

分类号

学校代码 10487

学号 M202172537

密级 公开

硕士学位论文

(学术型 ☐ 专业型 ☒)

低轨卫星网络路由策略及基于 TC 码的 传输方法

学位申请人：曹佩青

学 科 专 业：通信工程

指 导 教 师：魏蛟龙 教授

答 辩 日 期：2024 年 5 月 19 日

**A Dissertation Submitted in Partial Fulfillment of the Requirements
for the Master Degree in Engineering**

**Routing Strategies and Transmission Methods Based on
TC Code for Low Earth Orbit Satellite Networks**

Candidate :CAO Peiqing

Major : Communication Engineering

Supervisor :Prof. WEI Jiaolong

Huazhong University of Science and Technology

Wuhan 430074, P, R, CHINA

May, 2024

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，除文中已经标明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的研究成果。对本文的研究做出贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律结果由本人承担。

学位论文作者签名：曹丽青

日期：2024年5月19日

学位论文版权使用授权书

本学位论文作者完全了解学校有关保留、使用学位论文的规定，即：学校有权保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权华中科技大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

保 密 ☐，在_____年解密后适用本授权书。

本论文属于 不保密 ☒。

（请在以上方框内打“√”）

学位论文作者签名：曹丽青

日期：2024年5月19日

指导教师签名：张屹屹

日期：2024年5月19日

摘要

随着低轨卫星网络的快速发展，其网络拓扑日益庞大复杂，低轨卫星的高速运动导致了拓扑的高动态性，时变的流量和节点状态使得最优路由的计算变得困难。卫星网络所处的有损环境使得卫星数据传输变得不可靠，数据传输链路不稳定，链路的中断将导致超时重传，增加了端到端时延；不对称路径导致数据包不能按序抵达接收端，增加了接收端的处理时延。针对上述问题。本文从多路径路由和可靠传输两个方面展开了研究，主要内容如下：

（1）基于 SDN 的低轨卫星多径路由策略。针对低轨卫星网络的大规模拓扑结构以及卫星快速运动引起的拓扑变化难题，本文提出了一种利用软件定义网络（Software Defined Network, SDN）的多路径路由策略，该策略首先运用星历信息和时间片划分技术来预测未来的拓扑变化，计算出所有可能的星间传输路径，实现对动态拓扑的精确感知。通过 SDN 技术实时监控网络的负荷情况，综合考虑时延、丢包率、带宽和节点负载，构建链路传输代价计算模型，通过最小传输代价算法来实现多路径路由。本文设计了一种基于 SDN 的双层卫星网络架构，结合地面站和 GEO 卫星的优势，建立星地联合部署的控制器部署方式，LEO 卫星平面作为数据转发层，使用地面网络和三颗 GEO 卫星作为 SDN 控制平面来实现对网络的集中控制，结合路由算法可以实现高效的路由计算。

（2）基于 TC 码的卫星网络可靠传输算法。由于卫星所处环境恶劣，传输过程中存在损耗，可能会导致数据包的丢失，且当选择的链路之间耦合度较低时，不同路径之间存在较大时延差，会导致在接收端出现数据包的乱序问题。针对上述问题，本文提出基于 TC 码的卫星网络可靠传输算法，设计了 TC 码编解码传输方案，通过双向搜索动态迭代算法，寻找最优的编码数据包大小，通过轻量级的反馈就能实现编码批次大小对传输任务和传输环境的动态自适应调整，缩短联合传输时延。针对 TC 码的特点，设计了高解码率、低复杂度的 BLBP 译码算法，提升了传输效率。

关键词：低轨卫星网络；多路径路由；可靠传输；TC 码

Abstract

With the rapid development of low-earth orbit (LEO) satellite networks, their network topology is becoming increasingly large and complex. The high-speed movement of LEO satellites leads to the high dynamism of the topology, while the time-varying traffic and node states make the computation of optimal routing difficult. The lossy environment in which satellite networks operate makes satellite data transmission unreliable. Unstable data transmission links and interruptions in the links result in timeout retransmissions, increasing end-to-end latency. Additionally, asymmetric paths cause data packets to arrive at the receiving end out of order, further increasing processing latency at the receiver. To address these issues, this dissertation conducts research from two aspects: multipath routing and reliable transmission.

(1) Multipath routing strategy based on SDN (Software Defined Networking) for LEO satellites. Considering the vast topology of LEO satellite networks and the topology transformation caused by the rapid movement of satellites, this paper proposes a multipath routing strategy based on the Software Defined Networking (SDN) architecture. SDN is employed for network load awareness, while ephemeris-based technology is used for dynamic topology awareness. By considering factors such as latency, bandwidth, and node load, a link transmission cost calculation model is constructed, and a minimum transmission cost algorithm is used to achieve multipath routing. This paper designs a dual-layer satellite network architecture based on SDN, utilizing a ground station + GEO + LEO satellite joint deployment method. The LEO satellite plane serves as the data forwarding layer. Combined with routing algorithms, efficient routing calculations can be achieved.

(2) Reliable transmission algorithm based on TC (Triangular Code) for satellite networks. Due to the harsh environment in which satellites operate, there may be losses during transmission, leading to packet loss. Additionally, when there is low coupling between selected links, significant delay differences between different paths may result in packet disorder at the receiving end. To address these issues, this paper proposes a reliable transmission algorithm based on TC Code for satellite networks. Considering transmission scenarios without feedback links, a TC encoding and decoding transmission scheme is designed. Through bidirectional search dynamic iteration algorithms, the optimal coding

packet size is found to adapt to transmission tasks and environments, thereby reducing joint transmission latency. Leveraging the characteristics of TC, a BLBP (Bit-Level Belief Propagation) decoding algorithm with high decoding rate and low complexity is designed, improving transmission efficiency.

Key words: Low Earth Orbit (LEO) Satellite Networks, Multipath Routing, Reliable Transmission, TC Codes

目录

摘要.....	I
Abstract.....	II
目录.....	IV
1 绪论.....	1
1.1 研究背景与意义	1
1.2 国内外研究现状	2
1.3 论文内容安排	6
2 低轨卫星网络数据传输关键技术概述	7
2.1 卫星星座介绍	7
2.2 卫星网络路由技术	11
2.3 卫星网络可靠传输技术	12
2.4 本章小结	19
3 基于 SDN 的低轨卫星多路径路由策略	20
3.1 引言	20
3.2 算法设计考虑因素	20
3.3 基于 SDN 的低轨卫星多路由策略	21
3.4 基于 SDN 的双层卫星网络架构及实现	28
3.5 仿真与分析	39
3.6 本章小结	43
4 基于 TC 码的卫星网络可靠传输方法.....	45
4.1 引言	45
4.2 TC 码的优势	45
4.3 基于反馈的码长调整机制	46

硕士学位论文

4.4	基于 TC 码的卫星网络可靠传输方法	50
4.5	仿真与分析	54
4.6	本章小结	59
5	总结与展望.....	60
5.1	全文总结	60
5.2	后续工作展望	60
	致 谢.....	62
	参考文献.....	63

1 绪论

1.1 研究背景与意义

由于陆基设施的覆盖范围有限，无法向全球提供所需的通信服务，而低轨卫星网络能够实现全球覆盖^[1]，另外，低轨卫星系统通常具有高度的灵活性和可靠性，能够适应各种环境条件，这种全球覆盖的连通性和灵活性至关重要^[2]。

目前，各个国家及地区都通过构建天基卫星网络来与地面网络形成互补。一颗静止地球轨道卫星距离地面约 35786 公里，其波束能覆盖 40% 的地球表面^[3]。除此之外，卫星网络的部署不受地形条件的限制，也不易受到地震、海啸等自然灾害的破坏^[4]。相较于中高轨卫星，低轨卫星的轨道高度通常在几百到几千公里之间，信号的传播时延更小，意味着用户能够获得更快的响应时间。随着卫星技术的进步和制造成本的降低，低轨卫星网络的建设和维护成本逐渐降低，使其成为成本效益高的通信解决方案^[5]。

然而，卫星网络与地面网络在拓扑构造和底层链路特性上存在着明显的区别，这使得以往地面网络的研究成果无法直接适用于卫星网络的场景，需要进一步的研究和探索。由于低轨卫星的高速移动，其网络拓扑也会迅速变化，对卫星网络的路由带来了挑战^[6]。此外，无线传输过程中的空间干扰也会使传输线路变得不稳定，一旦链路节点出现中断，整条路径都会失效，如何动态维护链路的稳定性和传输的可靠性成为了一大难题^[7]。

目前大部分的网络通信的传输层协议都是 TCP 协议，TCP 协议为避免拥塞，设置了慢启动状态，当卫星链路环境较差时，TCP 将会长时间处于慢启动状态，无法进行正常速率的数据传输^[8]。若卫星网络的传输路径耦合度较低，可能会使通过不同路径传输的数据包到达时间相差较大，在接收端出现数据包乱序问题，增加接收端处理时延。多变的太空环境也会使通信过程中存在各种干扰，造成较高的误码率，从而导致重传，影响传输效率^[9]。

综上所述，随着低轨卫星互联网的兴起^[10]，传统的路由方法以及保障可靠传输

的机制都难以适应卫星网络传输场景^[11]。本文针对大规模低轨卫星网络中动态构建通信路径的挑战^[12]、难以保障传输可靠性^[13]等科学问题，以基于 SDN 的多路径路由策略、基于自适应 TC 码的网络传输方法为技术手段，为提升卫星网络服务质量探索解决方案。

1.2 国内外研究现状

在卫星通信中，数据首先由终端设备产生，通过 Wifi 等方式发送到地面站，地面站再借助星地上传链路将数据转发至最近的起始卫星 A，卫星 A 随后利用微波、毫米波、光纤或激光等技术，通过卫星间的连接链路逐步将数据传递给距离用户最近的目标卫星 B，卫星 B 再将数据通过星地下行链路发送给地面上的目标用户^[14]。本文的主要关注的是星间数据传输阶段。

1.2.1 卫星网络路由研究现状

实现全球卫星通信的无缝覆盖的需求促使各国积极部署了众多卫星节点，这使得卫星网络资源的管理变得复杂，也给如何优化网络传输带来了困难。快速变化的链路状态和高动态的网络拓扑使得路由算法的设计在低轨道卫星网络中的研究始终是个难题^[15]。

为了规避卫星移动带来的拓扑变化，一个经典的解决思路是使用虚拟拓扑，将卫星网络的变化周期进行离散，如图 1-1 所示。将卫星的位置都抽象为固定的节点，会得到一系列的静态拓扑，再由地面站提前计算好传输路径，将路径上传给卫星^[16]。如 Chang 等人提出的基于有限状态机的路由算法^[17]，文献[20]提出的快照序列算法，以及文献[21]提出的 CEMR (Compact Explicit Multi-path Routing) 算法，该算法利用全局已知路径标识符，在低轨卫星网络中实现 k-shortest 多路径路由。但随着卫星节点数目的不断增长，卫星需存储庞大的路由表，且星座在每个周期内的拓扑变化速度增快，对内存的需求也相应增大，传统的虚拟拓扑策略在这种情况下变得难以实施。

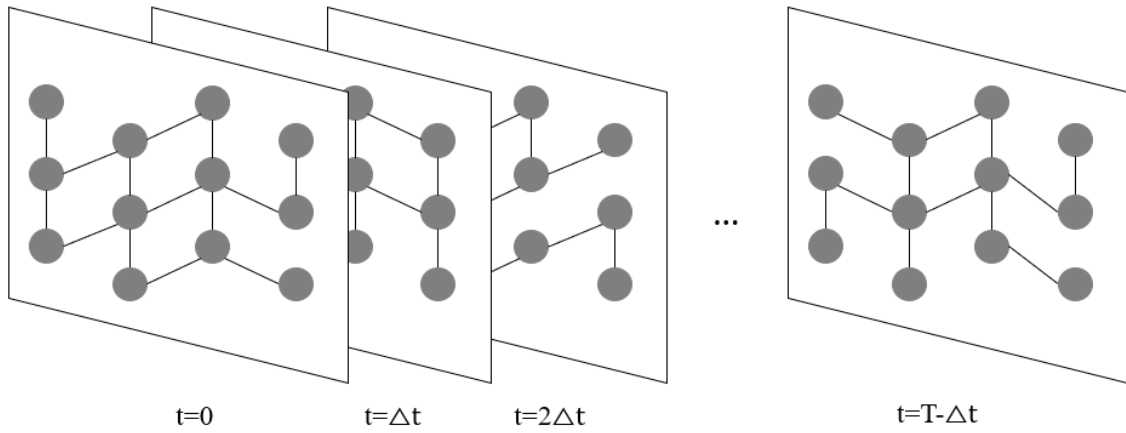


图 1-1 基于虚拟拓扑的路由

研究人员开始研究基于区域划分的路由算法，即虚拟节点^[22]，思路是根据卫星所在的轨道面和位置，给卫星分配一个逻辑地址，由于地面区域的逻辑地址始终是不变的，基于逻辑地址的路由是不变的，变化的只有逻辑地址与卫星之间的映射，将动态的卫星网络研究转化为静态的逻辑网络研究。如文献[23]、[24]提出的基于卫星网络的分布式数据包路由算法，该算法通过方向估计、方向修正、拥塞处理来计算路径，以及文献[25]提出的 DGRA（Distributed Geographic Routing Algorithm）、文献[26]提出的 LCRA（Low-Complexity Routing Algorithm）等。然而，现有的基于虚拟节点的路由思路都只针对与用户进行连通的切换，而没有将卫星间的连通情况进行分析，并且当卫星数量增加时，其所涵盖的范围也会随之发生重叠。这给地表覆盖面积的分割带来了很大难度。

由于卫星网络本质上仍是一种无线网络，已有很多研究尝试改变以往的静态方法，使各卫星之间进行状态信息的交流，从而获取最新的网络状态。文献[27]提出了 ELB（Explicit Load Balancing）算法，该算法运用在相邻卫星间显示地交换拥塞状态信息的思想，卫星不断地监视当前队列的大小，以确定它的状态，在即将出现拥塞时，卫星会将自身状态信息反映给邻近的卫星，从而减缓其传输速率，相邻卫星搜索其他不包含该拥塞卫星的备份路径进行传输，可以避免卫星的拥塞和数据包的丢失，但这种算法仍然存在通知延迟的问题，不适合解决实际网络中的拥塞问题。文献[28]提出了一种基于信号灯的智能路由策略，该策略是由地面调度中心预先计

算多条发送路径,并通过“红黄绿”信号灯的指示来判断其是否处于繁忙状态,并在此基础上实时调整下一步的选择。然而,TLR只能检测当前节点及下一跳的状态,无法了解整个链路的状态,很容易忽略其他阻塞节点的信息。ELB算法和TLR算法都是分布式地对节点的拥塞问题进行处理,缺乏全局计算能力,容易陷入次优循环以及级联拥塞^[29]。文献[30]设计了灵活的网络系统,解决了卫星节点移动导致的网络资源利用率低的问题,通过构建数据中心使地面网络与卫星网络能够很好地结合。实验显示,这种方法有效增强了网络的适应性和路由扩展性能。但由于资源的限制,在现有框架下配置和管理大规模网络仍然很困难。文献[31]提出了一种基于代理建模、聚类技术和博弈论的集群网络框架,以保证稳定性和可扩展性。文献[32]通过分层按需方式对天地一体化网络资源进行管理和调度,提出了一种高效的自适应星地协同传输方法。

然而,卫星通信网络的规模日益扩大,其异构性、复杂性和高度动态的拓扑结构给网络管理和资源调配带来了难题。此外,由于卫星通信环境的复杂性以及卫星自身计算和存储能力的局限,对卫星网络进行精准的资源配置显得至关重要^[33]。一些研究试图将SDN的架构融入卫星网络,其主要理念是将控制平面和数据平面解耦,SDN控制器能实时获取所有节点和链路的状态信息,形成网络的整体视图,以监控网络的动态变化,然后由控制器集中处理路由计算,并向底层卫星网络发送路由表,使得位于数据平面的卫星依据接收到的指令进行转发,显著减轻了卫星节点的计算负担,同时也缩减了网络中的信号交互成本。近年来,SDN卫星网络架构及其控制层配置引起了国内外学者的广泛关注和深入研究。为了能充分利用SDN技术整合卫星与地面网络的优势,文献[34]、[35]提出了一种软件定义的卫星网络架构,该架构将控制器部署在卫星和地面等不同位置。众多研究都致力于优化卫星网络中的控制器布局,目标是通过精准调控控制器的数量以实现网络性能的最大化,研究方向主要包括最小化整体的端到端延迟^[35];通信可靠性最大化^[36];部署成本最小化^[37]。文献[39]提出了HiQoS方案,利用SDN确保服务质量。该系统由差异服务模块和多路径路由模块组成。差异服务模块依据源节点的IP地址区别不同服务,通过SDN交换机的队列策略确保服务质量;多路径路由模块则动态监测网络状况,寻找符合服

务质量要求的多条路径。然而，该方法在执行多路径算法时依然沿用了最短路径原则，这可能导致在处理大规模网络拓扑时效率不高。文献[40]提出了基于 SDN 的动态多路径流量管理系统，利用 SDN 控制器实时监察交换机状态，并依据带宽指标重新计算最优传输路径。

1.2.2 卫星网络可靠传输研究现状

虽然已有研究关注于在网络状况理想时的卫星路由策略，但对于节点或连接故障等特殊情况考虑较少。考虑到卫星设备的资源限制及空间环境的严峻，卫星链路时常因环境干扰而发生中断^[41]，这使得理想状态下的路由算法的性能大大下降，无法保证传输的可靠性。传统的网络传输通过 TCP 协议作为可靠传输的保障，但由于卫星网络的传播距离通常较长，高传播延迟会使 TCP 一直处于慢启动状态。低轨道卫星达到 1Mb/s 和 10Mb/s 的时间分别需要 0.18s 和 0.35s^[42]。使用卫星网络传输的数据通常文件较小，传统 TCP 协议可能会使整个传输过程可能都处于慢启动状态，无法充分利用带宽。为提升卫星网络效能，有研究致力于优化 TCP 的差错控制和流量控制策略，比如在 TCP 连接建立时先探测路径上的带宽以加速慢启动过程^[43]。为了提高带宽利用率，一些策略着重于逐步增大拥塞窗口尺寸^[44]。针对卫星通信中常见的链路中断问题，研究者们专注于提升 TCP 在恢复阶段的适应性，通过估计链路容量来灵活设定慢启动阈值^[45]。

然而，由于卫星网络存在较高的误码率环境，TCP 协议的可靠传输机制会导致多次重传，重传的数据包在未完全抵达前，已接收的数据是无法被应用层读取的^[46]。因此，当前的地面可靠传输方法无法确保卫星网络的用户服务质量得到保障^[47]。

为减少重传带来的延迟影响，有些研究工作引入了前向纠错编码技术，通过向原始数据编码并加入冗余来对抗数据丢失。然而，许多传统的前向纠错编码算法通常针对特定网络环境设计，无法兼容不同的传输环境，并且大多采用固定码率，不完全适合无线通信需求。喷泉码在此背景下显得尤为适用^[48]，喷泉码是一种无率码，可以根据信道条件调整为合适的码率，因此可以良好地兼容各种传输环境，在接收方完成解码前，发送方会持续生成并发送编码包，这个过程无需反馈，因此能够极大地提升通信效率。

1.3 论文内容安排

本文共分五章，各章内容如下：

第一章为绪论，首先概述研究背景和国内外低轨卫星网络的最新进展，全面评述现有低轨卫星网络的路由策略和可靠传输的研究状况，同时简要说明论文的主要研究内容及章节布局。

第二章探讨卫星网络的关键技术，包括卫星轨道参数、星座构型和星间路由，进一步对经典星间路由算法进行分类讨论，最后针对卫星网络的链路中断问题，本章基于喷泉码对可靠传输技术进行了详细介绍。

第三章提出一种基于 SDN 的低轨卫星多路径路由策略，该策略设计了一种最小传输成本的路由算法，旨在解决链路拥堵和节点失效等问题，并详述了算法的传输成本模型以及网络负载感知和动态拓扑感知的方法。随后介绍了基于 SDN 的双层卫星网络的架构及实现，并在仿真平台上对算法进行了仿真及与其他算法的对比分析。

第四章针对不可预见的卫星网络传输丢包、链路中断、链路不对称等问题，提出了一种基于 TC 码的卫星网络可靠传输方法，该方法利用喷泉码稀疏编码和冗余编码所带来的优势，解决了卫星网络丢包等问题，该章还通过双向搜索动态迭代算法实现了 TC 码编码包大小的自适应，并针对 TC 码设计了优于现有译码算法的 BLBP 译码算法。最后，对算法进行仿真，并在不同的低轨卫星网络传输环境中与其他传输算法进行性能比较和分析。

第五章，总结和展望。总结了本文主要工作，并提出了进一步研究的方向。

2 低轨卫星网络数据传输关键技术概述

本章首先从轨道参数、星座构型和星间链路三个方面简要介绍了卫星星座相关知识，然后简述了低轨卫星网络中路由算法的分类，最后介绍了卫星网络可靠传输技术。

2.1 卫星星座介绍

卫星星座是由一组布局在多个轨道上的多颗卫星组成的网络，同一个星座中的卫星功能相似，用于协同实现特定的通信、导航、地球观测或其他目的。本节分卫星轨道参数、卫星星座构型、卫星星间链路三个方面来介绍。

2.1.1 卫星轨道参数

卫星轨道参数作为确定卫星在太空中运行的具体位置、形态以及方向的关键要素，是卫星导航和定位系统中不可或缺的一环^[49]。如图 2-1 所示，常见的卫星轨道参数共有六个，它们被统称为轨道六根数，具体包括轨道倾角、轨道半长轴、轨道离心率、升交点赤经、近地点幅角和真近点角，这六个轨道参数可以确定一颗卫星的运行轨道：

(1) 轨道倾角 η ：衡量卫星轨道平面与地球赤道平面之间夹角的重要参数，不仅决定了卫星的飞行路径，还影响着卫星的覆盖能力和通信性能。根据轨道倾角的具体数值，可以将卫星轨道分为赤道轨道、极地轨道以及介于两者之间的倾斜轨道等。

a) 赤道轨道：位于赤道轨道的卫星的轨道平面与地球的赤道平面完全重合，轨道倾角为 0° ，如常见的地球静止轨道卫星。它能够持续覆盖地球赤道附近的广阔区域，为这些地区的通信和数据传输提供稳定可靠的服务。

b) 极地轨道：指轨道倾角接近或达到 90° 的卫星轨道，与地球赤道平面几乎垂直，使得卫星能够穿越地球两极，实现全球范围内的覆盖。极地轨道卫星在极地地区以及中高纬度地区具有出色的覆盖性能，适用于气象观测、地球资源探测等需要全球数据收集的应用场景。

c) 倾斜轨道：在卫星轨道的分类中，除了赤道轨道和极地轨道，其余所有轨道均可归类为倾斜轨道，轨道倾角在 0° 至 90° 之间。

(2) 轨道半长轴：椭圆轨道的关键参数之一，其长度直接决定了椭圆轨道的大小。

(3) 轨道偏心率：即椭圆轨道两焦点之间的距离与长轴长度的比值，是描述椭圆轨道形状的重要参数。通过偏心率可以判断出椭圆轨道的扁平程度，进而分析卫星在轨道上的运动特性。

(4) 升交点赤经 Ω ：指的是卫星沿轨道由南向北运动时与赤道的交点，在赤道平面内与春分点对地心的夹角。这一参数主要用于确定卫星轨道平面相对于赤道平面的具体位置。

(5) 近地点幅角 θ ：描述的是近地点在轨道平面内与升交点相对于地心的夹角，可以确定卫星在轨道上的具体位置。

(6) 真近点角 ϕ ：是指卫星从近地点开始，沿轨道运行时扫过的径向角度。这一参数主要用于描述卫星在轨道平面上的相对位置，可用于分析卫星的运动状态和未来位置。

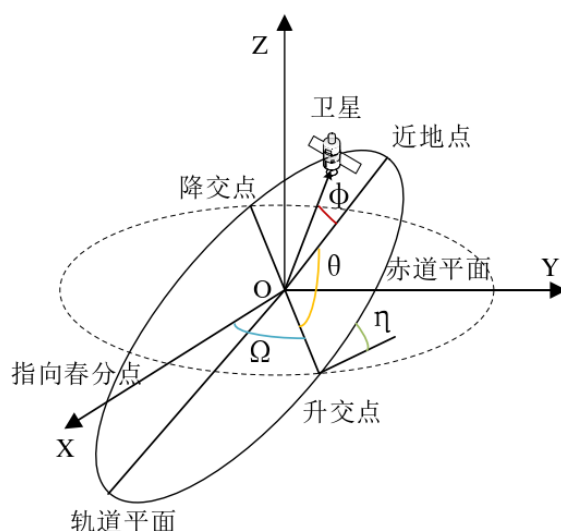


图 2-1 地心坐标系与卫星轨道参数

图 2-1 卫星轨道参数在地心坐标系下的详细示意图，其中，赤道平面为 XY 平面， X 轴指向春分点， Z 轴指向北极点。

2.1.2 卫星星座构型

卫星星座构型是指一组卫星的排列方式，旨在实现特定的通信、导航、地球观测或其他目的^[50]。构建卫星网络时，不同的适用场景下有不同类型的卫星星座构型。下面将介绍两种星座构型：极轨道星座和倾斜轨道星座。

（1）极轨道星座：极轨道星座中的卫星绕地球的北极和南极轨道运行，轨道倾角接近 90 度，每个轨道所含卫星数目相同，使得卫星可以覆盖地球的极地地区以及全球范围。卫星的运行轨道通常是近极轨道或太阳同步轨道，以实现对地球的全面观测和监测。顶视图 2-2 明确展示了其运行特性，在北极点东部，卫星沿由北至南的路径行进，而在北极点西部，它们则遵循南至北的轨迹。这两条靠近北极点的轨道之间形成了一道反向缝。通常，每颗极轨道卫星可与包括轨间与轨内的四颗卫星通过 ISL 连接，反向缝两侧卫星只能与两颗卫星通过 ISL 连接。

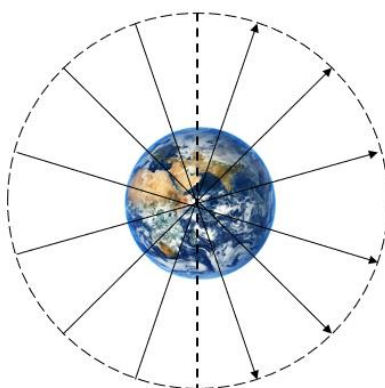


图 2-2 极轨道星座顶视图

（2）倾斜轨道星座：倾斜轨道星座中的卫星的轨道倾角相对于地球赤道倾斜，通常介于 20 度至 60 度之间。这种轨道构型可以提供广泛的地表覆盖范围，并且相对于极轨道而言，具有更低的轨道倾角。倾斜轨道卫星的顶视图如图所示，每颗卫星通常只与同轨道内的两颗卫星通过 ISL 连接，除此之外，两条轨道的交点也常被用于建立 ISL。如图 2-3 所示。

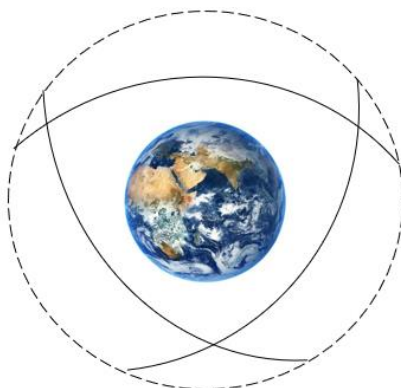


图 2-3 倾斜轨道顶视图

2.1.3 卫星星间链路

由于地形限制，地面站无法建立在海洋、沙漠等区域，只能建立在陆地区域，卫星通信作为地面通信的补充，能够实现全球覆盖，若将卫星视为网络中的节点，ISL 作为节点间的桥梁时，整个星座就构成了一个卫星通信网络，不同节点间可通过不同的路径来实现信息传输，能够为覆盖区域内任何信源和接收器提供服务。采用星间链路技术可以减少对地面站的依赖，提升卫星系统的抗干扰性能，从而提高了通信效率和稳定性。

星间链路作为卫星通信网络的关键组件^[51]，它如同星座中的无形纽带，将卫星节点编织成一个连通的整体网络。当我们将卫星视为网络中的节点，ISL 作为节点间的桥梁时，整个星座就构成了一个卫星通信网络。

星间链路技术的优点在于其能大幅度减少对地面站的依赖，有效降低信息传输的延迟时间，并增强卫星网络的抗干扰能力，进而优化通信效率和系统稳定性。

在 LEO 星座中，根据节点所在位置，可以将星间链路分为轨内星间链路和轨间星间链路两类^[52]。由于轨道面内卫星位置和链路状态相对稳定，因此轨内相邻节点间通常可以建立持续的链路；而轨间星间链路两侧卫星由于相对位置变化较快，且当卫星处于极地和反向缝时，难以建立星间链路。轨内星间链路和轨间星间链路如图 2-4 所示。

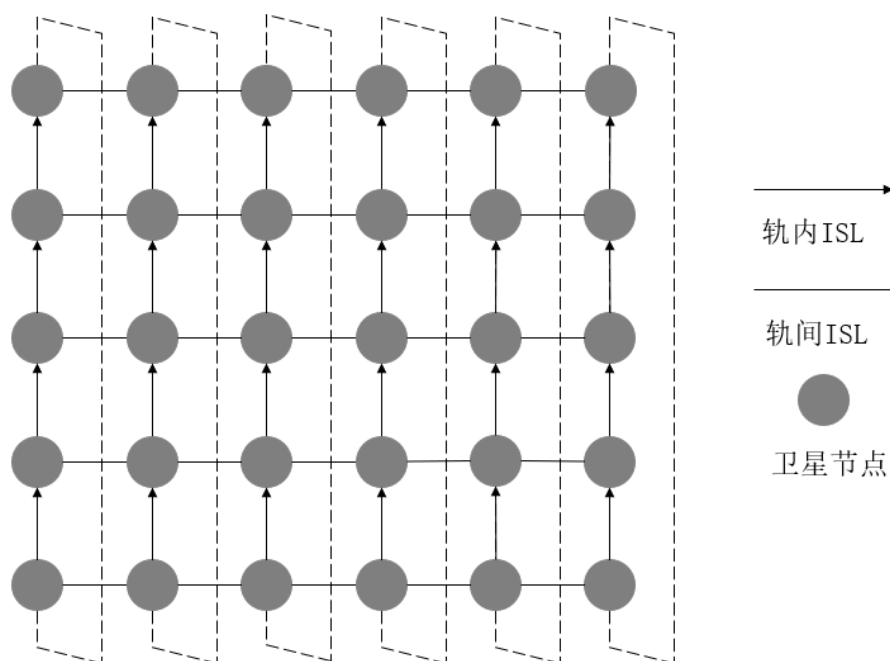


图 2-4 星间链路示意图

2.2 卫星网络路由技术

近年来，星间链路技术的出现使得多颗卫星可以实现相互间的通信。卫星路由算法在搜寻由源节点到目的节点的高效路径的同时，应最大限度地提升系统的资源利用效率与吞吐量，降低网络的阻塞几率，提升系统的可靠度。因此，卫星网络路由技术在卫星网络系统的发展中起着重要作用^[53]。

在当前卫星网络架构中，星间路由算法可按其策略机制可分为三种主要类型：单路径路由、备份多路径路由以及并行多路径路由。

单路径路由策略依据网络状态，如带宽和传输延迟作为路径选择的依据，选取总体权重最低的一条路径传送数据。经典算法包括 DOAR 算法^[54]和 LBRP-MA 算法^[55]，DOAR 算法定期检测网络链接状态，估算队列延迟，周期更新路由表，利用延迟信息作为权重，避开拥堵点。LBRP-MA 算法在 DOAR 基础上作了进一步的改进，通过各节点在全网洪泛移动代理进行链路状态搜集，各节点根据搜集的信息选择最佳下一跳，优化单路径分配。

备份多路径方法通常涉及多个传输路线。在主要路径流量过高或拥堵时，部分流量会被转移到备用路径。典型算法有 ELB^[27]、TLR^[28]等。ELB 通过监控本地队列，一旦超过阈值，即通告上游节点拥堵信息，并切换路径以分散流量。TLR 则同时使用主次路径，基于两条路径的队列和下一跳节点状态，采用交通信号灯模式指示状态。

并行多路径传输算法的思想是从源节点到目标节点，通过不同路径主动发送流量。如首次提出多路径数据传输的概念的 CEMR 算法^[22]，该算法通过周期性获取网络链路状况，找出最优的 k 条路径并发数据。HLBR 算法^[56]构建全局流量预测模型，预先在地面控制中心计算多路径路由，并根据实时局部网络状态灵活调整各路径的数据流量。

尽管上述三种路由算法各有其优势，但它们也各自存在一些局限性。单路径路由算法在可靠性方面存在不足，且在网络流量变化时，其实时调整能力有限，可能导致算法鲁棒性下降。备用多路径策略依赖于主路径达到阈值时才被动分流，可能导致链路利用率不均，甚至引发级联拥塞。而并行多路径路由算法在链路耦合度低或完全不交叠时，可能导致数据包接收端的乱序问题，进而影响网络吞吐量性能。现有算法都需要周期性地收集全网信息，随着卫星网络日渐变得复杂，星上计算和存储能力并不能满足由此产生的高网络开销。

除此之外，上述三种路由算法均未考虑到链路中断的情况，若某一条卫星链路突然中断，发送端等待接收端的反馈以及卫星端重新计算路径并重传将消耗大量时间，如何在恶劣的卫星网络环境中保证传输链路的可靠性问题是一大难点。

2.3 卫星网络可靠传输技术

随着卫星通信技术的不断发展和应用，卫星网络可靠传输算法正日益成为研究和应用的热点之一。传统的卫星通信系统在面对大规模数据传输时，常常受限于信道噪声、带宽限制和传输距离等因素，导致传输效率低下和系统性能受损。传统采用的信道编码技术包括 Turbo 码和 LDPC 码。Turbo 码的编码结构由两个部分编码器串行并联组成，其间以交织器相联，每个组件各自产生校验位。解码过程中，运用

迭代方式在各分量解码器间交换去除了正反馈的外部信息，以其出色的纠错能力和逼近香农限的性能而备受推崇。另一方面，LDPC 码作为前向纠错码，其关键特性在于其校验矩阵的稀疏结构，大多数元素为 0，少数为 1。这种特性确保了解码复杂度和最小码距随着码长的线性增长，同时简化了硬件实现，尤其适用于大规模通信系统。LDPC 码的解码算法基于稀疏矩阵的并行迭代，运算需求较低且错误平层更低，故在要求低误码率的应用中极具优势。然而，定码率的编码方式因编码率恒定，无法灵活适应信道条件的变化，从而在资源利用率和错误纠正效能上存在局限性。而基于喷泉码的传输算法通过其独特的纠错和编码特性，能够在不增加额外传输开销的情况下，实现高效可靠的数据传输。喷泉码通过随机生成的方式，产生无限长度的编码序列，从而为数据传输提供了更大的灵活性和容错能力。本节将探讨基于喷泉码的卫星网络传输算法的原理、编译码算法、以及性能分析。

2.3.1 喷泉码基本概念

喷泉码是 Byers 等人提出的一个数字内容分发概念。这个概念是基于一个类比，一个喷泉喷出的水滴，然后收集到一个桶，这转化为服务器喷射随机生成的数据片段，然后由接收器收集，当收集到足够数量的数据包时，可以对文件进行解码。

喷泉码是无速率的，速率不需要事先固定，编码符号是动态生成的，在编码前，编码器会将源数据等分成多个数据包，再进行编码，将这些编码包不断地发送给接收方，只要接收端正确接收到足够数量的编码包，就可以通过解码获得原始数据。对于好的喷泉码，解码所需的数据包的平均总大小接近文件的原始大小。数字喷泉的一个重要特征是它与接收到的特定数据包无关。一旦接收到一定数量的数据包，消息就有很大概率可以被解码。

LT 码是 Luby 提出的实用喷泉码，从理论上来讲，LT 码无需担心信道质量的好坏，也无需关心在信息传递过程中丢失哪些数据包，只要发送方源源不断地发送编码包，接收方经过一段时间一定能够解码成功，且在译码时无需关心数据包的先后顺序。除此之外，使用 LT 码作为信道编码的方式，也不会有 ARQ 机制导致的高延迟。LT 码具有编译码简单、译码开销低的优点，其计算复杂度为 $O(K \log K)$ ，其中

K 为码长。然而，由于编码的随机性，LT 码译码失败的概率较高^[59]，为了能够覆盖到所有原始信息，避免译码失败，研究人员在 LT 码基础上增加了预编码手段，即 Raptor 码。

Raptor 码可看做一种级联编码方式，源码首先以预编码的方式生成带有一部分冗余信息的中间符号，再将中间符号通过弱化的 LT 编码产生编码符号^[59]。这里的弱化 LT 编码旨在避免高连接度的编码符号生成，方法是尽量降低从中间表示生成编码符号时的度值，以此来减少异或运算的频率。尽管这可能导致解码过程中对中间表示的覆盖率不足，无法完全重构所有中间符号，但未恢复的中间符号仅占一小部分。由于中间符号在生成时就包含了冗余位，因此可以通过预编码的错误校正能力，从这些中间符号中有效地还原原始信息^[60]。其编码过程如图 2-5 所示。

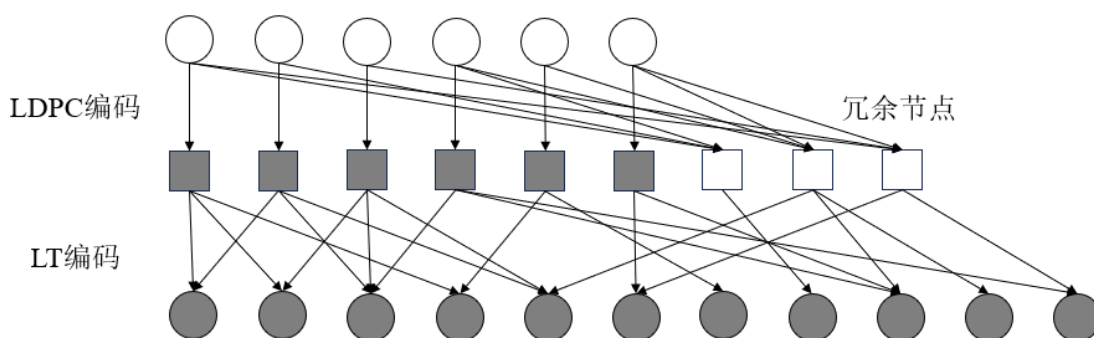


图 2-5 Raptor 码编码过程

Raptor 码以预编码加弱化的 LT 编码形式，加强了喷泉码的纠错能力，减少计算复杂程度，同时实现了线性编解码，其编解码算法复杂度为 $O(K \log \frac{1}{\epsilon})$ ，其中， ϵ 为译码冗余。然而，Raptor 码是高通公司的专利，因此不能被广泛使用^[58]。

2.3.2 喷泉码编译码算法

LT 码的编码过程相对简单。在发送端，将要传输的信息等分成 k 个源数据包， k 即为码长，编码过程演示如图 2-6 所示。

编码流程如下：

- 1) 根据度的概率分布随机确定一个整数 d ;

- 2) 在包含 k 个初始数据包的集合中，随机挑选出 d 个数据包；
- 3) 对选出的这 d 个数据包执行异或操作，由此产生一个新的编码包，这个编码包的度定义为 d ；
- 4) 重复步骤 1-3，不断地发出编码包，直到接收方译码成功。

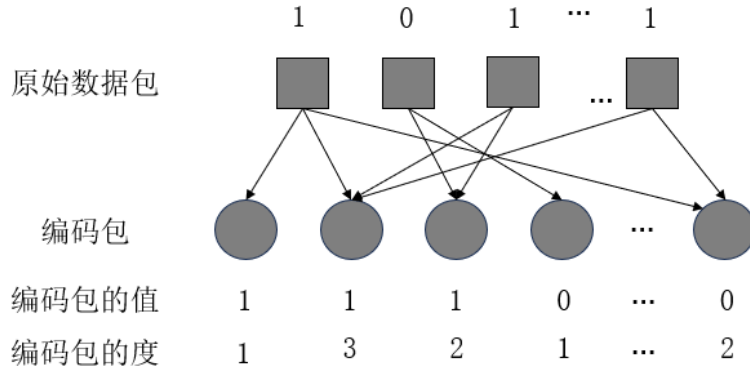


图 2-6 LT 码编码过程

度是指生成一个编码包所需原始数据包的个数，度分布则是指这种度数的分布情况，良好的度分布对于喷泉码的性能至关重要，一个理想的度分布应该能够最大程度地提高解码成功的概率，同时保持编码效率和解码复杂度在可接受的范围内。最具代表性的度分布有以下两种：

- a) 理想孤子分布 (Ideal Soliton Distribution)

$$\rho(i) = \begin{cases} \frac{1}{k} & i = 1 \\ \frac{1}{i(i-1)} & i = 2, 3, \dots, k \end{cases} \quad (2-1)$$

- b) 鲁棒孤子分布 (Robust Soliton Distribution)

$$\mu(i) = \frac{(\rho(i) + \tau(i))}{\beta} \quad i = 1, 2, \dots, k \quad (2-2)$$

$$\text{其中, } \tau(d) = \begin{cases} \frac{R}{d \cdot k} & d = 1, 2, \dots, \frac{k}{R}-1 \\ \frac{R \ln(\frac{R}{\delta})}{k} & d = \frac{k}{R} \\ 0 & d = \frac{k}{R} + 1, \dots, k \end{cases}, \text{ 式中 } R = c \cdot \ln(\frac{k}{\delta}) \cdot \sqrt{k}, \quad c > 0。$$

LT 码最具代表性的译码算法包括 BP 算法和 OFG 算法^[60]。在 BP 算法的解码过程中, 解码器在编码包集合中不断寻找度数为 1 的编码包, 每个解码的数据包都与连接到该数据包的所有编码数据包进行异或运算, 然后将相应的编码包的值替换为新的计算结果, 并删除它们之间的连接。

流程如下: 找到所有度为 1 的编码包, 并将其与与之相邻的数据包作异或运算, 所有相邻数据包的度减 1。重复上述过程, 直到编码包集合中没有度数为 1 的包, 如果所有数据包都恢复正常, 则解码成功。该算法的过程如图 2-7 所示。

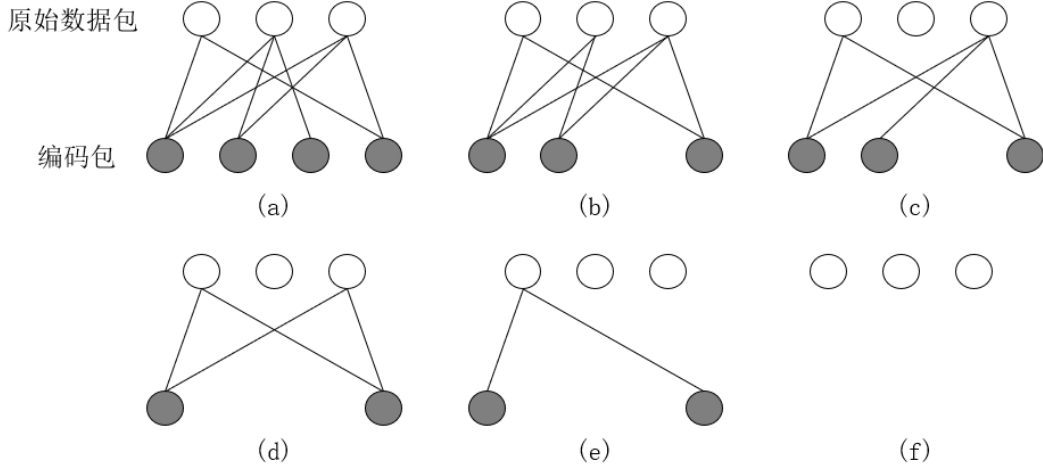


图 2-7 BP 译码过程

可见, BP 译码算法并不能挖掘数据包中的全部信息, 解码成功率的高低取决于度为 1 的数据包的个数, 若个数过多会造成信息冗余, 若个数过少会使解码过程中断。相较而言, OFG 译码算法作为一种改进的高斯消去算法, 能够挖掘数据包的全部信息, 大大提高译码成功率。其译码算法如下:

算法 1.1 OFG 译码算法

输入：编码数据包

输出：原始数据包

1. 将 $k \times k$ 的矩阵 G , k 维向量 Y , NumOnes 中的元素初始化为 0, EmptyRows 初始化为 k
 2. **while** $\text{EmptyRows} \neq 0$ **do**
 3. 接收一个由方程式 NewEq 生成的编码数据包 NewY , 其度数记作 d , NewEq 中第一个为 1 的位置为 s 。
 4. **if** $d = 0$ or $G[s][s] = 0$ **then**
 5. **if** $d > 0$ **then**
 6. $G[s] = \text{NewEq}$, $Y[s] = \text{NewY}$
 7. $\text{NumOnes}[s] = d$
 8. $\text{EmptyRows} = \text{EmptyRows} - 1$
 9. **else**
 10. 删除 NewEq
 11. **end if**
 12. **else**
 13. **if** $d < \text{NumOnes}[s]$ **then**
 14. 交换 NewEq 和 $G[s]$ 、 NewY 和 $Y[s]$
 15. $\text{NumOnes}[s] = d$
 16. 将 NewEq 与 $G[s]$ 进行异或运算, NewY 和 $Y[s]$ 进行异或运算
 17. 更新 NewEq 中第一个 1 的位置 s
 18. **end if**
 19. **end if**
 20. **end while**
 21. 通过行变换将矩阵 G 变为单位矩阵, 最终得到的 Y 就是原始数据包
-

2.3.3 LT 码性能仿真分析

本节在 MATLAB 上进行 LT 码的仿真及性能分析, 度分布采用的是鲁棒孤波分

布， c 和 δ 均取 0.05，译码算法分别采用 BP 算法和高斯消去算法，源数据包设置为 100 个，数据总长度设置为 30000 比特，在不同的误码率下进行了 1000 次实验，实验结果如图所示，图 2-8 是不同误码率下的丢包率曲线。图 2-9 是不同丢包率下的译码成功所需的代价。如图 2-8 所示，误码率越高，丢包率越高，即删除信道的删除概率越高。如图 2-9 所示，在信道条件很差的情况下，LT 码也可以译码成功，信道条件越差，丢包率越高，译码成功所需代价就越高，意味着发送端需要发送更多的数据包。

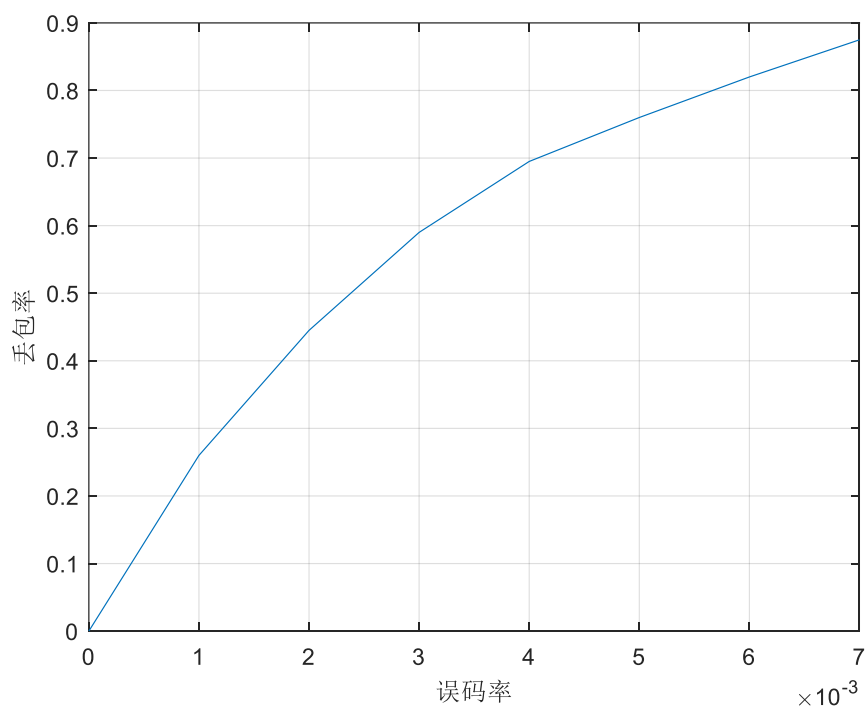


图 2-8 不同误码率下的丢包率

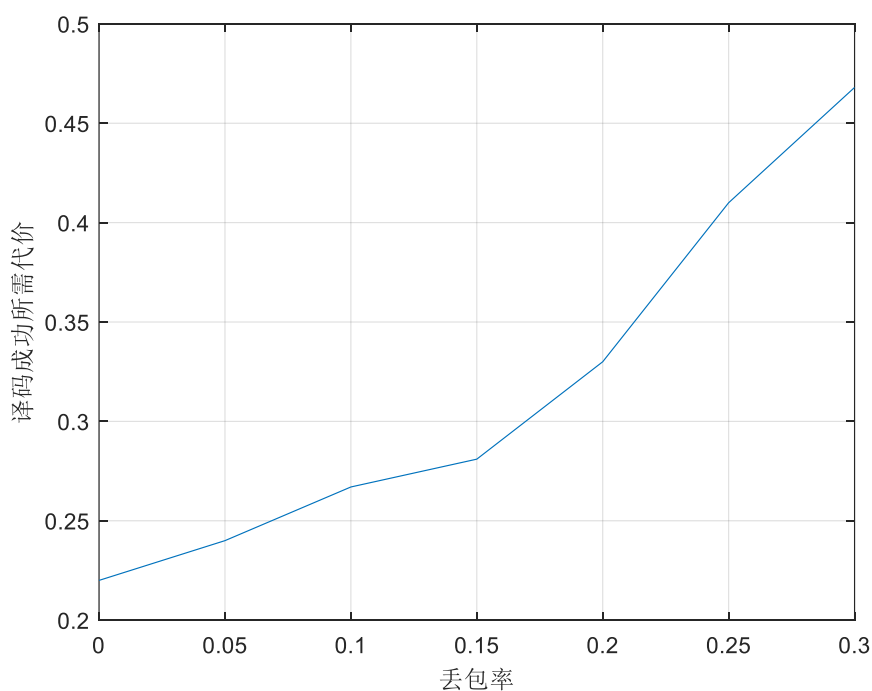


图 2-9 不同丢包率下译码成功的代价

2.4 本章小结

本章阐述的是低轨卫星网络传输技术相关内容，主要从三个方面展开：卫星星座研究、卫星网络路由研究和卫星网络数据传输方法研究。在第一部分的卫星星座研究中，主要介绍了卫星轨道参数、星座构型，以及卫星星间链路。第二部分是对卫星网络路由方法的分析，详述了单一路径路由、备用多路径路由以及并行多路径传输的算法优势与限制。在第三部分卫星网络数据传输方法研究中，重点研究了通过喷泉码来进行可靠数据传输的技术。

3 基于 SDN 的低轨卫星多路径路由策略

3.1 引言

在当今信息时代，数据传输的需求呈指数级增长，传统的单路径路由算法已经无法满足大规模数据传输的需求。因此，多路径路由技术应运而生，成为解决大规模数据传输和通信的重要手段之一。随着低轨卫星网络的发展，庞大的动态网络拓扑和多变的网络环境使得路由计算变得困难。软件定义网络（Software Design Network, SDN）是一种新型网络体系结构，适用于动态特性高的大型网络。现有的融合工作没有考虑如何将它们整合为一个整体网络，缺乏功能可配置的网络管理和卫星与地面网络之间的有效合作，因此本章提出了一种基于 SDN 的最低传输成本多路径路由策略，并设计了一种双层卫星网络架构。该策略综合考虑延迟、丢包率、带宽和节点负荷来建立链路传输成本模型，并通过持续监测网络状态和交换节点的负载，动态优化端到端的路由选择，以提升传输效能。

3.2 算法设计考虑因素

3.2.1 低轨卫星网络拓扑架构

本章建立的模型采用的是极轨道星座^[61]，参数如表 3-1 所示，极轨星座包含 66 颗卫星，6 个轨道面，每个轨道面 11 颗卫星，轨道高度为 780km，卫星倾角为 86.4°。由于每颗卫星最多与 4 颗卫星建立 ISL 链路，因此每颗卫星都配有 4 对收发机，每条链路都采用点对点全双工设计。

表 3-1 低轨卫星星座参数

参数名	参数值
轨道数	6
轨内卫星数量	11
卫星总数	66
卫星倾角/°	86.4

轨道高度/km	780
---------	-----

以下是星间链路建立的准则：

(1) 由于在反向缝的两侧的两颗卫星运动方向相反，相对运动速度高，因此不设立星间链路。

(2) 通常，处于同一轨道上的相邻卫星的物理位置相对固定，能够保持星间链路的持续连接。

(3) 高纬度地区的卫星会中断所有星间链路；当卫星离开高纬度地带并进入中低纬度地带时，会重新启动与相邻轨道卫星的星间链路。

3.2.2 算法设计需求

在卫星网络中，如果仅仅使用最短路径策略进行数据传输，可能会造成流量在返程至地面站时发生拥堵，进而影响网络的整体吞吐量。另外，由于卫星网络中的流量始终处于无法预知的动态变化状态，需要能够基于实时的网络状态信息，动态地调整路由。因此，设计多径路由算法时需要考虑到以下因素：

(1) 进行路由计算时，不仅要考虑到最短路径这个目标，同时需要考虑到实时带宽、节点负载对链路传输能力的影响，确保路由具有最低传输成本和最高传输效率。

(2) 提高整个卫星传输系统的鲁棒性，尽量避免规划的路径出现拥塞或者意外。

(3) 当星间流量处于动态变化状态时，路由算法需要能察觉网络状态的变化，依据流量及链路状况的实时变动，自主调整路由策略或平衡负载，以提升网络的动态适应性。

3.3 基于 SDN 的低轨卫星多路由策略

3.3.1 传输代价模型

为精确量化卫星网络的传输能力，本节在文献[29]的基础上，建立了一套链路传输成本的数学模型。

$$C_{mn}^{l_{mn}} = d_{mn} \times (1 + \zeta_{mn}) \times (1 + \frac{1}{1 + \frac{b_{mn}}{B_r}}) \times (1 + \max\{\frac{1}{Q_m \times (1 - load_m)}, \frac{1}{Q_n \times (1 - load_n)}\}) \quad (3-1)$$

在式(3-1)中， b_{mn} 为实时可用带宽， $1 + \frac{b_{mn}}{B_r}$ 表示实时带宽对传输成本 $C_{mn}^{l_{mn}}$ 的影响，

实时带宽可通过 SDN 南向 API 获得。 B_r 为默认带宽； b_{mn} 越接近 B_r ，链路传输成本越低。 ζ_{mn} 为链路丢包率，链路丢包率越高，传输代价越高，链路本选择概率越低。

$1 + \max\{\frac{1}{Q_m \times (1 - load_m)}, \frac{1}{Q_n \times (1 - load_n)}\}$ 表示卫星节点负荷对传输成本的影响，

其中 $load_m$ 和 $load_n$ 为节点 m 和 n 的实时负荷。

d_{mn} 表示相邻两个卫星节点 m 和 n 之间的延迟参数。如图所示，以同一轨道平面的相邻两颗卫星为例，利用 STK 仿真工具获取卫星轨道参数，然后用 MATLAB 分析得到链路距离变化图，如图 3.1 所示。

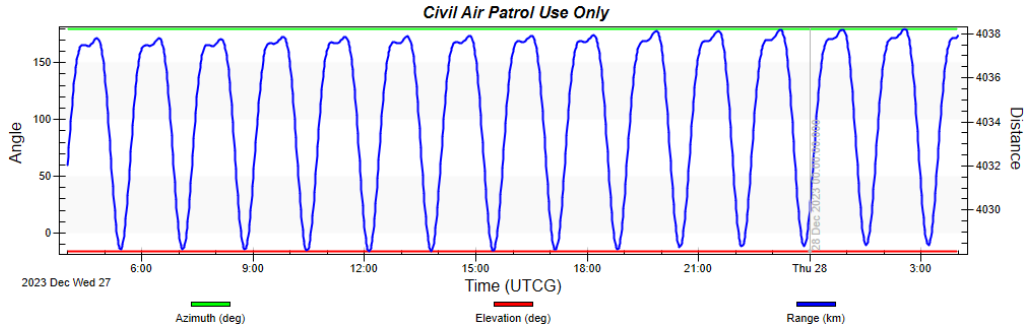


图 3-1 距离变化图

图 3-1 为铱星星座两个相邻卫星的间距变化示意图，模拟时间为一天。不难看出，星间距离变化具有周期性特征，变化周期约为 1.67 个小时，每隔十秒钟获取一次星间距离变化数据，结合无线信号的传播速度，相应的延迟变化方程为：

$$d_{mn} = \frac{Length}{v} \quad (3-2)$$

其中 $Length$ 是距离， v 是光速。

在卫星通信环境中，链路的延迟主要由传播延迟构成，这种延迟直接受到两链

路间距离的影响。依据式(3-2)，我们可以计算出传播延迟的具体数值。

3.3.2 网络负载感知

本小节将阐述如何获取式(3-1)中的实时带宽 b_{mn} 和节点负载 $load_m$ 、 $load_n$ 。交换机维护着各类统计信息，如流表、流条目、端口状态和队列详情，控制单元通过收集交换机的统计资料，利用内置的 API 接口，向交换设备发起 Port-Desc-Stats-Request 请求，并监听 Port-Desc-Stats-Reply 响应，以此来获取卫星链路的带宽 b_{mn} 。

节点的负载可被描述为：

$$load_m = \frac{q_m}{Q_m} \quad (3-3)$$

其中， q_m 代表了当前时间点交换节点需处理的数据包总量，而 Q_m 则代表交换机的最大处理能力，该值与设备的缓冲区容量密切关联。 q_m 越大，交换机越忙；否则，交换机处于空闲状态。 Q_m 越大，表示该交换机有更多的资源来存储数据包，反映了该交换机的处理能力。当 $q_m = Q_m$ 时，如果收到另一个报文，则丢弃该报文，在传输过程中表现为丢包。因此， $load_m$ 不仅代表交换机的负载，还反映了其处理能力，是一个综合性指标。

3.3.3 动态拓扑感知

鉴于卫星网络的周期变化的特性，我们可以预见下一时间片的网络拓扑结构。本小节借助 STK 导出的卫星轨道参数，运用 MATLAB 进行进一步的分析，生成卫星节点随时间变化的距离矩阵序列 L ，其中 t 时刻的矩阵 L_t 表现为：

$$L_t = \begin{bmatrix} 0 & L'_{(1,2)} & L'_{(1,3)} & \dots & L'_{(1,k)} \\ L'_{(2,1)} & 0 & L'_{(2,3)} & \dots & L'_{(2,k)} \\ L'_{(3,1)} & L'_{(3,2)} & 0 & \dots & L'_{(3,k)} \\ \dots & \dots & \dots & \dots & \dots \\ L'_{(k,1)} & L'_{(k,2)} & L'_{(k,3)} & \dots & 0 \end{bmatrix} \quad (3-4)$$

在时间点 t ，两节点间的距离被定义为 $L'_{(i,j)}$ ，若 $L'_{(i,j)}$ 为 0，则表示节点 i 与节点 j 在时间点 t 无连接关系，若 $L'_{(i,j)}$ 为非零值，该值就表示节点 i 与节点 j 在时刻 t 的间

距。

考虑到卫星的高度，每个节点通常能看见多个邻近节点，尽管可见，但这并不意味着存在有效的通信链路。通信链路通常只在一定范围内可用。为此，可以使用距离阈值 L_M 对矩阵 L 进行限制，从而得到 t 时刻的可见性矩阵 $A^t = L^t \leq L_M$ ，即：

$$A^t = \begin{bmatrix} 0 & A^t_{(1,2)} & A^t_{(1,3)} & \dots & A^t_{(1,k)} \\ A^t_{(2,1)} & 0 & L^t_{(2,3)} & \dots & A^t_{(2,k)} \\ A^t_{(3,1)} & A^t_{(3,2)} & 0 & \dots & A^t_{(3,k)} \\ \dots & \dots & \dots & \dots & \dots \\ A^t_{(k,1)} & A^t_{(k,2)} & A^t_{(k,3)} & \dots & 0 \end{bmatrix} \quad (3-5)$$

当 $A^t_{(i,j)}$ 为 1 时，表示节点 i 与节点 j 之间存在连接，否则表示没有连接。通过对每一时刻的可见性矩阵进行排序，形成可见性矩阵序列，可以反映网络拓扑结构的动态变化。控制器通过比较下一时刻与当前时刻的可见度矩阵差来决定是否更新流表。

3.3.4 多径路由算法

本节利用 3.3.2 节、3.3.3 的方法，获得相邻节点 i 和 j 的时延 d_{ij} 、实时带宽 b_{ij} 、负载 $load_i$ 和 $load_j$ ，并基于 3.3.1 节的传输代价模型设计了一种动态路由算法，称为 MTMR (Minimum Transmission-cost Multipath Routing Algorithm, MTMR)，一种三约束的最小传输代价多径路由算法。

此算法的核心思想是找出卫星间总传输成本最低的一组路径。路径的传输成本为路径上每一跳的传输成本的总和。首先，计算任意两点间链路的成本，构建一个传输成本矩阵，其中，节点和链路的状态信息由控制器获得。传输成本矩阵的构建过程如算法 3.1 所述。

算法 3.1 传输代价矩阵的构建

输入： 节点 i 、 j ，节点延迟 d_{ij} ，丢包率 ζ_{mn} ，实时带宽 b_{ij} ，负载 $load_i$ ，
 $load_j$

输出： 卫星网络传输代价矩阵 C

-
1. 卫星节点总数为 n
 2. **for** $i = 0$ to $n-1$ **do**
 3. **for** $j = 0$ to $n-1$ **do**
 4. 计算节点 i 、 j 之间的传输代价
 5. $C_{mn}^{l_{mn}} = d_{mn} \times (1 + \zeta_{mn}) \times (1 + \frac{1}{1 + \frac{b_{mn}}{B_r}}) \times (1 + \max\{\frac{1}{Q_m \times (1 - load_m)}, \frac{1}{Q_n \times (1 - load_n)}\})$
 6. **end for**
 7. **end for**
 8. **return** C
-

设 P_i 为第 i 条路径， $C_{mn}^{l_{mn}}$ 为链路 l_{mn} 的传输成本，则第 i 条路径的传输成本可表示为：

$$C^{P_i} = \sum_{\forall l_{mn}} C_{mn}^{l_{mn}} \quad (3-6)$$

完成传输成本矩阵的获取后，我们可进一步计算得到最低传输代价的路径集合。详细步骤如算法 3.2 所示。算法中使用的堆栈和图等数据结构以及相关方法如下所示：

- **pop(stack)**: 将栈顶元素弹出并以此作为返回值。
- **push(stack, e)**: 将元素 e 加入到栈顶。
- **adjacentNodes(node, G)**: 在无向图 G 中搜索节点 G 的所有相邻节点，并将其作为返回值。
- **nodes(path)**: 返回路径的所有结点。
- **extendPath(path, node)**: 将节点 $node$ 与路径 $path$ 连接起来，从而形成新路径，并将其作为返回值。

算法 3.2 最小传输代价算法

输入：源节点 u ，目的节点 v ，传输代价矩阵 C

输出：最小传输代价路由

-
1. **if** $u == v$ **then**
-

```
2.   return  $\emptyset$ 
3. end if
4.   paths = stack =  $\emptyset$ 
5.   构造只包含一个节点  $u$  的路径  $path_u$ 
6.   while stack  $\neq \emptyset$  do
7.     (node, path) = pop(stack)
8.     for next in adjacentNodes(node, G) - nodes(path) do
9.       newPath = extendPath(path, next)
10.      if next is  $v$  then
11.        paths = paths  $\cup$  newPath
12.      else
13.        push(stack, (next, newPath))
14.      end if
15.    end for
16.  end while
17.  for  $p_i$  in paths do
18.     $C^{P_i} = \sum_{l_{mn} \in P_i} C^{l_{mn}}$ 
19.  end for
20. 按照  $C^{P_i}$  值对路径进行排序
21. 找到前  $k$  条最优路径，组成新的路径组 newPaths
22. for each  $p_i$  in newPaths do
23.    $weight_{p_i} = 1 - \frac{C^{p_i}}{\sum_{t=1}^k C^{p_t}}$ 
24. end for
```

在卫星网络中，对于源节点 u 和目的节点 v ，存在多条路径。该算法首先找到两点之间所有路径集，然后利用传输代价矩阵计算每条路径 P_i 的传输代价。最后，对 C^{P_i} 进行排序，然后从新的有序集中选择传输代价最小的 k 条路径，形成新的路径集。

硕士学位论文

实现多路径路由需要以下步骤：首先，计算出可用路径集中每条路径的权值：

$$weight_{P_i} = 1 - \frac{C_{P_i}}{\sum_{i=1}^k C_{P_i}} \quad (3-7)$$

其中 P_i 为可用路径集中的一条路径， $weight_{P_i}$ 为对应的权值。最后，所有数据包将按照权重比例进行多路径调度。

每组数据包的调度过程如图 3-2 所示。

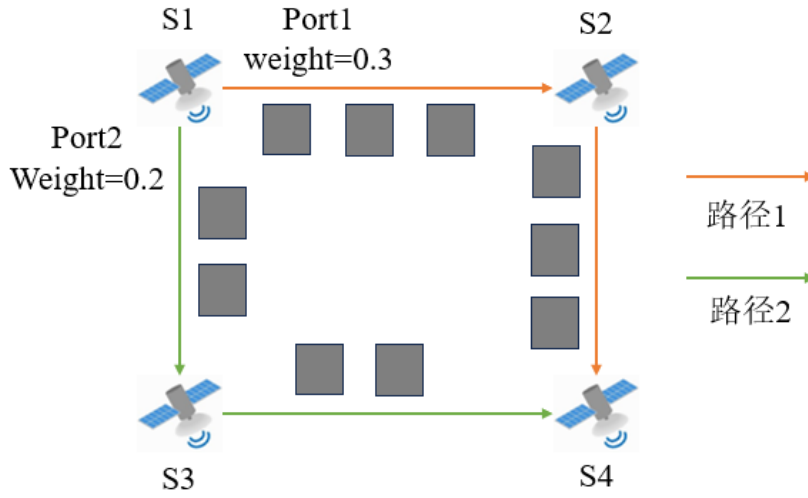


图 3-2 模拟传输场景

当数据包到达时，首先根据流条目中的匹配字段和优先级对数据包进行匹配；如果找到匹配项，则执行与该条目对应的指令。指令有多种动作，其中一种是分组，其功能是将数据包传输到分组表中，以供后续处理。交换机会根据组动作中给出的组 id 在组表中分组条目，然后根据组条目中的组类型值决定如何执行动作桶。操作桶可以理解为与组条目相关联的操作集。如果组类型为 all，则必须执行该动作集中的所有动作。如果选择组类型，则从动作集中选择一个动作，按照一定的规则执行。以图 3-2 场景为例，从卫星节点 S1 到 S2 有两条路径：路径 1（S1→S2→S4）和路径 2（S1→S3→S4）。为了支持向 S2 的多路径传输，可以将目的地址为 S2 的流表项与 S1 上选择类型的组表项关联，并在组表项中设置两个动作，分别对应端口 Port1 和 Port2。同时，每个动作都与一个计数器相关联，记录该动作处理的数据包数量。假设 Path1 的传输代价为 2，Path2 的传输代价为 3，根据式(3-7)，Port1 和 Port2 端

口的权值分别为 6 和 4。我们可以构造动作选择规则如下：对于每个数据包，选择权值最大且计数器值小于该权值的动作执行，当计数器和为 10 时，将每个计数器的值清零。这样，对于每 10 个数据包，前 6 个数据包通过 Path1 传输，后 4 个数据包通过 Path2 传输，从而根据权重调度两条路径。

3.4 基于 SDN 的双层卫星网络架构及实现

3.4.1 基于 SDN 的双层卫星网络架构

传统网络采用分布式控制，控制平面和数据平面紧密耦合。然而，对于当前网络复杂性和可扩展性的要求，传统网络逐渐暴露出架构上的缺陷，包括三个方面：封闭网络部署管理困难、流量控制要求网络运维成本高和分布式架构限制了网络配置和网络创新。

SDN 思想的核心是将网络的控制平面和数据平面分离，通过集中控制使网络具有开放性和可编程性，用一个中心控制器来管理网络中的所有设备，控制器可以获取全局网络拓拓扑信息，并制定路由策略和流量转发规则，并将这些信息发送给数据平面设备，数据平面负责实际的数据包转发。

SDN 架构依据控制器的配置，可分为三大类别：星上控制布局、地面控制布局以及星地协同控制布局^[57]。

星上控制平面通常设置于 GEO 卫星内^[63]，得益于其广阔的覆盖范围和出色的广播特性，只需三颗 GEO 卫星就能覆盖全球，实现全球通信。LEO 卫星能够向 GEO 卫星报告其自身状态信息，使得 GEO 可以获取网络状态和节点拥塞情况，GEO 卫星之间也能交互能形成全局网络状态视图，根据制定好的路由策略计算路由并告知 LEO 卫星，但由于星载处理器的存储能力和计算能力有限，面对 LEO 卫星告诉变化的拓扑，可能难以实时计算，且 GEO 卫星的维护成本较高，因此仅依赖星上的控制部署并不具备显著优势^[63]。

另一种选择是在地面网络中设立控制平面^[65]，地面网络的优势在于具有强大的存储和计算能力，然而，由于地理条件的限制，仅仅靠地面网络难以建立通信卫星的全局视图，且由于低轨卫星的高速移动，无法将流表控制信息实时传递给卫星。

星地联合控制平面部署是一种多层级策略，这种策略结合了 GEO 卫星覆盖范围广和地面存储计算能力强的优势，可以实现更全面、更高效的网络管理与优化，从而提升网络的可靠性和稳定性^[66]。

表 3-2 控制器部署方案对比^[67]

放置位置	层数	覆盖范围	计算速度	网络时延	网络动态性
地面	单层	部分	快	低	低
LEO	单层	全球	慢	较低	高
GEO	单层	全球	慢	较高	较低
地面+GEO	多层	全球	较快	较高	较低

经过表 3-2 的对比，本章节的控制器部署决定采取地面站与 GEO 卫星的混合配置策略，利用地基网络和三个 GEO 卫星构建 SDN 的控制层面，以达成网络的集中化管理，同时让 LEO 卫星星座承担数据传输的角色。如图 3-3 所示。

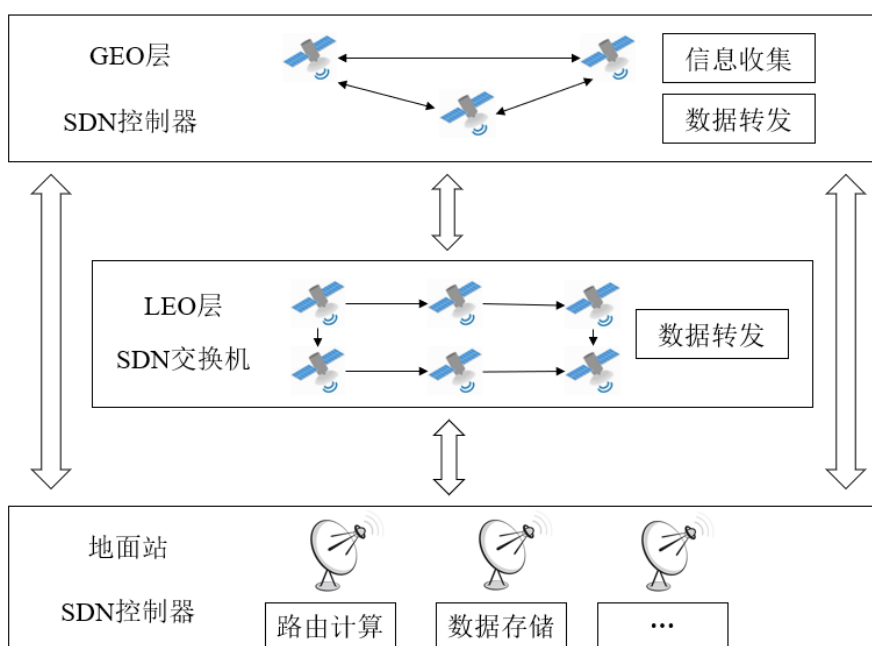


图 3-3 基于 SDN 的双层卫星网络架构

三个 GEO 卫星组成的从控制器负责监测各层网络中各节点的运行情况，并及时向地面观测站传递相关的信息。地面接收站是 SDN 的主控设备，它承担着数据的储存、路由等重要任务，它需要对低轨卫星的运行状况进行汇总，并根据路由算法计

算卫星路由。处于中间层的 LEO 卫星则遵循 OpenFlow 协议，作为数据转发平面，它们被用作交换机，依据 GEO 卫星下发的流表规则，快速高效地传输用户数据包。

数据传输流程如下图 3-4 所示：

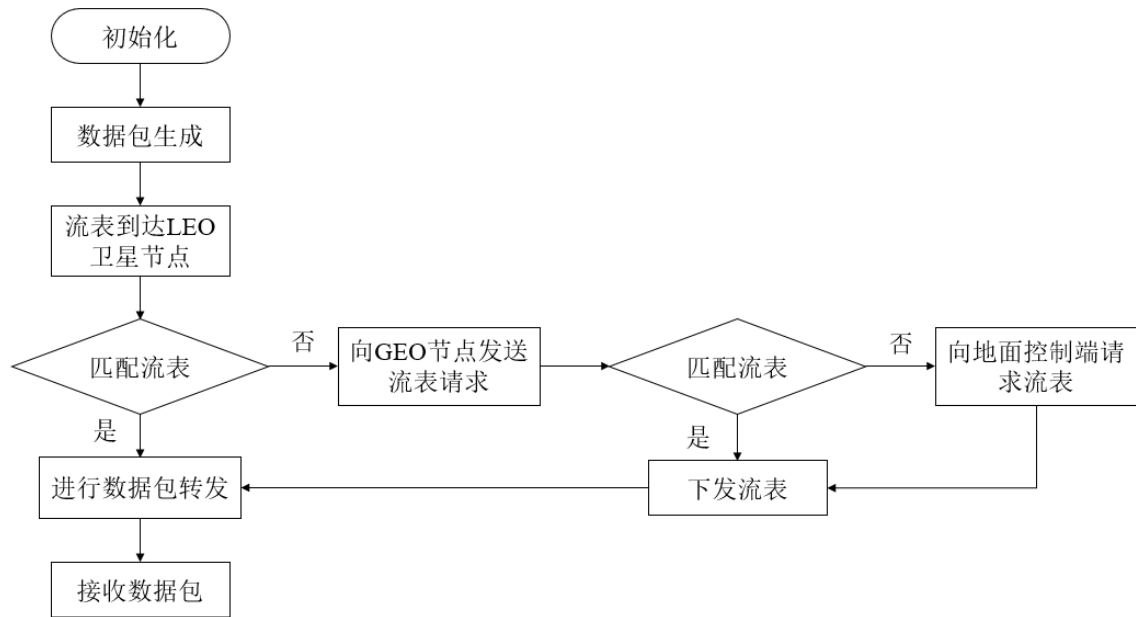


图 3-4 数据包传输过程

（1）初始化。GEO 卫星与其覆盖范围内的 LEO 卫星进行信息交互，LEO 卫星告知其所在的链路状况信息等。每颗 GEO 卫星采集其所在区域的低轨卫星的相关数据，构成局部的组网视图并报告给地面控制站，地面控制站实现对全局卫星节点及链路的实时监控。

（2）地面用户节点生成仿真数据包，依据信号强弱，将数据包发送给当前所在区域内距离最近的 LEO 卫星。

（3）LEO 卫星收到数据包后，会在缓存流表中查找是否有符合的转发规则，如果找到，那么就根据该流表规则转发数据包给下一个卫星节点，如果未找到，就将分组放置缓存中，向 GEO 卫星发出流表请求。

（4）当 GEO 卫星接收到 LEO 节点发出流表请求后，GEO 节点会启动流表匹配机制，一旦匹配成功，即根据流表执行相应的操作流程；若匹配不成功，则向地面控制中心发出路由请求。

硕士学位论文

(5) 地面控制中心接收到 GEO 卫星的请求后, 会解析请求内容, 包括数据包类型、源地址和目标地址等, 然后利用全球网络拓扑信息, 依据预设的路由策略生成新的路由表, 并将之传达给 GEO 卫星。接着, GEO 卫星群将更新的流表分发给 LEO 卫星, LEO 卫星则依据流表中的转发路径进行转发。

表 3-3 卫星节点参数^[61]

参数	LEO 卫星	GEO 卫星
轨道高度/km	780	35786
卫星轨道数	6	1
每条轨道上的卫星数	11	3
卫星总数	66	3
轨道倾角/°	86.4	0

在 STK 中，我们根据表 3-3 的参数模拟了卫星节点的仿真模型，确保 3 颗 GEO 卫星能全面覆盖 LEO 层。2D Graphics 如图 3-5 所示，将卫星轨道文件导入 OPENT 仿真软件后生成的卫星节点如图 3-6 所示。

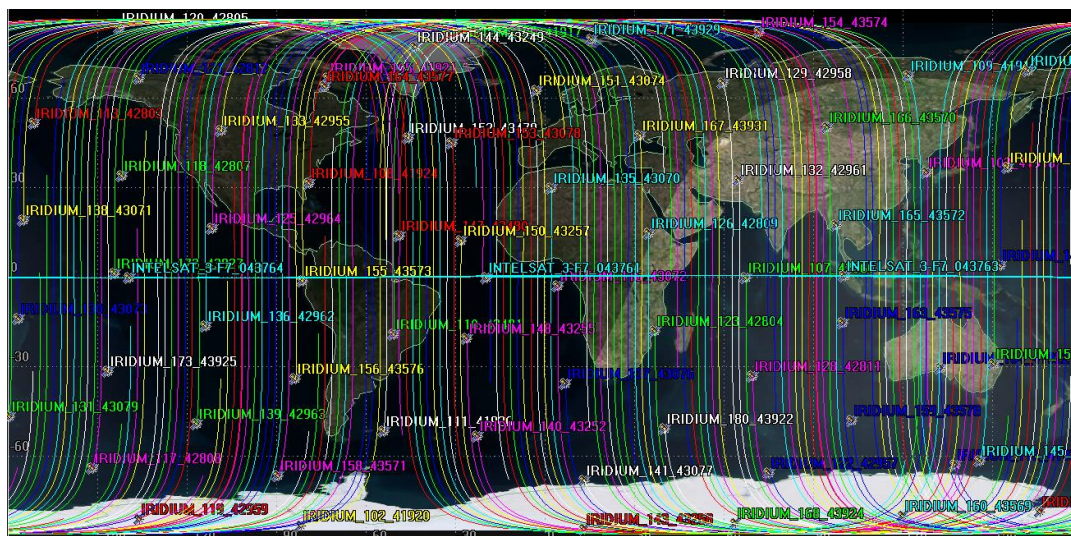


图 3-5 2D Graphics 仿真简图

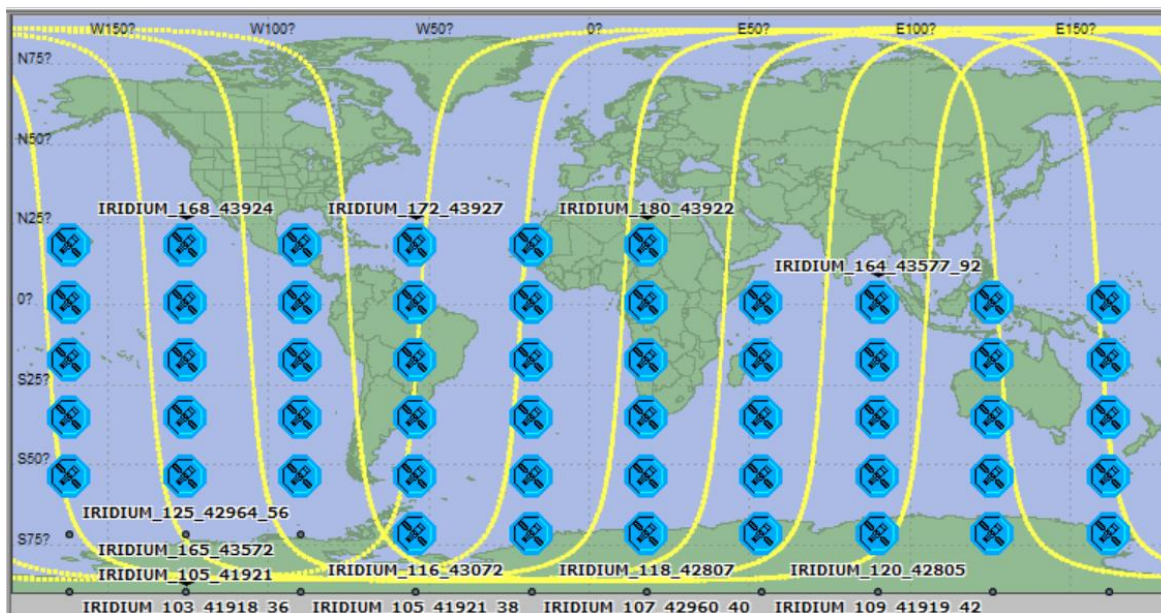


图 3-6 OPNET 卫星节点仿真

3.4.2 OPNET 节点仿真实现

(1) 用户节点

用户节点用于管理数据包的生成、接收过程，因此会包含数据包生成模块和数据包处理模块。当用户节点被多个 LEO 卫星覆盖时，用户节点会根据信号质量动态选择最适合的卫星进行连接，因此需要一组与 LEO 卫星交互地收发信机以及对应的天线，具体结构可参见图 3-7。

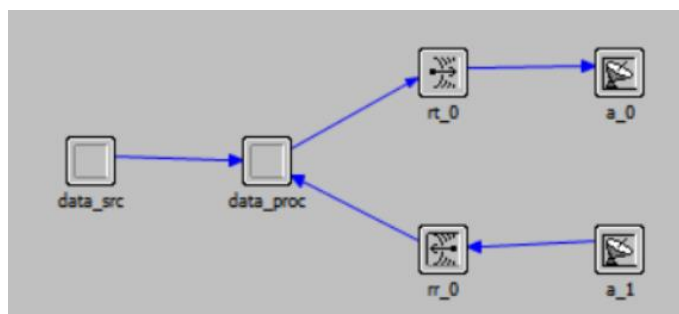


图 3-7 用户节点

用户节点包含两个模块：

a) 数据包生成模块 (data_src)：这个模块负责生成随机数据包。如图 3-8 所示为该模块的进程模型。

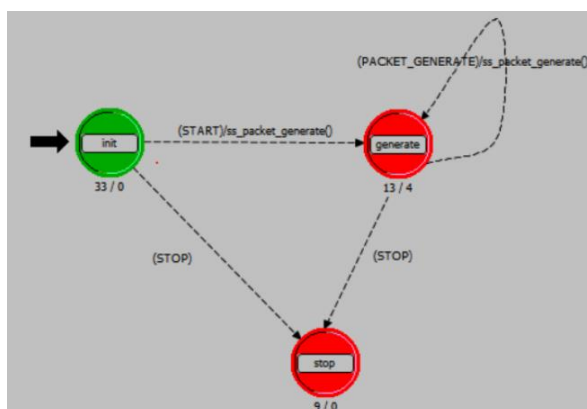


图 3-8 data_src 进程层建模

首先，init 状态负责完成初始化，获取该用户节点的标识，设置数据包生成的起止时间、发送速率和间隔等参数，generate 生成模式会根据起始时间持续创建数据包，并通过 ss_packet_generate()函数随机产生目标用户，到达终止时间后进入 stop 状态。在特定情况下，模块会从 generate 状态转至 stop 状态。

b) 数据处理模块 (data_proc)：该模块用于处理数据包，并管理用户节点的卫星连接。图 3-9 是该模块的进程层建模。

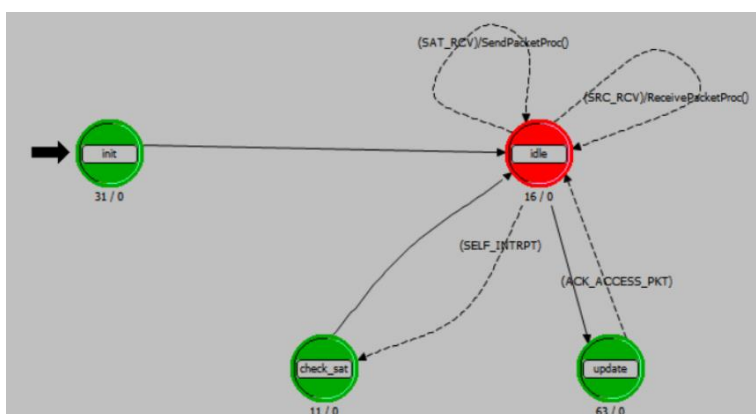


图 3-9 data_proc 模块

其中，init 状态获取用户节点的信道 ID，当发送数据包或者接收数据包时就进入空闲等待状态，该状态会统计丢包率等数据，等待状态会定时进入更新状态，该状态会在用户处于多个 LEO 卫星重叠区域时，根据信号强度优化卫星接入策略。idle 状态会定期进入 check_sat 状态，用户节点向 LEO 卫星发送探测消息，以维护当前卫星连接。

(2) LEO 卫星交换机节点

LEO 卫星节点在 SDN 架构中扮演着负责数据包转发的路由器功能。LEO 节点与地面用户节点直接相连，地面用户向 LEO 卫星发送数据包，因此会包含一组用于与地面用户节点交互的无线收发设备；接收到数据包后会根据缓存的流表进行匹配，若匹配成功则直接向下一跳卫星转发数据包，若匹配失败则向 GEO 卫星发送流表请求信息，因此，LEO 卫星模块也包含 4 组用于与其他 LEO 卫星通讯的无线收发设备。除此之外，LEO 卫星节点还需定期告知 GEO 卫星控制器自身的状态信息，如带宽、节点负载等。因此，还需要一组与 GEO 卫星节点沟通的无线收发设备，以及一个用于网络拓扑定期更新的模块，具体结构可参考图 3-10。

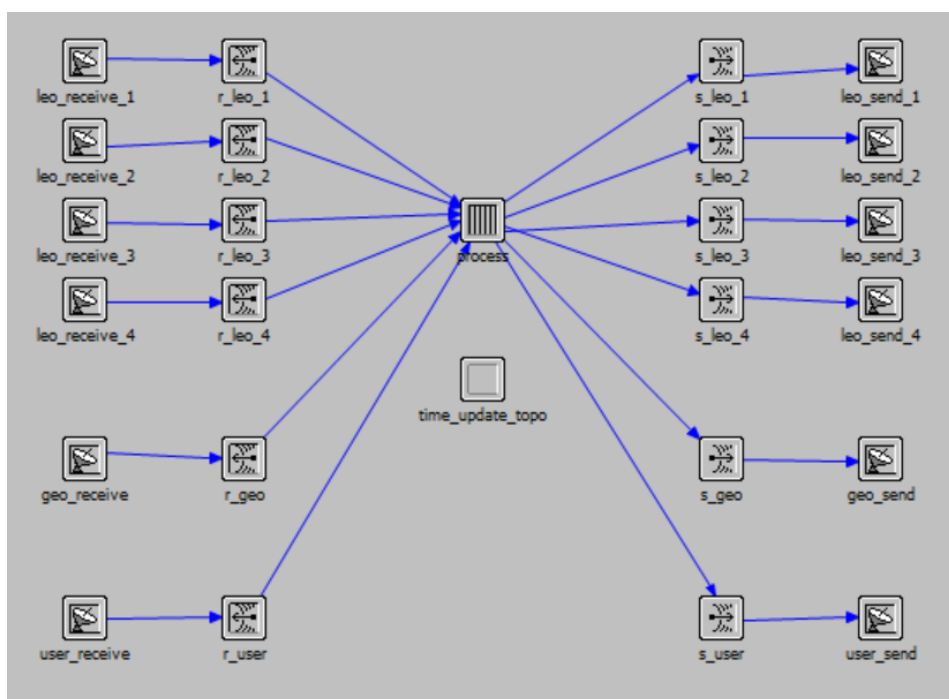


图 3-10 LEO 卫星交换机节点模块图

a) process 模块：该模块主要执行接收地面数据包的任务，依据流表进行操作，并且负责与其他 LEO 卫星建立连接，实现数据包的转发。同时，它也与 GEO 卫星保持通信，报告自身状态，接受流表更新等指令。该模块的进程图示例如图 3-11 所示。

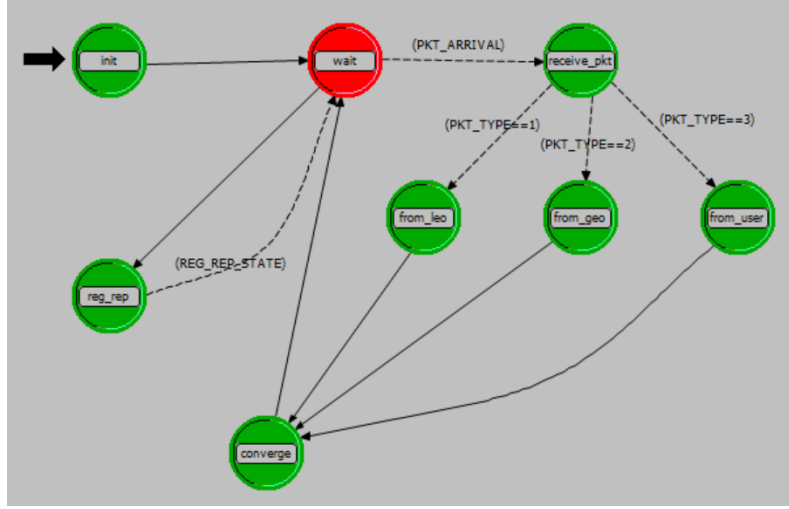


图 3-11 process 模块进程图

在 `init` 阶段，执行初始化相关参数变量，然后进入到 `wait` 等待状态。一旦卫星节点接收到消息，`receive_pkt` 状态会根据标识字段识别消息来源：若源于地面用户，则进入 `from_user` 状态，来自用户的数据包有两种类型，第一种是探测数据包，LEO 卫星配合作出响应即可，第二种是包含信息的数据包，那么 LEO 卫星会解析数据包的目的节点 ID，若缓存内有符合的流表信息则直接转发，若没有，就向 GEO 卫星发送流表请求并等待控制指令的下发；若源于其他 LEO 卫星，`from_leo` 状态会判断目标节点 ID 是否与本节点匹配，若匹配则接收，若不匹配则继续转发；若消息来自 GEO 卫星，则进入 `from_geo` 状态，来自 GEO 卫星的数据包也有两种类型，第一种是探测数据包，LEO 卫星向 GEO 反映自身带宽、节点负载等信息即可，第二种是 GEO 卫星下发的流表控制指令，LEO 卫星接收后更新自身流表。`converge` 为状态汇总节点。当本地节点的链路状态变化或即将出现拥塞时，会进入 `reg_rep` 状态下，该状态会通知 GEO 卫星节点进行相应的标记。

b) `time_uptopo` 模块：该模块主要负责 LEO 卫星网络拓扑结构的初始化和定期更新，其工作流程图如图 3-12 所示。

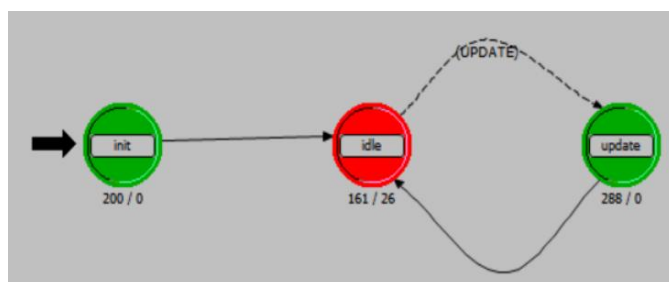


图 3-12 time_uptopo 模块

在 init 状态下，基于 STK 导入的链路信息设置参数，构建起卫星间的初步连接。随着时间的推移，update 状态会定期刷新卫星间的链路状态，确保网络连通性。

(3) GEO 卫星节点

GEO 卫星节点为从控制器，其任务包含搜集所覆盖区域内 LEO 卫星节点的信息，并把这些信息传达给地面站主控制器；同时，它们还要将地面控制中心制定的流表和转发策略广播至 LEO 卫星网络，因此，GEO 节点的进程层模型中会包含与 LEO 卫星和地面站交互通信的收发信机。GEO 节点也能充当中转站，将流表转送至其他 GEO 节点。GEO 卫星节点模块建模如图 3-13 所示。

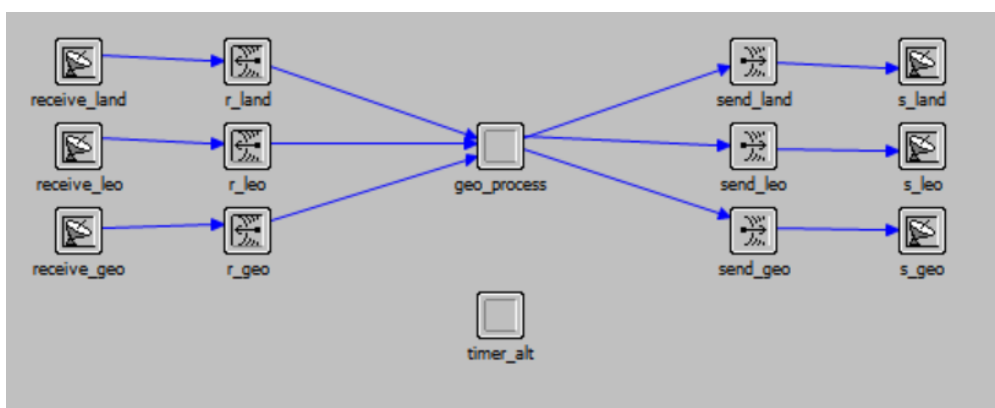


图 3-13 GEO 卫星节点

a) timer_alt 模块：该模块定期从 LEO 卫星获取实时状态更新，其中 init 为初始化阶段，state_up 为数据收集就绪状态，idle 为空闲状态，如图 3-14 所示。

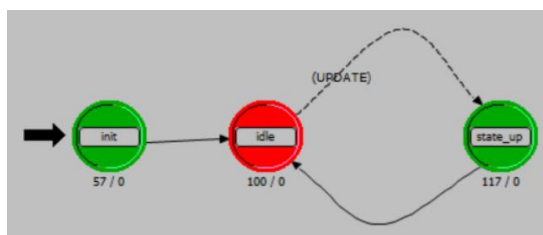


图 3-14 timer_alt 进程模型

b) geo_pro 模块：该模块作为 GEO 卫星节点的核心，有以下两个功能：一是将 LEO 卫星节点的状态信息传输至地面控制中心，二是接收并执行来自地面中心的指令，同时向 LEO 卫星发送流表。进程图如图 3-15 所示。

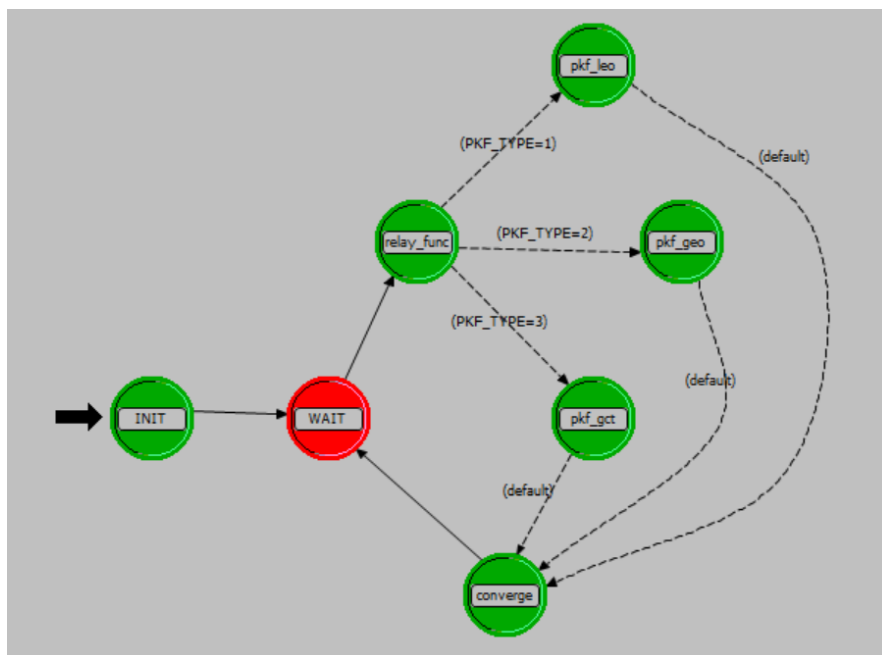


图 3-15 geo_pro 模块

在状态机中，init 阶段执行初始化设置。wait 状态是等待接收数据以进入下一步。relay_func 函数依据数据包的特定字段来区分信息源：若源于 LEO 卫星，则切换到 pkf_leo 状态，此状态下会区分处理自身状态报告和转发策略请求；若数据包来自 GEO 卫星，系统进入 pkf_geo 状态，处理 GEO 卫星间的转发数据包；若数据源自地面站，系统则进入 pkf_gct 状态，与地面站进行双向通信，包括向地面站报告收集到的 LEO 卫星状态，发起 LEO 卫星的转发策略申请，接收并广播地面站发出的流控命令至覆盖范围内的 LEO 卫星。

(4) 地面控制中心

地面控制中心为 SDN 架构的主控制器，主要功能包括：接收 GEO 卫星群的网络及节点的状态数据，构建出一个全面的网络视角，以此来达成对整个网络的监管和操作。利用全局网络视角，可以构造出源用户到目标用户之间的连接矩阵，并依据路由策略执行路径计算并生成流表。接着，这些流表经由 GEO 卫星节点被分发至

对应的 LEO 卫星节点。地面控制中心具有大容量的存储能力和强大的计算能力，能够为星间路由计算提供支撑。节点模型如图 3-16 所示。

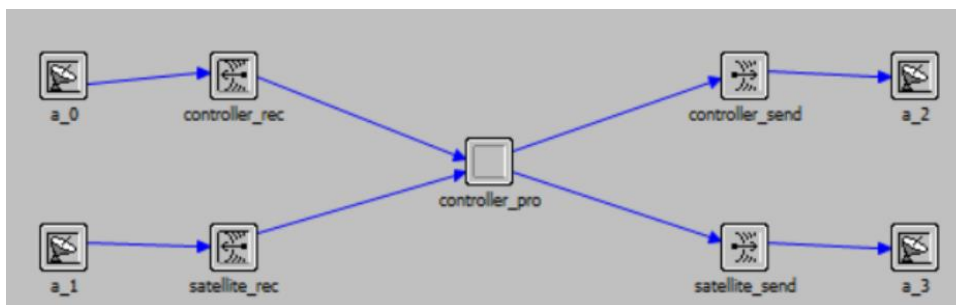


图 3-16 地面控制中心节点

`controller_pro` 模块进程层模型如图 3-17 所示。在初始化阶段，地面控制中心执行各种配置参数的设定，建立统计指标。当接收到 GEO 星座发送的即时 LEO 卫星节点和链路状况信息时，地面控制中心定期进入全局拓扑 `global_tobo` 模式，从而整合信息，构建整个网络的全景视图，实现卫星网络的集中式管控。在路由计算 `route_cal` 状态下，其核心任务是依据路由算法进行路由计算。

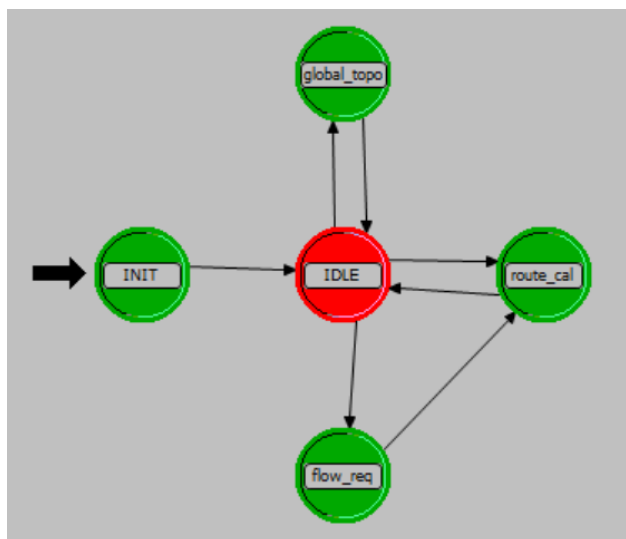


图 3-17 `controller_pro` 模型

在流请求 `flow_req` 状态下，地面控制中心接收 GEO 卫星转发的来自 LEO 卫星的流表请求，依据当前网络状态和路由策略计算出最优传输路径，随后将这些路由信息转化为流表，经由 GEO 控制网络传递给 LEO 卫星，以指导数据包的转发。

3.5 仿真与分析

3.5.1 仿真环境

本文的仿真实验环境是 STK 和 OPNET 平台。首先，以 LEO 铱星星座和 GEO 卫星集群为基础，在 STK 进行仿真，构建了覆盖全球的虚拟卫星网络，卫星参数如表 3-3 所示。从 STK 中提取星间和星地连通性信息以及相关距离数据，生成一系列的矩阵模型，如图 3-18 所示。在 OPNET 环境中构建包含卫星节点和地面站的网络，通过实时监控连接矩阵来动态捕捉网络拓扑的变化，调整节点间的连接关系和距离长度。同时，控制器通过南向接口实时获取网络运行状态，包括链路延迟、带宽利用情况以及交换节点的负载状况，根据路由算法计算出新的转发策略，并将流表下发到 GEO 卫星及 LEO 卫星交换机。实验框架如图 3-3 所示。IRIDIUM-102 号卫星与 GEO 卫星的链路保持时间如图 3-19 所示，链路保持时间报告如图 3-20 所示。

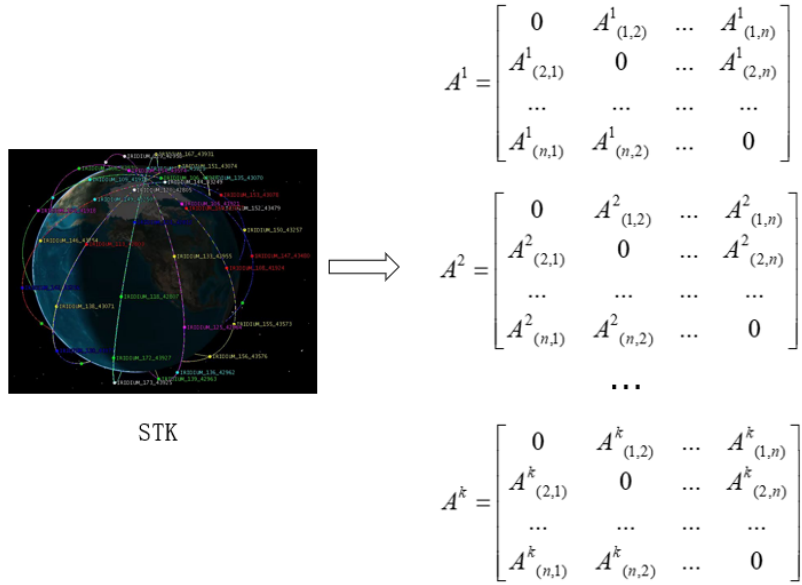


图 3-18 从 STK 导出卫星可见性矩阵

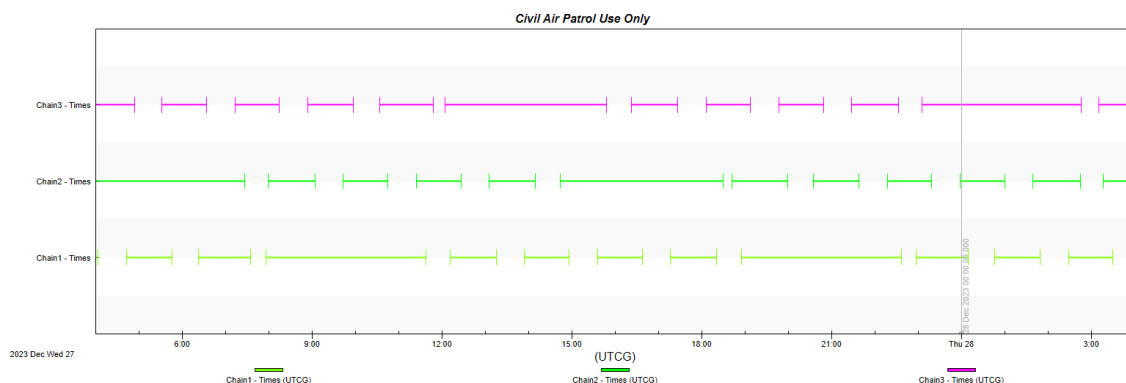


图 3-19 IRIDIUM-102 与 BEIDOU-3-G1 等 GEO 卫星之间链路保持时间

Civil Air Patrol Use Only

Chain-Chain1: Complete Chain Access

Access	Start Time (UTCG)			Stop Time (UTCG)			Duration (sec)
1	27 Dec 2023	04:00:00.000		27 Dec 2023	04:03:56.810		236.810
2	27 Dec 2023	04:43:35.076		27 Dec 2023	05:46:26.839		3771.764
3	27 Dec 2023	06:23:03.078		27 Dec 2023	07:34:40.019		4296.940
4	27 Dec 2023	07:56:47.491		27 Dec 2023	11:37:38.632		13251.142
5	27 Dec 2023	12:11:14.387		27 Dec 2023	13:16:01.707		3887.320
6	27 Dec 2023	13:54:53.983		27 Dec 2023	14:56:23.116		3689.132
7	27 Dec 2023	15:36:25.310		27 Dec 2023	16:37:50.611		3685.301
8	27 Dec 2023	17:16:49.103		27 Dec 2023	18:21:18.278		3869.175
9	27 Dec 2023	18:55:18.290		27 Dec 2023	22:37:17.683		13319.393
10	27 Dec 2023	22:57:37.031		28 Dec 2023	00:10:06.393		4349.362
11	28 Dec 2023	00:46:27.863		28 Dec 2023	01:49:27.237		3779.374
12	28 Dec 2023	02:29:01.899		28 Dec 2023	03:30:12.365		3670.465
Global Statistics							

Min Duration	1	27 Dec 2023	04:00:00.000	27 Dec 2023	04:03:56.810		236.810
Max Duration	9	27 Dec 2023	18:55:18.290	27 Dec 2023	22:37:17.683		13319.393
Mean Duration							5150.515
Total Duration							61806.179

图 3-20 IRIDIUM-102 与 BEIDOU-3-G1 卫星之间的链路保持时间报告

3.5.2 对比算法

ELB 算法^[27]: 一种基于卫星状态感知的备份多路径路由策略。卫星不断地监视当前队列的大小, 以确定它的状态, 在卫星即将面临拥塞时, 通知上游卫星使用不包含该拥塞卫星的备份路径进行数据传输。该策略可以避免卫星的拥塞和数据包的丢失。

TLR 算法^[29]: 一种基于“交通灯”概念的路由策略。该策略利用用红黄蓝三种状态来反映节点的处理状态。绿灯表示当前节点可正常收发数据, 黄灯表示该节点已接近满负荷, 不再接收新数据, 而红灯则警告若负载过高, 节点可能会丢弃数据包。数据包按照预计算的路径行进, 其吞吐量随时间波动, 路径会依据沿途节点的

“交通灯”颜色实时调整。

MTMR 算法：本文的 MTMR 多径路由算法综合考虑了链路延迟、丢包率、带宽和节点负载，因此称为四约束的最小传输成本多径路由算法。

3.5.3 结果与分析

本小节对比分析了 ELB 算法、TLR 算法和 MTMR 算法三种算法的丢包率、端到端时延以及负载均衡指数三个方面的性能。

(1) 丢包率

丢包率是反应网络状态的重要指标。在网络仿真过程中，网络总流量输入范围设置为 5T bit/h~8T bit/h，仿真周期为铱星星座变化周期 1.67 小时；星间链路带宽设置为 25Mbps，和星地链路带宽均设为 200Mbps，各个队列缓存长度设置为 200 个包。图 3.6 描述了 ELB 算法、TLR 算法和 MTMR 算法在仿真中的丢包率随网络总流量输入变化。如图 3-21 所示，在 6.5T bit/h 的输入下，ELB 算法、TLR 算法、MTMR 算法的丢包率分别为 12.53%，9.36%，5.98%；在 8T bit/h 的输入下，ELB 算法、TLR 算法、MTMR 算法的丢包率分别为 21.63%，18.39%，14.87%。不难看出，无论在多大的输入流量下，MTMR 算法相较于其他两种算法都具有更低的丢包率，这是因为 ELB 算法是一种“被动式”的拥塞控制路由算法，它只在节点即将面临拥塞时采取措施，而将该拥塞节点的流量转移到其他节点也容易引起其他节点的级联式拥塞，从而导致丢包率较高；TLR 算法仅仅考虑到节点负载问题，未能考虑节点的处理能力；而本文算法采取的是“主动式”流量控制，且将卫星交换机的缓冲容量考虑进节点负载，因此可以获得较低的丢包率。

(2) 端到端时延

图 3-22 所示为在不同的传输数据量下三种算法的端到端时延对比。MTMR 算法性能最好，BFS 算法次之，TLR 算法性能最差，随着传输数据量的增加，ELB 算法和 TLR 算法比 MTMR 算法需要更多的传输时间。当网络输入总流量为 8Tbit/h 时，ELB 算法的端到端时延比 MTMR 算法长 22.7 ms。

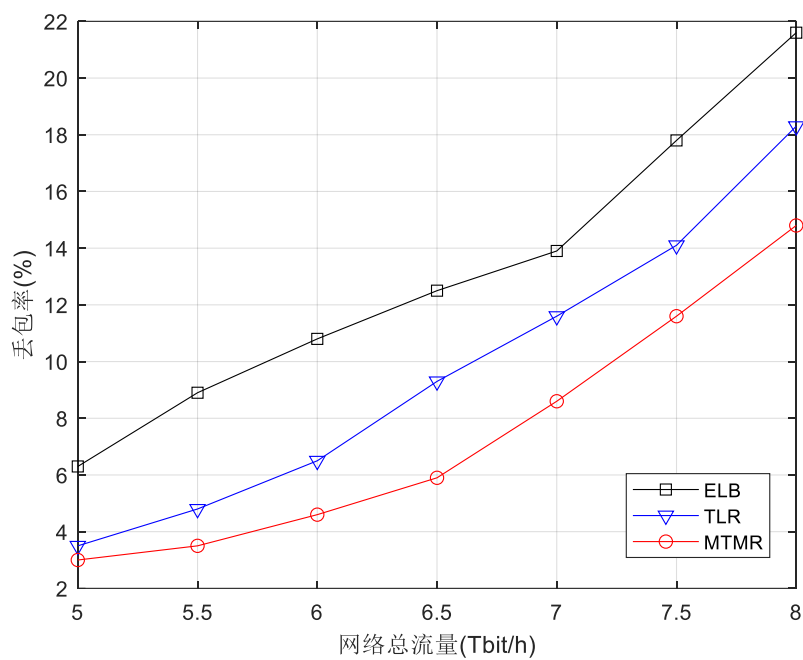


图 3-21 丢包率随网络总流量变化图

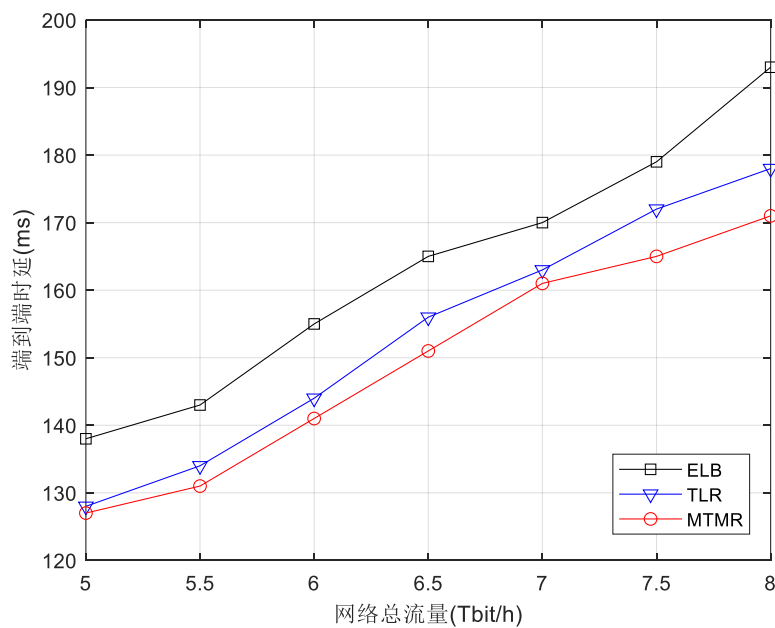


图 3-22 端到端时延随网络总流量变化图

(3) 负载均衡指数

本文采用流量分配指数来表示负载均衡^{错误!未找到引用源。}，对三种方法的负载均衡性能进行评价，计算方法为：

$$\text{负载均衡指数} = \frac{(\sum_{i=1}^n x_i)^2}{n \sum_{i=1}^n x_i^2} \quad (3-8)$$

其中 n 为路径数, x_i 为第 i 条路径上传输的数据包数。负载均衡指数取值范围为 $(0, 1)$, 负载均衡指数越接近 1, 说明多径路由算法的均衡性越好; 负载均衡指数越接近 0, 说明多径路由算法的均衡性越差。

图 3-23 为在不同的传输数据量下三种算法的负载均衡指数对比。从实验结果可以看出, ELB 算法的均衡性是最差的, 其次是 TLR 算法, MTMR 算法最优。这是因为 ELB 算法每当在节点过载时, 都会将该节点的所有流量转移到备份路径上; 而 TLR 只有在节点的“信号灯”为“绿色”时才转发数据包, 否则流量将被转移到另一条路径上, 因此, ELB 算法和 TLR 算法的流量都在多条路径间频繁迁移, 导致不同路径上传输的数据量差异较大。对于 TLR 算法而言, 随着传输数据量的增加, “信号灯”从“绿灯”变为“黄灯”或“红灯”的概率就越大, 这将进一步加剧流量迁移。

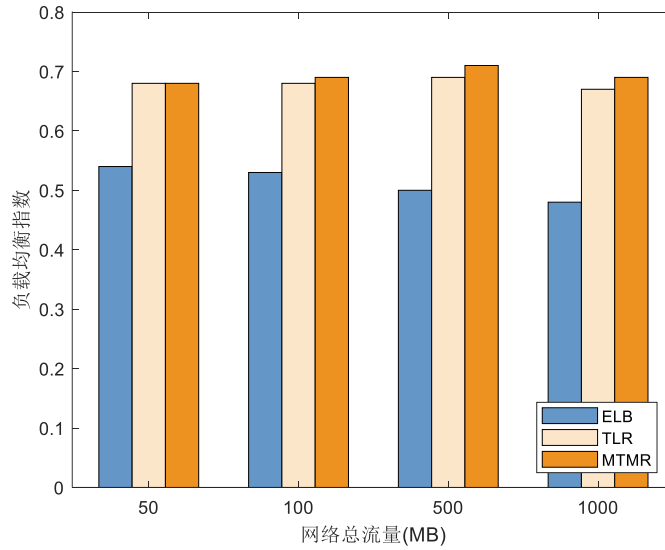


图 3-23 负载均衡指数随网络总流量变化图

3.6 本章小结

本章提出了一种基于 SDN 的最小传输代价多路径路由策略, 该策略利用 SDN 易

于获取网络状态的优点，给出了一种综合考虑传输时延、丢包率、带宽和节点负载的最优路径集的搜索算法，并设计了一种基于 SDN 的双层卫星网络架构，利用地面+GEO 卫星的星地联合部署方式，LEO 卫星平面作为数据转发层，使用地面网络和三颗 GEO 卫星作为 SDN 控制平面来实现对网络的集中控制，并在 STK 和 OPNET 平台上进行了仿真并与 ELB 算法、TLR 算法进行了对比，验证了本文路由策略的先进性。

4 基于 TC 码的卫星网络可靠传输方法

4.1 引言

卫星通信作为地面通信的有效补充，在全球通信中发挥着重要作用。在卫星通信网络中，由于通信资源有限，数据传输链路不稳定，卫星通信存在信号衰减和传输延迟等问题，如何实现稳定、高效的数据传输是卫星通信网络面临的主要挑战。除此之外，在使用多路径算法时，当链路耦合度低甚至完全不相交时，两条路径的时延差较大，导致两条路径传输的数据包在接收端产生乱序，增大了端到端时延，降低了网络吞吐量性能。当前的研究多是从考虑链路的对称性和链路的耦合度入手，计算出多条时延差较小的对称链路，但这种被动方法依然会增加一定的端到端时延。

基于此，本章考虑使用 TC (Triangular code) 码作为信道编码方法，由 TC 码编码算法产生的数据包不存在先后顺序，因此可以从编码的角度来解决链路时延差导致的接收端数据包乱序问题。另外，在有损的卫星传输链路中，TC 码可以利用冗余编码包来补偿丢失的编码包，避免了传输过程中频繁的反馈，从而抵消了链路丢包对传输可靠性的影响，减少了接收延迟。

4.2 TC 码的优势

由 Luby 提出的 LT 码在实际实现中具有很高的接收冗余。我们用以下例子来说明它的局限性，对于 n 个原始数据包 $\{P_1, P_2, P_3, \dots, P_n\}$ ，最多生成 n 个线性无关的编码包，假设有三个原始数据包 P_1, P_2, P_3 ，最多生成三个互相线性无关的编码包 $\{P_1 + P_2, P_1 + P_3, P_2 + P_3\}, \{P_1, P_2 + P_3, P_3\} \dots$ ，当发送端进行 LT 编码时，极有可能随机生成多个线性相关的编码包，而在接收端，只有当接收到 n 个线性无关的编码包才能译码成功，因此会存在发送端发送的编码包冗余的情况。即使是优化的 LT 代码，对于 $5 \leq k \leq 20$ 的小数据包批大小，也具有至少 40% 的高冗余^[68]， $k = 1024$ 时冗余度为 9%^[69]。

针对这个问题，一种新的喷泉码 TC 码 (Triangular code, TC) 被提出^[70]，TC 码

是一种非线性编码，使用简单的移位和异或运算进行编码，并使用反向替换算法进行解码，TC 码理论上可以生成无数多个线性无关的编码包，大大降低了接收端的冗余度。如图 4-1 所示，两个源数据包 P_1, P_2 通过 TC 编码生成了四个线性无关的编码包 C_1, C_2, C_3, C_4 。

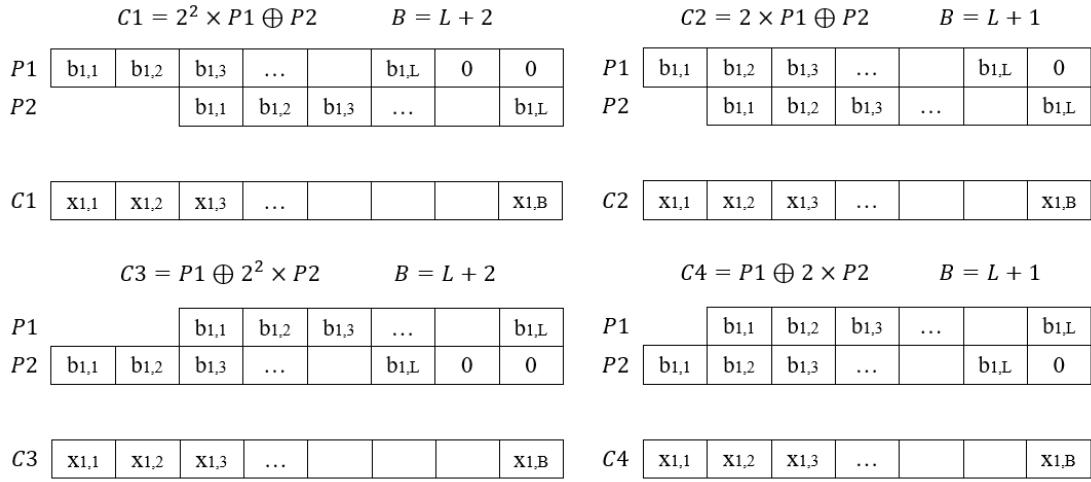


图 4-1 TC 编码示意图

虽然它在分组级别上是非线性代码，但在位级别上原子化后仍然是线性代码。TC 码相较于 LT 码的优势在于，TC 码具有近零冗余，编解码计算复杂度为 $O(kL \log k)$ ，其中 k 为数据包批大小， L 为数据包数据长度^[71]。与以往在渐近假设下显示码的最优性能不同，TC 即使在非渐近设置下也具有接近零的冗余。这些特性使 TC 能够最大限度地减少由于其低解码延迟而导致的延迟。当 k 达到 1024 时，TC 将 LT 码的冗余减少了 68%-99%。

4.3 基于反馈的码长调整机制

在卫星通信网络中，当传输信道条件不稳定时，采用固定的编码分组大小可能会导致丢包率高，从而使数据的有效传输速率降低。因此，根据数据大小和网络环境自适应动态调整编码包的大小就显得尤为重要，本节将提出基于双向搜索动态迭代算法的自适应码长 TC 码，码长可以在传输过程中实时调整。

定义传输数据文件的大小为 M ，分割后的信息包数为 n ，则每个信息包的大小

为 $\left\lceil \frac{M}{n} \right\rceil$ 。采用 TC 码进行编码，数据包需要移位，设单个信息包的最大移位数为 k ，单个编码包中编码参数值为 a ，则每个编码包中加入的附加信息大小为 an ，每个编码包的实际大小为 $\left\lceil \frac{M}{n} \right\rceil + an + k$ 。

由于卫星传输信道的不稳定和不可靠，数据传输过程中每一个比特都可能出现错误，因此需要在编码的数据包中添加检查信息，以确定单个数据包是否正确传输。为每个编码包添加循环冗余检查。校验码大小为 b 字节，因此单个编码数据包的大小为 $\left\lceil \frac{M}{n} \right\rceil + an + k + b$ ，整个数据文件的总数据大小为 $n(\left\lceil \frac{M}{n} \right\rceil + an + k + b)$ 。

卫星网络数据传输过程中存在延迟，包括传输延迟、传播延迟和处理延迟。传输延迟是从发送数据帧开始到完成数据帧所需的总时间。传输延迟 T_t 表示为

$$T_t = \frac{\left\lceil \frac{M}{n} \right\rceil + an + k + b}{r} \quad (3-9)$$

其中 r 表示发送端与接收端之间的数据传输速率。处理延迟是编码和解码所需的时间，处理延迟 T_p 用公式 $T_p = \beta M$ 表示，其中 β 为数据编码和解码的计算能力，单位为 ms/MB，即处理 1MB 数据所需的时间。

算法的目标是使总延迟最小化，提高数据的有效传输率，由于每个编码数据包的大小为 $8(\left\lceil \frac{M}{n} \right\rceil + an + k + b)$ 位，因此单个编码数据包传输失败的概率为式(3-10)所示，其中 p 为误码率。

$$P = 1 - (1 - p)^{8(\left\lceil \frac{M}{n} \right\rceil + an + k + b)} \quad (3-10)$$

接收端接收到的编码数据包数 $n_s = n(1 + \varepsilon)$ ， ε 是冗余。期望接收端接收到的编码报文数为 $(1 - P)E(n_s)$ ，其中 $E(n_s)$ 为发送的编码包数。期望接收端接收到的编码报文数等于最终接收到的编码报文数，即：

$$(1 - P)E(n_s) = n(1 + \varepsilon) \quad (3-11)$$

硕士学位论文

编码包总数期望值的表达式为：

$$E(n_s) = n(1 + \varepsilon) \frac{1}{1 - P} \quad (3-12)$$

总传输延迟 $E(T_t)$ 的期望值表示为 n_s 的期望值乘以传输延迟 T_t ，表示为式(3-13)。

$$E(T_t) = E(n_s)T_t = n(1 + \varepsilon) \frac{1}{1 - P} T_t \quad (3-13)$$

由于计算和传输是同步的，发送端不断地生成和发送数据，所以总延迟只需要加往返链路延迟。总延迟的表达式为：

$$T = E(T_t) + d = n(1 + \varepsilon) \frac{1}{1 - P} T_t + d \quad (3-14)$$

其中 d 为往返时间。

最小化总传输延迟期望即 $\min(E(T))$ ，综上， $E(T)$ 的最终表达式为：

$$E(T) = \frac{(1 + \varepsilon)n \left(\left\lceil \frac{M}{n} \right\rceil + an + k + b \right)}{r(1 - p)^{8 \left(\left\lceil \frac{M}{n} \right\rceil + an + k + b \right)}} + d \quad (3-15)$$

由于上式有向上取整运算，对其近似等式的分析如下：

$$E(T) = \frac{(1 + \varepsilon)n \left(\frac{M}{n} + an + k + b \right)}{r(1 - p)^{8 \left(\frac{M}{n} + an + k + b \right)}} + d \quad (3-16)$$

对上式求一阶导后，得到：

$$\frac{dE(T)}{dn} = \frac{(1 + \varepsilon)(2an + k + b + 8 \ln(1 - P)) \left(\frac{M^2}{n^2} + \frac{(k + b)M}{n} - a^2 n^2 - a(k + b)n \right)}{r(1 - p)^{8 \left(\frac{M}{n} + an + k + b \right)}} \quad (3-17)$$

为了使总传输延迟的期望值最小，需要获取 n 的最优值。令 $\frac{dE(T)}{dn} = 0$ ，得到下式：

$$\begin{aligned} & -8 \ln(1 - P) a^2 n^4 + (2a - 8 \ln(1 - P) a(k + b)) n^3 + (k + b) n^2 + 8 \ln(1 - P) b M n \\ & + 8 \ln(1 - P) M^2 = 0 \end{aligned} \quad (3-18)$$

由于 n 为整数，上式难以解决，因此采用启发式算法来求解，令

硕士学位论文

$$\begin{aligned} h(n) &= (1-p)^{8(\frac{M}{n}+an+k+b)} \\ g(n) &= an^2 + (k+b)n + M \\ f(n) &= \frac{g(n)}{h(n)} \end{aligned} \quad (3-19)$$

上述问题等价于求解 $f(n)$ 的最小值。对于分子 $g(n)$ ，由于 $g(n)$ 是二次函数且 $n > 0$ ，在 $n \rightarrow 0$ 时取得最小值，在区间 $(0, +\infty)$ 上单调递增。对于分母 $h(n)$ ，由平均不等式可得 $\frac{M}{n} + an \geq 2\sqrt{Ma}$ ， $n = \sqrt{\frac{M}{a}}$ 时取等号，此时 $8(\frac{M}{n} + an + k + b)$ 取得最小值，由于 $1-p < 1$ ， $h(n)$ 取得最大值，所以 $h(n)$ 在区间 $(0, +\infty)$ 内有下界，可以得出 $f(n)$ 在 $(0, +\infty)$ 内有下界。

当 n 从正方向趋近于 0 时， $h(n)$ 趋近于 0， $g(n)$ 是正的，因此 $f(n)$ 趋近于无穷，另外，因为 $g(n)$ 在区间 $[\sqrt{\frac{M}{a}}, +\infty)$ 中单调递增， $h(n)$ 单调递减，因此 $f(n)$ 在区间 $[\sqrt{\frac{M}{a}}, +\infty)$ 中单调递增，可得当 $n = \sqrt{\frac{M}{a}}$ 时， $f(n)$ 取得在区间 $[\sqrt{\frac{M}{a}}, +\infty)$ 中的最小值，因此， $f(n)$ 在区间 $(0, \sqrt{\frac{M}{a}}]$ 中取得最小值。

本节采取双向搜索动态迭代的方法来确定最优码长，算法流程如下：

算法 4.1 双向搜索动态迭代算法

输入： 文件大小 M ，编码参数 a, b ，误码率 p ，搜索步长 Δ ，最小门限 t

输出： 近似最优码长 m

1. 初始化： $n_1 = 1, n_2 = \left\lceil \sqrt{\frac{M}{a}} \right\rceil, \nabla_1 = \nabla_2 = t + 1$
 2. **while** $\nabla_1 > t$ **do**
 3. 根据式(3-15)计算延迟 d_1
 4. 根据式(3-17)计算 dn_1
 5. $n_1 = n_1 - dn_1, n_1 = \lceil n_1 \rceil$
-

-
6. 根据式(3-15)计算 d_1' , $\nabla_1 = |d_1 - d_1'|$
 7. **end while**
 8. $m_1 = \left\lceil \frac{M}{n_1} \right\rceil$
 9. **while** $\nabla_2 > t$ **do**
 10. 根据式(3-15)计算延迟 d_2
 11. 根据式(3-17)计算 dn_2
 12. $n_2 = n_2 - dn_2$, $n_2 = \lceil n_2 \rceil$
 13. 根据式(3-15)计算 d_2' , $\nabla_2 = |d_2 - d_2'|$
 14. **end while**
 15. $m_2 = \left\lceil \frac{M}{n_2} \right\rceil$
 16. **if** $n_1 < 0$ **then**
 17. $m = m_2$
 18. **else**
 19. $m = \min\{m_1, m_2\}$
 20. **end**
 21. **return** m
-

经过上述过程，可以求解出近似最优的信息包大小，并可以根据大小进行分组。将数据包分割后的数据用 TC 码进行编码，其中 TC 码的信息包数为算法 4-1 计算出的近似最优 n 值。因此，我们在 4.4 节中提出了一种自适应数据包大小 TC 码的传输算法。

4.4 基于 TC 码的卫星网络可靠传输方法

4.4.1 编码传输算法

经过 4.3 节的过程，可以求解出近似最优的信息包大小，将数据包分割后用 TC

码进行编码，图 4-2 为 TC 编码算法流程，图 4-3 为传输算法流程。在发送端，数据被细分为多个数据段，并被存入发送端的缓冲区，作为 TC 编码的输入。这些编码的长度由专门的码长调控单元来操纵。编码后的数据段随后通过多种传输路径发送。每一条路径在传输过程中，都会运用一种轻量级的反馈策略，以使码长能够根据传输条件进行自动调整。

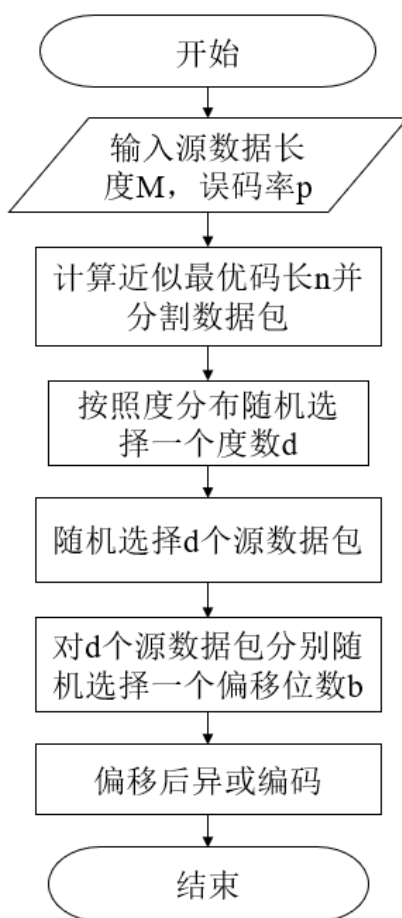


图 4-2 TC 编码算法流程

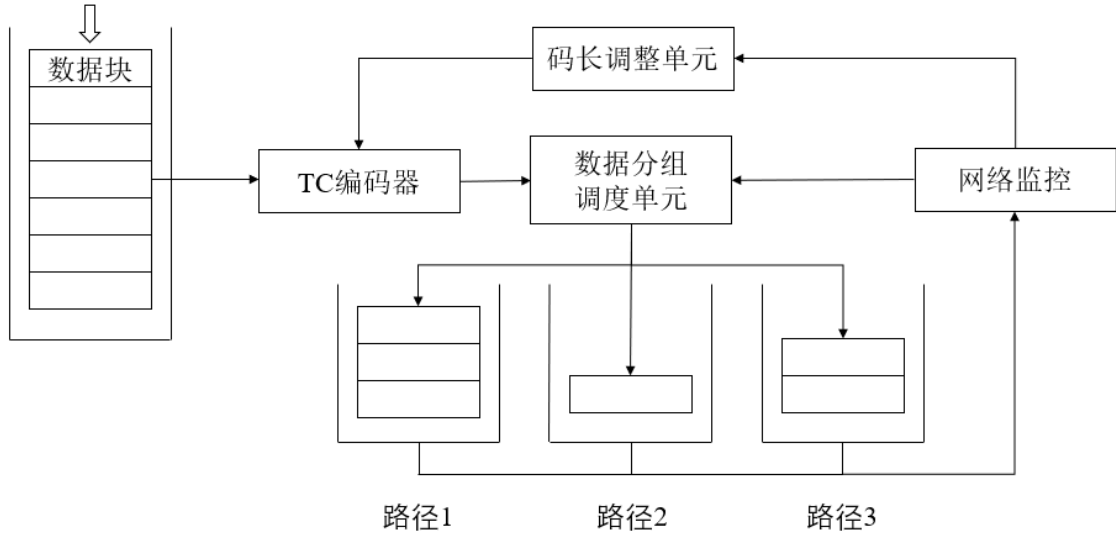


图 4-3 发送端传输算法流程

4.4.2 BLBP 译码算法

接收端会从多个路径接受数据包，运行译码算法来恢复原始数据。传统 LT 码的解码算法常用 BP 算法^[72]、增强 BP 算法^[73]和 OFG 算法^[60]，BP 算法的计算复杂度为 $O(kL \log k)$ ，其中 k 为码长， L 为包长，可见 BP 算法复杂度低，但由于 BP 算法未利用到编码包的全部信息，可能会导致解码失败；增强 BP 算法在保持和 BP 算法同样低复杂度的同时，提高了译码成功率，但仍然不尽如人意；OFG 算法解码成功率虽然高，但计算复杂度 $O(k^2 L)$ 也相对较高，如图 4-4 所示，译码开销即译码成功所需冗余数据包，可以看到，当译码开销为 0 的时候，即接收数据包数量等于源数据包数量时，BP 算法、EBP 算法和 OFG 算法的译码成功率均不到 30%，当有 5% 的译码开销后，OFG 算法的译码成功率才接近于 1，而此时 BP 算法的成功率仍然不到 20%，在接收到 50% 的冗余数据包后，BP 算法才能达到 95% 以上的成功率，而 EBP 算法也只有在接收到 15% 的冗余数据包后才能达到接近 100% 的译码成功率。

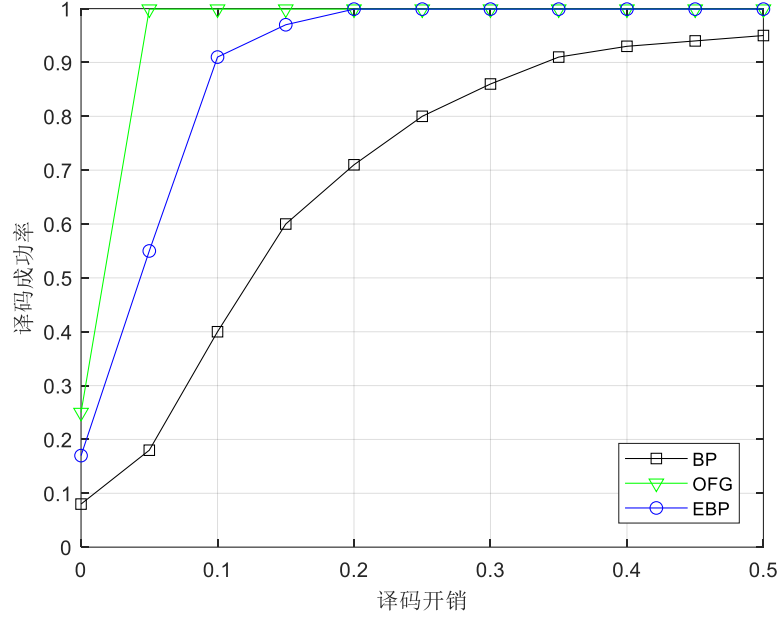


图 4-4 BP、OFG、EBP 算法在不同译码开销下的译码成功率

对此，本节针对 TC 码的天然优势，提出了 BLBP 译码算法，先在包级使用 BP 算法解码，当包级解码失败时，再触发位级反替换解码（Bit-Level Back Propagation, BLBP）算法，该算法的计算复杂度为 $O(kL \log k)$ ，最终可以达到低复杂度和高译码成功率的目的。BLBP 算法的流程如下所示：

算法 4.2 BLBP 译码算法

输入：编码包 $\{X_1, X_2, \dots, X_n\}$

输出：源数据包 $\{P_1, P_2, \dots, P_m\}$

1. $flag = 0$
2. $ripple = \emptyset$
3. **while** $flag == 0 \ \&\& \ n \geq m$ **do**
4. **for** $i = 1:n$ **do**
5. **if** $\{X_i \text{ 的度为 } 1\}$ **then**
6. $ripple = ripple \cup X_i$
7. **end if**
8. **end for**

```
9.  while  $ripple \neq \emptyset$  do
10.    for  $X_i$  in  $ripple$  do
11.      将  $X_i$  移位后与其所有相邻数据包作异或运算
12.      断开  $X_i$  与所有相邻编码包的连接，并将它们的度减 1
13.      将  $X_i$  从  $ripple$  中删除
14.      将新产生的度为 1 的编码包加入  $ripple$  中
15.    end for
16.  end while
17.  while {存在度为 1 的编码位  $x_{d,c}$ } do
18.    解码源编码位  $b_{i,j} = x_{d,c}$ 
19.    使用 for 循环替换解码位
20.  end while
21.  if {所有编码包都被译出} then
22.    flag = 1
23.  else
24.    额外接收一个编码包
25.  end if
26. end while
```

4.5 仿真与分析

4.5.1 BLBP 译码算法性能比较

为了验证本文所提出的 BLBP 译码算法的有效性，在不考虑丢包的情况下，采用相同码长对 BP、OFG、EBP、BLBP 算法的译码性能进行比较，在不同译码开销下的译码成功率如图 4-5 所示，可以看到，当译码开销为 0 时，其他三种算法的译码成功率均低于 30%，而 BLBP 算法的成功率可以达到 65%；当译码开销为 0.05 时，BLBP 算法的成功率略低于 OFG 算法，而远高于其他两种算法；当译码开销达到 0.1

时，BLBP 算法的成功率稳定在了 95%以上，说明本文算法在低译码开销下也能保证高译码成功率。

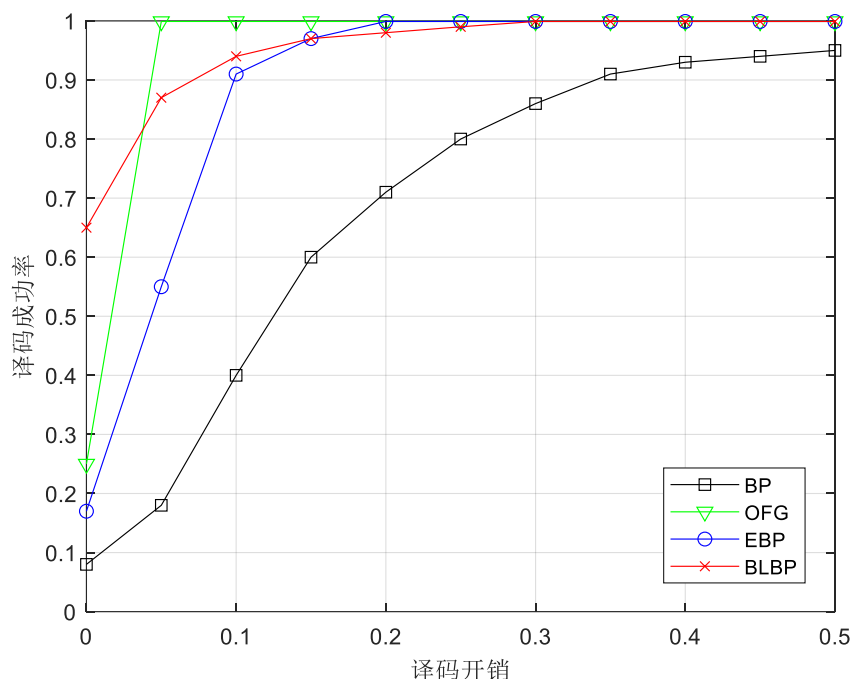


图 4-5 BP、OFG、EBP、BLBP 算法在不同译码开销下的译码成功率

4.5.2 自适应 TC 码性能比较

为了比较本文提出的自适应码长 TC 码相较于固定码长 TC 码的性能，本小节在 MATLAB 平台上模拟数据传输过程，数据大小设为 4MB，分别采用固定 TC 编码算法和自适应 TC 编码。在固定码长 TC 编码算法中，我们将信息包大小分别设置为 64 KB、128 KB 和 256KB，并比较了它们在不同的误比特率下的数据传输延迟和有效数据传输率。

(1) 延迟比较

本小节比较固定码长 TC 码和自适应码长 TC 编码在不同传输环境下的总延迟。设 $a=0.125$ ^[71]，在仿真中，往返时间设置为 200ms。根据单个数据包的大小和链路的传输速度计算传输延迟。链路传输速率设置为 1MB/s。

图 4-6 为不同误码率下自适应 TC 码编码算法、固定信息包大小为 64K 的编码算法和固定信息包大小为 128K 的编码算法的总延迟比较结果。当误码率特别低时，

时延受编码包的大小的影响较小，因为在信道条件好的时候，编码包均能无误码地到达接收端。随着误码率的增加，自适应数据包大小编码算法对延迟的降低效果更强，这是因为在信道条件变差时，若编码包过大，则出现误码的概率增加，会导致整个编码包不能被成功接收，需要发送方发送更多的冗余包来让接收方复原原始数据。当误码率为 10^{-6} 时，固定信息包大小为 256 K 的编码算法的延迟约为自适应 TC 码编码算法的 2.5 倍，固定信息包大小为 128K 的编码算法的延迟约为自适应 TC 码编码算法的 2 倍。结果表明，在传输信道条件不稳定的情况下，该算法能有效地降低时延。

（2）数据有效传输效率比较

本小节比较的是固定码长 TC 码和自适应码长 TC 编码的有效数据传输速率，有效数据传输率是文件总大小与全部接收时传输的数据包总大小的比值。图 4-7 为不同误码率下固定码长 TC 编码与自适应码长 TC 码编码算法有效数据传输率的比较结果。从图中可以看出，自适应数据包大小编码算法的有效数据传输率几乎始终优于固定信息数据包大小算法的有效数据传输率，保持在 80%以上。当误码率特别低时，由于鲁棒孤子分布的随机性，有效数据传输速率会出现波动；当误码率为 10^{-6} 时，自适应码长 TC 码编码算法的有效数据传输率比固定码长为 128 K 的编码算法高 20%左右，比固定信息包大小为 256K 的编码算法高 60%左右。结果表明，在传输信道条件不稳定的情况下，该算法能更有效地提高有效数据传输率。

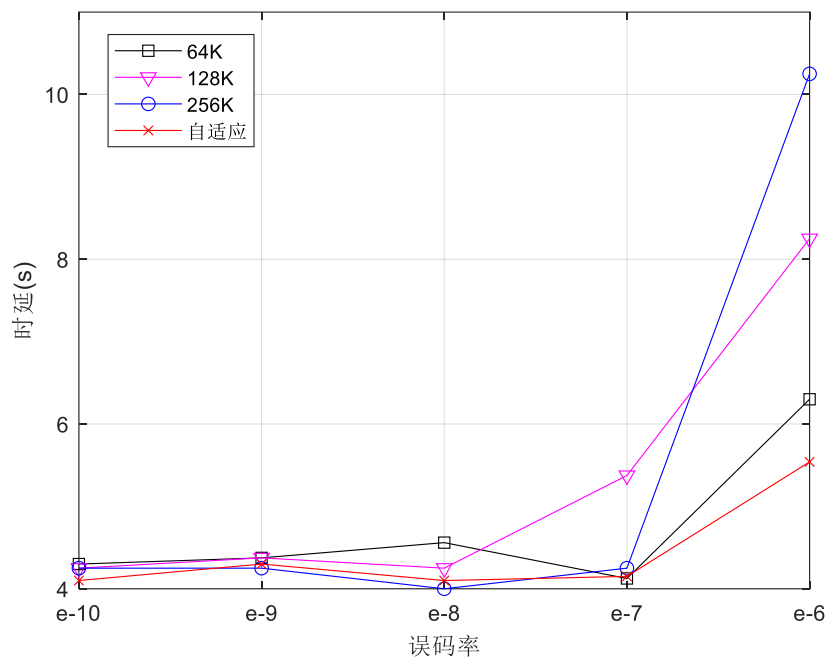


图 4-6 时延比较

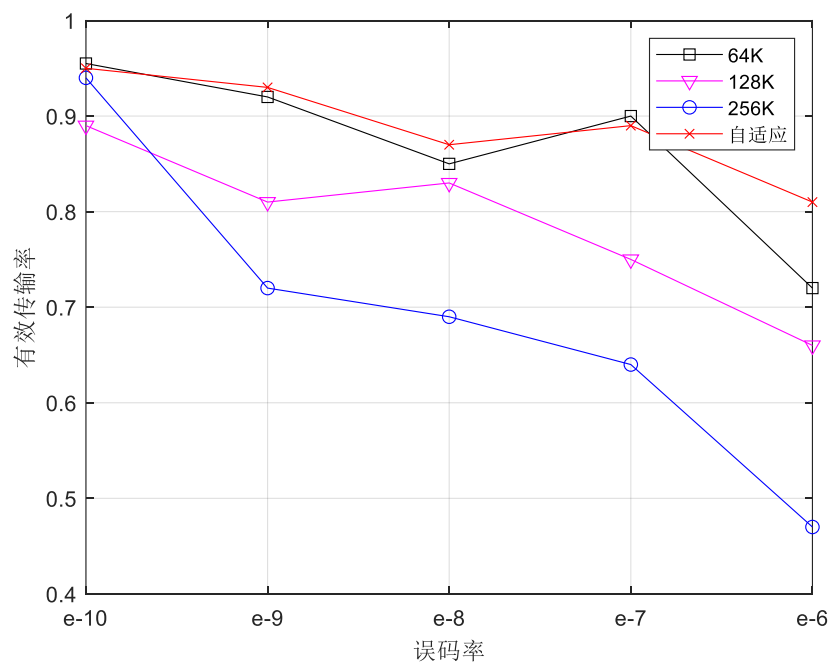


图 4-7 有效传输率比较

4.5.3 可靠传输算法性能比较

鉴于卫星网络的星间链路常受各种干扰影响，并不稳定，我们模拟了在 10%至

35%平均丢包率环境中，结合 TC 码与 MTMR 多路径传输策略的性能进行了评估。在 MATLAB 平台和 3.4 节中的 OPNET 软件进行了联合仿真，原任务数据大小为 2MB，低轨卫星处理和转发编码报文时，发送带宽为 20Mbps。如图 4-8 所示，ELB、TLR、MTMR 算法的任务交付率都随着丢包率的增加而下降，这是由于丢包导致数据缺失，而 TC+MTMR 方法和 LT+MTMR 方法始终都能保持 100%的任务交付率，这是喷泉码冗余编码的优势使接收方可以无视链路丢包所带来的影响。

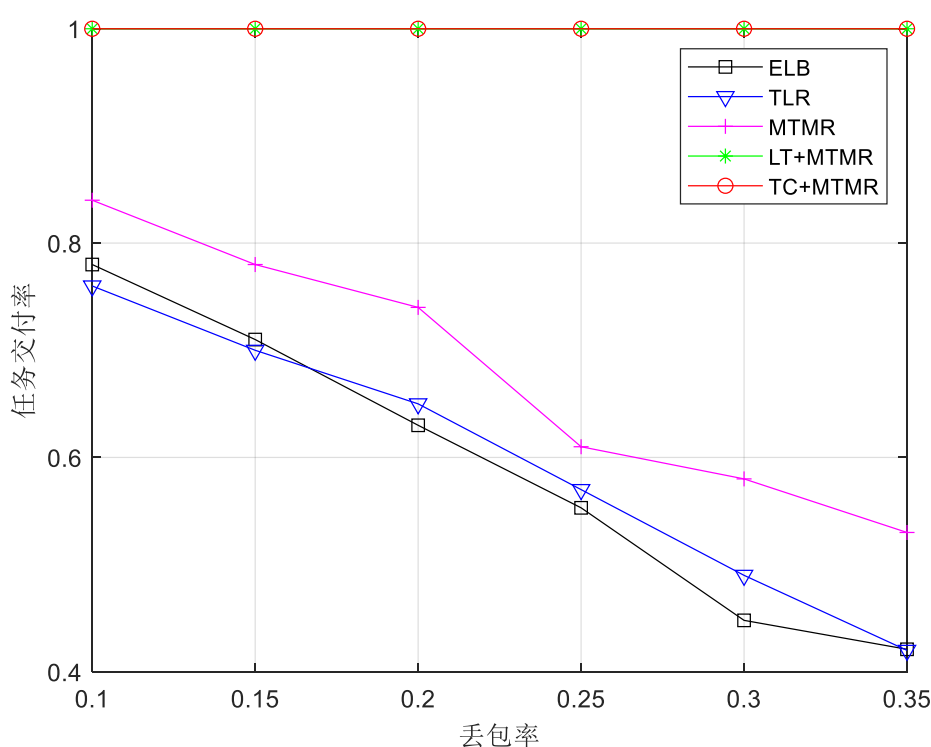


图 4-8 不同丢包率下 TC+MTMR 和对比算法的任务交付率比较

针对 ELB、TLR 和 MTMR 这三个不能保证可靠传输的路由算法，可通过重传机制来处理这种意外的传输失败。如图 4-9 所示，所有传输策略的传输延迟均随丢包率上升而增加，原因在于丢包率提高使发送端需发送更多冗余数据以弥补链路损失，而在所有传输方法中，TC+MTMR 一直保持着最低的传输时延。从图 4-9 中可以看出，LT+MTMR 的时延始终略高于 TC+MTMR，这是因为 LT 码存在很高的接收冗余，当数据量较小时，码长 n 较小，易出现编码包之间线性相关的情况，因此发送端需要发送更多编码包，接收端才能成功解码。ELB 算法、TLR 算法的传输时延始

终高于 TC+MTMR 的一倍以上，这是因为当存在丢包情况时，ELB 和 TLR 算法只能通过重传来弥补链路的损失，这样会大大增加传输时延。相比之下，TC+MTMR 不需要重传，还能在合理的多径流量调度的基础上，在保证较低的冗余度的同时，保持较低的任务时延，另外，BLBP 译码算法也提升了解码的成功率。

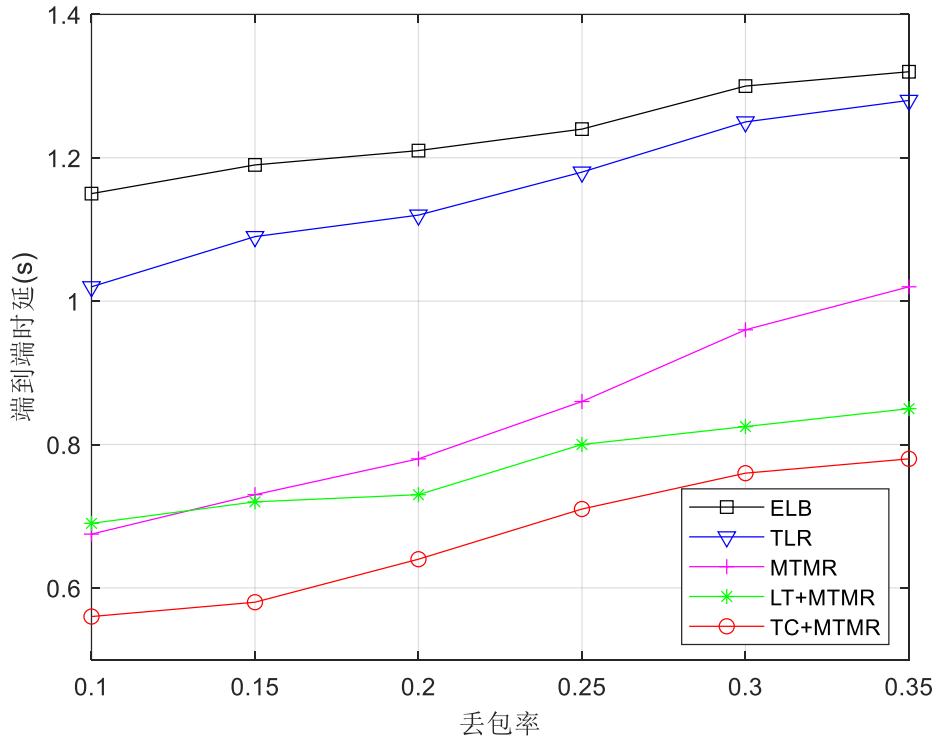


图 4-9 不同丢包率下 TC+MTMR 和对比算法的传输时延

4.6 本章小结

本章探讨了在低轨卫星网络场景下的可靠传输问题，针对卫星网络星间链路的损耗导致的数据丢失问题，提出基于 TC 编码的可靠卫星网络传输方法，在最小传输代价多路由算法的基础上，借助 TC 码的冗余编码的特性，增强了在有损传输环境和缺乏回传链路条件下的传输可靠性，保障了低轨卫星网络的传输性能。除此之外，本章还设计了能够动态调整 TC 码编码包大小的算法和高解码成功率的 BLBP 译码算法。最后，通过与相关算法进行对比，实验结果表明所提出的 TC+MTMR 算法能够对有效适应卫星网络动态恶劣传输环境，实现卫星网络可靠传输。

5 总结与展望

5.1 全文总结

低轨卫星网络解决了地面网络覆盖范围有限、抗灾能力差的缺陷，能够实现全球通信，然而，目前低轨卫星网络传输还存在一些挑战，首先，卫星网络的不断发展使得网络拓扑日益庞大复杂，低轨卫星的高速运动导致了拓扑的动态性很强，且时变的流量和节点状态使得最优路由的计算变得困难。其次，卫星网络所处的有损环境使得卫星数据传输变得不可靠，数据传输链路不稳定，链路的中断将导致超时重传，严重影响了端到端时延，再加上不对称路径导致数据包不能按序抵达接收端，增加了接收端的处理时延。针对上述问题，本文从多路径路由和可靠传输两个方面入手，展开了如下研究工作：

(1) 针对低轨卫星高动态、多节点的网络拓扑难以计算合适路由的问题，本文提出了一种基于 SDN 的最小传输代价多路由策略，该策略利用 SDN 易于获取网络状态的优点，给出了一种综合考虑传输时延、丢包率、带宽和节点负载的最优路径集的搜索算法，并设计了一种基于 SDN 的双层卫星网络架构，利用地面+GEO+LEO 的星地联合部署方式，LEO 卫星平面作为数据转发层，使用地面网络和三颗 GEO 卫星作为 SDN 控制平面来实现对网络的集中控制，并在 STK 和 OPNET 平台上进行了仿真和对比，验证了本文算法的先进性。

(2) 针对卫星链路不可预见的中断、有损环境下的高丢包率导致的高时延问题，本文提出了基于 TC 码的卫星网络可靠传输算法，设计了 TC 码编解码传输方案，给出了双向搜索动态迭代算法。通过寻找最优的编码数据包大小，通过轻量级的反馈就能实现编码批次大小对传输任务和传输环境的动态自适应调整，缩短了联合传输时延。设计了高解码率、低复杂度的 BLBP 译码算法，提升了传输效率。

5.2 后续工作展望

由于时间有限，本文所做的研究依然存在不足，后续可从以下这两个方面作进一步的研究和改善：

(1) 本文未考虑到地面用户流量分布区域性集中的问题, 以及当大量星间流量通过就近地面站进行星地传输时会引起的空间段动态漏斗型拥塞问题, 若能结合用户流量分布和地面站的漏斗形拥塞设计路由算法, 性能将会再次提升。

(2) 大多数应用的传输层算法仍采用的是 TCP 协议, 该协议虽能保证可靠传输, 但在复杂的卫星网络环境下, TCP 协议中包含的慢启动机制、ACK 确认机制以及超时重传机制都会大大限制传输效率, 若能结合本文所提出的可靠传输算法, 完善协议层面的设计, 将更有现实应用价值。

致 谢

至此，我已完成在华中科技大学七年的学习生涯，回顾硕士这三年，很感恩自己进入了一个氛围和谐的课题组，非常非常感谢我的导师魏蛟龙老师和唐祖平老师，魏老师不仅在学业上帮助我、教导我，也教会了我为人处世的道理，帮助我更好地向社会过渡，是一位令人尊敬的老师；唐老师会在我不懂的地方不厌其烦地教我，他专业的知识储备和和蔼的为人让学生如沐春风。

很感谢我同门师兄师姐师弟师妹，在我遇到困难时帮助我，在我情绪低落时安慰我。最后要感谢我的家人，对我有着百分百的支持和信任。

大学和高中不同，高中频繁的大考小考能够在短时间内形成正向反馈，让人不断受到刺激、充满动力。但到大学，反馈周期逐渐拉长，本科期间是每学期一次的期末考试，研究生期间是以年为单位的论文写作，所以研究生更像一个不知道方向和尽头的苦行僧，而我终于顺利度过了这个阶段，其间的苦辣终是如迷雾消散。

即将告别学生身份，走向社会，我对下一段全新的旅程充满了期待。

参考文献

- [1] 卢勇, 赵有健, 孙富春, 等. 卫星网络路由技术[J]. 软件学报, 2014, 25(05): 1085-1100.
- [2] 晏坚. 低轨卫星星座网络 IP 路由技术研究[D]. 清华大学, 2010.
- [3] 况鸿凤. 低轨大星座系统路由策略研究[D]. 四川: 电子科技大学, 2018.
- [4] 许晋. 天地一体化网络可信身份认证机制研究[D]. 北京邮电大学, 2019.
- [5] 仇陈. 大规模低轨卫星组网方法与性能评估[D]. 南京大学, 2021. DOI: 10.27235/d.cnki.gnjiu.2021.001159.
- [6] Joo C, Choi J P. Dynamic cross-layer transmission control for station-assisted satellite networks[J]. IEEE Transactions on Aerospace and Electronic Systems, 2015, 51(3): 1737-1747.
- [7] Fraire J A, Finochietto J M. Design challenges in contact plans for disruption-tolerant satellite networks[J]. IEEE Communications Magazine, 2015, 53(5): 163-169.
- [8] Giambene G, Luong D K, de Cola T, et al. Analysis of a packet-level block coding approach for terrestrial-satellite mobile systems[J]. IEEE Transactions on Vehicular Technology, 2019, 68(8): 8117-8132.
- [9] Li X, Wang J. Elastically reliable video transport protocol over lossy satellite links[J]. IEEE Journal on Selected Areas in Communications, 2018, 36(5): 1097-1108.
- [10] 晓春. OneWeb 太空互联网低轨星座的新进展[J]. 卫星应用, 2016 (6): 75-77.
- [11] 刘沛龙. 低轨卫星星座网络分布式路由算法研究[D]. 中国科学院大学, 2018.
- [12] Xue K, Meng W, Zhou H, et al. A lightweight and secure group key based handover authentication protocol for the software-defined space information network[J]. IEEE Transactions on Wireless Communications, 2020, 19(6): 3673-3684.
- [13] Yang J, Li D, Jiang X, et al. Enhancing the resilience of low earth orbit remote sensing satellite networks[J]. IEEE Network, 2020, 34(4): 304-311.
- [14] 黄昀辉. 卫星网络智能传输研究[D]. 中国科学技术大学, 2023. DOI: 10.27517/d.cnki.gzkju.2023.000021.
- [15] Saeed N, Almorad H, Dahrouj H, et al. Point-to-point communication in integrated

- satellite-aerial 6G networks: State-of-the-art and future challenges[J]. IEEE Open Journal of the Communications Society, 2021, 2: 1505-1525.
- [16] Zhou D, Sheng M, Wang X, et al. Mission aware contact plan design in resource-limited small satellite networks[J]. IEEE Transactions on Communications, 2017, 65(6): 2451-2466.
- [17] Chang H S, Kim B W, Lee C G, et al. Topological design and routing for low-earth orbit satellite networks[C]//Proceedings of GLOBECOM'95. IEEE, 1995, 1: 529-535.
- [18] Chang H S, Kim B W, Lee C G, et al. Performance comparison of static routing and dynamic routing in low-earth orbit satellite networks[C]//Proceedings of Vehicular Technology Conference-VTC. IEEE, 1996, 2: 1240-1243.
- [19] Chang H S, Kim B W, Lee C G, et al. FSA-based link assignment and routing in low-earth orbit satellite networks[J]. IEEE transactions on vehicular technology, 1998, 47(3): 1037-1048.
- [20] Gounder V V, Prakash R, Abu-Amara H. Routing in LEO-based satellite networks[C]//1999 IEEE Emerging Technologies Symposium. Wireless Communications and Systems (IEEE Cat. No. 99EX297). IEEE, 1999: 22.1-22.6.
- [21] Jianjun B, Xicheng L, Zexin L, et al. Compact explicit multi-path routing for LEO satellite networks[C]//HPSR. 2005 Workshop on High Performance Switching and Routing, 2005. IEEE, 2005: 386-390.
- [22] Henderson T R, Katz R H. On distributed, geographic-based packet routing for LEO satellite networks[C]//Globecom'00-IEEE. Global Telecommunications Conference. Conference Record (Cat. No. 00CH37137). IEEE, 2000, 2: 1119-1123.
- [23] Ekici E, Akyildiz I F, Bender M D. A distributed routing algorithm for datagram traffic in LEO satellite networks[J]. IEEE/ACM Transactions on networking, 2001, 9(2): 137-147.
- [24] Ekici E, Akyildiz I F, Bender M D. Datagram routing algorithm for LEO satellite networks[C]//Proceedings IEEE INFOCOM 2000. Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (Cat. No. 00CH37064). IEEE, 2000, 2: 500-508.
- [25] Henderson T R, Katz R H. On distributed, geographic-based packet routing for LEO satellite networks[C]//Globecom'00-IEEE. Global Telecommunications Conference.

- Conference Record (Cat. No. 00CH37137). IEEE, 2000, 2: 1119-1123.
- [26] Liu X, Yan X, Jiang Z, et al. A low-complexity routing algorithm based on load balancing for LEO satellite networks[C]//2015 IEEE 82nd Vehicular Technology Conference (VTC2015-Fall). IEEE, 2015: 1-5.
- [27] Taleb T, Mashimo D, Jamalipour A, et al. Explicit load balancing technique for N GEO satellite IP networks with on-board processing capabilities[J]. IEEE/ACM transactions on Networking, 2008, 17(1): 281-293.
- [28] Song G, Chao M, Yang B, et al. TLR: A traffic-light-based intelligent routing strategy for N GEO satellite IP networks[J]. IEEE Transactions on Wireless Communications, 2014, 13(6): 3380-3393.
- [29] Tang F .Dynamically Adaptive Cooperation Transmission among Satellite-Ground Integrated Networks[C]//IEEE INFOCOM 2020 - IEEE Conference on Computer Communications.IEEE, 2020.DOI:10.1109/INFOCOM41043.2020.9155485.
- [30] Feng B, Zhou H, Zhang H, et al. HetNet: A flexible architecture for heterogeneous satellite-terrestrial networks[J]. IEEE network, 2017, 31(6): 86-92.
- [31] Geng S, Liu S, Fang Z, et al. An agent-based clustering framework for reliable satellite networks[J]. Reliability Engineering & System Safety, 2021, 212: 107630.
- [32] F. Tang, Dynamically adaptive cooperation transmission among satelliterground integrated networks
- [33] Sheng M, Wang Y, Li J, et al. Toward a flexible and reconfigurable broadband satellite network: Resource management architecture and strategies[J]. IEEE Wireless Communications, 2017, 24(4): 127-133
- [34] Yang B, Wu Y, Chu X, et al. Seamless handover in software-defined satellite networking[J]. IEEE Communications Letters, 2016, 20(9): 1768-1771.
- [35] Li T, Zhou H, Luo H, et al. SERvICE: A software defined framework for integrated space-terrestrial satellite communication[J]. IEEE Transactions on Mobile Computing, 2017, 17(3): 703-716.
- [36] ul Huque M T I, Si W, Jourjon G, et al. Large-scale dynamic controller placement[J]. IEEE Transactions on Network and Service Management, 2017, 14(1): 63-76.
- [37] Killi B P R, Rao S V. Optimal model for failure foresight capacitated controller placement in software-defined networks[J]. IEEE Communications Letters, 2016,

- 20(6): 1108-1111.
- [38] ul Huque M T I, Jourjon G, Gramoli V. Revisiting the controller placement problem[C]//2015 IEEE 40th conference on local computer networks (LCN). IEEE, 2015: 450-453.
- [39] Yan J, Zhang H, Shuai Q, et al. HiQoS: An SDN-based multipath QoS solution[J]. China Communications, 2015, 12(5): 123-133.
- [40] Zaw H T, Maw A. Traffic management with elephant flow detection in software defined networks (SDN)[J]. International Journal of Electrical and Computer Engineering, 2019, 9(4): 3203.
- [41] Y. Wei et al., "Routing Based on Dynamic Reliability in Massive LEO Satellite Optical Networks," 2023 Opto-Electronics and Communications Conference (OECC), Shanghai, China, 2023, pp. 1-3, doi: 10.1109/OECC56963.2023.10209572.
- [42] Partridge C, Shepard T J. TCP/IP performance over satellite links[J]. IEEE network, 1997, 11(5): 44-49.
- [43] Jacobson V. Congestion avoidance and control, SIGCOMM'88: Symposium proceedings on Communications architectures and protocols[J]. 1988.
- [44] Caini C, Firrincieli R. TCP Hybla: a TCP enhancement for heterogeneous networks[J]. International journal of satellite communications and networking, 2004, 22(5): 547-566.
- [45] Wang R, Valla M, Sanadidi M Y, et al. Adaptive bandwidth share estimation in TCP Westwood[C]//Global Telecommunications Conference, 2002. GLOBECOM'02. IEEE. IEEE, 2002, 3: 2604-2608.
- [46] Akyildiz I F, Morabito G, Palazzo S. TCP-Peach: a new congestion control scheme for satellite IP networks[J]. IEEE/ACM Transactions on networking, 2001, 9(3): 307-321.
- [47] Lai Y C, Lai C N. DCCP partial reliability extension with sequence number compensation[J]. Computer Networks, 2008, 52(16): 3085-3100.
- [48] MacKay D J C. Fountain codes[J]. IEE Proceedings-Communications, 2005, 152(6): 1062-1068.
- [49] S. K. Kartal, Y. B. Kaya, Y. Yilmaz, E. Özbağ, A. Z. Abdurahman and T. Dar, "Perturbation Effect to Orbit Motion of Satellite for Positioning System," 2023 10th

- International Conference on Recent Advances in Air and Space Technologies (RAST), Istanbul, Turkiye, 2023, pp. 1-6, doi: 10.1109/RAST57548.2023.10197851.
- [50] P. Liu, J. Liang, X. Qian and Y. Xu, "Research on link load-balancing between marine LEO satellite constellations based on SDN," 2021 2nd International Conference on Electronics, Communications and Information Technology (CECIT), Sanya, China, 2021, pp. 977-982, doi: 10.1109/CECIT53797.2021.00175.
- [51] X. Xu, C. Wang and Z. Jin, "An analysis method for ISL of multilayer constellation," in *Journal of Systems Engineering and Electronics*, vol. 33, no. 4, pp. 961-968, August 2022, doi: 10.23919/JSEE.2022.000093.
- [52] 张涛,柳重堪,张军. 卫星时变拓扑网络最短路径算法研究[J]. 计算机学报,2006,29(3):371-377. DOI:10.3321/j.issn:0254-4164.2006.03.003.
- [53] 朱润涛.基于流量预测的卫星网络负载均衡路由关键技术研究[D].北京邮电大学,2017.
- [54] LIU Liang, ZHANG Tao, and LU Yong. A novel adaptive routing algorithm for delay-sensitive service in multihop LEO satellite network[J]. *KSII Transactions on Internet and Information Systems*, 2016, 10(8): 3551–3567.doi: 10.3837/ tiis.2016.08.007
- [55] RAO Yuan, WANG Ruchuan, and XU Xiaolong. Load balancing routing for single-layered satellite networks[J]. *The Journal of China Universities of Posts and Telecommunications*, 2010, 17(2): 92–99.doi: 10.1016/S1005-8885(09)60453-2
- [56] LIU Peilong, CHEN Hongyu, WEI Songjie, *et al.* Hybrid-traffic-detour based load balancing for onboard routing in LEO satellite networks[J]. *China Communications*, 2018, 15(6): 28–41.doi: 10.1109/CC.2018.8398222
- [57] 沈亦昼. 天地一体化网络下网关和 SDN 控制器联合部署的研究[D]. 上海:东华大学,2022.
- [58] Qureshi J, Foh C H. Triangular code: Near-optimal linear time fountain code[J]. *Digital Communications and Networks*, 2023, 9(4): 869-878.
- [59] 马梦宇,罗长洲,梁春瑞,等.RaptorQ 喷泉码编译码算法分析及优化[J].系统工程与电子技术,2024,46(04):1440-1447.
- [60] Lu F, Foh C H, Cai J, et al. LT codes decoding: Design and analysis[C]//2009 IEEE

- International Symposium on Information Theory. IEEE, 2009: 2492-2496.
- [61] Y. Jia and Z. Peng, "The analysis and simulation of communication network in Iridium system based on OPNET," 2010 2nd IEEE International Conference on Information Management and Engineering, Chengdu, China, 2010, pp. 68-72, doi: 10.1109/ICIME.2010.5477454.
- [62] Li T, Chen J, Fu H. Application scenarios based on SDN: an overview[C]//Journal of Physics: Conference Series. IOP Publishing, 2019, 1187(5): 052067.
- [63] Wang F, Jiang D, Qi S. An adaptive routing algorithm for integrated information networks[J]. China Communications, 2019, 16(7): 195-206.
- [64] Liu Z, Dong X, Wang L, et al. Satellite network task deployment method based on SDN and ICN[J]. Sensors, 2022, 22(14): 5439.
- [65] Sheng M, Wang Y, Li J, et al. Toward a flexible and reconfigurable broadband satellite network: Resource management architecture and strategies[J]. IEEE Wireless Communications, 2017, 24(4): 127-133.
- [66] Shi Y, Cao Y, Liu J, et al. A cross-domain SDN architecture for multi-layered space-terrestrial integrated networks[J]. IEEE Network, 2019, 33(1): 29-35.
- [67] 韩珍珍,赵国锋,徐川,等. 基于时延的LEO卫星网络SDN控制器动态放置方法[J]. 通信学报,2020,41(3):126-135. DOI:10.11959/j.issn.1000-436x.2020054.
- [68] Hyytia E , Tirronen T , Virtamo J .Optimal Degree Distribution for LT Codes with Small Message Length[C]//IEEE Infocom -IEEE International Conference on Computer Communications.IEEE, 2007.DOI:10.1109/INFCOM.2007.324.
- [69] Yen K K , Liao Y C , Chang H C .Design of LT code degree distribution with profiled output ripple size[C]//IEEE Workshop on Signal Processing Systems.IEEE, 2015.DOI:10.1109/SiPS.2015.7344985.
- [70] Qureshi J, Foh C H. Triangular code: Near-optimal linear time fountain code[J]. Digital Communications and Networks, 2023, 9(4): 869-878.
- [71] Xu X, Wang Q, Liu C, et al. A satellite network data transmission algorithm based on adaptive LT code[C]//2021 International Conference on Space-Air-Ground Computing (SAGC). IEEE, 2021: 100-105.
- [72] Fan Z, Lixin X, Xi P. Comparison of BP and Gauss code base on Fountain

- Code[C]//2011 Third International Conference on Measuring Technology and Mechatronics Automation. IEEE, 2011, 1: 737-740.
- [73] Zhang L, Su L. Design of Improved BP Decoders and Corresponding LT Code Degree Distribution for AWGN Channels[J]. Wireless Communications and Mobile Computing, 2020, 2020: 1-16.