9 路由信息包格式

业务类型	会话序号	源卫星	目的卫星	包序号	路径方向	业务包
------	------	-----	------	-----	------	-----

详细描述算法前，首先定义几个术语，



上游：同一轨道面某路径所经历的卫星节点编号从大到小的方向定义为上游。

下游：同一轨道面某路径所经历的卫星节点编号从小到大的方向定义为下游。

路由方向：同一轨道面路径的方向（上游或者下游）。

在卫星节点中存储简化的卫星网络拓扑表，业务分组根据当前时刻拓扑表、目的卫星编号和节点自身独立进行同轨路由或异轨路由。

**同轨路由：**源卫星和目的卫星属于同一轨道面即为同轨路由。路由时只考虑同轨星间链路，同轨卫星对之间的路径只有两条，上游或者下游，如图 3-9 表示轨道面 1，卫星运行方向为顺时针。假如源卫星为 Sat11，目的卫星为 Sat14，源一目的卫星的路径如下示：

上游方向：Sat11→ Sat12→ Sat13→ Sat14

下游方向：Sat11→ Sat18→ Sat17→ Sat16 → Sat15→ Sat14

源卫星根据路由信息帧中目的卫星编号确定路由方向。统计得到同轨道各卫星对在上下游方向路由跳数相差为 2、4 或者 6 跳不等。具体执行如下：

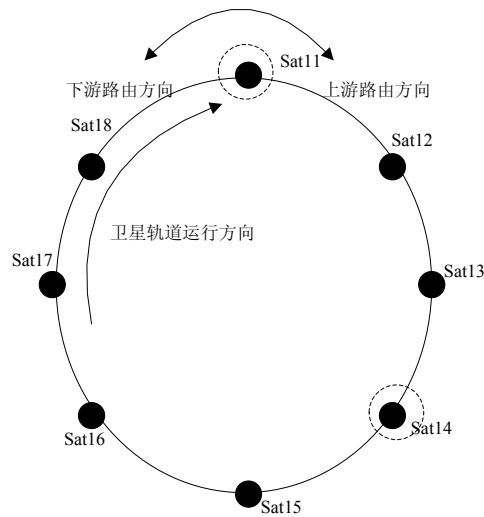


图 3-9 同轨路由示意图

第一步：源卫星中计算上下游两条路径相差的跳数，如果为 6，选择最少跳数路径，转入第三步。

第二步：估算上下游两条路径的星间端到端时延：

每颗卫星定期在同轨道面广播当前时刻的业务数量，因此，每颗卫星了解同轨道其余卫星当前时刻的队列情况。计算同轨路径时根据路径的跳数得到总的传播时延，根据经历的各卫星节点的队列情况估算队列时延和处理时延，两者之和即为星间端到端时延。比较上下游两路径的星间端到端时延，选取时延较小的路径。转入第三步。

第三步：在分组头部路由方向中标记所选路径的方向，在下颗卫星节点时只需要根据路由方向标记选择下一跳。当到达目的卫星后，把分组传递到地面。

为避免耗费带宽资源和减少路由开销，仅在同轨道卫星之间传递各卫星业务状态信息，保证端到端时延性能，同时大大降低了网络的负担和信令开销。

**异轨路由：**源卫星和目的卫星属于不同轨道面即为异轨路由。路由时，只选择源卫星和目的卫星所在两个轨道面卫星节点构成的链路，路径中只经过一条异轨星间链路。在各卫星节点计算路径时，记录所选路径中经过的异轨链路编号，并累加所用次数。

第一步：卫星根据此刻的拓扑表计算最少跳数的路径，如果只有一条，在卫星中记录所经过的异轨链路，转入第三步。如果有多条最短路径，转入第二步。

第二步：比较各条路径中的所经过的异轨链路在此卫星中记录的次数，选择记录次数最少的异轨链路的路径，进入下一步。

第三步：根据路径确定下一跳卫星节点。分为两种情况：当目的卫星与自身节点属于异轨卫星则转入第一步；如果目的卫星与自身节点属于同轨卫星，则进行同轨路由。

异轨路由中选择最短路径时对业务进行分流，降低了发生链路拥塞的概率。

当星间链路由于意外失效时，地面信关站需要及时广播此信息给各卫星节点，并更改各节点中存储的拓扑表；同理，当链路恢复时，信关站需要发送信息给各卫星节点还原各拓扑表。通过信令建立语音业务连接或数据业务直接路由时，选择路径时都会参考信关站的路径耗费信息列表优化路径，也可以为系统中切换等策略作为参考。

DRAST 策略中拓扑更新平缓，结构简单，路由计算简单，即不要求卫星大容量存储空间，也无需较高的处理能力，符合卫星硬件受限的要求，具备稳健性。

### 3.5 仿真设计和结果分析

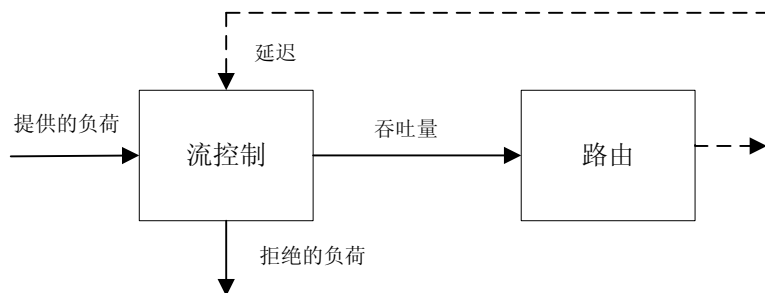


图 3-10 路由和流量控制相互作用

路由算法对于两个主要的性能量度——吞吐量和平均延迟时间有相当大的影响。通过图 3-10 所示的反馈机制路由协议协同流量控制以决定这两个性能量度。当外部子网提供的通信量负荷相对较低时，网络将全部接收该通信量，也就是说

$$\text{吞吐量} = \text{提供的负荷 (offered load)}$$

当提供的负荷过量时，部分负荷将被流量控制丢弃，从而

$$\text{吞吐量} = \text{提供的负荷} - \text{丢弃的负荷}$$

被网络接收的通信量将经过一段平均延迟时间，这一延迟主要取决于通过路由算法选择的路径。但是，吞吐量也会在很大程度上（间接地）受路由算法的影响，这是

因为典型的流控制方案工作在维持吞吐量和延迟之间平衡的基础上（也就是说，当延迟过大时系统将开始丢弃提供的负荷）。因而，路由算法将延迟控制得越好，流控制将允许更多的通信量进入网络。由于延迟和吞吐量之间精确的平衡是由流控制决定的，在提供负荷量较大的条件下，好的路由将对于同一流控制实现一条更好的延迟—吞吐量曲线，如图 3-11 所示。

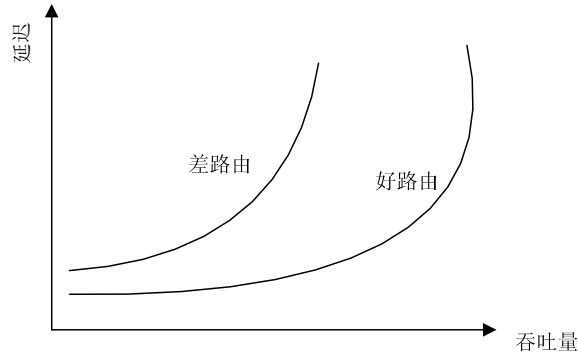


图 3-11 好路由和差路由情况下延迟—吞吐量操作曲线

### 3.5.1 仿真模型设计

为验证本文提出的路由算法的性能，在建立的 OPNET 低轨卫星网络仿真平台中进行仿真。为侧重仿真路由协议，其仿真平台基于第二章进行修改，整个网络模型由卫星子网、3 个信关站节点和多个用户终端节点（均为固定节点）组成，卫星子网由 24/3/1 Walker 星座构成，其轨道模型由 STK 导入，如图 3-12 所示。

#### （1）仿真场景和仿真目的

在网络中信息的流通和交互在软件中主要以包流（即包中断）和 ICI 流实现，包的传输在用户链路、星间链路和馈电链路以无线传输实现，而地面信关站之间以有线传输实现包的流通。此模型重点在于星间路由策略的仿真，根据图 3-11 中的曲线描述，基于对实际的低轨卫星网络系统通信协议的简化，在平台中主要实现以下几个场景的仿真：

场景一：两个远距离用户在网络吞吐量较低状态下数据、语音包的传输；

场景二：两个远距离用户在网络吞吐量较高状态下数据、语音包的传输；

场景三：两个近距离用户在网络吞吐量较低状态下数据、语音包的传输；

场景四：两个近距离用户在网络吞吐量较高状态下数据、语音包的传输。

在上述场景中分别在卫星节点运行 DRAST 策略和基于快照序列路由算法，对两种算法进行性能比较，通过统计在不同吞吐量状态下端到端的时延、星上路由处理时间、掉包率等性能，验证 DRAST 路由协议的优越性。

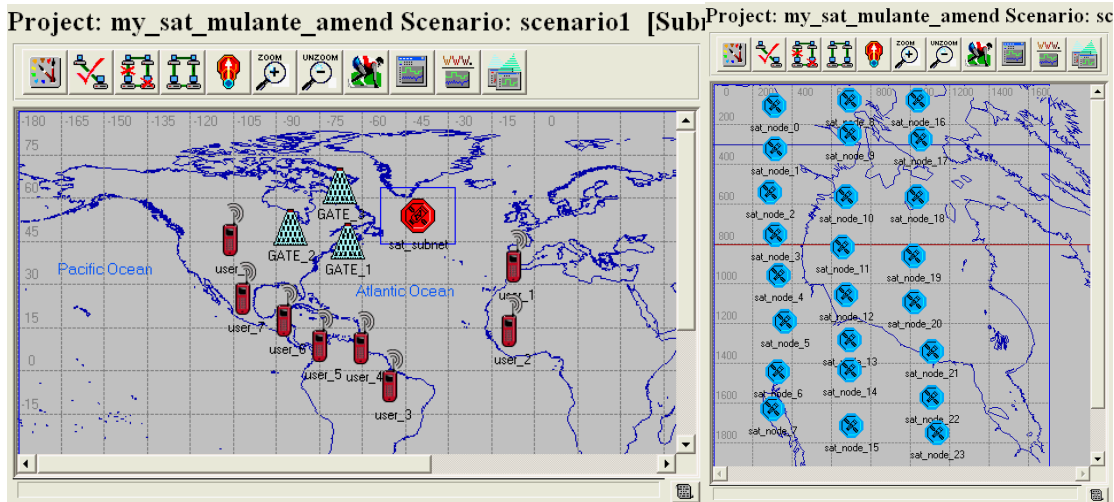


图 3-12 OPNET 中低轨卫星网络和卫星子网模型

### (2) 无线信道配置

根据描述的仿真场景以及仿真目的,模型中理想化物理层的实现,即在各种链路的无线传输中认为无信道衰落、无信道误码率、无各种干扰以及噪声。在 OPNET 仿真模型中,各节点的无线收发信机均涉及信道参数的设置问题,为实现方便,简化处理,设定各节点模型的收发信机调制方式为 QPSK,并添加自定义参数“Communication Direction”和“Link Type”。网络中卫星节点根据 3.4.1 和 3.4.2 中的拓扑结构,对可以建立星间链路的同轨道卫星和异轨道卫星之间建立收发信机对,其对应的信道进行完全匹配(频率、调制方式、带宽、链路类型、通信方向等),执行一对一收发业务包。针对异轨星间链路的关闭与连接,卫星节点链路层根据存储的拓扑表设置。

用户通过计算与卫星节点的距离,选择最短距离卫星实现接入,当选择接入卫星后其收发信机信道配置与接入卫星匹配,即可实现一对一收发业务包。信关站同样根据最近距离选择其所覆盖的卫星,与用户不同的是可与之通信的卫星为两到三颗,因此信关站需要设置两到三对收发信机,并一一根据可通信卫星配置其信道参数。

综上所述,卫星各信道参数配置固定,用户和信关站根据最短距离接入原则更新信道配置;同轨星间链路对应的卫星对收发信机信道匹配且固定,仿真中持续可通信;异轨星间链路根据全拓扑表设置可通信卫星的收发信机,使对应的收发信机信道完全匹配(信道配置固定不变),物理上链路持续可通信,实际仿真中异轨星间链路的连接取决于当前时刻的拓扑表,此决策由卫星节点的链路层实现。当运行基于快照序列路由算法时,卫星节点根据轨道周期内的全拓扑表(48 张全拓扑表)确定异轨星间链路的连接,而运行 DRAST 策略时根据简化拓扑表(24 张拓扑表)确定异轨星间链路的连接。模型中物理层各链路的可靠性通信是仿真的基础。

### (3) 基于包的通信过程

仿真中采用基于包的通信机制,根据 2.1.3 节中描述的语音和数据通信流程,包在

仿真模型中的流程如图 3-13 所示：

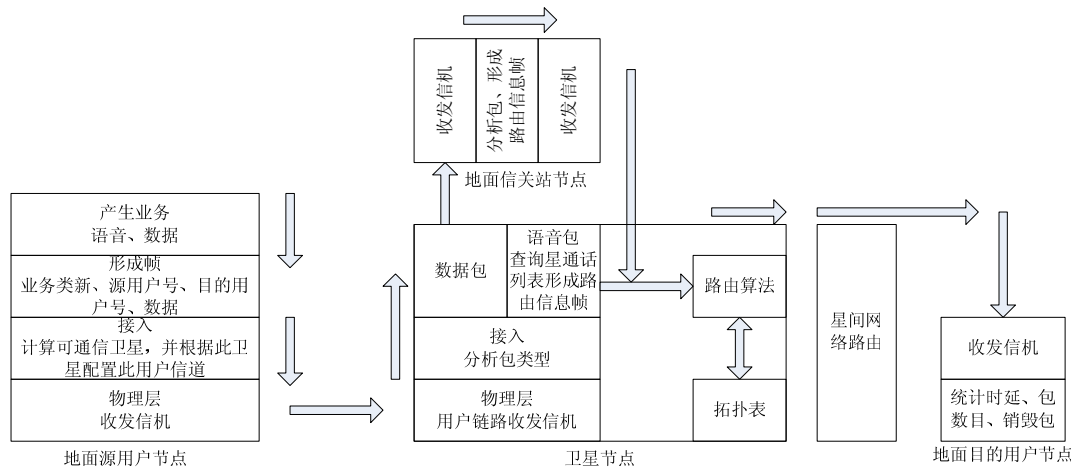


图 3-13 仿真模型中业务包流程

数据包：

源用户（产生业务分组）→接入卫星（源卫星）→信关站（形成路由信息帧）→卫星→星间路由（星间链路）→目的卫星→目的用户；

语音业务首先根据 2.1.3 节中描述的流程建立逻辑连接，其所需的信令和格式本文不再详细描述，建立连接后语音包的流程如下：

源用户（产生业务分组）→源卫星（查看星上会话列表，形成路由信息帧）→星间路由（星间链路）→目的卫星→目的用户。

#### （4）节点和进程模型设计

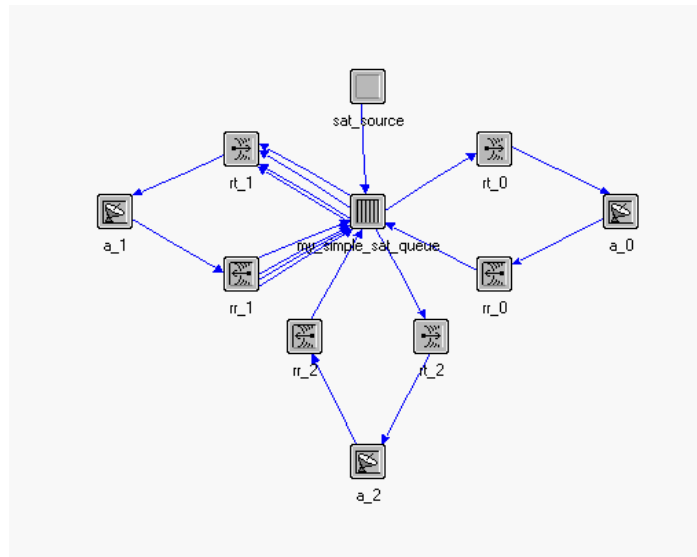


图 3-14 卫星节点模型

根据上述信道配置和包通信流程，网络中卫星节点如图 3-14 所示。卫星节点中各模块描述如表 3-10 所示。

表 3-10 卫星节点模块描述

模块名称	功能描述
rt_0/rr_0、rt_1/rr_1、rt_2/rr_2	用户链路、星间链路及馈电路径收发信机
a_0、a_1、a_2	用户链路、星间链路及馈电路径全向天线
my_simple_sat_queue	分析包、存储拓扑表、路由计算、组帧送往下一个节点
sat_source	星间网络业务流模拟

无线管道模型在节点的天线和收发信机模块设置，无线信道配置如上所述。

为了模拟星间路由算法在不同网络业务量情况下的性能，卫星节点中增加 **sat-source** 模块，模拟用户终端自主产生业务量(业务的目的地均匀分布于 24 颗卫星)，仿真模型中仅布置两对用户终端。此方法不但能加快仿真速度，也较为真实模拟网络处于不同业务流量下算法的性能。

为了编写代码方便，卫星节点直接利用进程模型模拟卫星功能，用多个状态机划分各协议层次，因此，节点中的 **my\_simple\_sat\_queue** 模块承担卫星节点物理层以上的所有功能，其进程模型如图 3-15 所示。

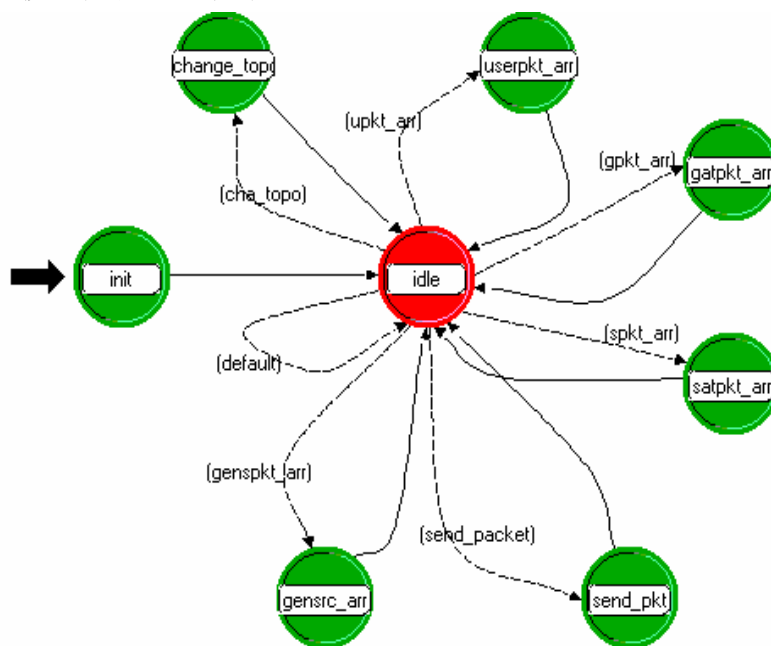


图 3-15 卫星进程模型

进程模型中各状态机的说明如表 3-11 所示。此进程融合卫星的接入层、链路层以及交换路由层，实现解包、转发包、形成路由信息帧、路由计算、分组交换调度等功能。进程的 FB 函数中实现路由算法的编码，是本仿真中的核心。

表 3-11 my\_simple\_sat\_queue 进程模块的状态描述

状态	功能描述
init	初始化节点属性、状态变量以及各种参数
change-topo	存储周期内拓扑表并根据拓扑时刻和仿真时间更新
userpkt-arr	接收处理用户链路收信机的包、路由计算其下一跳、存储到相应输出链路缓存区
gatpkt-arr	接收处理星间链路收信机的包、路由计算其下一跳、存储到相应输出链路缓存区
satpkt-arr	接收处理馈电链路收信机的包、路由计算其下一跳、存储到相应输出链路缓存区
send-pkt	轮询调度各输出口缓存区业务包到对应发信机
gensrc-arr	处理卫星自己产生的包并路由计算下一条、存储
FB	在进程模型中的FB函数中编写路由算法代码

地面的用户节点和信关站节点功能比较简单。用户主要产生两种业务：一种为数据业务，到达为泊松过程（模型中设置分组到达时间参数、分组长度）；另外一种为语音业务，不仅需要产生业务包，同时为建立逻辑连接需要产生各种信令包，信令包的种类和格式请参看2.1.3节的流程描述。语音业务包的属性如下图3-16所示，包括：呼叫到达率、呼叫持续时间、呼叫间隔时间，呼叫分组长度以及信令与业务包之间的关系参数等。用户节点还需要统计时延、发送包数目和接收包的数目等。

Attribute	Value
name	User_1
model	my_user_test1
x position	-94.4
y position	27.6
threshold	0.0
icon name	fixed_comm
TIM source	none
altitude	0.0
altitude modeling	relative to subnet-plat
condition	enabled
financial cost	0.00
priority	0
user id	0
connect.Call Arrival Rate	50
connect.Call Duration Time	160
connect.Call Interarrival Time	160
connect.Call Packet Length	272
connect.Originate Call Symbol	1
rr.channel [0].min frequency	promoted
rr.channel [1].min frequency	promoted
rt.channel [0].min frequency	promoted
rt.channel [1].min frequency	promoted

Extended Attrs. ☐ Apply Changes to Selected Objects ☒ Advanced

Details Promote Cancel OK

图 3-16 用户节点属性

模型中信关站节点辅助完成端到端通信：查找目的用户所在卫星覆盖区域、路由各种信令信息、形成数据业务路由信息帧等。地面信关站之间传递用户终端所在卫星覆盖区域和卫星所在信关站覆盖区域的信息，用于查找源和目的用户所在位置，保证可靠的通信流程。



### 3.5.2 仿真结果

模型中各参数设置如下：

卫星节点：缓存区总容量为 60kB，每颗卫星分配一个用户链路缓存区、一个馈电链路缓存区和四个星间链路缓存区，平均每个缓存区容量为 10kB。

设置两个固定用户通信，产生语音业务和数据业务，用户业务（语音业务或数据业务）平均速率为 9.6kbps（OPNET 中各节点无线信道速率统一设置为此值）。数据业务（泊松分布）到达率为 0.6，长度为 480 字节。语音业务每次通话时间持续为 160s，通话期间平均 0.8 秒产生一个包，分组长度为 272 个字节。语音业务的产生同样为泊松过程，到达的负指数分布为 0.8。

实际低轨卫星网络链路带宽较宽，业务量较大，为了加快仿真速度，在软件仿真中对真实系统的业务通信量、带宽容量和卫星节点缓存区容量按照同比例降低。因此，仿真模型中用户链路带宽设置为 1Mbps，馈电链路和星间链路带宽为 2Mbps。

定义单个卫星节点吞吐量 10kbps 为网络较闲状态，单个卫星节点吞吐量 10Mbps 为网络较忙状态；当两个用户经纬度分别为  $(-80, 22)$ ， $(8, 6)$  时定义为远距离用户，当用户的经纬度分别为  $(-95, 21)$ ， $(-77, 22)$  时定义为近距离用户。在 3.5.1 节中描述的场景中分别运行 DRAST 策略和基于快照序列路由算法，通过下述仿真结果比较两种算法的性能：

- (1) 统计分组从源用户到达目的用户的平均端到端时延；
- (2) 统计源用户发送的分组数目，以及目的用户接收的分组数目，统计掉包率；
- (3) 统计程序运行时间粗略比较算法计算的复杂度。

➤ 场景一仿真结果：远距离用户/单个卫星节点吞吐量 10kbps

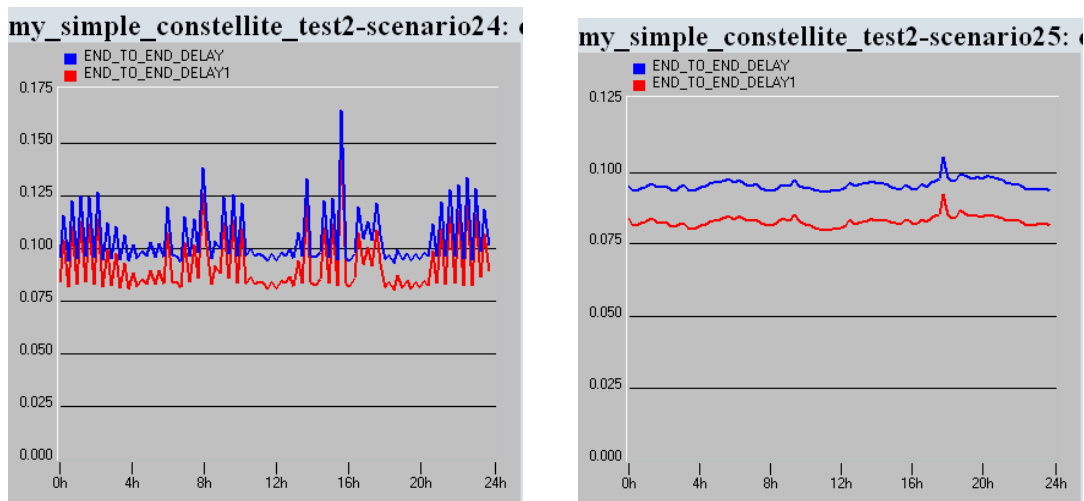


图 3-17 (a) 快照序列路由语音/数据分组时延

(b) DRAST 策略语音/数据分组时延

场景一中端到端时延性能如图 3-17 所示。横坐标表示仿真时间(单位：小时)，纵坐标表示业务分组从源用户到达目的用户端到端的时延（单位：秒）。(a) 图表示基于快照序列路由算法时语音/数据业务分组的平均时延曲线，由于吞吐量较低，时延



低于 0.2s。(b) 图表示执行 DRAST 策略时语音/数据业务的平均时延曲线。从图中可以看出右图中时延稍微低一点, 最重要的是时延曲线平缓, 表明时延抖动小。与快照序列路由策略比较, 结果显示 DRAST 算法时延性能较好。另外上述两图中的曲线显示(时延较大的蓝曲线代表的数据业务, 时延较低的红曲线代表语音业务)面向连接的语音业务时延性能比无连接的数据业务时延性能优越。场景一仿真时间为 24 小时。

➤ 场景二仿真结果: 远距离用户/单个卫星节点吞吐量 10Mbps

用户位置不变, 当卫星网络业务量增大, 单个卫星节点吞吐量为 10Mbps (网络较繁忙) 时, 时延曲线如图 3-18 所示。图中显示当业务量增大时, 端到端时延变大。执行基于快照序列路由算法时, 由于没有考虑队列时延和执行分流机制, 业务集中于一部分星间链路, 导致较大时延抖动, 如 (a) 图所示, 时延最大到达 2s 左右 (为了显示曲线的精细度, 截掉了时延最大到 2s 的峰值), 引起掉包现象。因为卫星节点业务量较大带来排队时延的增长, (b) 图显示时延比 3-18 (b) 图有所增加。在网络较为繁忙状态下 DRAST 策略能保证良好的时延性能。如场景一所示, 下图中显示语音业务时延性能比数据业务时延性能较好。场景二仿真时间为 10 小时。

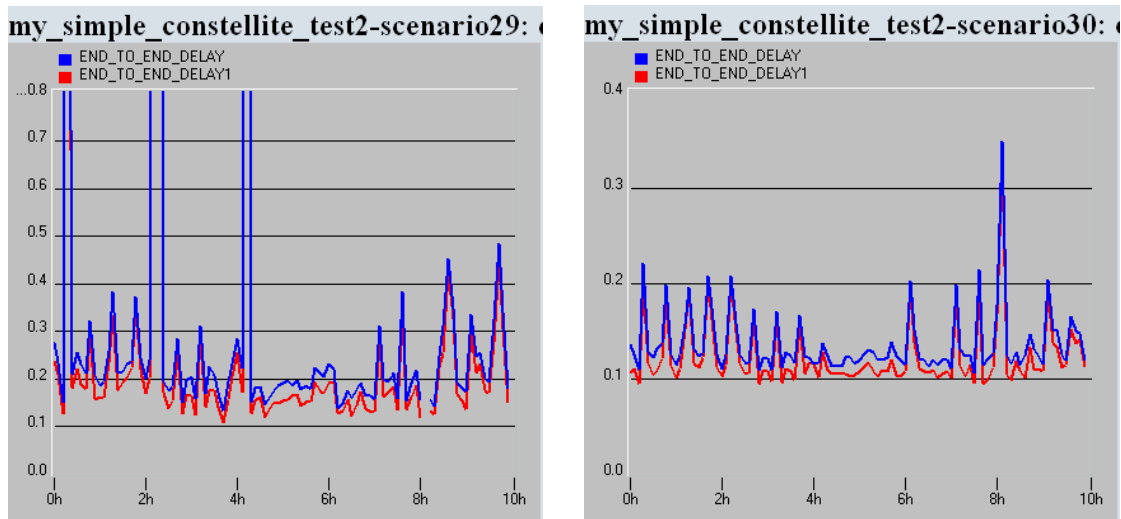


图 3-18 (a) 快照序列路由语音/数据分组时延

(b) DRAST 策略语音/数据分组时延

DRAST 算法执行简单, 其复杂度为  $O(0.14N^2)$ 。表 3-12 通过比较不同算法下程序运行仿真的物理时间可以粗略衡量处理程序的复杂程度 (OPNET 中仿真时间只是设定的仿真实际系统运行的时间, 根据仿真模型复杂程度, 运行仿真需要的物理时间可以比仿真时间长或者短。因此, 当在同一台电脑以及同一仿真模型中, 当仿真输入和场景设置相同, 如果模型中仅仅某一部分处理程序不同, 我们可以通过比较运行仿真的物理时间粗略衡量处理程序的复杂程度)。同时, 表 3-12 中列出上述两种场景中的掉包率。由于星间链路切换频率降低, 相对基于快照序列路由, DRAST 策略掉包率较低。

表 3-12 两种算法不同吞吐量下性能比

算法类型	吞吐量(10kbps)/仿真时间:24hours		吞吐量(10Mbps)/仿真时间:10hours	
	掉包率	运行物理时间	掉包率	运行物理时间
快照序列路由算法	0.07%	37 分 7 秒	4.93%	7 小时 50 分
DRAST 算法	无	20 分 1 秒	无	4 小时 58 分

同时,本文在 OPNET 的 ODB 调试中打印出从仿真开始运行到 28 分钟时的拓扑更新(由于软件限制,不能在同一界面打印出一个轨道周期内的运行情况),以及各更新时刻路径所经历的卫星节点编号。图 3-19 为 DRAST 算法的结果,显示拓扑更新 6 次,路径中异轨链路经历 1 次切换;图 3-20 为基于快照序列路由算法的结果,显示拓扑更新 12 次,路径中异轨链路经历 3 次切换。仿真结果表明 DRAST 算法有效降低了拓扑更新频率和链路切换频率。

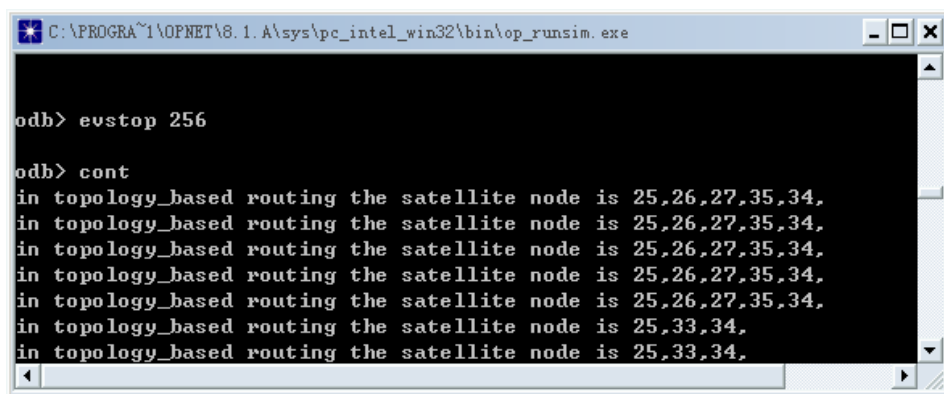


图 3-19 DRAST 算法路径经历的卫星节点

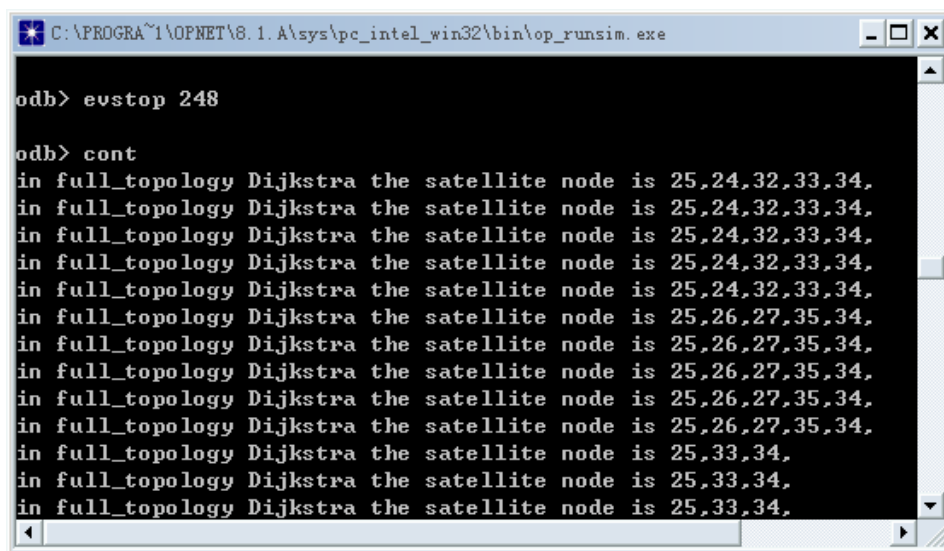


图 3-20 快照序列路由算法路径经历的卫星节点

#### ➤ 场景三仿真结果: 近距离用户/单个卫星节点吞吐量 10kbps

在场景三仿真中两个用户距离缩短,首先在单个卫星节点吞吐量为 10kbps 下仿真,其端到端时延性能如图 3-21 所示。仿真时间为 4 小时,DRAST 策略时延较低(平均低于 100ms)。在此场景中无论执行哪种路由策略都无掉包现象发生。同理,任何

策略执行下语音业务时延性能略好于数据业务。

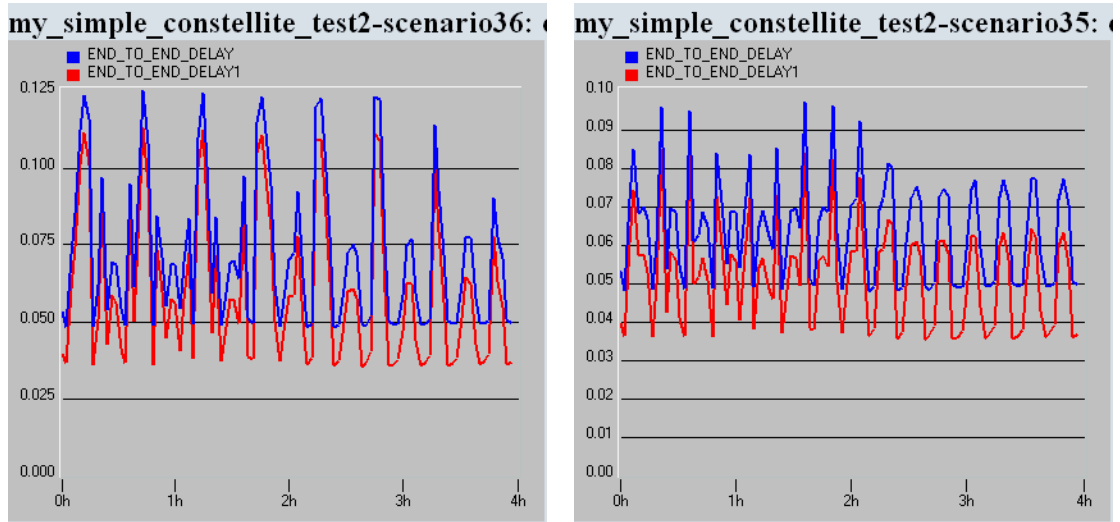


图 3-21 (a) 快照序列路由语音/数据分组时延

(b) DRAST 策略语音/数据分组时延

➤ 场景四仿真结果：近距离用户/单个卫星节点吞吐量 10Mbps

当卫星网络节点吞吐量为 10Mbps 时，基于快照序列路由算法中最大时延达到 2s 左右，为了清晰显示平均时延截掉曲线图顶端（图中有到达无穷顶点的延伸），且时延抖动厉害，由于没有考虑节点中的队列时延，导致拥塞带来 1% 的掉包率。DRAST 算法时延性能比网络吞吐量小时稍有加大，时延抖动平缓，无掉包，结果显示 DRAST 算法符合图 3-11 所示好的路由算法标准（吞吐量加大，时延性能维持整体较好的趋势）。

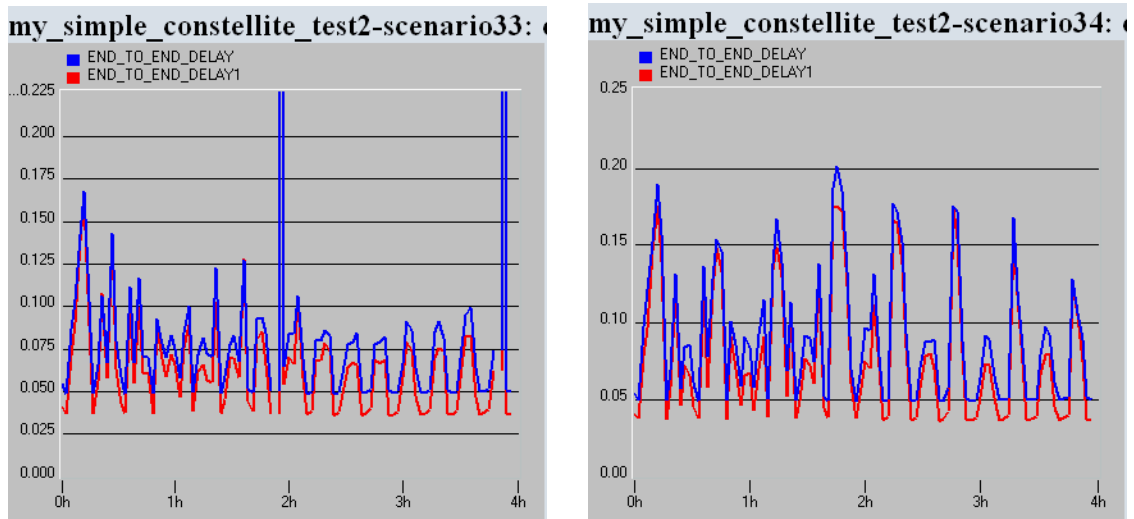


图 3-22 (a) 快照序列路由语音/数据分组时延

(b) DRBST 策略语音/数据分组时延

综合上述仿真结果，DRAST 算法有以下几方面的优势：

1. 拓扑更新频率降低，且结构简单，有效降低星间链路切换频率；
2. 保证良好的时延性能；
3. 计算复杂度低，对卫星存储空间和硬件处理能力的要求降低；
4. 路由协议开销小，有效利用卫星局部业务状态信息；

5. 通过分流避免了链路拥塞;
6. 系统能根据拥塞、故障等链路状态更新拓扑表, 及时调整分组路由, 所以具有非常好的鲁棒性。

下一步将结合调度策略进行路由, 合理分配星间各链路带宽, 保证不同类型的业务的 QoS 性能。

### 3.6 本章小结

本章重点讨论了 24/3/1 Walker 低轨卫星网络的拓扑结构, 根据其特征对拓扑进行了简化, 针对简化后的拓扑结构提出 DRAST 策略; 对卫星网络中端到端时延性能进行了理论的估算, 针对较为复杂的队列时延参数, 采用排队理论建立多优先级模型, 给出了求解方法和结果, 详细描述了 DRAST 的执行; 同时在 OPNET 低轨卫星通信网络仿真平台中针对路由策略建立了多个仿真场景, 在场景中设置不同的输入参数和条件, 分别运行 DRAST 算法和基于快照序列路由算法, 分析并总结仿真结果, 验证了 DRAST 算法的优越性。

本章的主要创新工作包括:

(1) 针对高动态变化的低轨卫星网络, 分析了星间链路的变化和特征, 基于分析简化了卫星网络拓扑结构, 在此基础上, 提出了 DRAST 策略。

(2) 建立了端到端时延参数的马尔可夫多级排队论模型, 给出了端到端时延参数理论上的求解方法。

(3) 在 OPNET 中建立了低轨卫星通信网络仿真模型, 此模型在实现语音、数据等通信流程基础上能有效验证路由算法的性能。

路由算法的有效性对网络设计以及优化至关重要, 在路由中主要考虑的是各业务的时延性能, 在 DRAST 策略中, 为保证端到端时延性能, 不仅仅考虑了节点之间的传播时延, 还考虑了分组在节点中的排队和处理时延, 选择最佳路径时, 避免选择队列较长的节点, 能避免某些节点业务量过多造成拥塞。

从上述分析看出, 有效地转发卫星节点缓存区各队列中的分组很重要, 如果分组及时从缓存区输出到目的链路, 即能有效地降低其在队列中的等候时间, 同时节省缓存区空间, 避免拥塞, 分组调度算法的重要性不言而喻, 本文下一章提出了有效的适合星载交换机的分组调度算法。

## 第四章 低轨卫星通信网络中的分组调度算法

星上具有交换功能,使得信息选路更加方便,并可减少信息传播中使用地球中继站的次数。星载交换技术作为通信星座系统与卫星有效载荷的核心技术,用于完成通信星座系统空间段的信息交换和组网功能,为用户终端、信关站和卫星提供维持星座业务正常运行所需业务的传送承载交换通路。星座网络除了支持系统的业务信息传送之外,还需要提供星座系统信令传送通路。同时为了实现对整个星座网络的测控和载荷的网络管理功能,需要利用星座网络对测控/管理信息进行路由和交换,以完成对地面测控站覆盖范围之外的星座卫星的测控和业务管理。

对于空间交换网络,由于每颗卫星和每条星际链路的业务量大且时段分布不均匀,网络拓扑结构随时间变化,为了有效地利用空间资源,针对具体的低轨星座系统和业务模型,需要在合适的星上交换机结构中应用复杂度低的调度算法,保证低时延并充分理利用有限的带宽资源,降低掉包率,达到用户的 QoS 要求,提高吞吐量。

本章首先针对星座系统特征和业务模型提出合适的星载交换机结构。分析现有的应用于卫星网络的分组调度算法,根据星载交换机结构和业务类型提出了一种新的 SDRR (Subdivision Deficit Round Robin) 调度算法,详细描述了算法以及在星载交换机中的具体实现,建立时延分析模型。最后基于 OPNET 卫星网络仿真平台建立验证调度算法的模型,仿真并验证了本 SDRR 调度算法。

### 4.1 星载交换机结构

#### 4.1.1 交换机结构概述

基于交换的分组排队在交换机中的位置,交换机结构可以分为三类:输出排队 OQ (output queue) 结构、输入排队 IQ (input queue) 结构和联合输入输出排队 CIOQ (combined input output queue) 结构。

OQ 结构在输出端口设置缓存器,是比较传统的交换机结构,该结构易于控制和调度。但在极限情况下,其内部运行速度为线速 (link speed) 的  $N$  (表示输出端口数目) 倍,不利于扩展,不能适用于高速交换机。

IQ 结构在输入端口设置缓存器,内部工作在线速,具有良好的扩展性。例如,对于线速为 10Gbit/s 的  $32 \times 32$  交换机, OQ 结构要求缓存器在 1.6ns 之内完成一次读写操作,而 IQ 结构要求 51.2ns。

由于采用 FIFO (first in first out) 排队的 IQ 机构存在输入缓存中头分组堵塞问题,通过率只有 58.6%<sup>[111]</sup>。为此人们曾提出在交换机交叉节点 (crossbar) 配置缓存器,以及同时在输入输出端口配置缓存器的 CIOQ 结构,提高内部工作速度的方法。CIOQ 结构分组时延和吞吐量等性能是内部加速因子的函数,加速因子在 1 和  $N$  (输出端口

数)间取值。为了能在较短时间内达到输入到输出的最大化双向匹配,需要选取合适的加速因子,调度算法复杂度也较高,同时内部工作速度也需大于线速。端口数越多交换矩阵以平方值增长,存在类似于 IQ 的问题。并且其应用主要是针对定长分组来传输。

针对 IQ 结构的头阻塞问题,直到采用虚拟输出队列<sup>[11]</sup> (VOQ, virtual output queue) 的提出,才使问题得到彻底解决,吞吐量可达到 100%。所谓 VOQ 结构就是在 IQ 结构的每一个输入端口,物理地或逻辑地配置 N 个去往不同输出端口的缓存器,到达的分组依照其去往的输出端口,在相应的缓存器中排队。

VOQ 的关键问题是对分组传输顺序的控制较为困难和复杂,要执行三次竞争裁决才能完成一个分组的调度。1) 分组在 VOQ 队列中竞争成为头信元,目前绝大多数算法采用了 FIFO 的策略,即基于端口 (port based) 的算法; 2) 输入端口中各 VOQ 的头分组竞争离开输入端口,即输入竞争; 3) 赢得输入竞争的分组竞争其目的输出端口,即输出竞争。这些竞争之间相互关联和制约,各端口竞争裁决结果往往出现冲突。若要提供确定性 QoS 保障,则需要保持输入输出竞争结果之间的协调一致,保证分组能够适时传输。

星上交换机由于资源受限 (体积、重量、执行和处理速度),且不可重构。因此为了适应高速环境下的应用,本文选取近年来研究较多的 CICQ (combined-input-crosspoint-queue) 结构作为星载交换机的结构。CICQ 是一种特殊的 IQ 结构,在每个输入端口对应于不同输出端口设置 VOQ 队列,特殊之处在于在交叉交换节点上配置一定尺寸的缓存区。如图 4-1 所示。

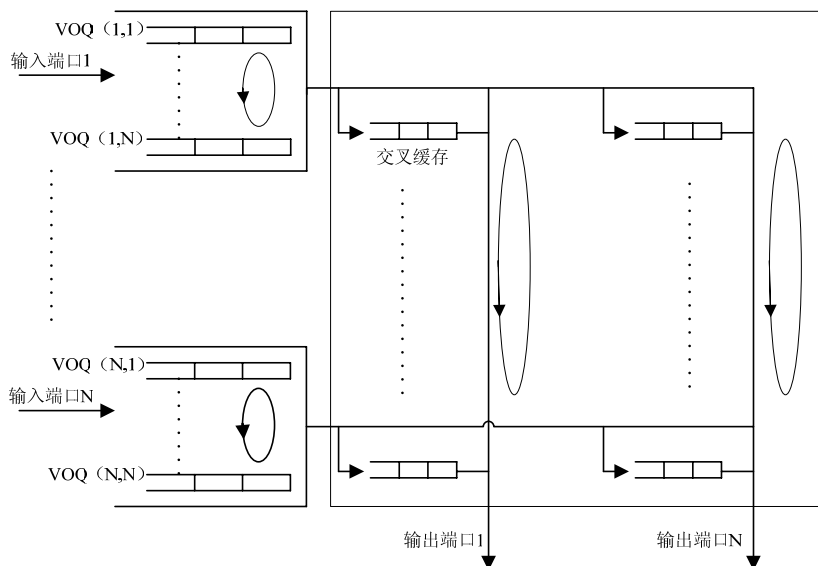


图 4-1 CICQ 结构

CICQ 中分组经过输入竞争传输到交叉节点缓存,再从交叉缓存中通过输出竞争传输到输出端口,竞争在各端口独立地裁决。因此,CICQ 能够去除 VOQ 结构中

三类制约关系中的两类，即输入/输出端口之间的关联性。为防止分组溢出，在输入端口与交叉点缓存之间要采用流量控制。目前一般采用基于信用（credit）的方法，即只有缓存器没有满的时候才可以从输入端口传输分组到缓存。

CICQ 中分组调度研究的主要内容就是协调或同步输入输出竞争的裁决。因为分组经过输入竞争进入缓存，若输出竞争裁决不能及时地将其选中输出，其时延就可能较大。在极限的情况下，该分组将不被选中传输，从而导致交换机的不稳定。当容量有限的缓存被占满时，输入竞争就受到输出竞争的影响。而对于要求确保时延的分组，两者之间的关系就更加重要。所以实际应用中，CICQ 中分组调度的性能由算法和交叉节点缓存容量决定。考虑到在星载交换机中物理实现的条件，交叉点缓存区的缓存容量设置有限。为了加深输入/输出端口之间的隔离程度，使业务分组获得 100% 通过率，调度算法是关键。所以，星载 CICQ 交换机中分组调度是一个具有极大挑战性的难题。

#### 4.1.2 星载交换机模型

如第一章所述，星座通信系统由空间段、地面段和用户段组成。分组交换模式的业务流程如图 4-2 所示。

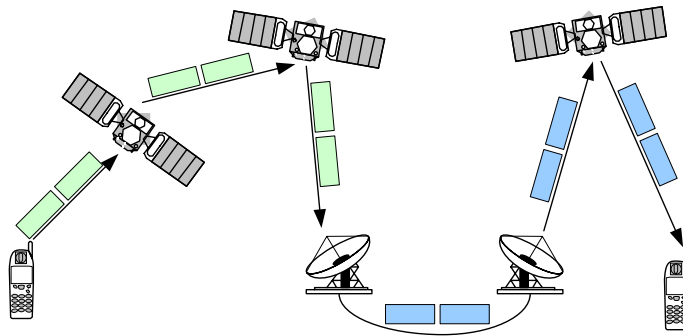


图 4-2 分组交换模式下的业务流程示意图

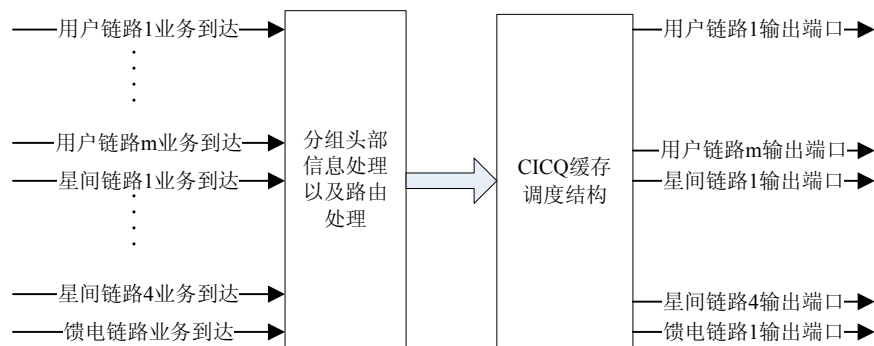


图 4-3 星载交换机结构简图

星载交换机框图如 4-3 所示。在星载交换机中有多条用户链路端口、4 条星间链路端口和 1 条馈电链路，业务具有突发性，不仅有实时和非实时业务信息，还有各种

信令信息、测控信息和管理信息，不同类型信息服务要求不同。

星载交换机与地面交换机不同之处：

- (1) 卫星对应的是星地、星间的无线传输信道，输入输出端口对应用户链路、星间链路和馈电链路。信道误码率高，并有多径效应；
- (2) 位置相关且时变的无线链路；
- (3) 稀有的带宽资源；
- (4) 星上功率、体积以及处理能力受限；
- (5) 星载交换机各输出口链路带宽不一致，需要合理分配输入/输出端口处理速度和缓存区大小；
- (6) 业务种类多，各业务对时延性能要求不同，系统中根据信令类、测控类以及业务类分为三种长度不一的分组。

因此，星上交换系统设计中涉及的问题包括：通信协议、交换体制、业务信息流程、信令系统、星地功能分配等。由于星上存储器容量不像地面那样宽裕，且空中接口的往返传播时延较大，设计简单有效的调度算法，能更好地控制流量，降低分组在交换机中的排队时延，减少对缓存区容量的要求，同时避免拥塞引起掉包。

## 4.2 典型的星载交换机调度算法

下面详细描述几种比较经典的星上调度算法。

### (1) 类 GPS 算法描述

在类 GPS 算法中最具代表性的是 WFQ<sup>[57]</sup>算法，其系统虚拟时间函数  $v(t)$  如式 (4.1) 所示。 $B(t)$  是在时刻  $t$  中存储在服务器中所有的会话， $r$  代表总的带宽， $r_i$  代表会话  $i$  获得的服务带宽，则  $\phi_i = r_i/r$  表示会话  $i$  在服务器中所占的服务额度。

$$\frac{dv(t)}{dt} = \frac{r}{\sum_{i \in B(t)} r_i} = \frac{1}{\sum_{i \in B(t)} \phi_i} \quad (4.1)$$

维持系统虚拟时间的复杂度为  $O(V)$ ， $V$  表示在此刻加入或离开系统的业务流（会话）。当新的分组到达系统时更新系统虚拟时间函数，并根据此函数计算各分组的虚拟完成时间（称为时间戳）。

1. 第  $i$  个会话的第  $k$  个分组表示为  $p_i^k$ ，在时刻  $t_{a_i}^k$  到达时计算时间戳  $F_i^k$ ：

$$F_i^k = \frac{L_i}{r_i} + \max\{F_i^{k-1}, v(t_{a_i}^k)\} \quad (4.2)$$

$$F_i^0 = 0 \quad (4.3)$$

式中  $L_i$  表示分组长度。

2. 服务器根据 (4.2) 式计算的虚拟完成时间递增的顺序选择分组传输。

当某会话占用的带宽较大时，WFQ 会在很长的一段时间内仅仅服务于此会话，



导致其余业务被延迟。针对上述缺点，WF<sup>2</sup>Q<sup>[58]</sup>被提出，与 WFQ 不同之处在于仅仅选择正在开始接受服务的分组，而不是选择队列中的所有分组，其公平性大大提高。但无论 WFQ 还是 WF<sup>2</sup>Q 计算复杂度较大，一旦有分组到达，需要更新系统虚拟时间，重新计算并对比时间戳选择分组。

为降低算法复杂度，提出了各种类 GPS 算法：VC<sup>[59]</sup>（Virtual Clock）使用实际时间作为系统虚拟时间函数，算法复杂度仅为  $O(1)$ ，但公平性较差；SCFQ<sup>[60]</sup>（Self-Clocked Fair Queueing）根据当前时刻正在接受服务分组的虚拟完成时间估计系统虚拟时间函数，导致不同会话的虚拟完成时间一致。

### （2）极大匹配算法描述

PIM<sup>[63]</sup>算法通过请求、允许和接收三步平行迭代匹配交换机输入/输出口，迭代中选择匹配的输入/输出口时采用随机策略，因此需要经过  $O(\log N)$  次迭代才能达到最大匹配，导致不公平性。针对 PIM 中的高复杂度和不公平性，RRM<sup>[64]</sup>在三步迭代算法中采用指针选择可匹配的输入/输出口，但出现同步现象，当输入为 i.i.d 均匀伯努力流时，单次迭代不能达到 100% 的吞吐率。为消除同步现象，iSLIP<sup>[65]</sup>算法改进了 RRM 算法指针的更新策略，iSLIP 算法避免了饥饿现象，消除了同步。

类 GPS 和极大匹配算法执行和计算都比较复杂，不适用于星载交换机调度。

### （3）轮询调度算法

轮询调度算法中最具代表的就是 WRR<sup>[72]</sup>（weight round robin）调度，其基本结构如图 4-4 所示。假定输出链路速率为  $r$ ，分配给各业务流的速率定义为  $r_i (i=1, 2, \dots, N)$ ，则可以得到：

$$r = \sum_{i=1}^N r_i \quad (4.4)$$

$$W = \sum_{i=1}^N w_i \quad (4.5)$$

$W$  是单次循环中服务器可以传输的最大分组数目或最大比特数。由此得到：

$$\frac{w_i}{W} = \frac{r_i}{r}, i=1, 2, \dots, N. \quad (4.6)$$

定义  $\phi_i = w_i/W$ ，表示业务流  $i$  在链路带宽中所占的比例，而  $w_i$  称为服务额度，服务器根据既定的顺序和服务额度循环服务各业务流。

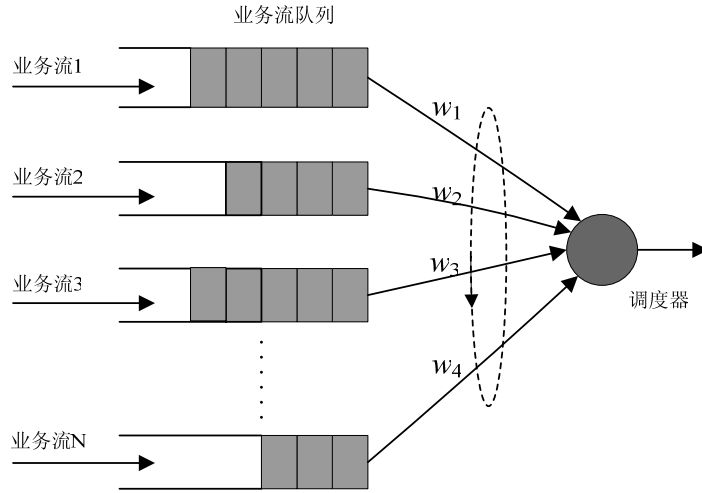


图 4-4 轮询调度结构图

MCCR<sup>[79]</sup> (minimized cycle round robin) 是 WRR 简单的扩展, 通过统一比例缩小各业务流服务额度, 达到最小化一轮传输的业务总量。此算法最坏状态下时延为:

$$\theta_i = (W - w_i) \frac{L}{r} = W(1 - \phi_i) \frac{L}{r} \quad (4.7)$$

JVCRR<sup>[74]</sup> 结合了轮询调度策略和类 GPS 算法的特点, 其服务顺序固定且循环, 类似轮询调度, 但服务额度模仿跳变 VC 策略动态地根据输入输出状态调整。

轮询算法复杂度低, 但时延性能和公平性比类 GPS 差, 针对此缺陷本文提出 SDRR (Subdivision Deficit Round Robin) 调度策略, 时延性能提高, 复杂度不变, 适合星载交换机。

### 4.3 SDRR 调度算法

#### 4.3.1 DRR 算法描述

针对 WRR 中分组长度不一带来的不公平性, Deficit Round -Robin(DRR)算法被提出。每一轮服务时, 如果业务流没用完分配的服务额度, 则用  $DC$  (deficit) 记录剩余的额度, DRR 将叠加  $DC$  到下一轮服务中, 算法描述如下。

描述之前定义几个参数:

$Q_i$ : 调度器分配给业务流  $i$  在每轮服务中可传输的比特数 (即服务额度);

$Q_i(k)$ : 业务流  $i$  在第  $k$  轮调度时可用的服务额度;

$Q_{\min} = \min_i(Q_i)$ : 最小的服务额度值;

$f_i = Q_i / Q_{\min}$ : 业务流  $i$  的服务权重;

$bits_i(k)$ : 业务流  $i$  在第  $k$  轮时被传送的比特数;

$DC_i(k)$ : 业务流  $i$  在第  $k$  轮调度后的剩余额度。

业务流  $i$  在第一轮服务中允许传输的比特数限制为:  $bits_i(1) \leq Q_i$ , 每一轮调度

后业务流  $i$  的剩余额度更新为:  $DC_i(k) = Q_i(k) - bits_i(k)$ , 下一轮服务额度更新为:  $Q_i(k+1) = Q_i(k) + DC_i(k)$ 。

轮询服务各业务流时, 为了避免遇到空队列 (即业务流中无业务), 在 DRR 中维持服务列表, 记录所有非空队列 (至少存储一个分组) 序号。每一轮调度开始时, 服务器根据列表头部记录的序号调度对应队列的分组, 如果完成此轮调度后该队列还有分组, 则把队列序号移到服务列表末端; 如果该队列无后续分组 (变为空队列),  $DC_i$  重新设置为 0, 并把队列序号从服务列表中删除。

DRR 算法缺点是交换分组较长, 时延变大, 时延为:

$$\Theta_i \leq \frac{(F - f_i)M + (n-1)(m-1)}{r} + (m-1)\left(\frac{1}{\rho_i} - \frac{1}{r}\right) \quad (4.8)$$

其中  $F$  为  $f_i$  总和,  $M$  是实际到达的最长分组比特数,  $m$  为实际被服务的最长分组比特数,  $r$  为链路速率,  $\rho_i$  为分配给业务流  $i$  的带宽速率。

#### 4.3.2 SDRR 算法描述

在 DRR 中, 如果某业务流有较高的服务额度将大大影响竞争同一输出链路的其余业务流的时延, 导致服务额度低业务流时延较大; 且当某类分组长度较长, 时延也较大。为克服上述的缺陷, 本文提出 SDRR, 能保证不同长度分组以及不同服务额度业务的时延性能, 提高公平性, 且算法复杂度不变。

SDRR 核心思想是把 DRR 中的每一轮服务 (在此称为外轮询) 分解为多轮服务 (称作内轮询), 每次内轮询服务顺序不变, 执行过程与 DRR 算法相同, 不同之处在于内轮询中各业务流的分配的固定服务额度降低。下面具体描述 SDRR 算法 (下面提到的各种参数请参考 DRR 描述)。

**第一步:** 对比各业务流的服务额度, 获得最小服务额度值  $Q_{\min}$  和最大服务额度值  $Q_{\max}$ , 计算  $n = \lceil Q_{\max} / Q_{\min} \rceil$  (不大于  $Q_{\max} / Q_{\min}$  的整数), 表示每次外轮询可以分为  $n$  次内轮询。

**第二步:** 开始一轮新的外轮询调度 (第一次外轮询各  $DC_i$  初始化为 0), 根据 DRR 中的原则维持外轮询服务列表, 根据列表确定每轮外轮询服务的业务流。同时, 内轮询中设置如下参数:

内轮询服务列表: 根据既定顺序记录在此次外轮询中可以继续进行内轮询调度的各业务流序号;

$Q_{i\text{odd}}$ : 记录各业务流进行内轮询时在本次外轮询中所剩服务额度, 初始化为  $Q_{i\text{odd}} = Q_i$ ;

$Q_{i\text{in}}$ : 内轮询时各业务流的服务额度, 新一轮外轮询开始时初始化为  $Q_{\min}$ ;

$DC_{i\text{in}}$ : 表示内轮询中的剩余额度, 开始新一轮外轮询时其值初始化为 0。

**第三步:** 如果内轮询服务列表为空, 转入第四步; 否则, 开始内轮询, 根据内轮询服

务列表服务各业务流，服务情况分两种：

**情况 A:**

假如某业务流的  $Q_{i\text{odd}} < Q_{\min}$ ，意味着此服务流结束本次外轮询中的内轮询调度，并删除内轮询服务列表中此业务流的序号。如果此业务无后续分组则设置剩余额度： $DC_i = 0$ ，并从外轮询服务列表中删除该业务流序号；否则，统计该业务流的剩余额度： $DC_i = Q_{i\text{odd}} + DC_{i\text{in}}(cnt)$ ；

**情况 B:**

否则，业务流被内轮询调度，执行调度后，内轮询剩余额度更新为  $DC_{i\text{in}}(cnt) = Q_{i\text{in}}(cnt) - \text{bits}_{i\text{in}}(cnt)$ ， $\text{bits}_{i\text{in}}(cnt)$  表示在第  $cnt$  内轮询中传送的比特数；下一轮内轮询服务额度更新为  $Q_{i\text{in}}(cnt+1) = Q_{\min} + DC_{i\text{in}}(cnt)$ ； $Q_{i\text{odd}}$  更新为  $Q_{i\text{odd}} = Q_{i\text{odd}} - Q_{\min}$ ；业务流序号移到内轮询服务列表末端。

继续服务其余业务流，重新开始执行第三步。

**第四步:** 内轮询调度结束，更新各业务流的服务额度  $Q_i = Q_i + DC_i$ ，转入第一步。

值得注意的是为保证每次服务时至少可以传输一个分组，无论是 DRR 算法或 SDRR 算法都必须保证  $Q_{\min}$  比所有业务流中最长分组大。

SDRR 策略细化外轮询，分散传输各业务流，能有效降低交换机一直服务于额度较大的业务流，大大降低时延与服务额度的关联性，各业务平均时延为：

$$\Theta_i \leq \frac{(n-1)(m-1)}{r} + \frac{1}{r} \sum_{j \in Q_{\text{end}}} f_j \cdot F + \frac{1}{r} \sum_{j \in (n-Q_{\text{end}})} \left[ f_i \right] \cdot M + \left( \left[ f_i \right] - 1 \right) \cdot M \left( \frac{1}{r} - \frac{1}{\rho_i} \right) + (m-1) \left( \frac{1}{\rho_i} - \frac{1}{r} \right) \quad (4.9)$$

上式中各字母表示的意义与 (4.8) 相同， $Q_{\text{end}}$  表示在每次外轮询中内轮询结束的业务流。

考虑两种边界情形：如果  $f_i < f_j$ ，则  $F > f_i$ ，同时  $Q_{\text{end}}$  为空的。比较 (4.8) 和 (4.9) 式，易于证明 SDRR 算法对于流  $i$  延时低于 DRR 算法。如果  $f_i > f_j$ ，除了流  $i$  外， $Q_{\text{end}}$  包括其余  $n-1$  业务流，同样通过比较延时计算公式，SDRR 算法的延时降低。

算法公平性如下式：

$$UI = \frac{\sum_{n=1}^M \left| \frac{\Delta \omega_i^n}{C(\tau_n - \tau_{n-1})} - \frac{\phi_i}{\sum_{j \in B_n} \phi_j} \right|}{N_n} \quad (4.10)$$

$\tau_n, n=0,1,\dots,M$ ，表示时间系统状态改变。 $B_n$  表示在时间段  $[\tau_{n-1}, \tau_n)$  中繁忙的队列， $N_n$  是  $B_n$  中包含的分组数目。 $C$  表示链路容量， $\Delta \omega_i^n$  表示队列  $i$  在  $[\tau_{n-1}, \tau_n)$  期间的被服务量， $\phi_i$  是分配给队列  $i$  的服务速率。

下面通过简单实例说明 SDRR 策略和 DRR 算法的区别。如图 4-5 所示。

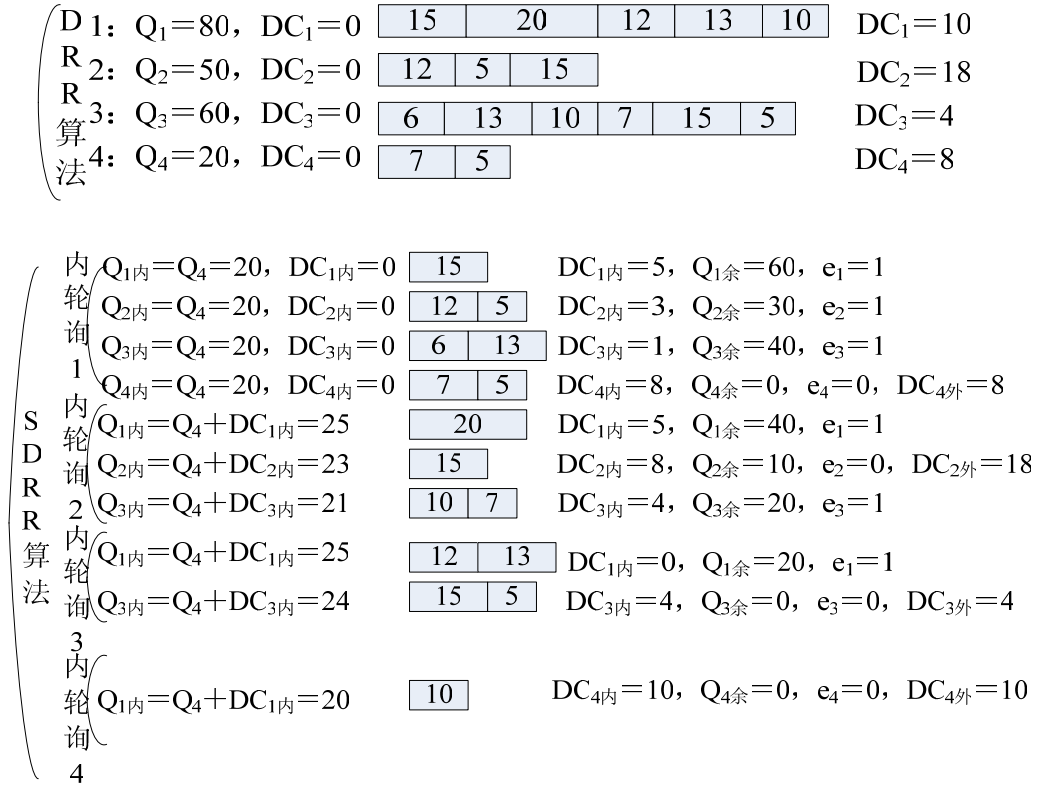


图 4-5 DRR 算法和 SDRR 算法简单实例

上图表示 DRR 算法，左边列出 4 个业务流服务额度  $Q_i$  的值和  $DC_i$  初始化值，中间代表业务流的分组，右边为经过一次轮询后的剩余额度。SDRR 算法如下图所示，显示了一次外轮询过程，其分解为 4 次内轮询， $Q_{i内}$  初始化为最小服务额度  $Q_4$ ， $DC_{i内}$  初始化为 0，经过一次内轮询后，更新  $DC_{i内}=Q_{i内}-bits$  ( $bits$  表示此次内轮询传送的比特数)，以及更新  $Q_{i内}=Q_{i内}+Q_4$ 。并通过设置剩余额度  $Q_{i余}$  标记此业务流是否可以继续内轮询调度，当某业务流  $Q_{i余}$  小于  $Q_4$  时，设置指针  $e_i$  为 0，指示队列序号从内轮询服务列表中删除，并统计外轮询的  $DC_{i外}$ 。

### 4.3.3 调度算法仿真模型

为验证 SDRR 算法的有效性，用 OPNET 网络仿真软件进行仿真。仿真模型如图 4-6 所示，模型由两类模块组成：业务源模块和调度模块，模块描述如表 4-4 所示。模型中包括两级调度：第一级调度器 scheduler\_0 和第二级调度器 scheduler\_1。scheduler\_0 的输入为五路相同的业务源：src\_0~src\_4，其业务到达为泊松分布，分配各业务的权重分别为：0.1、0.4、0.15、0.2、0.15。当各业务流在 scheduler\_0 调度后传输到 scheduler\_1，同时进入 scheduler\_1 的还有三路 on-off 突发业务源：src\_5~src\_7，到达 scheduler\_1 的业务源是非均匀分布的，由于从 scheduler\_0 传输过来的业务流较大，因此在 scheduler\_1 中调度时分配的权重高达为 0.64，其余几个业务流的权重统

一为 0.12。各业务流经由 scheduler\_1 调度后传输到终点 sink，统计业务分组从源端到目的端的调度时延和吞吐量，并销毁分组。同时在各级调度器中分别统计各业务流的时延。

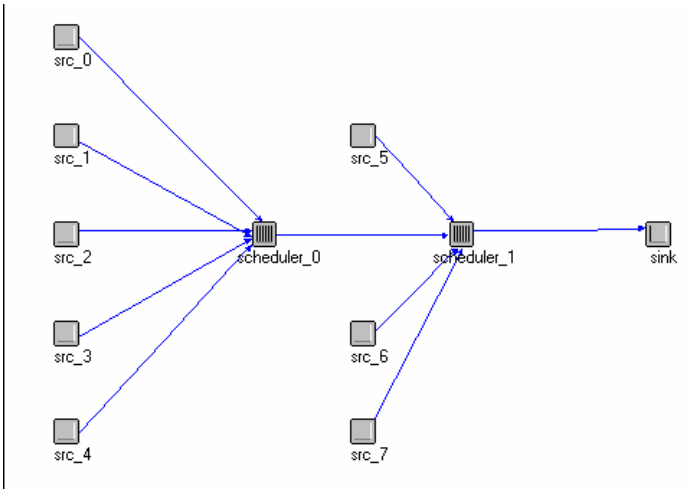


图 4-6 算法仿真模型

表 4-4 调度算法仿真模型模块描述

模块名称	功能描述
src_0、src_1、src_2、src_3、src_4	第一级调度业务源
src_5、src_6、src_7	第二级调度业务源
scheduler_0	第一级调度器
Scheduler_1	第二级调度器
Sink	终点

为验证 SDRR 算法性能优越于 DRR 算法，在模型中的两个调度器中分别执行 DRR 算法和 SDRR 算法。SDRR 算法的进程模型如图 4-7 所示。进程模型中各状态机的说明如表 4-5 所示。

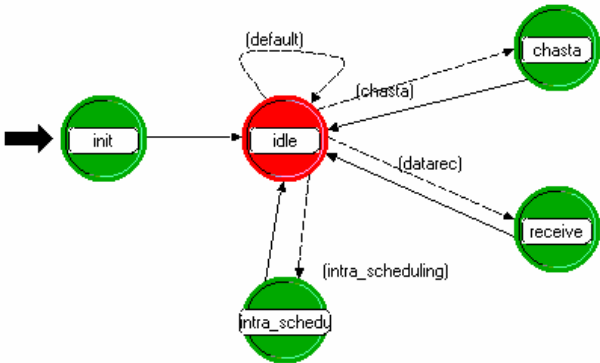


图 4-7 SDRR 进程

表 4-5 SDRR 算法进程模块的状态描述

状态	功能描述
init	初始化节点属性、状态变量以及各种参数
receive	接收各业务流分组并组成队列
chasta	更新外轮询参数并开始外轮询服务
intra-schedul	进行内轮询，并更新内轮询参数

在仿真模型中，根据业务量大小，合理设置调度器的处理速度以及调度周期。调度周期是指每一次轮询的服务时间，等于每一轮服务中传输的总比特数与处理速度的比值（模型中处理速度根据硬件速度以及输入输出端口数目合理分配）。由于 OPNET 仿真是基于离散事件驱动的模拟机理，因此在分组调度时根据设置的轮询周期，通过时间器以自中断进行内轮询和外轮询调度，为了较为真实模拟实际调度，每次仿真时间设置较长，为 16hours。

#### 4.3.4 SDRR 算法仿真结果及分析

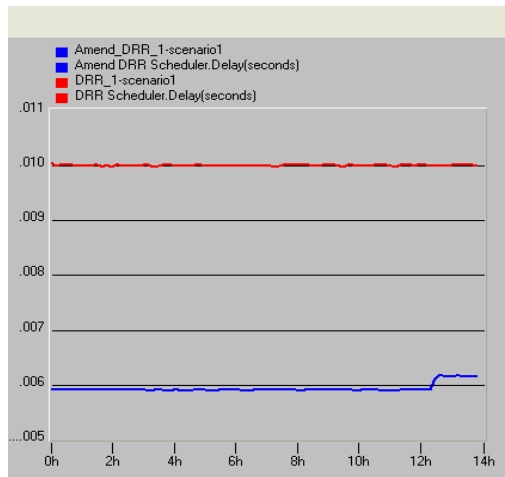


图 4-8 (a) DRR/ SDRR 时延结果比较

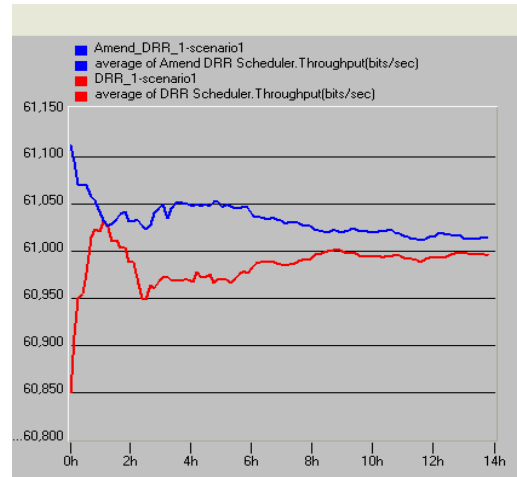


图 4-8 (b) DRR/ SDRR 吞吐量结果比较

首先针对 4.3.3 描述的仿真模型，只采用一级调度，即仿真时只执行图 4-6 中五路泊松到达业务源以及一级调度器，在调度器 scheduler\_0 中分别执行 DRR 和 SDRR。模型中各模块参数设置如下：业务分组长度在 24~220bits 均匀取值，处理速度 150kbps，调度周期为 0.02s。仿真结果如图 4-8 所示，左图纵坐标表示调度时延，单位为秒，右图纵坐标表示吞吐量，单位为 bits，两图中横坐标表示的是仿真时间，单位为小时。图中曲线显示 SDRR 算法时延比 DRR 算法低 0.004s（左图所示），吞吐量较大（右图所示），仿真中无掉包发生。

当分组长度加大，在 240~5000bits 均匀取值，处理速度为 8Mbps，调度周期为 0.04s。仿真结果如图 4-9 所示，其坐标表示与 4-8 相同，图中显示 SDRR 算法平均时延比 DRR 算法低 0.007s（左图所示），吞吐量几乎一样（右图所示），同样无掉包发生。

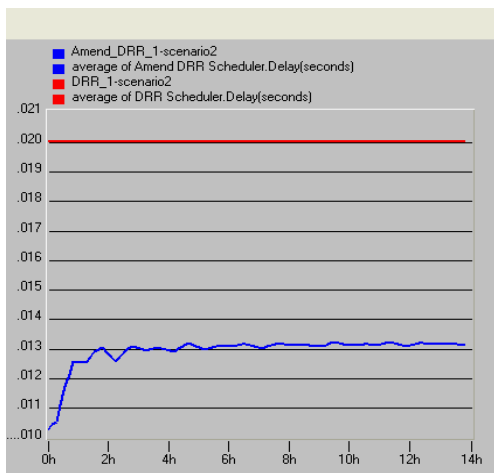


图 4-9 (a) DRR/SDRR 时延结果比较

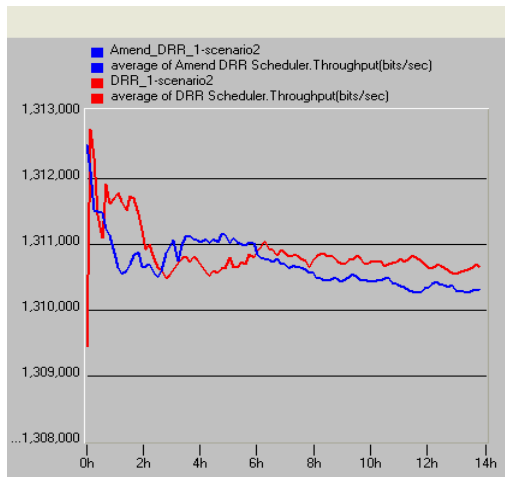


图 4-9 (b) DRR/SDRR 吞吐量结果比较

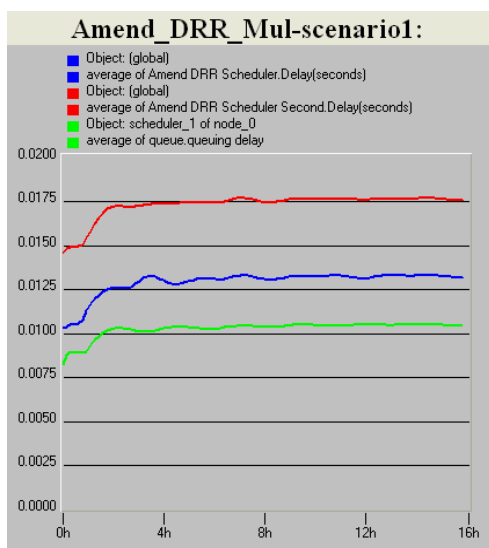


图 4-10 (a) SDRR 各级时延

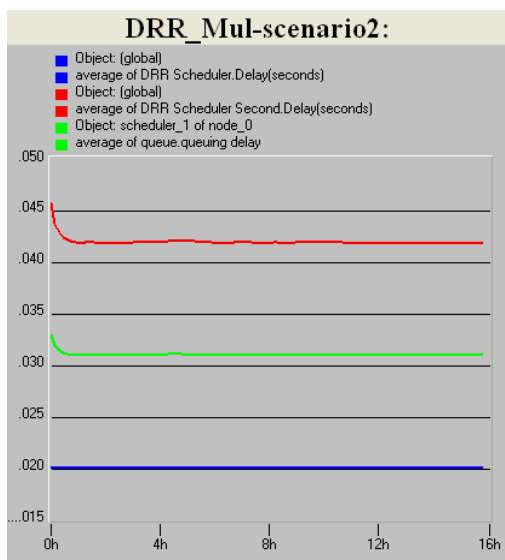


图 4-10 (b) DRR 各级时延

下面将在 4.3.3 提出的整个模型中运行仿真。业务源分组长度在 240~4880bits 之间均匀取值，仿真中为保证无掉包现象发生，根据业务流量和现有的硬件选取合适的调度器处理速度和调度周期，在两种算法执行中仿真参数以及业务源等输入设置相同，结果如图 4-10 和图 4-11 所示。图中横坐标表示仿真时间，单位为小时 (hours)，纵坐标为分组从产生到传输到 sink 模块的时延，单位为秒 (seconds)。

图 4-10 分别统计不同业务流状态下调度器执行 SDRR/DRR 算法的平均时延。图 4-10 (a) 和 (b) 中时延最低的曲线表示五路泊松到达的业务源从产生到调度到模型的 sink 模块时的平均时延，图中显示执行 SDRR 时平均时延比执行 DRR 时低 7ms 左右。

图 4-10 (a) 和 (b) 中间的时延曲线表示所有业务经过两级调度后在调度模块 scheduler\_1 中统计的平均时延性能，结果显示执行 SDRR 算法时平均时延比执行 DRR 算法低 20ms 左右。经过第一次调度后，业务源被塑性，各业务流量不一，并且具有突发性，DRR 算法时延性能大大下降。图 4-10 (a) 和 (b) 中时延值最高的曲线，



即两图中纵向上顶端的曲线统计所有业务源从产生到最后到达 sink 后的平均时延性能，曲线图显示 SDRR 算法时延低于 DRR 算法 26ms 左右，正好是上述两时延差别之和。

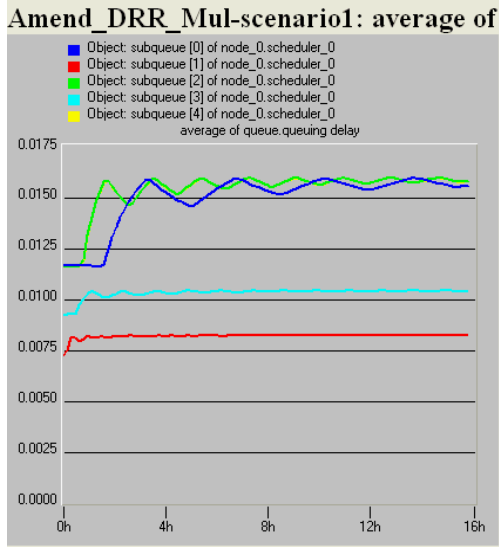


图 4-11 (a) SDRR 第一级各业务时延

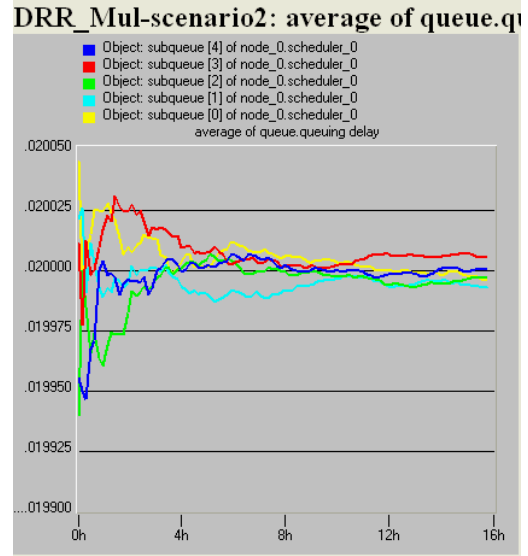


图 4-11 (b) DRR 第一级各业务时延

图 4-11 (a) 和 (b) 表示执行两种算法下五路泊松到达的业务经历第一级调度后各自的平均时延，进入 scheduler-0 的各业务权重相差不大，对比显示执行 SDRR 时，各业务流时延性能稍有区别（权重较大时延较低），而 DRR 中各业务时延性能一致。

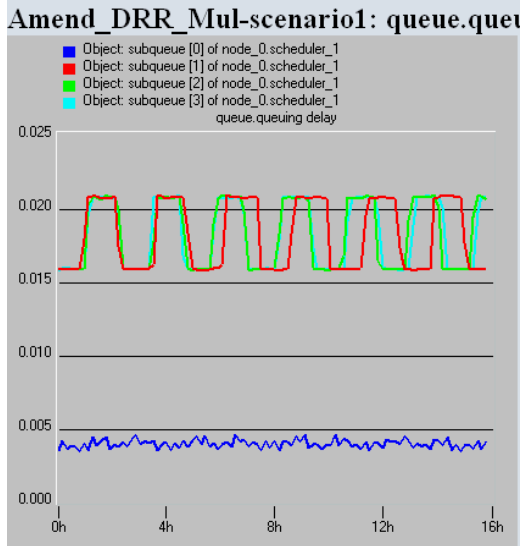


图 4-12 (a) SDRR 第二级各业务时延

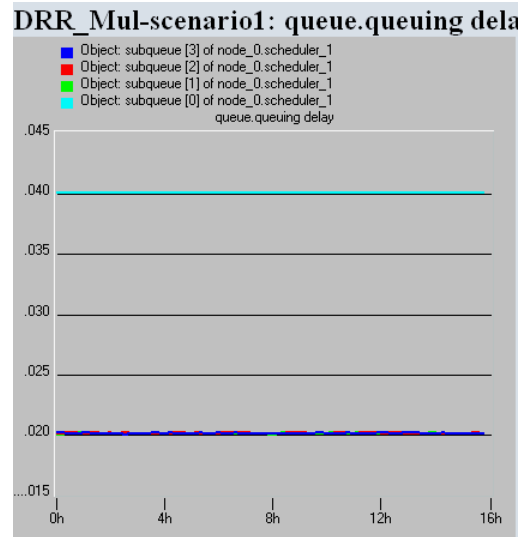


图 4-12 (b) DRR 第二级各业务时延

图 4-12 (a) 和 (b) 表示当分别执行 SDRR/DRR 时在模型的 scheduler\_1 单独针对非均匀及突发各业务流的时延性能结果。由于突发业务在调度中所得的服务权重大大低于 scheduler\_0 输出的泊松业务，因此，图中曲线显示无论执行哪种算法服务额度大的业务时延性能较好，相同权重的其余业务流时延性能几乎一致。值得注意的是图 (a) 显示权重大的业务时延比权重小的业务平均低 0.015s，而图 (b) 显示不同权

重的业务时延相差 0.02s, 表明 SDRR 改善了 DRR 中因服务额度相差大导致的不公平的缺点。

从上述结果分析可知,

- 1) SDRR 算法与 DRR 算法的复杂度都为  $O(1)$ , 仿真运行消耗的实际时间相同;
- 2) 业务分组长度较大时, SDRR 算法时延明显低于 DRR;
- 3) 对于服务权重不同的业务流, SDRR 即能保证各业务的公平性, 也能保证权重较大 (也指优先级较高) 的业务按时延性能, 充分利用资源保证不同业务的需求;
- 4) 针对非均匀业务流量, SDRR 相对于 DRR 算法时延性能大大提高。

SDRR 算法复杂度低, 时延性能较好, 能有效调度长度不一的分组, 适合于业务繁多、资源受限的星载交换机。

#### 4.3.5 SDRR 在星载 CICQ 交换机中的执行

星载交换机有  $N$  个输入和输出端口, 在每个输入端口有  $N$  个 VOQ 队列。输入端口  $i$  去往输出端口  $j$  的分组存在  $VOQ(i, j)$  队列中, 对应的交叉点的缓冲称为  $CPB(i, j)$ 。对于  $CPB(i, j)$  的尺寸用  $k$  表示, 意味着能储存的最大分组比特数。

RTT (round trip time) 指示往返程时间, 是从进入交换机队列到离开交换机的时间段, 包括输入端口调度 IS (input schedule) 时间, 输出端调度 OS (output schedule) 时间, 分组从输入到交叉点的传输时间 ( $d1$ ) 和流控制信息从交叉点反馈到输入端口的时间 ( $d2$ ) 总和, 因此:

$$RTT = IS + OS + d1 + d2 \leq k \quad (4.18)$$

交叉缓冲区的尺寸至少保证在一个 RTT 时间内不处于空闲状态, 而且要保证能存储业务流中最大尺寸的分组, 星上由于硬件条件限制, 不能无限制增大缓冲区尺寸, 权衡考虑确定尺寸大小为 RTT 与最大分组长度相加。

应用 SDRR 调度算法在星载交换机之前, 需要确定各输入条件。

- 1) 确定输入端口/输出端口数目 (假设都为  $n$  个端口);
- 2) 分组长度的确定, 星载交换机分三种长度的分组: 信令消息、业务消息和测控消息;
- 3) 根据不同链路带宽和数据速率设置各 VOQ 队列缓存区容量;
- 4) 分配不同类型业务不同的服务额度或服务权重;
- 5) 根据用户链路、馈电链路和星间链路带宽资源和速率比设置各队列调度权重, 譬如, 针对用户链路输入端口的调度速度低于星间链路输入端口, 而具体针对某用户链路输入端口中各 VOQ 队列的调度权重根据各队列前往的输出端口链路带宽资源和速率比配置。

在 CICQ 星载交换机的输入端口以及输出端口中执行 SDRR 调度, 每个输入端口

同步执行调度，从  $n$  个 VOQ 队列中选择分组传输，每个输出端口同步执行调度，从  $n$  个 CPB 队列中选择分组传输。

对应每个 VOQ  $(i, j)$  队列，相应的 CPB  $(i, j)$  有  $CPB\_Credit_{ij}$  (初始化为 0) 参数反馈给 VOQ  $(i, j)$ ，如果 CPB  $(i, j)$  缓存区满， $CPB\_Credit_{ij}$  变为 1，如果当此交叉缓存有空间可以容纳分组时，则  $CPB\_Credit_{ij}$  更新为 0。根据  $CPB\_Credit_{ij}$  的值确定是否可以继续传输分组到达交叉缓存区。输入调度中，如果轮询指针指向某个队列，而此队列的反馈参数指示不能传输分组，则累加此次的服务额度到剩余额度中，用于下一次内轮询或外轮询。

在整体调度中，星载交换机设置专门模块统一调配各输入/输出端口调度指针，使各调度指针指向不同的端口，降低同步现象，避免缓存区拥塞，但增加了一定的复杂性。譬如，同一时刻，输入端口 1 的服务指针指向输出口 1，而输入口 2 指向输出口 2，依此类推，但是，由于各链路调度周期不一致，很难达到完全解同步，不过结合初始化解同步策略以及统一调配策略可以降低输入输出同步调度现象，大大加速分组的传输。

#### 4.4 星载交换机 SDRR 调度算法仿真

##### 4.4.1 星载交换机仿真模型

针对上述设置，在 OPNET 中建立星载交换机模型，如图 4-13 所示。图中由业务源模块、输入队列模块、输出队列模块和终点模块组成，各模块描述如下。

###### 业务源模块

$p\_0 \sim p\_65$  是业务源模块，产生不同类型和不同密度的业务。每个输入端口有三类业务：信令分组（包括上面提到的测控信息），以泊松分布到达；语音业务分组，到达模式和语音持续周期为泊松分布；数据业务分组为 on-off 突发模型。各类型分组长度不一。

###### 输入/输出模块

设置每颗卫星 21 个输入/输出口，对应图中 input-0/output-0~input-20/output-20 队列模块：端口 0~15 对应用户链路，端口 16~19 对应星间链路，端口 20 对应馈电链路。每个输入端口模块设置 21 个 VOQ 队列，每个输出端口模块设置 21 个 CPB 队列，对应实际的 CICQ 结构。

###### 终点模块

SINK 模块中销毁各分组并统计时延、吞吐量等性能结果。

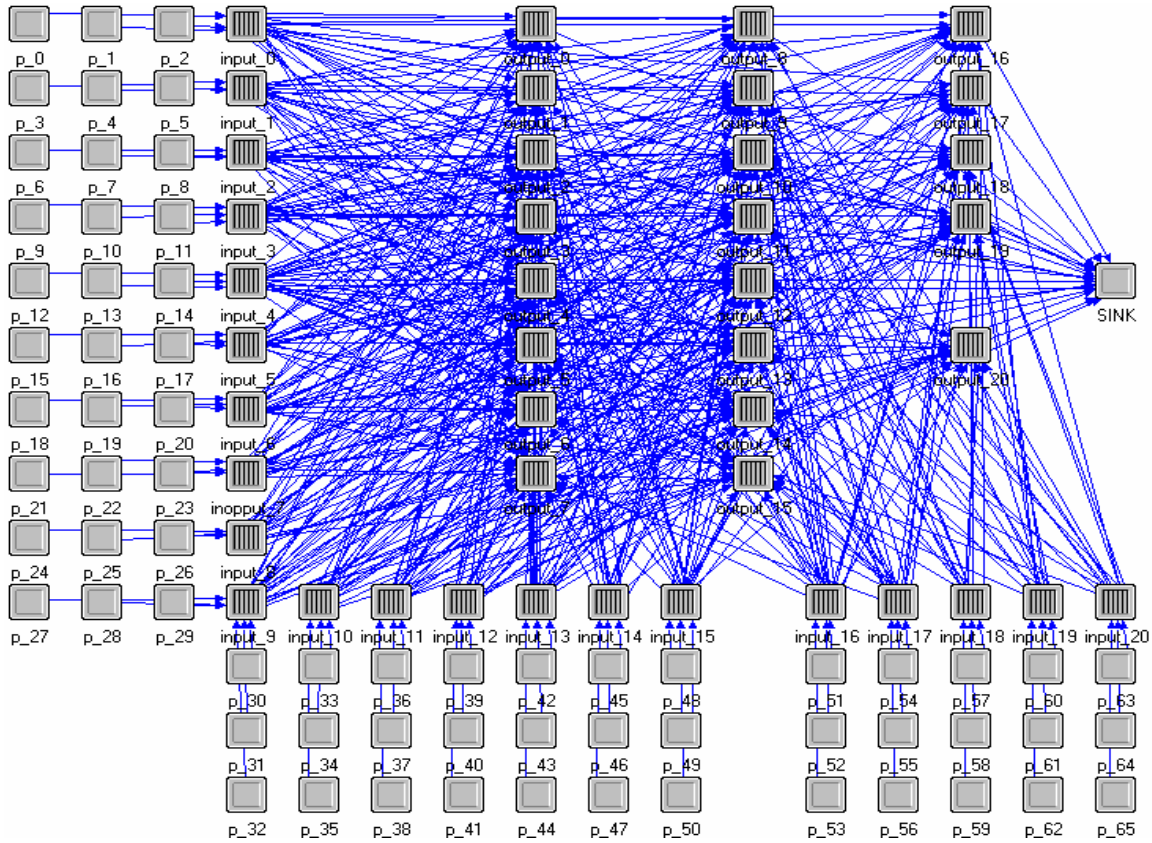


图 4-13 星载交换机模型

以下将详细描述仿真中所设置的参数。

#### 1) 输入/输出端口调度周期

调度周期具体指完成一次外轮询调度耗费的时间。根据实际系统中各链路可用带宽设置对应用户链路端口的调度周期为星间链路和馈电链路端口调度周期的十分之一。

#### 2) 各端口业务密度

链路带宽以及数目的不同导致对应端口业务的密度为非均匀分布。对应星间链路端口的业务量大约为用户链路端口的 4 倍，馈电链路端口的业务量大约为用户链路端口的 16 倍。

#### 3) 其余参数

业务到达密度、分组长度、输入缓存区容量、交叉缓存容量、各端口及处理速度。各 VOQ 队列缓存容量设置相同，交叉缓存区容量设置为最长分组长度的两倍。业务长度根据类型设置，总共分为三种长度类型。

分组在交换机的流程如下图所示。

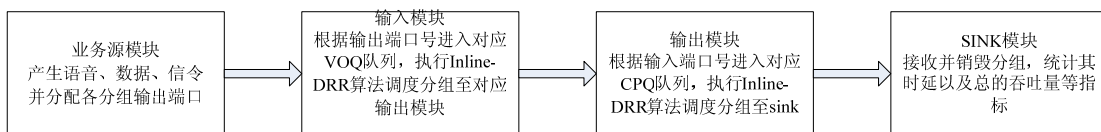


图 4-14 星载交换机模型分组流程图

#### 4.4.2 调度算法与路由策略的结合

执行调度时,各业务的服务权重分配(各业务在竞争同一输出链路时所占带宽比)影响时延性能,从第三章中分析得到分组在卫星网络路由时端到端时延包括各节点中的队列时延和处理时延,因此,权重的分配影响调度算法的时延,同样,也影响到路由策略的时延性能。从这点出发,根据各业务所选择的路径以及对性能的要求分配其权重,均衡路由中的传播时延和各节点中的调度交换时延。

因此,执行 SDRR 算法时其业务调度权重分为 6 类,如图 4-15 所示。图中分为两个等级业务:语音和信令为高优先级分组,数据为较低优先级分组,高优先级业务权重大于低优先级;在上述的两等级业务中又根据各业务路径经历的跳数细分为三类,跳数越多权重越大,则在一轮服务中占用的服务额度就越大。根据上述原则在仿真中,语音和信令业务权重值分别为:0.3、0.2、0.14,数据业务权重分别为:0.18、0.1、0.08。权重分配合理能提高调度算法的时延性能,从而提高路由协议中端到端的时延性能。

语音业务 信令信息	路由跳数大于6跳	数据业务	路由跳数大于6跳
	路由跳数为4~6跳		路由跳数为4~6跳
	路由跳数低于3跳		路由跳数低于3跳

图 4-15 业务带宽分配比例示意图

仿真中,各输入口根据队列中的业务以及各分组的路由状态确定其权重比例(路由跳数由随机序列产生),然后根据权重比确定各 VOQ 队列能用的带宽资源;为简化系统处理,输出调度中的权重仅仅根据不同的输入链路配置,对应用户链路输入口的交叉缓存队列权重设置 0.04,则 16 个用户链路占用的总权重值为 0.64,馈电链路和星间链路的权重设置为 0.072,则总共的权重值为  $5 \times 0.072 = 0.36$ 。所有参数设置完备后,运行不同场景下的仿真,结果如下。

#### 4.4.2 仿真结果分析

仿真中,调整合适的业务流量、调度速度、调度周期和队列缓存容量等仿真参数,并保证仿真过程中无掉包发生,分析在一定的业务吞吐量下,执行 SDRR 的 CICQ 星载交换机的时延性能指标以及所需带宽资源。

本文在分组长度不同和到达强度不一的两种场景下仿真,仿真时间为 5hours。图 4-16 所示为第一种场景,信令分组 64bits,语音分组 128bits,数据分组 640bits。图 4-17所示为第二种场景,信令分组长度不变,语音分组为 256bits,数据分组为 1920bits,适量降低各分组到达率,保证两种场景下吞吐量一致。图 a 中上面曲线纵坐标表示时延,单位为秒,下面的曲线表示吞吐量,单位为 bits,图 b 中纵坐标表示时延,图中

横坐标表示的都是仿真时间，单位为小时。4-16 和 4-17 图 a 中显示，两者的总时延和吞吐量一致，但分组长度变大时，为达到同样性能，各端口调度速度和缓存空间也需变大，需提高 15% 左右。图 b 所示为两种场景下，三类业务的时延曲线，从图 4-17 (b) 中可看出，数据分组长度变大时，延时较大。

On\_Board\_Switch\_Nest\_DRR-scenario3:

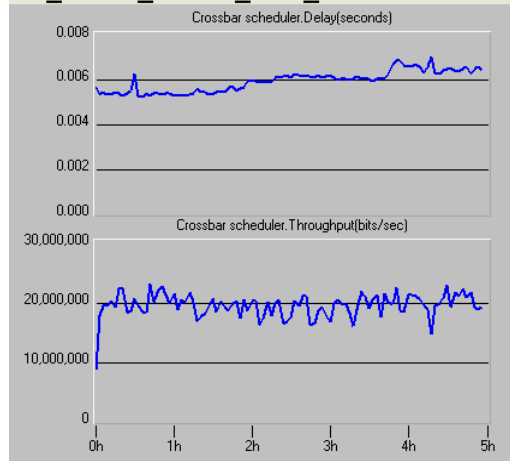


图 4-16 (a) 短分组的时延/吞吐量

On\_Board\_Switch\_Nest\_DRR-scenario3:

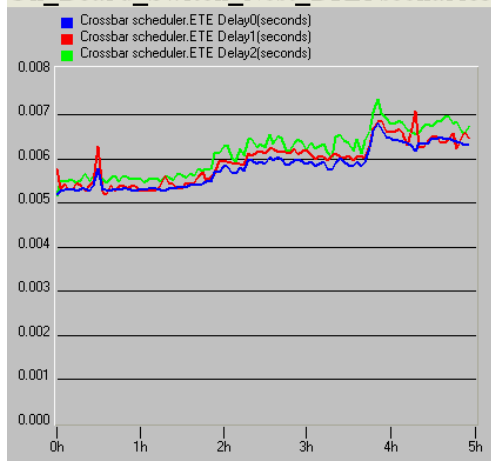


图 4-16 (b) 短分组各类业务时延比较

On\_Board\_Switch\_Nest\_DRR-scenario3:

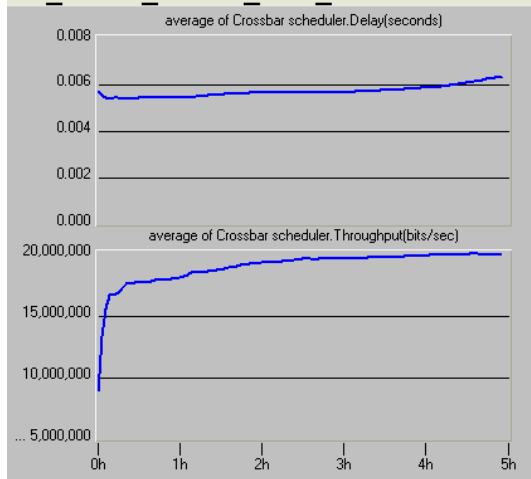


图 4-17 (a) 长分组的时延/吞吐量

On\_Board\_Switch\_Nest\_DRR-scenario3:

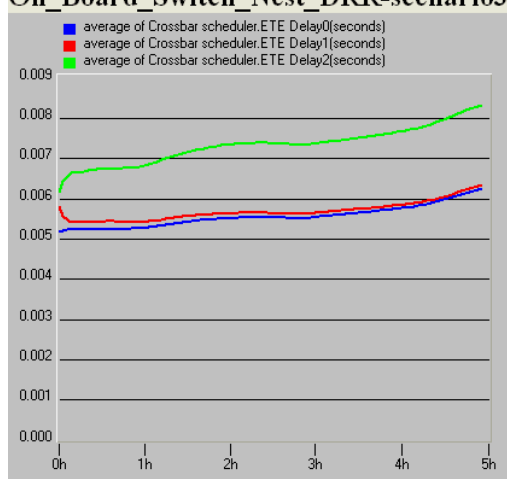


图 4-17 (b) 长分组各类业务时延比较

在上述仿真场景中，不同类型链路端口的业务量大小不同，但是对于同一链路端口业务是均匀分布的，而在实际系统中，会遭遇到用户链路业务分布不均，继而导致星间和馈电链路上业务分布不均，星载交换机不可重构，需要考虑这些特殊情况，合理设置交换机参数。

因此，在仿真模型中调整业务源的分布参数并降低调度速度，其结果如图 4-18 (a) 所示，时延稍微变大。在同样参数配置下，设置同类型端口业务到达为非均匀分布，进行仿真。其结果如图 4-18 (b) 所示，时延抖动变大，同时产生丢包现象。为保证无掉包现象，调度速度需要提高 50%，缓存区容量加大 28%，其时延和吞吐量结果如图 4-19 (a) 所示，结果显示丢包可以忽略不计，图 4-19 (b) 表示在此仿



真场景中的掉包统计（三根曲线表明在不同缓存区的掉包统计），从图中可看出，掉包可忽略不计。

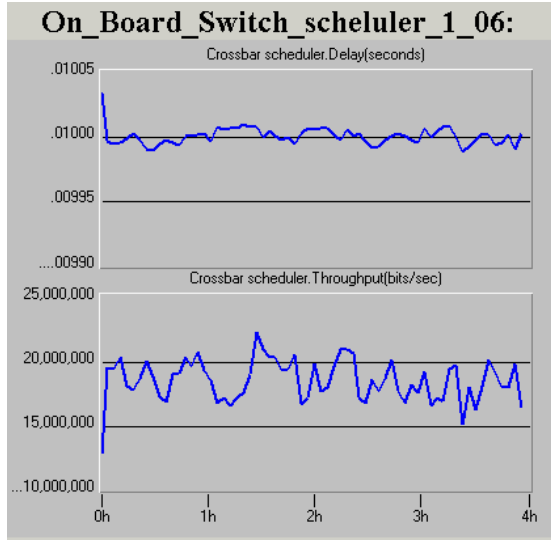


图 4-18 (a) 均匀流量时延/吞吐量仿真结果

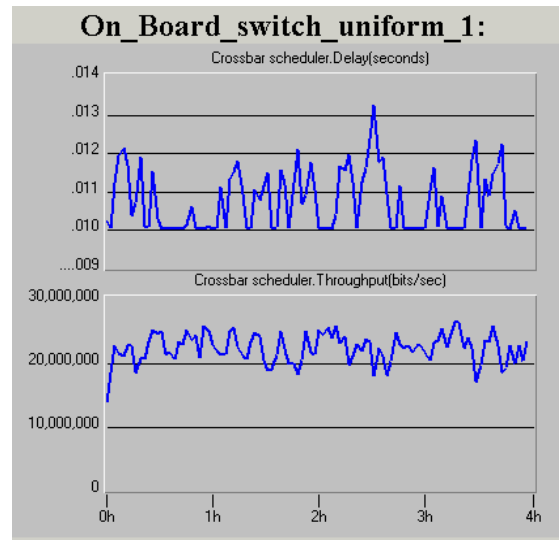


图 4-18 (b) 非均匀流量时延/吞吐量仿真结果

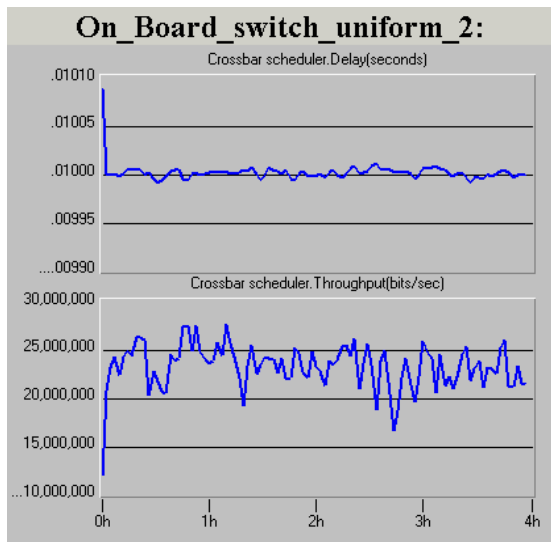


图 4-19 (a) 非均匀流量时延/吞吐量仿真结果

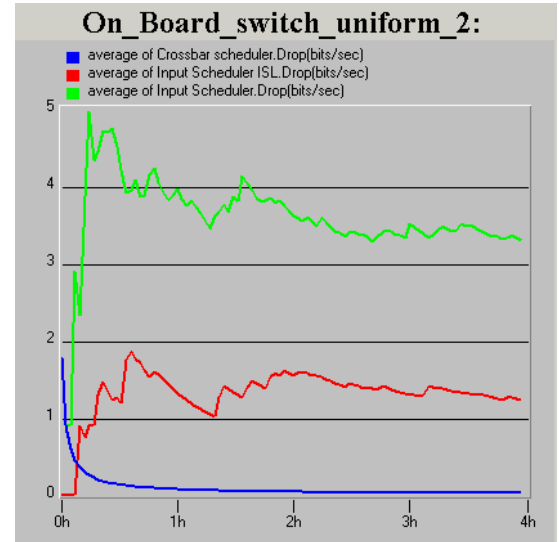


图 4-19 (b) 非均匀流量掉包仿真结果

地面交换机为保证时延，吞吐量等性能，根据业务特性，可较为方便加大缓存区容量，而星载交换机不可重构，为保证性能，需考虑系统业务繁忙或分布非均匀等各种特殊情况，设计交换机调度速度和存储空间，留出充分的裕量。现在地面一般的交换机处理速度可达 40Gps 左右，本文以 CICQ 结构为星载交换机，选取复杂度低的 SDRR 算法，在各种业务流量和分布下，通过仿真，合理设计各链路端口处理速度和缓存区容量，可以在星上资源受限的环境下易于实现并保证良好的性能。

#### 4.5 本章小结

本章重点讨论了星载交换机的结构，以及交换机中的分组调度策略。通过比较已有的星上调度策略，分析得到轮询调度策略适合星上交换调度；基于对传统的 DRR 策略进行修改和提高，设计适合星载交换机的 SDRR 调度算法；在 OPNET 中建立仿

真调度算法的模型，在不同业务到达模式下仿真，结果验证 SDRR 调度算法时延性较好、复杂度低，适用于不定长分组的交换调度；针对星载交换机业务类型和链路特点，建立 CICQ 星载交换机模型，在此模型中运行 SDRR 算法，从不同场景的仿真结果中验证本文提出的分组调度策略适合星载交换机。

本章的主要创新工作包括：

(1) 提出了采用 CICQ 作为星上交换机的结构；基于 CICQ 星载交换机结构提出了 SDRR 分组调度策略，分析了此策略的性能；在 OPNET 中建立验证分组调度策略的模型，并在模型中运行 DRR 算法和 SDRR 调度策略，有效验证各算法的性能。

(2) 在 OPNET 的卫星网络仿真平台中建立卫星交换机的仿真模型。此模型可以根据星座系统业务的特点设置交换机的输入，根据硬件限制和对性能指标的需求设置交换机调度算法、处理速度、处理周期、缓存区容量以及分组的长度，从仿真结果中分析其可行性。且通过结果分析能优化设置星载交换机各端口时隙的数目，充分利用星上有限资源，保证各链路即不空闲也不拥塞，简化星地可靠性传输。

分组调度策略大大影响卫星网络中时延性能，且对分组合理有序的调度能保证各类型业务顺利送达交换机输出端口，降低对卫星缓存区容量的要求，调度中结合路由和业务类型分配各业务可用带宽的权重，保证 QoS 要求。但由于低轨卫星固有的快速移动，当用户离开与之相连的卫星时，需要进行切换。此外，星上路由与切换两个过程本身就是一个有机整体，切换发生的同时往往也要求重新路由，因此，在交换和路由中需要考虑切换带来的影响，基于路由与切换相结合，下一章提出合适的切换策略。



## 第五章 低轨卫星通信网络中的切换策略

低轨卫星移动速度很快，其典型的运行周期为 100 分钟左右，这就意味着地面终端对于单颗卫星的可见时间为十来分钟，为保持服务持续而不被中断，用户从一颗卫星的“覆盖区”移动到另外一颗卫星的“覆盖区”，引起了网络进行切换操作，如图 5-1 所示。为了减少切换引起的时延和相应的信令开销，应当减少切换的次数以及处理时间，第三章所述的路由策略中，为了降低星间链路切换次数简化卫星网络拓扑，以其作为衡量算法性能优越的一个标准。可以看出，路由中不可避免涉及各种切换，为了保证持续性通话以及良好的通话质量，切换策略成为关键因素。

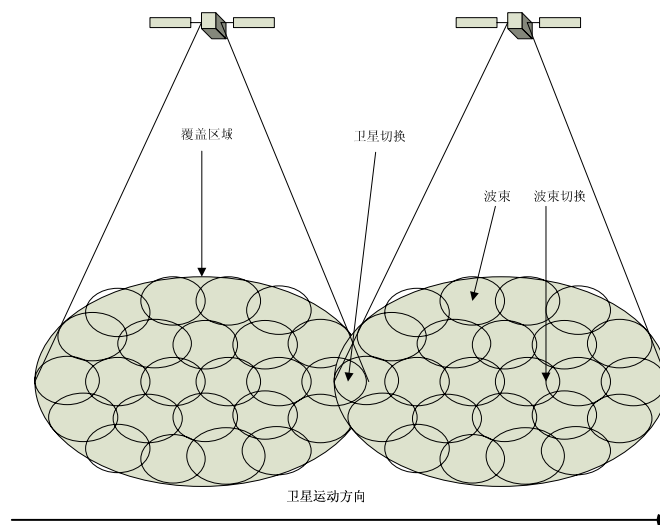


图 5-1 卫星波束覆盖及切换示意图

### 5.1 切换执行

切换包括切换执行、切换测量以及切换策略。切换执行包括从发起切换请求到最终完成切换的过程；切换测量决定在什么条件下发起切换请求；切换的重点在于切换策略，其决定用户将被切换到哪颗卫星或波束。下面将简要描述在低轨卫星网络中波束切换和卫星切换的执行。

图 5-2 描述了波束切换执行流程：

- (1) 用户持续进行切换测量，根据切换测量准则发起切换请求信息到通信卫星。
- (2) 卫星根据用户位置和各波束信道占用情况进行分配（利用切换策略），并发送切换指示消息到达用户。
- (3) 用户发送切换完成消息到达卫星。
- (4) 卫星释放原信道资源，并发起信道分配确认消息到信关站，信关站中进行资源信息列表的更新。

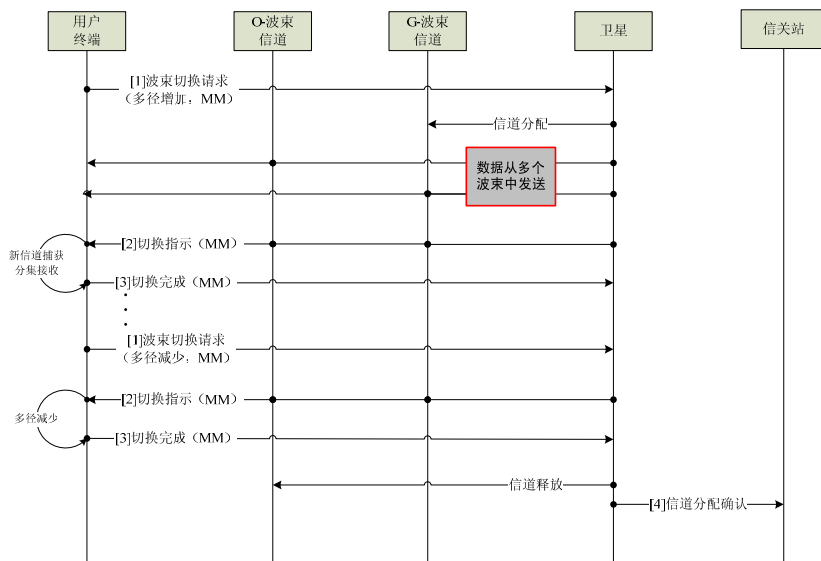


图 5-2 星上处理的波束间切换流程图

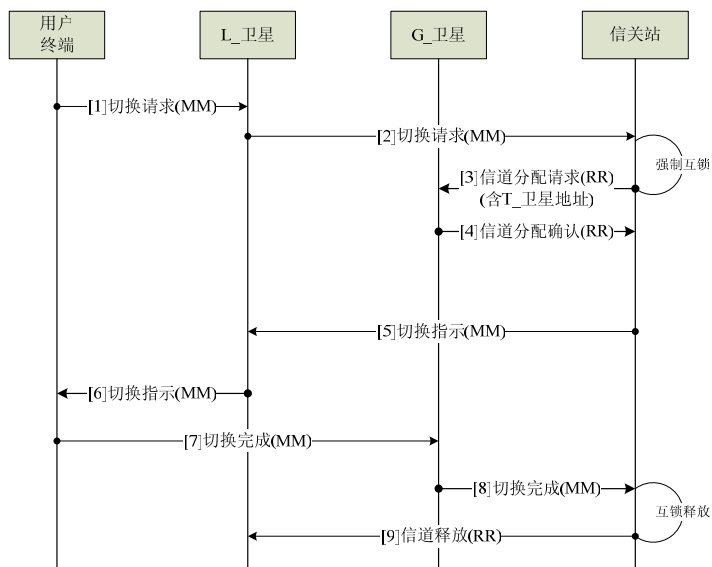


图 5-3 星上处理的星间切换流程图

图 5-3 描述了卫星切换执行流程：

- (1) 用户发起切换请求到原卫星。
- (2) 切换请求信息传输到信关站。
- (3) 信关站根据切换策略以及系统资源信息列表确定切换到哪颗卫星，并发送信道分配请求消息到对应的卫星。
- (4) 卫星发送回信道分配确认消息到信关站。
- (5) 信关站发起切换指示消息到达原卫星。
- (6) 切换指示消息经由原卫星到达用户。
- (7) 用户发送切换完成消息到达切换卫星。
- (8) 切换完成消息送达信关站，信关站更新资源信息列表。
- (9) 信关站发起信道释放消息，指示原卫星释放信道。

## 5.2 切换测量

地面无线 CDMA 系统中切换测量以导频信号强度为标准。在卫星上使用多波束天线模型,多波束天线的应用不仅能够提高天线增益和频率复用率,还能大大增加系统容量,已经成为卫星移动通信系统中不可或缺的重要组成部分。当用户远离星下点到达卫星覆盖区边缘时,功率控制已经达到最大值,之后随着卫星进一步向前运动,用户看卫星的仰角继续变小,接收到的下行链路导频能量也减弱。导频能量减小的主要原因有:自由传播损耗的增加,天线方向图的主瓣衰落。

地面终端到同轨道相邻的两颗星之间的距离为

$$\begin{aligned} d_1 &= \sqrt{R_e^2 + R_o^2 - 2R_e R_o \cos(\omega t)} \\ d_2 &= \sqrt{R_e^2 + R_o^2 - 2R_e R_o \cos(\theta_n - \omega t)} \end{aligned} \quad (5.1)$$

式中  $\theta_n$  为同轨卫星间夹角,  $\omega$  为卫星绕地角速度。

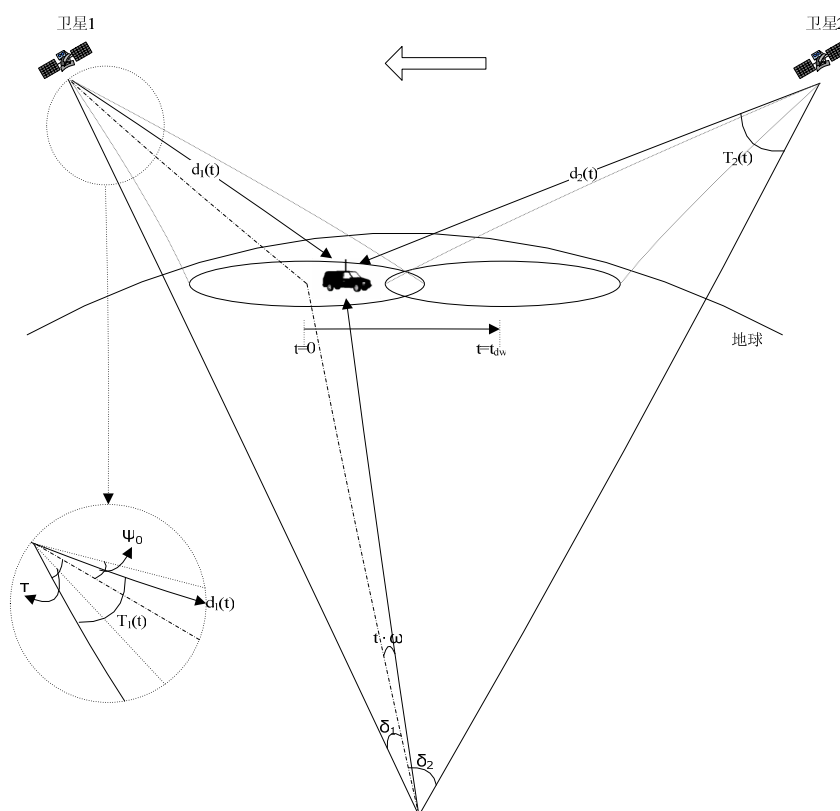


图 5-4 同轨面两颗星与地面终端的几何关系

地面终端位于星下点时为 0 时刻。以衰减 10dB 为接收门限则卫星过顶时间约 20 分钟, 两星覆盖区重叠时间约 8 分钟; 以最短距离原则选择卫星则过顶时间约 14 分钟。

为了降低卫星和地面终端的功耗,提高低仰角时通信质量,星座系统采用的多波束天线由 61 个阵元合成 16 个波束,通过调整合成参数,能够补偿自由传播损耗的差异。

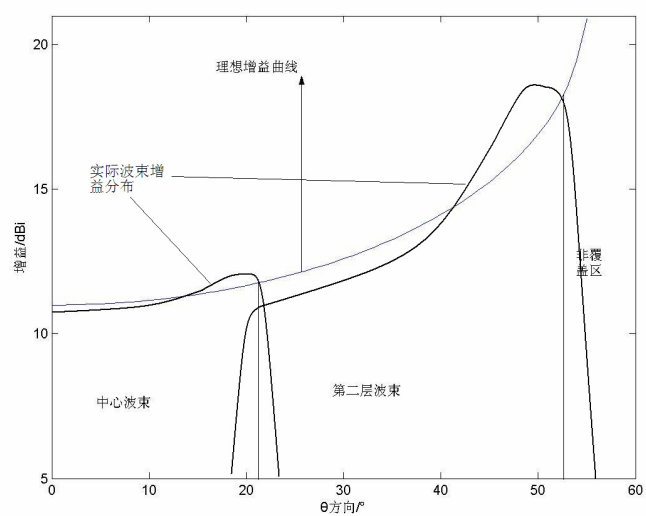


图 5-5 多波束接收天线增益剖面图

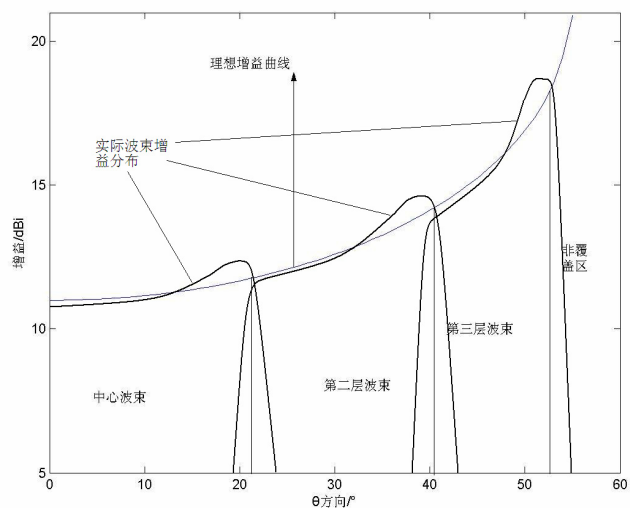


图 5-6 多波束发射天线增益剖面图

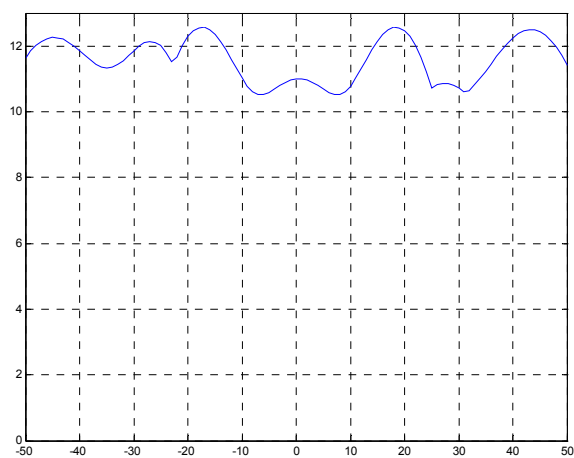


图 5-7 多波束下行链路覆盖区增益

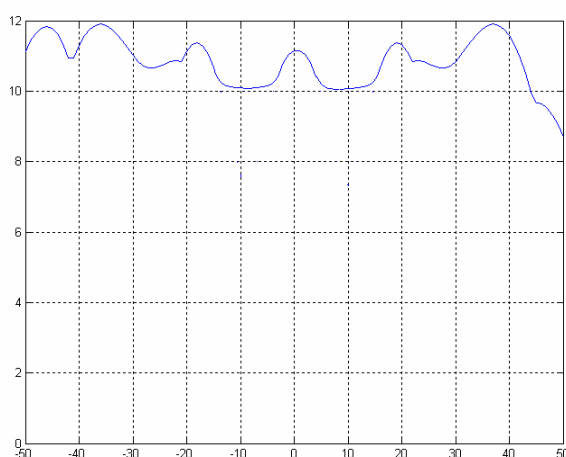


图 5-8 多波束上行链路覆盖区增益

将多波束天线增益和自由传播损耗叠加起来就得到整个卫星覆盖区的增益图。张角  $53^\circ$  范围内，补偿了自由传播损耗差异的卫星覆盖区增益如下图，补偿后的增益差异小于 2dB。

在地面蜂窝系统中，用户终端和基站 A 的距离可能从几米慢慢变到几百米，同时和基站 B 的距离从几百米慢慢变成几米，在这个过程中用户终端接收到的基站 A、B 的导频功率有连续但高达几十 dB 的剧烈变化，因此导频功率能很好地反映用户终端到基站的 A、B 距离，可作为切换判断的依据。

而多波束天线对传播损耗的补偿给导频测量辅助的切换方案带来了负面影响。一般的，卫星下行链路导频按固定功率发送，则地面终端在卫星覆盖范围内( $53^\circ$ )接收到的两颗卫星的导频功率基本上一样大，无法作为距离的判断依据；只有当地面终端离开覆盖区时方向图主瓣的快速衰落才会使导频功率迅速下降。

因此，在本系统中执行切换的前提条件：用户能自主定位，或采取 GPS 的办法，或采用文献所述的通过测多普勒频偏对用户精确定位。用户根据自己的位置信息以及各卫星的星历判断执行切换，可以通过计算仰角、距离星下点位置或者距离各卫星波束增益中心位置判断可以切换的卫星或波束。本文中采用用户距离卫星各波束中心位置的切换测量办法。

低轨卫星网络中地面信关站中存储卫星的星历信息，其中包括卫星的位置、卫星对应星下点的位置、卫星的运动状态和轨迹以及卫星各时刻信道占用信息等，此星历信息较短周期内更新。网络中的用户自主定位，并周期性发送自己的位置信息到信关站。信关站周期检测各用户与可通信卫星及波束的位置关系。如图 5-9 所示，左图表示卫星波束在地球投影的简单示意图，右图为波束投影在地面的截面图，其包括一个中心波束和 15 个边缘波束，图中虚线表示各边缘波束的投影中心，中心波束 0 的中心对应左图中的  $t$ （两图中对应相同的点字母标注相同）。

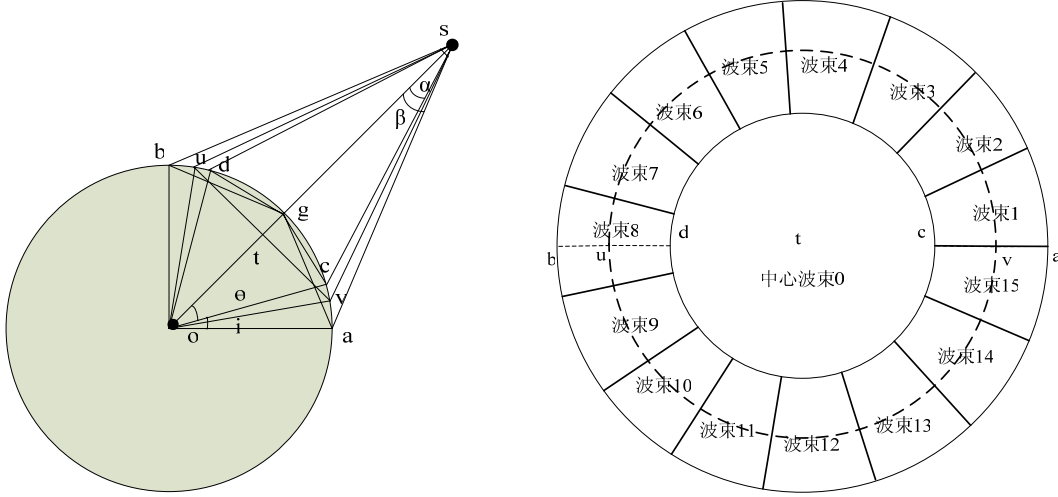


图 5-9 卫星投影图和多波束天线覆盖图

卫星  $s$  的星下点为  $g$ ，已知卫星天线张角  $\beta$  和中心波束的张角  $\alpha$ ，因此，根据卫星的位置计算星下点的位置， $H$  表示卫星轨道高度， $R_e$  表示地球半径， $x_s, y_s, z_s$  表示卫星在地心坐标系中的坐标，则  $x_g, y_g, z_g$  表示星下点坐标。

$$\begin{aligned} x_g &= \left(1 - \frac{H}{H + R_e}\right) x_s \\ y_g &= \left(1 - \frac{H}{H + R_e}\right) y_s \\ z_g &= \left(1 - \frac{H}{H + R_e}\right) z_s \end{aligned} \quad (5.2)$$

在  $\Delta ocs$  利用正弦定理得到  $\angle \theta$  的值，同理在  $\Delta oas$  中得到  $\angle \theta + i$  的值，则可计算出弧度  $\angle gc$ 、 $\angle ga$  和  $\angle ca$  的值，获得中心波束和边缘波束的投影宽度分别为  $\angle gc$  和  $\angle ca$ 。

$$\begin{aligned} \angle gc &= R_e \times \angle \theta \\ \angle ga &= R_e \times \angle (\theta + i) \\ \angle ca &= R_e \times \angle i \\ \angle gv &= (\angle gc + \angle ga) / 2 \end{aligned} \quad (5.3)$$

推导得到：

$$\begin{aligned} \angle sov &= \angle gv / R_e \\ \overline{gv} &= R_e \times \sqrt{2 \times (1 - \cos(\angle sov))} \\ \overline{uv} &= R_e \times \sqrt{2 \times (1 - \cos(2\angle sov))} \end{aligned} \quad (5.4)$$

其中  $\overline{gv}$  和  $\overline{uv}$  表示图中对应两点之间的直线距离。中心波束 0 在地表投影的中心

点为星下点  $g$ ，为了推导出边缘各波束在地表投影中心点的位置，首先需要确定上两图中  $v$  的坐标。根据星下点的坐标获得其对应的经纬度值：

$$\begin{aligned} long_g &= \arctan(x_g/y_g) \\ lat_g &= \arcsin(z_g/\sqrt{x_g^2 + y_g^2 + z_g^2}) \end{aligned} \quad (5.5)$$

容易得到  $v$  的经纬度值：

$$\begin{aligned} long_v &= long_g \\ lat_v &= lat_g + \angle sov \end{aligned} \quad (5.6)$$

由此得到  $v$  的笛卡儿坐标：  $x_v, y_v, z_v$ ，设置边缘波束中心点坐标为  $x_j, y_j, z_j$ ，下表  $j$  ( $j=1,2,3\dots15$ ) 表示第几个波束，则在左图地心坐标以及右图中的平面坐标中建立方程：

$$\begin{aligned} x_j^2 + y_j^2 + z_j^2 &= R_e^2 \\ (x_j - x_g)^2 + (y_j - y_g)^2 + (z_j - z_g)^2 &= (\overline{gv})^2 \\ (x_j - x_v)^2 + (y_j - y_v)^2 + (z_j - z_v)^2 &= 2 \times (\overline{uv}/2)^2 \times [1 - \cos(j \times 24^\circ)] \end{aligned} \quad (5.7)$$

通过 (5.7) 式可以求得卫星各波束投影在地表的中心位置。

根据用户与各波束投影中心的距离确定何时发起切换请求，其测量步骤如图 5-10 所示：

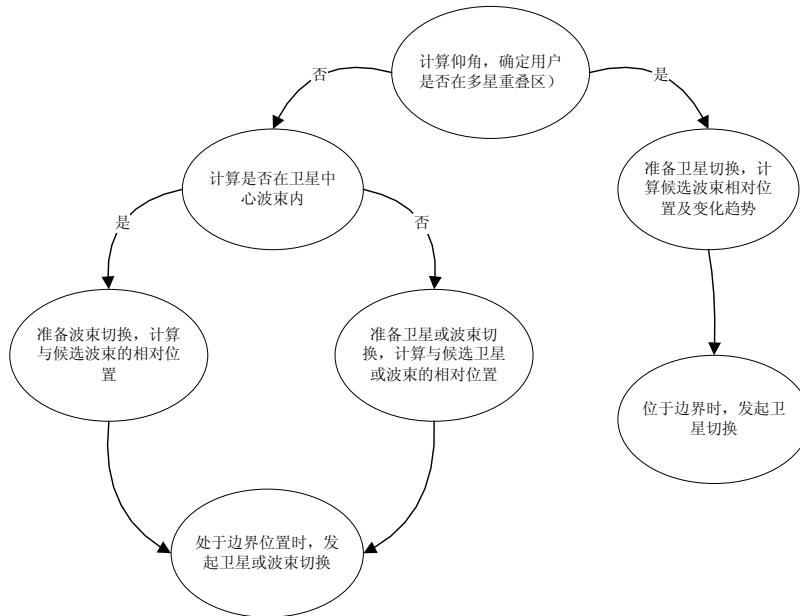


图 5-10 系统切换位置测量流程示意图

1. 根据用户与卫星仰角确定是否是多颗卫星重叠覆盖区。
2. 如果单颗卫星覆盖时，且位于中心波束内，暂且只考虑波束切换，计算用户与

此卫星中心波束和边缘波束的相对位置以及变化趋势，当距离中心波束中心位置某范围时，发起切换请求。如果位于边缘波束覆盖范围内，需要考虑卫星切换，计算用户与此卫星以及候选卫星的波束相对位置关系，当距离当前波束中心位置某范围时发起卫星切换或波束切换请求。

4. 如果位于多颗卫星重叠区，则同步骤 3，检测用户与各卫星波束中心位置相对位置关系和变化趋势。

5. 根据位置关系和趋势以及其余考虑的因素（波束中空闲信道等）选择可进行切换的卫星或波束，及时发起切换请求并预定信道，如果波束中无空闲信道则进行 FIFO 排队，为了降低掉话率，每个波束预留 6%~25% 的信道留给切换请求。

### 5.3 权值优化切换策略

当切换发生后，需要重新进行路由选择，所以切换过程和路由是一个有机整体。但已有的切换策略往往将路由与切换分开考虑，着重讨论切换本身，不能得到最优的系统性能。本文将路由与切换结合的基础上，提出了权值优化的切换策略。

本文提出的 DRAST 路由策略对网络拓扑进行了简化，有效降低了星间链路切换频率，为了降低用户链路切换频率，选择服务时间最长的波束执行切换，同时，为了平衡全网络业务分配，考虑各波束信道占用情况，因此，采用权值优化切换策略，使用多个准则选取切换的卫星或波束，其准则如下所述：

- 1) 选取最大服务时间的卫星或波束，设系统中最大波束服务时间即为用户可以在一个波束中逗留的最长时间  $T_b$ ，则针对服务时间  $T_i$  ( $i=1,2,3\dots$ )，归一化为

$$T_i/T_b;$$

$$T_b = T_{orb} \times \frac{L(i)}{2 \times \pi \times R_e} \quad (5.8)$$

$T_{orb}$ ：轨道围绕地球一圈的周期；

$L(i)$ ：卫星波束的直径（即波束覆盖区域最大宽度）。

- 2) 选取最小仰角的卫星，设系统最小仰角为  $E_{\min}$ ，则对仰角  $E_i$  ( $i=1,2,3\dots$ ) 归

一化为  $E_i/E_{\min}$ ；

- 3) 选取距离波束中心位置最近的波束，假设波束为圆形，则离波束最远距离为波束的半径： $L(i)/2$ ，则对用户距离波束中心的距离  $U_{dis}(i)$  ( $i=1,2,3\dots$ ) 的归

一化为： $(1-U_{dis}(i))/L(i)$ ；



4) 选取信道最多的卫星或波束, 设卫星波束信道占用率为 $\eta_i$  ( $i=1,2,3\dots$ )。

可通过对上述各准则进行加权构造一个卫星目标函数, 最优化权值进行选择, 目标函数如下:

$$W_i = s_T \times T_i / T_b + s_E \times E_i / E_{\min} + s_D \times (1 - U_{dis}(i)) / L(i) + s_\eta \times 1 / \eta_i \quad (5.9)$$

此函数可将最近卫星准则 (或最大仰角准则)、最长可视时间准则、最强信号准则和最小负荷准则统一如下:

$$s_T = s_D = s_\eta = 0, s_E = 1 \quad \text{最近卫星优先策略}$$

$$s_E = s_D = s_\eta = 0, s_T = 1 \quad \text{最长可视时间优先策略}$$

$$s_E = s_T = s_\eta = 0, s_D = 1 \quad \text{最强信号优先策略}$$

$$s_E = s_T = s_D = 0, s_\eta = 1 \quad \text{最小负荷优先策略}$$

$$s_E \neq 0, s_T \neq 0, s_D \neq 0, s_\eta \neq 0 \quad \text{最优化权值选择策略}$$

目标函数中是第一项和第三项主要考虑的是用户被服务的时间较长, 为有效降低用户链路切换次数, 设置其权值较大, 因此最优化权值方案为:  $s_T > s_D > s_E > s_\eta$ , 具体值根据系统轨道特征、新呼叫的阻塞率 $P_b$ 和呼叫终止概率 $P_f$ 进行设置。因此, 要求解阻塞率和掉话率设置合理权值, 下文通过排队理论估算各概率值。

## 5.4 马尔可夫链分析

模型中分为卫星切换和波束切换, 如图 5-11 所示, 左图表示用户在边缘波束时平衡状态, 右图表示用户在中心波束时平衡状态。

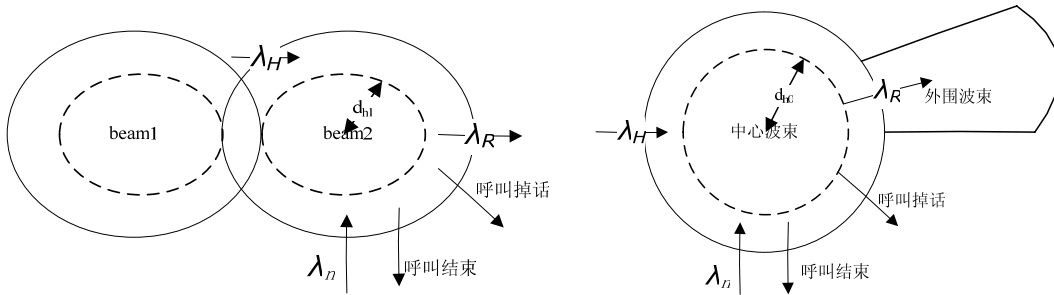


图 5-11 业务呼叫平衡状态示意图

以往文献主要通过设置发起切换请求的时刻评估性能, 本文主要通过计算用户与波束相对位置决定何时发起切换请求。从上图看出当用户距离中心波束中心点距离为 $d_{h0}$ 时发起切换请求, 当用户位于边缘波束时距离此波束中心为 $d_{h1}$ 时发起切换请求。

如果切换距离门限过大则不能及时为用户预留信道，带来较大的掉话率，而门限过小，信道被预留时间较长，虽然能保证切换成功，但引起较高的新的呼叫阻塞率。因此，距离门限值根据系统性能要求设置。

网络中移动特性定义为用户的平均呼叫持续时间与用户在一个波束中可以逗留的时间的比值：

$$\gamma = \frac{T_{call}}{T_b} \quad (5.10)$$

式中  $T_{call}$  表示用户呼叫持续时间。则用户发生切换概率与参数  $\gamma$  相关，在原波束中发生第一次切换的概率  $P_{h1}$  为<sup>[112]</sup>：

$$P_{h1} = \gamma \times (1 - e^{-(1/\gamma)}) \quad (5.11)$$

而在随后的波束中再次发生切换的概率为：

$$P_{h2} = e^{-(1/\gamma)} \quad (5.12)$$

当  $\gamma$  增加时，切换概率增加（意味增加系统移动性）。从图 5-12 得到用户在波束中达到均衡状态的方程式，式中  $\lambda_n$  表示新呼叫到达率（其到达为泊松分布）， $\lambda_H$  表示从其余波束到达本波束的切换请求到达率（假定为独立的泊松到达）， $\lambda_R$  为向外发起切换请求率，

$$\lambda_n \times (1 - P_b) \times P_{h1} + \lambda_H \times (1 - P_b) \times P_{h2} = \lambda_R \quad (5.13)$$

不管切换到达还是发起切换要求，结果都会占用信道，因此上式中  $\lambda_H$  相当于  $\lambda_R$ ，则推导得到：

$$\lambda_H = \lambda_n \times \frac{(1 - P_b) \times P_{h1}}{1 - (1 - P_b) \times P_{h2}} \quad (5.14)$$

系统中因新的呼叫用户的到达每个波束总的业务密度为：

$$\rho_n = \lambda_n \times T_{call} \quad (\text{erlang}) \quad (5.15)$$

推导中，本文考虑实际的用户到达模式（没有使用以往文献中的“街道模型”<sup>[81]</sup>）。在切换策略中使用位置测量，地面信关站设置位置管理设备（mobility position management），每隔小段周期  $\Delta t$  计算用户与仰角范围内卫星各波束中心点的距离，如果位于某几个波束覆盖范围内，则连续检测一段时间  $T_h$ ，跟踪相对位置变化趋势，

当距离本波束距离越来越远且超过门限时（图 5-11 所示的  $d_{h0}$  和  $d_{h1}$ ）则根据权值优化切换策略选择波束发起切换请求预留信道。

波束中总的呼叫平均到达率定义为  $\lambda_t$ ：

$$\lambda_t = \lambda_n + \lambda_H \quad (5.16)$$

虽然切换的到达与新的呼叫相关，分析中假定其为泊松到达，独立于新呼叫到达。

定义波束中信道被占用时间为  $1/\mu$ （负指数分布），则：

$$\frac{1}{\mu} = P_1 E[T_{H1}] + P_2 E[T_{H2}] \quad (5.17)$$

式中

$P_1$ ：信道被新呼叫用户占用时间的概率；

$P_2$ ：信道被切换用户以及切换请求占用时间的概率。

$$P_1 = \frac{\lambda_n \times (1 - P_b)}{\Lambda} \quad (5.18)$$

$$P_2 = \frac{\lambda_H \times (1 - P_b)}{\Lambda}$$

$\Lambda$  表示总的业务的平均值：

$$\Lambda = \lambda_n \times (1 - P_b) + \lambda_H \times (1 - P_b) \quad (5.19)$$

平均占用信道时间如下式：

$$E[T_{H1}] = T_{call} \times [1 - P_{h1}] \quad \text{表示在源波束中用户占用信道时间} \quad (5.20)$$

$$E[T_{H2}] = T_{call} \times [1 - P_{h2}] \quad \text{表示在切换波束中用户占用信道时间}$$

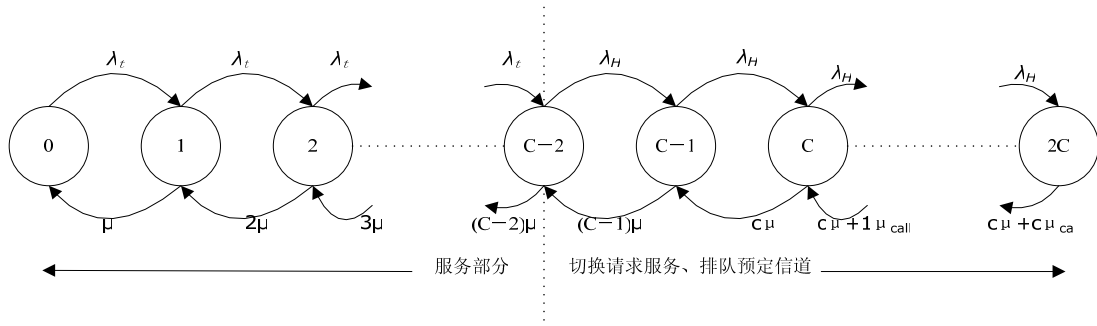


图 5-12 马尔可夫状态转移图

从上述的假设和推理中，每个波束可以模型为 M/M/C/S 排队系统。队列中的各状态定义为被服务的用户数、切换请求以及切换到达占用信道的数目。第一个  $M$  表示用户和切换的到达是独立的无记忆的泊松过程，第二个  $M$  表示服务时间的概率为负指数

分布,  $C$  表示每个波束有  $C$  条信道,  $S$  表示系统中的容量。系统中每个波束有  $C$  个信道, 则切换请求预定的信道不能超过此容量, 因此  $S = 2C$ 。马尔可夫链如图 5-12。

为了降低掉话率, 每个波束预留 6%~25% 个信道给切换请求的用户 (图 5-12 中预留 2 个信道), 预留的信道数目根据系统特征以及用户分布调整。如果状态  $n$  相当或超过容量  $C-2$ , 表明波束中所有信道都处于繁忙状态, 此时拒绝接入新的呼叫用户, 只接纳切换用户和切换请求。

图中  $c\mu$  表示波束中任一的呼叫结束, 而  $i\mu_{call}$  表示切换预留信道的请求因为呼叫终止清除, 则,

$$\mu_{call} = 1/T_{call} \quad (5.21)。$$

当系统达到稳定状态时, 从状态  $n$  转移到状态  $n+1$  的概率必然等于系统从状态  $n+1$  转移到状态  $n$  的概率。

$$P_n P_{n,n+1} = P_{n+1} P_{n+1,n} \quad (5.22)$$

否则系统不可能稳定。系统的稳态全局平衡方程为:

$$\begin{aligned} \lambda_t \times P_{n-1} &= n\mu P_n & n < C-2 \\ \lambda_H \times P_{n-1} &= n\mu P_n & C-2 \leq n \leq C \\ \lambda_H \times P_{n-1} &= [C\mu + (n-C)\mu_{call}] P_n & C < n \leq 2C \end{aligned} \quad (5.23)$$

可得到系统的稳态状态概率为:

$$P_n = \begin{cases} \frac{\lambda_t^n}{n!\mu^n} P_0, & n < C-2 \\ \frac{\lambda_H^n}{n!\mu^n} P_0, & C-2 \leq n \leq C-1 \\ \frac{\lambda_t^C \lambda_H^{n-C}}{C!\mu^C \prod_{j=1}^{n-C} [C\mu + j\mu_{call}]} P_0, & C \leq n \leq 2C \end{cases} \quad (5.24)$$

系统完全空闲状态概率  $P_0$  为:

$$P_0 = \left\{ \sum_{n=0}^{C-3} \frac{\lambda_t^n}{n!\mu^n} + \sum_{n=C-2}^{C-1} \frac{\lambda_H^n}{n!\mu^n} + \sum_{n=C}^{2C} \frac{\lambda_t^C \lambda_H^{n-C}}{C!\mu^C \prod_{j=1}^{n-C} [C\mu + j\mu_{call}]} \right\}^{-1} \quad (5.25)$$

当波束中所有信道都繁忙的概率则为新呼叫阻塞率  $P_b$ :

$$P_b = \sum_{n=C-2}^{2C} P_n \quad (5.26)$$

权值优化切换策略中利用预留一定比例的信道为需要切换的用户提供及时的服务并通过队列对各切换请求进行预定信道,模型中假定切换到达同样为独立的泊松到达,切换请求与新的呼叫请求的优先级一样,则掉话率或切换失败概率的计算如下:

$$P_f = \frac{\omega P_b P_{h1}}{1 - (1 - \omega P_b) P_{h2}} \quad (5.27)$$

$$\omega = \frac{(C-m)\lambda_n}{(2 \times C)\lambda_H}$$

式中  $m$  表示为切换请求预留的信道数目,则  $\omega$  表示切换所用信道与新呼叫用户所用信道比率。从公式 (5.17) ~ (5.19) 中看出  $\mu$  的计算依靠  $P_b$ , 为了计算  $P_b$  需要使用递归的方法, 已知系统的输入参数:  $C, \rho_n, T_{call}$  和  $\gamma$ , 递归迭代的方法如下图 5-13 所示。

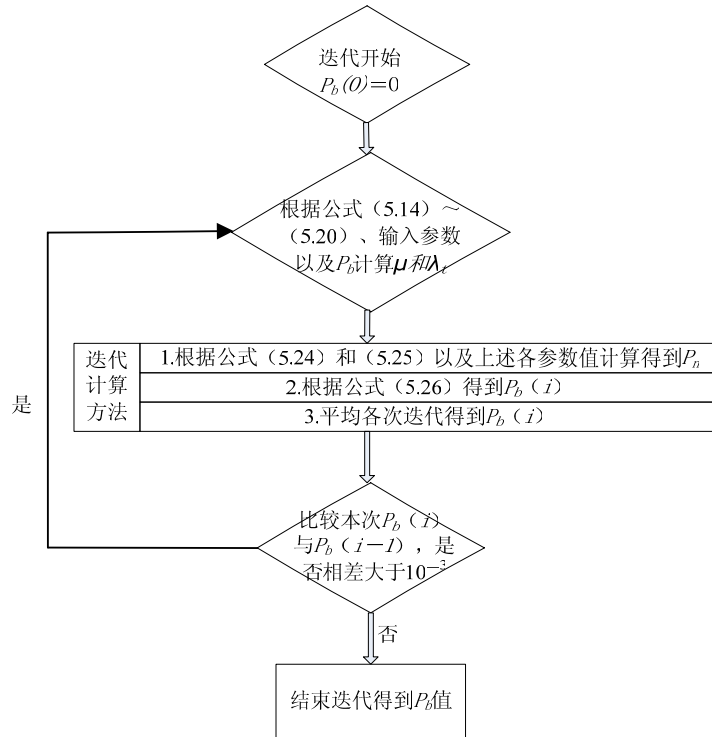


图 5-13 计算新呼叫阻塞概率递归流程

在卫星网络中设置业务 QoS 要求函数如下:

$$Q_{QoS} = w_f \times P_f + w_b \times P_b \quad (5.28)$$

根据系统的要求设置新呼叫阻塞率和切换失败概率各自的权值  $w_f$  和  $w_b$ , 同时对阻塞率  $P_b$  和掉话率  $P_f$  采用上述的估算, 由此可从理论上评估系统中切换策略的性能。同

时，还能为合理设置切换策略中各权值提供参考。

综上所述，低轨卫星网络中为降低掉话率，通过合理的路由策略和切换策略减少链路切换次数；执行切换时由于卫星导频信号的变化趋于平缓，采用位置测量，根据用户距离波束中心远近和相对位置变化趋势作为执行切换的基准；根据排队论估算的掉话率和阻塞率配置权值优化切换策略的各权值大小；通过预留 6%~25% 个信道提高切换成功率。此切换执行较为简单，但需要地面信关站时刻检测并传递各种辅助信息，切换策略性能在 OPNET 中建立的仿真平台中进行了简单的仿真。

## 5.5 仿真结果

为了估计权值优化策略的性能，根据星座模型设置各输入参数值：

轨道高度：1450Km；

呼叫持续时间：180S；

波束长度：1026Km，（因此  $T_b = 2.98 \text{ min}$ ， $\gamma = 1.007$ ）；

每个波束业务容量  $\rho_n$  从 6 到 14 erlangs 变化；

每个波束信道数：12；

每个波束用户数大约为：1900 个。

图 5-14 显示分别预留 1、2 和 3 个信道时，当波束中业务量密度变化时新的呼叫阻塞率。图中显示，同一业务量密度时信道预留越多阻塞率越大，针对预留同一信道的情况时，业务量越大阻塞率越高。图 5-15 显示分别预留 1、2 和 3 个信道时通话终止的概率，为切换预留信道越多在同一业务量密度下掉话率越低。

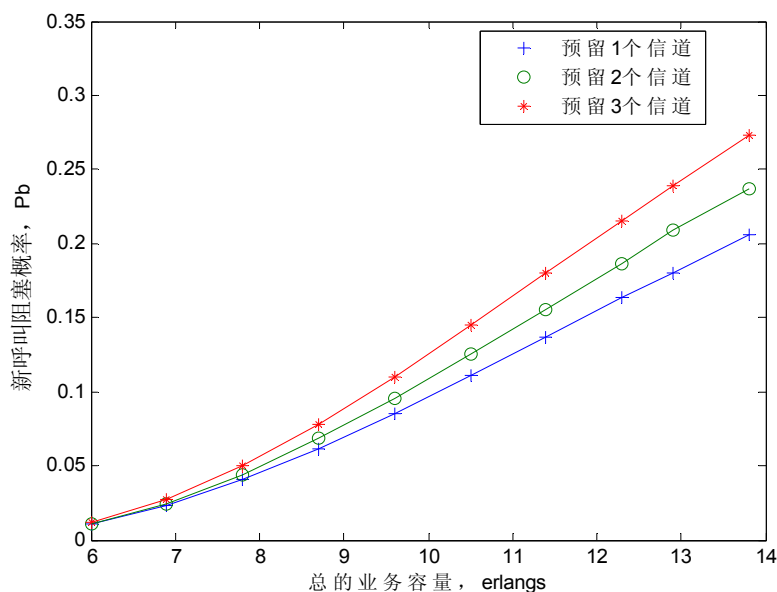


图 5-14 各信道预留下的阻塞率

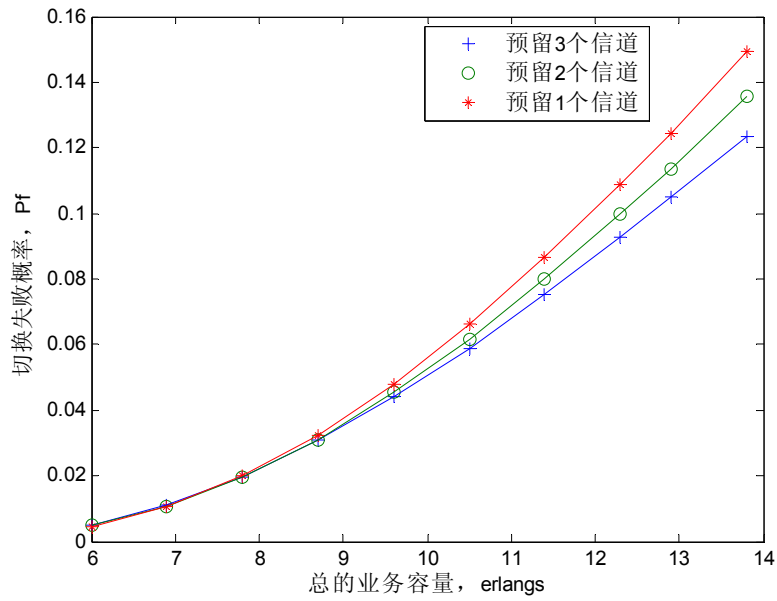


图 5-15 各信道预留下掉话率

整体显示当业务量密度小于 8erlangs 时阻塞率约为 5%，切换失败率低于 2%。通过上述的理论估算能有效地分配系统的信道资源，指导切换策略灵活设置预留信道数调节阻塞率和掉话率，并根据其概率估算值调节权值优化切换策略中的距离阈值，根据系统的要求增大或减小阈值。

同时，运行权值优化切换策略于低轨卫星网络仿真平台中，其权值设置为  $s_E = 0.1, s_T = 0.5, s_D = 0.4, s_\eta = 0$ ，为了简化仿真过程，仿真中重点验证了用户链路卫星切换的性能，因此其权值设置没有考虑各卫星波束中信道的占用率。仿真结果如下图所示。

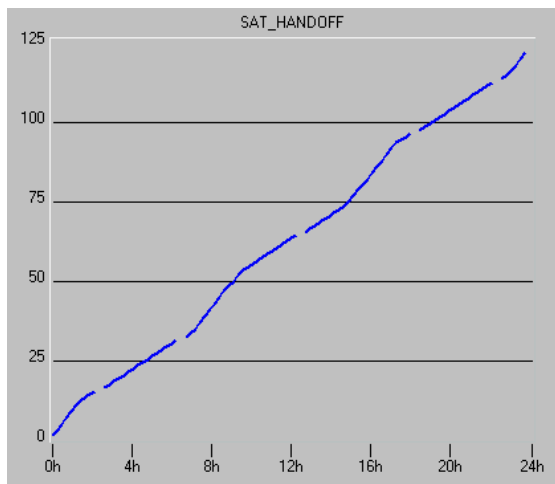


图 5-16 轨道周期内经历的卫星切换次数

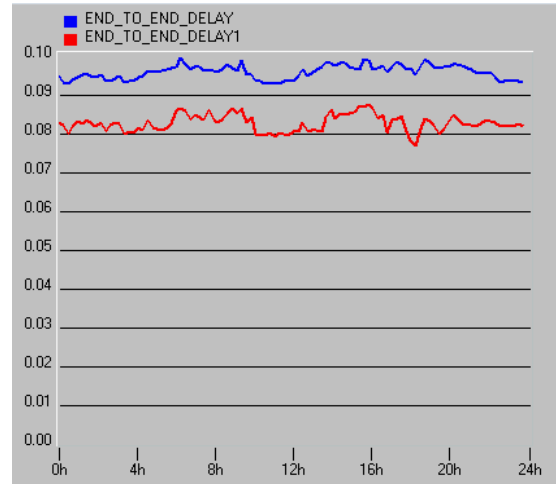


图 5-17 不同业务量下语音业务端到端时延

图 5-16 显示在卫星轨道周期内（24 小时）一直通信的用户经历的卫星切换次数，分析已知用户在单波束中的逗留时间为 2.98 分钟，在单星中逗留的时间大约 11 分钟，估算得到轨道周期内一直通信的用户经历的卫星切换次数平均为 131 次，图 5-16 中

显示经历了 120 次卫星切换，表明权值优化切换算法能有效降低卫星切换次数。图 5-17 显示了在不同业务容量下，语音分组端到端的时延性能，时延较平缓，没有较大时延抖动。当业务流量较大时，掉包率为 5.3%，与上述数学推导的估算相近。仿真结果证明了此切换策略能够保证较好的时延性能和较小的切换频率。

## 5.6 本章小结

本章首先分析了低轨卫星网络的波束切换和卫星切换执行流程，提出了权值优化切换策略。通过排队论分析了衡量切换策略性能的掉话率和阻塞率的估算，基于 OPNET 卫星网络仿真平台验证权值优化切换策略保证了较好的时延性能以及较小的切换频率。

本文的创新工作包括：

(1) 重点研究了低轨卫星网络切换策略中很少涉及的切换测量，提出并分析了位置测量的策略；基于位置测量提出了权值优化切换策略；

(2) 摆脱大多文献中分析切换算法时所用的低轨卫星网络简化的“街道模型”，在实际的卫星多波束天线覆盖模型中建立马尔可夫链，分析单个波束处于各状态的概率，理论推导出估算新呼叫阻塞率和切换失败概率的公式和递归方法，用于权值优化切换策略中各参数设置的参考。



## 第六章 全文总结

卫星通信网络按照空间信息资源的最大有效综合利用原则,实现互通互联,构成有机的智能化体系,具有自主的信息获取、传输、存储、处理、融合及分发能力,能与陆、海、空基的信息系统互联,以实现信息的多元、立体共享。低轨卫星通信网络是未来实施空间综合信息网的空间基础设施,是夺取空间信息优势的关键。

路由协议和分组调度是卫星通信网络研究中的关键技术,是实现卫星网中高效、可靠信息传输的基础。课题自开展研究以来,在借鉴国内外研究成果的基础上,针对系统通信协议、卫星网络拓扑特征、QoS 路由协议的设计与优化、交换调度算法的设计和优化以及切换策略上进行了大量的创新工作,取得了以下研究成果:

(1) 针对本文研究的星座模型和业务类型制定了通信协议架构和交换体制,基于此提出一种基于拓扑简化的分布式路由算法(DRAST)。从减少星间链路切换数目出发,对卫星网络拓扑分析并简化,并基于此设计了一种新的分布式路由策略。此算法能保证良好的时延性能,路由协议开销小,不用维护复杂的路由表,屏蔽了卫星网络的动态性。同时,卫星能够根据局部路由信息,以及拥塞、故障等链路状态更新拓扑表,及时调整分组路由,所以具有非常好的鲁棒性。建立了端到端时延参数的优先级排队模型,给出端到端时延参数的求解方法。

(2) 提出了合适的 CICQ 结构作为星载交换机。借鉴国内外研究成果,结合星载交换机结构以及业务特征和要求设计了一种新的 SDRR 调度算法,不仅保持了轮询调度算法复杂度低的优点,并提高了时延性能,交换机无须大容量的缓存和高的处理速度,适合资源受限的星载交换机。提出了路由协议与分组调度算法结合的思想,当某业务路径较长时分配其较大的服务额度,降低队列时延和处理时延,保证端到端的时延性能。

(3) 分析了低轨卫星网络的切换测量;为减少路由中切换的次数,设计了合适的权值优化切换策略;并利用马尔可夫链建立模型,获得了系统阻塞率和掉话率的求解,根据此解优化切换策略。

(4) 搭建了基于 OPNET 环境的卫星网络软件仿真平台,全面研究了网络拓扑结构实现、网络接入策略、邻居卫星节点发现方式、卫星运行轨道自主维护和星际链路模拟的实现方式;在软件仿真平台的基础上,分析并比较了 DRAST 算法和基于拓扑快照算法在不同业务流量下的时延性能、丢包率以及算法复杂度等性能;同时,在 OPNET 中建立了验证调度算法的模型,比较和分析 SDRR 算法与 DRR 算法在不同仿真场景下的性能;应用 SDRR 调度算法于卫星网络软件仿真平台中,证明了此调度算法的优越性。最后针对权值优化切换策略中的卫星切换进行了简要仿真和验证

本文内容较多,涉及范围较广,提出了多项具有一定创新性的设计方法和算法,并且在仿真方面也做了大量工作。除了使用专用的卫星仿真软件、网络仿真软件进行

相关的仿真外，还开发了大量的仿真程序（VC++的源代码数千行），形成了功能较为齐全的卫星通信仿真软件，并作为课题组的一项研究成果供实验室其他人员研究使用，本文的多数仿真结果就是来自于此软件。

卫星通信技术涉及非常广泛，而且处于不断发展之中。在以后的研究中，需要进一步研究以下几个方面的问题：

- （1）基于卫星的 TCP/IP 协议改进及相关算法设计，实现卫星互联网技术；
- （2）在卫星通信网络中通过无线资源管理算法的优化设计实现网络对各种业务的端到端 QOS 保证，并保证各算法在星上的可行性；
- （3）宽带多媒体移动卫星通信、星地一体化，在卫星通信网络中实现对不同类型业务的服务。

## 参考文献

- [1] 余金培, 李国通, 梁旭文, 杨根庆. 现代通信小卫星技术的发展与对策. 电子学报, 2001, No. 3, Vol. 29:100-105.
- [2] 余金培, 杨根庆, 梁旭文, 现代小卫星技术与应用, 上海科学普及出版社, 2004.
- [3] 张乃通, 刘会杰, 初海彬. 对发展卫星移动通信的几点思考. 电子电气教学学报, 2002.
- [4] 张更新, 张杭等. 卫星移动通信系统, 人民邮电出版社, 2001.
- [5] 吕海寰, 蔡剑明, 甘仲民等. 卫星通信系统. 人民邮电出版社, 1996.
- [6] Del Re E, Pierucci L. Next-Generation Mobile Satellite Networks. IEEE Communications Magazine. September 2002. pp.150-159.
- [7] Jamalipour A, Tung T, Giordano S. The Role of Satellites in Global IT: Trends and Implications. IEEE Personal Communications. June 2001. pp.5-11.
- [8] Theodore B. Zahariadis, Konstantinos G. Vaxevanakis, Christos P. Tsantilas, et al. Global Roaming in Next-Generation Networks. IEEE Communications Magazine. February 2002. pp.145-151.
- [9] Wood L. Internetworking with Satellite constellations. [PHD Thesis] Centre for Communication Systems Research, University of Surrey. 2001.
- [10] Lutz E, et al. Satellite Systems for Personal and Broadband Communications. Springer, 2000.
- [11] 张乃通, 张中兆, 李英涛等. 卫星移动通信系统. 电子工业出版社, 2000.
- [12] Giovanni Santella<sup>1</sup>, Roberto De Martino<sup>1</sup>, Enrico Russo. The European satellite telecommunications regulatory framework for broadband communications[A]. Baveno, Italy, 8th Ka Band Utilization Conference, Sept. 2002, 25-27.
- [13] Iida T, Suzuki Y. Communication satellite R&D for next 30 years. In: AIAA 19th ICSSC Conference, Toulouse, France, April 2001
- [14] Pelton J N. Satellite communication 2001. International Engineering Consortium, Chicago, Illinois, Dec 2000 vol.24, NO.1: 1-6
- [15] Jing Chen, Abbas Jamalipour; Adaptive Channel Management for Routing and Handoff in Broadcast WATM Mobile Satellite Network. Communications, CC 2001. IEEE International Conference on , Volume: 9 , 11-14 June 2001. 2928 ~ 2932.
- [16] Werner W, Jahn A, Lutz E, et al. Analysis of System Parameters for LEO/ICO-Satellite Communication Networks. IEEE Journal on Selected Areas in Communications, 1995, 13(2):371-381.
- [17] 刘刚, 苟定勇, 吴诗其. 低轨卫星星座网的切换研究. 通信学报, 2004 年 4 月, 25 卷 4 期.
- [18] Werner M, Delucchi C, Vogel H, et al. ATM-based Routing in LEO/MEO Satellite Networks with Intersatellite Links. IEEE Journal on Selected Areas in Communications, 1997, 15(1):69-82.
- [19] Werner M, A Dynamic Routing Concept for ATM-based Satellite Personal Communication Networks. IEEE Journal on Selected Areas in Communications, Vol.15, No.8, pp. 1636-1648, October 1997.
- [20] Chang H S, Kim B W, Lee C.G, S.L.Min, Y.Choi, H.s.kim, C.S.Kim, Performance

Comparison of Static Routing and Dynamic Routing in Low Earth Orbit Satellite Networks. Proceedings of IEEE VTC'96, pp.1240-1243, 1996.

[21] Chang H S, Kim B W, Lee C.G, et al. FSA-based Link Assignment and Routing in Low-earth Orbit Satellite Networks. IEEE Transactions on Vehicular Technology, 1998, 47(3):1037-1048.

[22] Vidyashankar V Gounder, Ravi Prakash, Hosame Abu-Amara. Routing in LEO-based Satellite Networks. Proceedings of IEEE Emerging Technologies Symp :Wireless Communications and Systems, April 1999:91-96.

[23] H. Uzunalioglu, Probabilistic Routing Protocol for Low Earth Orbit Satellite, Proc, ICC 98, pp.89-93. 1998.

[24] Admela Jukan, Hoang Nam Nguyen and Harmen R. van An approach to QoS-based for LEO Satellite Networks, 2000 International Conference on Communication Technology (ICCT 2000) IEEE . Vol .1(2000).ISBN 0-7803-6394-9:S. 922-929, 21.08.2000-25.08.2000.

[25] Roy Kucukates, Cem Ersoy, High Performance Routing in LEO Satellite Networks. Proceedings of The English International Symposium on Computers and Communication IEEE 2003. ISCC'03.

[26] M. Mohorcic, A. Svirgelj, G. Kandus, and M. Werner, Performance evaluation of adaptive routing algorithm in packet-switched intersatellite link networks, Intern. J. of Satellite Commun., vol.20, pp.97-120, 2002.

[27] M. Mohorcic, M. Werner, A. Svirgelj, and G. Kandus, Adaptive Routing for Packet-oriented Intersatellite Link Networks: Performance in Various Traffic Scenarios, IEEE Trans, Wireless Commun., vol.1, no.4, pp.808-818, October 2002.

[28] Jun Sun and Eytan Modiano. Routing Strategies for Maximizing Throughput in LEO Satellite Networks, IEEE Journal On Selected Areas In Communications, 22(2):273-286, Feb 2004.

[29] G. Kandus S. Kos M. Pustisek J. Bester A. Svirgelj, M. Mohorcic. Routing in ISL networks Considering Empirical IP Traffic. IEEE Journal On Selected Areas In Communications, 22(2):261-272, Feb 2004.

[30] Q.Huang, B.S Yeo and P.Y. Kong, A routing algorithm to provide end-to-end delay guarantee in low earth orbit satellite networks, to appear in IEEE VTC 2004' Spring, 2004.

[31] Qijie Huang, Boon Sain Yeo and Peng-Yong Kong, "An Enhanced QoS Routing Algorithm of End-to-End Delay Guarantee in Low Earth Orbit Satellite Networks", IEEE Communications Society/WCNC 2005.

- [32] E. Papapetrou and E-N. Pavlidou, A Proposal of Optimal Routing Techniques for non-GEO Satellite Systems. *International J. of Wireless Inform. Networks*, vol.8, no.2, pp.75-83, April 2001.
- [33] M.Mohorcic, M.Werner, A.Svigelj, G. Kandus. Alternative Link Routing for traffic engineering in packet-oriented ISL Networks. *Int. J. Satell. Commun.*, vol.19,pp.463-480.2001.
- [34] M.Mohorcic, A.Svigelj, G. Kandus, Y.F.Hu and R.E. Sheriff, Demographically weighted traffic flow models for adaptive routing in packet switched non-geostationary satellite meshed networks, in *Computer Networks*, pp.113-131, Apr.,16 2003.
- [35] Gragopoulos, I., Papapetrou, E., and Pavlidou, F-N(2000) Performance study of adaptive routing algorithm for LEO Satellite Constellations under self-similar and poisson traffic. *Space Communications*, 16 2000,15-22.
- [36] Papapetrou, E., Gragopoulos, I., and Pavlidou, F-N(1999) Performance evaluation of LEO Satellite Constellations with inter-satellite links under self-similar and poisson traffic. *International Journal of Satellite Communications*,17,1(1999),51-64
- [37] Werner, M., Wauquiez, F., Frings, J., and Maral, G.(1999) Capacity dimensioning of ISL networks in broadband LEO satellite systems, In *Proceeding of the International Mobile Satellite Conference*, 1999,334-341.
- [38] S.A.M. Makki, Niki Pissinou, and Philippe Daroux, A New Routing Algorithm for Low Earth Orbit Satellite Networks. *IEEE* 2001.
- [39] Janoso, R.F.(1996) Performance analysis of dynamic routing protocols in low earth orbit satellite data networks. Master's thesis, Air Force Institute of Technology, Wright-Patterson Air Force Base,1996.
- [40] Raines, R.A., Janoso, R.F., Gallagher, D.M and Coulliette, D.L.(1997) Simulation of two routing protocols operating in a low earth orbit satellite network environment. In *Proceedings of the IEEE Military Communication Conference*,1997,429-433.
- [41] Hashimoto Y. Design of IP-based Routing in A LEO Satellite Network. *Proceedings of the 3rd International Workshop on Satellite-Based Information Services*, 1988:81-88.
- [42] Ekici E, Akyildiz I F, and Bender M D. Bender, Datagram routing algorithm for LEO satellite networks. In *Proc, IEEE INFOCOM*, vol.2, Mar,2000,pp.500-508.
- [43] Ekici E, Akyildiz I F, and Bender M D. Bender, A Distributed Routing Algorithm for Datagram Traffic in LEO Satellite Networks. *IEEE/ACM Transactions on Networking*, 2001,9(2):137-147.
- [44] Mauger R, Rosenberg C. QoS guarantees for multimedia services on a TDMA-based

- satellite network. IEEE Communication Magazine, 1997, 35:56-65.
- [45] Berndt G, Werner M, and Edmaier B. Performance of Optimized Routing in LEO Intersatellite Link Networks. IEEE Proceedings of 47th Vehicular Technology Conference, 1997, 1:246-250.
- [46] P.Narvaez et al, Internet Routing over LEO Satellite Constellations, Third ACM/IEEE International Workshop on Satellite-based Information Services, October 1998.
- [47] Randy H. Katz Thomas R.Hendersom. "On Distributed, Geographic-Based Packet Routing for LEO Satellite Networks." IEEE Global Telecommunications Conference, 2(270):1119-1123, December 2000.
- [48] R. Kucukates and C.Ersoy. High Performance Routing in a LEO Satellite Networks. The Eighth IEEE International Symposium on Computers and Communications (ISCC'03), KemerAntalys, Turkey, pages 1403-1408, 2003.
- [49] R.Mauger and C.Rosenberg. QoS Guarantee for Multimedia Services on a TDMA-Based Satellite Networks. IEEE Communications Magazine, 35(7):56-65, July 1997.
- [50] L.Wood etc. IP Routing Issue In Satellite Constellation Networks. International Journal of Satellite Communications, 19(10):69-92, January/February 2001.
- [51] Yurong Hu and Victor O.K.Li. Satellite-Based Internet: A Tutorial. IEEE Communication Magazine, 39(3):154-162, 2001.
- [52] Y. Hu. And V.O.K. Li, Logic Topology-Based Routing in LEO Constellations, Proceedings of IEEE International Computer and Communications. Helsinki. Finland. 2001.
- [53] Uzunalioglu H, Akyildiz I.F, Yesha Y, et al. Footprint handover rerouting protocol for low earth orbit satellite networks[J]. ACM Wireless Networks, 1999.327-337.
- [54] 王亮, 张乃通. LEO 网络中卫星切换的动态概率路由优化策略, 通信学报, 2002 年 9 月第 23 卷 第 9 期。
- [55] 白建军, 卢锡城, 彭伟, LEO 卫星网络中一种简洁的星上分布式路由协议, 软件学报, 2005 vol.16, No12.
- [56] Henderson, T.R.(1999) Networking over next-generation satellite systems. Ph.D. dissertation, University of California at Berkeley, 1999.
- [57] Abhay K.Parekh and Robert G.Gallager, A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks: The Single-Node Case. IEEE/ACM Transactions on Networking, Vol.1. NO.3, June 1993.
- [58] Jon C.R. Bennett and Hui Zhang, WF<sup>2</sup>Q: Worst-case Fair Weighted Fair Queueing. Proceedings of IEEE INFOCOM '96, pp.120-128, March 1996.

- [59] L.Zhang, Virtual Clock: A New Traffic Control Algorithm for Packet Switching. ACM Transactions on Computer systems, vol.9, no.2, pp.101-124, May 1991.
- [60] S.J.Golestani, A Self-Clocked Fair Queueing Scheme for Broadband Applications. Proceedings of IEEE INFOCOM '94, pp.636-646, April 1994.
- [61] Subhash Suri, George Varghee and Girish Chandranmenon. Leap Forward Virtual Clock: A New Fair Queueing Scheme with Guaranteed Delays and Throughput Fairness. Proceedings of IEEE INFOCOM '97, 1997.
- [62] D.Stiliadis and A.Varma, Efficient Fair Queueing Algorithms for ATM and Packet Networks. Technical Report UCSC-CRL-95-59, December 1995.
- [63] ANDERSON T, et al. High speed switch scheduling for area networks[J]. ACM Trans Comput Syst, 1993, 11(4):319-352.
- [64] MCKEOWN N. Scheduling in Input-Queued Switch[D]. University of California at Berkeley, 1995.
- [65] MCKEOWN N. The iSLIP scheduling algorithm for input-queued switches[J]. IEEE/ACM Trans on Networking, 1999, 7(2):188-201.
- [66] SERPANOS D N, et al. FIRM: A class of distributed scheduling algorithms for high-speed ATM switches with multiple input queues[A]. IEEE INFOCOM'00[C]. Tel Aviv, Israel, 2000.548-555.
- [67] JIANG Y, HAMDI M. A 2-stage matching scheduler for a VOQ packet switch architecture[A]. IEEE ICC'02[C]. New York, NY, USA, 2002.2105-2110.
- [68] MCKEOWN N. et al. A starvation-free algorithm for achieving 100% throughput in an input-queued switch[A]. IEEE ICCCN'96[C]. Rockville, Maryland, USA, 1996.226-231.
- [69] MEKKITTIKU A, et al. A practical algorithm to achieve 100% throughput in input-queued switches[A]. IEEE INFOCOM'98[C]. San Francisco, CA, USA, 1998.792-299.
- [70] LIU N H, et al. Scheduling algorithm for input-queued switches with virtual output queueing[A]. IEEE ICC'01[C]. Helsinki, Finland, 2001.2038-2042.
- [71] TABATABAEE V, et al. MNMCM a new class of efficient scheduling algorithms for input-buffered switches with no speedup[A]. IEEE INFOCOM'03[C]. San Francisco, CA, USA, 2003.1406-1413.
- [72] M. Katavenis, S. Sidiropoulos, and C. Courcoubetis, Weighted Round-Robin Cell Multiplexing in a General-Purpose ATM Switch Chip, IEEE Journal on Selected Areas in Communication Vol. 9, No.8, pp.1265-1279, Oct.1991.
- [73] S.S Kanhere, H. Sethu, and A.B. Parekh. Fair and efficient packet scheduling using elastic round robin. IEEE Trans. Parallel Distrib. Syst., 13(3):324-336, Mar.2002.
- [74] Lain-Chyr Hwang, Steen J. Hsu, San-Yuan Wang and Yong-Hua Huang, A Hybrid Scheduling Algorithm with Low Complexity: Jumping Virtual Clock Round Robin, Proceeding of the 25th IEEE International Conference on Distributed Computing Systems Workshops, 2005.
- [75] Roberto Roja-cessa, High-Performance Round-Robin Arbitration Schemes for Input-crosspoint Buffered Switches. IEEE 2004.
- [76] Viet Linh Do, Practical Scheduling Algorithm in Packet Switch[D], thesis, UMI Number:3071047, University of California . San Diego..
- [77] M. Shreedhar and George Varghese, "Efficient Fair Queueing Using Deficit Round

- Robin”, IEEE/ACM Transactions on Networking, VOL.4,NO.3, June 1996.
- [78] Lucinao Lenzini, Enzo Mingozzi and Giovanni Stea, Tradeoffs Between Low Complexity, Low Latency, and Fairness With Deficit Round-Robin Schedulers. IEEE 2004.
- [79] Yao Liang, A Simple and Effective Scheduling Mechanism Using Minimized Cycle Round Robin. IEEE/ACM Transactions on Networking, 2002.
- [80] Aysegul Tuysuz, Fatih Alagoz “Satellite Mobility Pattern based Handover Management Algorithm in LEO Satellites” IEEE ICC 2006 proceedings
- [81] G.Maral, J. Restrepo et al “Performance analysis for a guaranteed handover service in an LEO constellation with a ‘satellite\_fixed cell’ System” IEEE Transaction on Vehicular Technology,47(4):1200-1214,November 1998.
- [82] Y. XU, Q.Ding, and C. Ko. “Elastic handover scheme for LEO satellite mobile communication systems.” In IEEE Global Telecommunications Conference, pages 1161-1165, San Francisco, CA,USA,27 November -1 December 2000.
- [83] L.Boukhatem, G. Pujolle,and D.Gaiti. “A time-based reservation scheme for managing handovers in satellite systems.” International Journal of Network Management, 13(2):139-145,March/April 2003.
- [84] E.Papapetrou and F-N. Pavlidous. “QoS handover management in LEO/MEO satellite systems”. Wireless Personal Communications,24(2):189-204,February 2003.
- [85] V.Santos, R.Silva, M.Dinis, and J.Neves. “Performance evaluation of channel assignment strategies and handover policies for satellite mobile networks.” In Annual International Conference on Universal Personal Communications, pages 86-90, Tokyo,Japan,6-10 November 1995.
- [86] E.D.Re, R. Fantacci and G.Giambene. “Handover queuing strategies with dynamic and fixed channel allocation techniques in low earth orbit mobile satellite systems” IEEE Transactions on Communication, 47(1):89-102, January 1999.
- [87] Y.Lin, S.Mohan, and A. Noerpel. “Queuing priority channel assignment strategies for PCS hand-off and initial access.” IEEE Transactions on Vehicular Technology, 43:704-712,August 1994.
- [88] S.Olariu, S.R.Rizvi, R.Shirhatti, and P.Todorova, “Q-WIN- A new admission and handoff management scheme for multimedia LEO satellite networks,” Telecommunication Systems, 22(1-4),2003,15 1-168.
- [89] Mertzanis I, Sfikas G, Tafazolli R, et al. Protocol Architectures for Satellite ATM Broadband Networks. IEEE Communications Magazine, 1999, 37(3):46 – 54.
- [90] Papapetrou E, Karapantazis S, Dimitriadis G, et al. Satellite Handover Techniques for LEO Networks. International Journal of Satellite Communications and Networking, 2004, 22(2):231-245.
- [91] 卞佳丽 等, 现代交换原理与通信网技术, 北京邮电大学出版社, 2005 年.
- [92] Hu Y, Li V O K.Satellite- based internet: A tutorial[J].IEEE Communications Magazine, 2001; 39( 3) : 154~162.
- [93] Toh Chai- Keong, Li Victor O K.Satellite ATM network architectures: an overview[J]. IEEE Network, 1998; 12( 5) : 61~71.



- [94] Akyildiz Ian F, Jeong Seong- Ho. Satellite ATM networks: A survey[J]. IEEE Communications Magazine, 1997; 35( 7) : 30~43.
- [95] I Mertzanis et al. Protocol Architectures for Satellite ATM Broadband Networks[J]. IEEE Commun Mag, 1999: 46~54.
- [96] Landry Randall, Scher David, Burdin Tared et al. An efficient packet switching architecture for next- generation military satellite networks[C]. In: Proceedings- IEEE Military Communications Conference MILCOM, 2003; 1: 268~273.
- [97] 王勇前, 马正新, 曹志刚. 一种基于星上处理的卫星 ATM 网络动态带宽分配策略. 通信学报. 2001, 22(4): 7~8.
- [98] 王勇前, 马正新, 曹志刚. 卫星 TCP/ATM 传输中的明确窗口指示拥塞控制策略研究, 电子学报, 2001 年 1 月第 1 期.
- [99] Sriram Subramanian, TCP Performance Issues In Satellite Networks. Thesis of master of science in Dalhousie University. 2004.12.
- [100] 顾明, 张军. 适用于卫星网络的 TCP 跨层改进机制, 电子与信息学报, 2008 年 8 月第 8 期.
- [101] 王平, 顾学迈. LEO 卫星网络中 TCP 协议性能及路由策略研究, 南京理工大学学报, 2007 年 2 月第 31 卷第 1 期.
- [102] 刘炯, 曹志刚. 一种适合卫星网络的拥塞控制算法, 宇航学报, 2007 年 5 月第 28 卷第 3 期.
- [103] Evangelos Papapetrou, Fotini-Niovi Pavlidou. "Distributed Load-Aware Routing in LEO Satellite Networks" IEEE GLOBALCOM 2008.
- [104] Dongni Li, Xin Wang and Ya Meng. "A Destruction-resistant Routing Algorithm in Low Earth Orbit Satellite Networks" IEEE Wireless Communication 2007.
- [105] Derya Yiltas, A, halim Zaim. A Dynamic Routing Algorithm in LEO Satellites System Estimating Call Blocking Probabilities[J]. IEEE 2007
- [106] Yan He, Susanna Pelagatti, "CRT: an Adaptive Routing Protocol for LEO Satellite Networks" IEEE 2006.
- [107] Wang Ping, Gu Xue-man and Liu Gong-liang, "Multi-Qos Routing for LEO Satellite Networks" ICACT 2007.
- [108] ZhiGang Zhao, A Robust ATM-Based Algorithm in LEO Satellite Communication Networks[J]. IEEE 2007
- [109] Stylianos karapantazis, Evangelos Papapetrou and Fotini-Niovi Pavlidou, On-Demand Routing in LEO Satellite Systems[J]. IEEE Communications Conference, July 2007.
- [110] Trivedi K S. Probability & Statistics with Reliability, Queuing, and Computer Science Application. Prentice-Hall, Inc. 1982.
- [111] 熊庆旭, 输入排队结构交换机分组调度研究, 通信学报, 2005 年 6 月, 第 26 卷第 6 期.
- [112] E. Del Re, R. Fantacci, and G. Giambene, Efficient dynamic channel allocation techniques with handover queuing for mobile satellite networks. IEEE J. Sel. Areas Commun., vol. 13, pp. 397-405, Feb. 1995.

## 作者攻读博士学位期间发表的论文

- [1] Zeng Yuan, Liang Xuwen, Li Yong, A Distributed Routing Algorithm Based-on Simplified Topology in LEO Satellite Networks, High Technology Letters, 2010, No 1.
- [2] Zengyuan, et al, LEO Satellite Scheduler Model and Simulation , The 4th IEEE International Conference on Wireless Communications, Networking and Mobile Computing, October 12-14, 2008, Dalian, China.
- [3] 曾媛, 龚文斌, 刘会杰, 梁旭文, 低轨卫星交换机的建模与仿真, 系统仿真学报, 2009 年第 6 期.
- [4] 曾媛, 龚文斌, 刘会杰, 梁旭文, 适合星载交换的一种调度算法, 计算机工程, 2009 年第 6 期.
- [5] 曾媛, 龚文斌, 刘会杰, 梁旭文, 基于 OPNET 的低轨卫星网络建模与仿真, 卫星与网络, 2008 年第 10 期.

## 致 谢

首先非常感谢我的导师梁旭文研究员，在我四年的博士生涯中给予我的关心与帮助。梁老师对学术前沿的精确把握、在科研工作中敏锐的洞察力、对工程技术的深刻理解和丰富的实践经验等无一不体现了他渊博的知识以及严谨的治学态度。梁老师除了在学习、科研工作上关心我外，还在生活中关心、帮助我，使我能够全身心投入学习和科研。在治学和为人方面，梁老师树立了很好的榜样，对我以后的学习、工作将产生深远的影响。在此，谨向他表示最诚挚的谢意。

感谢通信技术室主任刘会杰研究员和副主任龚文斌老师，他们严谨的治学态度、缜密的思维能力、深厚的理论造诣让我受益匪浅，获得了很多的启发。教研室研究生众多，科研和日常管理工作十分繁重，即使如此两位老师仍然时常关心我的研究工作，督促我抓紧时间，保质保量地完成工作，给我提供了良好的研究环境，是你们的辛勤劳动使我的论文得以顺利完成。在此向刘老师、龚老师致以衷心的感谢。

感谢梁继业、帅涛、任前义、赵国栋、陈晓挺、姚晔、江玉洁、朱丽、刘翔、贾铂琦、金萍、梁广、马陆、胡泽鑫、吕源、赵星惟、丁晟、刘俊、顾菁华、赵君伟、李理敏、杜志远、许培培、姜兴龙、那顺巴雅尔、姜泉江等同学，无论在学习还是生活中，我们互相帮助，互相支持，与你们共度的美好时光使我终生难忘，深深感谢你们，愿我们的友谊地久天长。

感谢所研究生部余翔波老师和小卫星综合办周素华老师在工作、生活上的关心和帮助，让我们这些身在外地的学子时时刻刻都感受到大家庭的温暖。

我要特别感谢我敬爱的父母，感谢你们的理解和无私的奉献；特别感谢我的丈夫李小卫，谢谢你对我生活和工作上的支持、帮助和鼓励。

最后感谢所有参加论文评审的专家，感谢你们在百忙之中对我的论文进行批评指正。

## 作者简历

姓 名 曾媛  
性 别 女  
籍 贯 湖南省涟源市  
出生日期 1979.6.27

### 主要简历:

2005/09-2009/06 中国科学院上海微系统与信息技术研究所  
通信与信息系统 工学博士  
2004/07~2005/09 中国科学院上海微系统与信息技术研究所  
传感器国家重点实验室 助理工程师  
2001/09-2004/07 中国科学院武汉物理数学研究所  
应用物理专业 理学硕士  
1997/09-2001/07 湖南师范大学  
电子信息工程 工学学士  
1994/09 - 1997/07 湖南省涟源一中

### 参加研究课题情况

2005 年 10 月开始参加中国科学院创新工程重大项目某卫星的研制工作,编写验证通信协议的软件;

2006 年 4 月-2007 年 5 月

参加十一五低轨卫星系统预研项目,参与设计了低轨卫星网络通信协议架构的工作,完成了导师布置的设计低轨卫星网络通信协议、星上交换体制和切换策略设计。

2007 年 2 月-2008 年 3 月

负责预研项目中低轨卫星网络的星载交换机结构设计,并设计了交换分组调度算法,通过 OPNET 软件进行了仿真和验证。

2007 年 8 月-2008 年 5 月

负责预研项目中低轨卫星网络路由协议的设计,并对路由协议进行了仿真和验证。

2007 年 6 月-2008 年 12 月

负责预研项目中软件仿真总体工作,编写了功能较为齐全的低轨卫星网络软件仿真平台。

---

## 中国科学院上海微系统与信息技术研究所

### 学位论文独创性声明

本人声明所呈交的学位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的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得中国科学院上海微系统与信息技术研究所或其它教育机构的学位或证书而使用过的材料。与我一起工作的同志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。

研究生签名 \_\_\_\_\_ 日 期 \_\_\_\_\_

## 中国科学院上海微系统与信息技术研究所

### 学位论文使用授权声明

本人完全了解中国科学院上海微系统与信息技术研究所有关保留、使用学位论文的规定，即研究所有权保留送交论文的复印件，允许论文被查阅和借阅；可以公布论文的全部或部分内容，可以采用影印、缩印或其他复制手段保存论文。本人电子文档的内容和纸质论文的内容相一致。保密的论文在解密后遵守此规定。论文的公布(包括刊登)授权中国科学院上海微系统与信息技术研究所人才教育处。

研究生签名 \_\_\_\_\_ 导师签名 \_\_\_\_\_ 日 期 \_\_\_\_\_