大规模低轨卫星星座区域划分和负载均衡分段路由策略

郑恺填[1]，陈晨[1]，杜平[2,3,4]，俱莹[1]，徐宁[1,5]
（1.西安电子科技大学通信工程学院，陕西 西安 710071；2.上海卫星互联网研究院有限公司，上海 200120；
3.卫星互联网重点实验室，上海 200120；4.上海市卫星互联网重点实验室，上海 200120；
5.重庆卫星网络系统研究院有限公司，重庆 401135）

摘要：大规模低轨（LEO）卫星星座中，节点故障、局部业务突增和链路拥塞容易造成星间转发层负载不均衡，进而影响端到端传输性能。针对非均匀业务压力下的星间路由负载均衡问题，提出一种业务压力驱动的高需求区域划分与结构化分段路由算法。该算法利用公开统计数据构造具有空间异构性的地面业务压力背景，并将其映射为卫星节点负载，完成高需求节点标记和高需求区域划分；在此基础上，为不同节点类型分配分段标识（SID），并结合源宿节点位置构造分层航点序列，使业务流尽量绕开高需求区域。基于NS-3平台的重复仿真结果表明，与对比算法相比，所提算法在多数负载区间内具有较低端到端时延和丢包率、较高吞吐量和流量分布指数，并在单链路及多链路故障场景下表现出较快的路由恢复能力。

关键词：大规模低轨卫星星座；负载均衡；区域划分；SRv6；多级航点路由
中图分类号：TP309
文献标志码：A
doi：10.11959/j.issn.1000–436x.TXXB260237

Region division and load balancing segment routing strategy for large-scale low earth orbit satellite constellations

Zheng Kaitian[1], Chen Chen[1], Du Ping[2,3,4], Ju Ying[1], Xu Ning[1,5]
1. School of Telecommunications Engineering, Xidian University, Xi'an 710071, China
2. Shanghai Satellite Network Research Institute Co., Ltd., Shanghai 200120, China
3. Key Laboratory of Satellite Network, Shanghai 200120, China
4. Shanghai Key Laboratory of Satellite Network, Shanghai 200120, China
5. Chongqing Satellite Network System Research Institute Co., Ltd., Chongqing 401135, China

Abstract: In large-scale low earth orbit (LEO) constellations, node failures, traffic surges, and link congestion unbalance inter-satellite forwarding loads and degrade end-to-end performance. For the inter-satellite routing load balancing problem under non-uniform service pressure, a traffic-pressure-driven algorithm was proposed for high-demand region partitioning and structured segment routing under non-uniform pressure. Public statistics constructed a heterogeneous traffic-pressure profile, mapped it to satellite loads, and marked high-demand nodes for partitioning. It assigned segment ID by node type and built hierarchical waypoints from endpoints to bypass high-demand regions. NS-3 simulations show lower end-to-end delay and packet-loss rate, higher throughput and traffic distribution index (TDI) than benchmarks across most load ranges, and faster recovery under single- and multi-link failures.

Key words: large-scale LEO satellite constellations, load balancing, region division, SRv6, multi-level waypoint routing

收稿日期：2026–04–30；修回日期：2026–06–14
通信作者：陈晨，cc2000@mail.xidian.edu.cn
基金项目：国家重点研发计划基金资助项目（No.2025YFF0514803）；陕西省重点研发计划基金资助项目（No.2021ZDLGY02-09, No.2023-GHZD-44, No.2023-ZDLGY-54）
Foundation Items：The National Key Research and Development Program of China (No.2025YFF0514803), The Key Research and Development Program of Shaanxi Province (No.2021ZDLGY02-09, No.2023-GHZD-44, No.2023-ZDLGY-54)

0 引言

作为6G“空天地海一体化”通信网络中的重要组成部分，低地球轨道（LEO）卫星接入网主要面向海洋、沙漠、灾害应急等地面网络难以连续覆盖的场景，并正朝着大规模、多层化方向演进[1]。在这一背景下，本文聚焦于完成星地接入之后的星间转发层负载均衡问题，研究如何在动态拓扑和非均匀业务压力下实现热点绕行与链路拥塞缓解。低轨卫星的高速运动使网络拓扑持续动态演化[2]：卫星波束覆盖区域随时间变化，星间链路频繁通断[3]，高纬度区域的星间链路按连接策略周期性关闭[4]。这些因素会加剧卫星节点和链路负载波动。与此同时，即使卫星业务主要面向偏远覆盖场景，地面业务压力仍具有明显的空间差异。

本文采用人口、经济等公开统计量构造具有现实地理差异特征的非均匀业务压力背景[5]，并不等价于真实卫星接入需求，而是用于验证路由算法在负载不均衡条件下的有效性。

现有算法通常通过区域划分降低卫星星座的拓扑管理和路由计算复杂度，但不同划分方式在动态场景下仍存在局限：基于地理坐标或固定规则的静态划分开销较低，但当跨域业务集中涌入或局部链路失效时，易使局部链路持续承压；基于聚类或全局优化的动态划分能够利用状态信息重构区域，但通常依赖频繁状态收集与全局计算，在大规模星座下难以及时跟踪拓扑演化；已有分段路由（segment routing, SR）研究多围绕极区断链绕行或骨架路径构建展开，对业务压力驱动的动态热点识别和主动负载疏导考虑不足[6-11]。因此，面向非均匀业务压力场景，如何构建能够感知高需求区域并快速调整转发路径的轻量化路由机制，成为提升星间网络服务能力的关键。

1 相关工作

1.1 分区分域算法

大规模低轨卫星星座面临着巨型拓扑的动态变化和负载不均衡两大挑战，分区分域算法可降低网络管理复杂度，分段路由技术能够提高路径选择的灵活性。本节回顾了分区分域算法和分段路由技术的研究现状。

分区分域算法可以分为基于静态规则、基于动态优化和基于需求的区域划分算法。表1为分区分域算法的划分依据和优缺点。

1.2 分段路由

分区分域算法极大地降低了网络复杂度，为高效路由提供了可能，如何为数据流灵活、高效地选择一条传输路径，则成为下一个问题。分段路由是互联网领域一项基于源路由的新型网络技术，可以简化中间节点状态维护并提高路径可编程性[6]。赵鑫[4]设计了基于航点的星间分段路由算法（WSR），核心目标是绕过极区星间链路（ISL）周期性断开造成的拓扑空洞。Hu等[11]采用骨架图和分层转发机制（LGSR）降低低轨星座中的路径计算复杂度；Zhao等[15]提出了面向卫星网络的数据平面协议（STSR），利用位图表示节点和路径，但在超大规模场景下头部开销与扩展性仍受限制。Tu等[16]提出象限分段路由算法（QMSR），通过静态象限几何划分构造固定规则的分段路径，可在较低复杂度下完成大规模星座中的分段转发，但其区域划分

表1 分区分域算法的划分依据和优缺点

分类	子类	划分依据	优点	缺点	文献
基于静态规则	基于地理空间	使用北斗网格地理编码进行固定区域划分	规则简单，开销小；支持快速空间索引和查询	难以适应网络拓扑动态变化；灵活性差	文献[8]
	基于固定规模	将星座划分为包含固定数量卫星的簇	实现分布式管理，降低计算复杂度；易于部署	仅依赖星座规模，忽略流量需求；可能负载不均衡	文献[9]
	基于拓扑特性	根据卫星轨道坐标划分为4个象限区域	划分规则固定，简化路由计算；适用静态拓扑	无法适配链路动态变化；极区边缘易形成瓶颈	文献[10]
基于动态优化	聚类算法	根据网络实时状态动态聚类卫星	灵活性高，能响应网络变化；优化资源分配	计算复杂度较高；难以满足时延敏感业务	文献[12]
	优化算法	根据区域需求定制划分，多目标优化调整	提高资源利用率；适配业务需求变化	依赖实时数据，可能引入时延；难以应对突发流量	文献[13]
基于需求	基于地面属性	根据应用场景、地理位置等细粒度划分	以用户需求为中心，优化网络性能和用户体验	划分复杂度高；需大量地面数据支持，实施难度大	文献[14]

主要依赖几何位置关系，难以随业务压力变化自适应调整。现有研究表明，SR 适合用于动态卫星网络，但面向业务压力驱动高需求区域的确定性多级航点选择仍有研究空间。

SRv6 是基于 IPv6 的 SR 技术，利用 IPv6 地址空间和扩展头机制承载分段路由信息，更适用于最前沿的空间网络。戴天等[17]对 SRv6 技术应用于空间网络路由的可行性进行分析，提出了基于 SRv6 的空间网络模型，详细描述了基于 SRv6 的空间网络数据包路由和传输的过程。现有 SR 机制适用于动态星座场景，但其关注点并不相同：WSR 主要面向极区拓扑空洞绕行，QMSR 侧重静态几何划分下的固定规则分段转发，LGSR 侧重通过骨架图降低大规模路径计算复杂度，STSR 强调面向卫星网络的数据平面表示效率。总体来看，现有算法尚未显式结合业务压力驱动的高需求区域识别与多级航点协同选择来进行主动流量疏导。

1.3　本文工作

综上，现有分区分域算法在应对大规模 LEO 星座的动态性和不均衡业务需求时通常存在两方面不足：一方面，静态几何划分算法实现简单，但难以反映非均匀业务压力下的局部承载失衡；另一方面，复杂动态重构算法虽然灵活，但在大规模动态拓扑下往往伴随较高的在线计算与状态收集开销。与此同时，现有分段路由研究更多面向极区断链绕行或骨架路径构建，对业务压力驱动的高需求识别及其与路由决策之间的协同关系关注不足。为了综合动态区域划分与 SRv6 灵活可编程的优点，本文聚焦于星地接入关系与业务落地关系已知条件下的星间转发层负载均衡问题。本文的核心贡献主要体现在以下 4 个方面。

（1）提出面向星间转发层的业务压力感知高需求区域划分模型，用公开统计数据构造空间异构的非均匀业务压力背景，并将其映射为卫星节点负载和转发压力的不均衡性。

（2）提出高需求区域划分机制，根据高转发压力节点在卫星逻辑拓扑中的连通关系形成高需求区域，并为不同节点类型设计分段路由标识（SID），为后续结构化分段路径构造提供节点和区域基础。

（3）将源宿业务的分段路由过程建模为高需求区域协同的结构化分段路径构造问题，并设计一级出口航点、一级入口航点和二级绕行航点的分层求解机制。与主要面向拓扑空洞绕行的 WSR[4]、依赖骨架图进行分层转发的 LGSR[11]以及采用静态几何分区的 QMSR[16]不同，本文航点由业务压力驱动的高需求区域边界确定，分别服务于高需求区域的快速退出、跨区绕行和目标区域定向进入，从而改善星间网络的负载均衡与抗毁性能。

（4）通过 NS-3 仿真验证，在统一仿真设置下，所提算法相较于基准算法在端到端时延、吞吐量、丢包率和流量分布指数方面表现更优，具有更快的路由恢复能力，同时结合复杂度分析、运行开销对比和消融实验，验证各模块在性能收益与系统代价之间具有较合理的权衡关系，初步说明其在星间转发层中的有效性和可执行性。

与已有算法相比，本文的创新不在于单纯采用 SRv6 作为分段路由载体，也不在于沿用静态几何划分、骨架图抽象等已有思路，而是在控制面构建从高需求区域识别到边界提取和结构化分段决策的协同机制。进一步地，高需求区域划分的作用是为后续分段路由提供边界候选集合。高需求区域边界可用于一级出口航点、一级入口航点和二级绕行航点选择，从而减少业务流在高需求区域内部的停留和转发竞争。因此，区域划分、边界提取和多级航点选择共同构成面向星间转发层负载均衡的连续决策过程。具体而言，本文以业务压力驱动的高需求区域为输入，在相关区域边界候选集合上选择具有快速退出、跨区绕行和定向进入功能的多级航点，其优势不在于协议封装方式本身，而在于高需求区域规避与结构化路由决策机制。需要指出的是，本文工作的重点不在于求解完整的端到端路由问题，多星可视下的卫星接入、落地与信关站选择问题不在本文当前建模范围内。

2　业务压力驱动的高需求区域划分模型

本节构建业务压力驱动的高需求区域划分模型，该模型旨在将地面业务需求的空间不均衡性转化为卫星节点负载差异，并进一步形成高需求区域和普通区域，为分段路由实现和航点选择算法提供输入。模型分为 3 步：利用公开统计数据构造地面网格需求，建立卫星覆盖与负载映射关系，通过负载感知的波束分配优化完成区域划分。表 2 为定义的符号说明。

表2 定义的符号说明

符号	描述
$G_{1\times M}$	地面网格集合，M为网格数量
$\mathrm{POP}_{1\times M}$	网格人口数量
$\mathrm{GDP}_{1\times M}$	网格GDP值
$\mathrm{DIS}_{M\times M}$	网格之间的距离集合
α,β,γ	重力模型参数
$d_{1\times M}$	地面网格流量需求
H	地面高需求网格集合
$S_{1\times N}$	高需求卫星集合，N为卫星数量
$\boldsymbol{C}_{N\times K}$	高需求卫星覆盖网格集合
$\boldsymbol{B}_{N\times K}$	卫星波束集合，K为单星波束数
$b_{\max}$	单卫星频率带宽资源大小
$\boldsymbol{\delta}_{N\times K\times K}\in\{0,1\}$	$\boldsymbol{C}$与$\boldsymbol{B}$的连接关系矩阵
$D_{1\times N}$	各卫星负载集合
$\langle x,y\rangle$	卫星逻辑坐标<所处轨道面序号，轨道面内的序号>
h	卫星间逻辑距离

2.1 需求收集和地面数据网格化处理

本文采用全球人口和经济发展数据作为需求代理，用于构造具有空间异构性的地面业务压力背景，该需求代理不直接等价于真实卫星的接入流量，仅用于生成非均匀业务压力分布。

将地面被星座覆盖的区域均匀划分为等面积方形网格，为了保证用户与卫星波束间的对应关系，地面网格区域和波束覆盖区域均用正方形表示，地面区域集合映射到一维集合$G=\{g_1,g_2,\cdots,g_M\}$。

重力模型基于“相互吸引”的假设，认为两个区域之间的流量与各自的GDP值和人口数量成正比，与两地之间的距离成反比，可用式(1)表示。

$$d_{ij}=\frac{\mathrm{POP}_i^{\alpha}\cdot\mathrm{POP}_j^{\alpha}\cdot\mathrm{GDP}_i^{\beta}\cdot\mathrm{GDP}_j^{\beta}}{\mathrm{DIS}_{ij}^{\gamma}} \tag{1}$$

其中，α、β、γ为重力模型参数，具体取值根据国际电信联盟（ITU）*Measuring digital development: Facts and figures 2024*给出的典型国家/地区移动宽带流量数据，结合世界银行公布的人口、GDP及地理距离数据，对式(1)进行对数化处理后采用多元线性回归进行拟合获得[5]；$\mathrm{DIS}_{ij}^{\gamma}$为网格$i$与$j$中心点的球面距离。网格$g_i$的互联网流量需求为$d_i=\sum_{j=1}^{M}d_{ij}$，$i=1,2,\cdots,M$。需要强调的是，本文采用上述统计量主要表示在给定业务压力背景下形成的相对高负载区域，并非对所有实际卫星业务热点的普适预测，而是构造具有空间异构性和业务压力聚集特征的地面业务压力背景，用于比较不同路由算法在非均匀业务注入条件下的负载疏导能力。

随后，将该权重映射得到卫星节点负载，根据负载大小将卫星集合分为高需求节点S和普通节点；进一步地，将相邻高需求节点在卫星逻辑拓扑中连通并合并形成的区域称为高需求区域H，其余区域称为普通区域。本文高需求节点和高需求区域是给定业务压力背景下的相对负载概念，其作用在于为星间转发层路径绕行提供边界候选集合，而不是对真实卫星接入热点进行绝对判定。在实际系统中，该业务压力背景可由运营统计、接入负载观测或历史流量数据进行更新，以适应不同业务分布条件下的路由优化需求。

2.2 建立卫星与地面连接关系的数学模型

本文以不过极区的沃克德尔塔（Walker-Delta）星座作为验证平台[18]，其目的并非完整复现多层复杂巨型星座的接入与馈电体系，而是用于在可控拓扑下验证本文所提出的区域划分与星间分段路由框架。假设源节点与目的节点已经完成终端接入卫星选择及业务落地关联，后续数学模型只刻画非均匀业务压力映射到星间转发层时的负载均衡路由问题；多层可见卫星接入决策、层间路由及信关站馈电链路负载均衡属于更高层次的联合优化问题，不在本文当前建模范围内，将在后续工作中进一步研究。

通过卫星两行轨道根数（TLE）数据和时间，可以根据第四版简化通用摄动（SGP4）模型计算得到每个卫星的覆盖区域，如图1所示。将与高需求地面网格G存在覆盖关系的卫星作为候选承载卫星$S=\{s_1,s_2,\cdots,s_N\}$，地面覆盖网格用集合$\boldsymbol{C}_{N\times K}$表示，其中$K$为每个卫星的波束数量，波束集合$\boldsymbol{B}_s=\{b_{s,1},b_{s,2},\cdots,b_{s,K}\}$，单卫星频率带宽资源上限为$b_{\max}$。问题转化为计算$\boldsymbol{C}$与$\boldsymbol{B}$的三维连接关系矩阵$\boldsymbol{\delta}_{N\times K\times K}\in\{0,1\}$，$\boldsymbol{\delta}_{i\times j\times k}$的含义是在卫星$i$的覆盖区域内，地面区域$j$被波束$k$所覆盖。

值得注意的是，虽然$\boldsymbol{C}$与$\boldsymbol{B}$矩阵都是$N\times K$大小，但由于相邻卫星的覆盖区域之间存在重合区域，因此$\boldsymbol{C}$集合并集的大小小于$\boldsymbol{B}$集合，存在一个区域被多个波束覆盖的可能。如图2所示，两个卫星覆盖区域如菱形网状区域所示，若矩形网状区域

需求较大，两个卫星可以同时分配较多波束服务高需求区域，一个区域可能被多个波束覆盖。

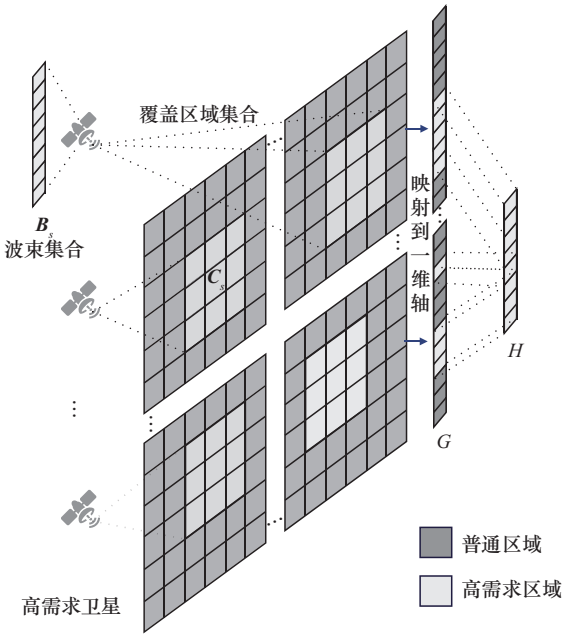


图1　卫星波束与地面连接关系数学模型

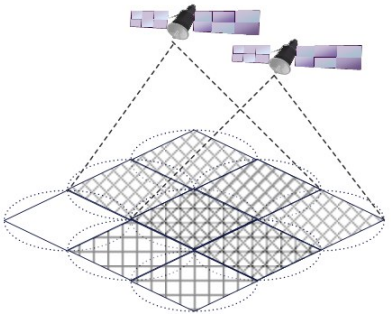


图2　一个区域可能被多个波束覆盖

2.3　业务压力驱动的高需求区域划分算法

本节将高需求区域划分问题转化为负载感知的波束分配优化问题：在满足覆盖和资源约束的前提下，调整波束分配使各卫星总负载尽量均衡，从而将需求的空间不均衡性映射为星座中高需求卫星与普通卫星的划分结果。

目标函数，卫星间负载均衡为：

$$\min \frac{1}{N}\sum_{i=1}^{N}\left(D_i-\frac{1}{N}\sum_{i=1}^{N}D_i\right)^2 \tag{2}$$

其中，D_i为单卫星负载，计算式为：

$$D_i=\sum_{j=1}^{K}\sum_{k=1}^{K}\boldsymbol{\delta}_{i,j,k}\boldsymbol{B}_{i,k} \tag{3}$$

每个卫星波束只覆盖一个区域：

$$\forall i\in[1,N],k\in[1,K],\sum_{j=1}^{K}\boldsymbol{\delta}_{i,j,k}=1 \tag{4}$$

每个区域至少连接一个波束：

$$\forall j\in[1,K],\sum_{i=1}^{N}\sum_{k=1}^{K}\boldsymbol{\delta}_{i,j,k}\geqslant 1 \tag{5}$$

连接关系矩阵由0和1组成：

$$\forall i\in[1,N],\forall j\in[1,K],k\in[1,K],\boldsymbol{\delta}_{i,j,k}\in\{0,1\} \tag{6}$$

单卫星负载小于频率带宽上限：

$$\forall i\in[1,N],D_i=\sum_{j=1}^{K}\sum_{k=1}^{K}\boldsymbol{\delta}_{i,j,k}\boldsymbol{B}_{i,k}\leqslant b_{\max} \tag{7}$$

每个区域所接收的频率带宽大于需求：

$$\forall j\in[1,K],\sum_{i=1}^{N}\sum_{k=1}^{K}\boldsymbol{\delta}_{i,j,k}\boldsymbol{B}_{i,k}\geqslant \boldsymbol{C}_{i,j} \tag{8}$$

对于上述混合整数非线性规划（MINLP）问题，由于连续变量与整数变量耦合，且目标函数非凸，难以直接采用标准凸优化方法求解。为此，本文采用启发式两阶段迭代优化方法。第一阶段进行可行初始分配，依据覆盖关系和区域需求占比，为各卫星波束分配服务区域与频谱资源，得到满足覆盖约束和带宽约束的初始解。第二阶段为局部调整阶段，以当前负载均衡目标函数作为评价指标，在相邻高需求卫星之间交换部分波束或带宽资源，当且仅当目标函数值下降且满足约束条件时调整。为避免过度迭代，当连续一轮不存在可接受调整动作或迭代次数达到上限时，算法终止。由于每次被接受的局部调整均使目标函数单调下降，且目标函数存在下界，因此算法在有限步内收敛。该方法通过在可行解空间中寻找局部改进来逼近最优解，兼顾求解质量与在线计算开销，具体实现如算法1所示。

算法1　业务压力驱动的高需求区域划分算法

初始化　波束集合$\boldsymbol{B}_s$、地面覆盖网格集合$\boldsymbol{C}_s$、连接矩阵$\boldsymbol{\delta}$、卫星负载D_s

1) 输入高需求区域H和网络需求d，高需求卫星S_N，频率带宽$b_{\max}$，波束数量K

// 第一阶段：基于覆盖和需求占比的初始分配

2) for i = 1:1:N

3)　计算卫星坐标和地面覆盖网格集合$\boldsymbol{C}_i$

4)　for j = 1:1:K

5)　　计算需求d_j在$\boldsymbol{C}_i$中的占比

6)　　分配波束数量和频率带宽$\boldsymbol{B}_{i,j}$

7)　　波束数量分配结果→$\boldsymbol{\delta}$

8)　　if $\boldsymbol{B}_{i,j}$ == 0

9)　　调配波束频率使每个区域都被覆盖
10)　else
11)　　将各卫星覆盖区内的网格需求累加后作为卫星负载
12)　end if
13) end for
14) end for
// 第二阶段：基于负载均衡的迭代优化
15) for i = 1:1:H
16)　while（更新值≤原值）
17)　　计算当前卫星负载均衡值
18)　　if $d_i > \boldsymbol{B}_{i,j}$
19)　　　调配波束频率使相邻卫星负载均衡
20)　　　更新卫星负载均衡值
21)　　end if
22)　end while
23) end for

完成负载感知的波束分配优化后，得到卫星负载集合D_s，设节点在基准源宿路径集中的路径穿越次数为T_i，定义节点i的转发压力为$P_i = \bar{D}_i + \bar{T}_i$，其中，$\bar{D}_i$表示归一化后的注入负载，$\bar{T}_i$表示归一化后的过境转发强度，同时反映节点的本地承载压力和中继承载压力。

在此基础上，将转发压力集合P_s降序排列，取前$\lceil \rho \cdot N \rceil$个节点标记为高需求节点，其中$\rho \in (0,1)$为高需求节点比例阈值。等价地，设$Q_{1-\rho}$为负载集合的$(1-\rho)$分位数，则卫星$i$被标记为高需求节点当且仅当$P_i \geqslant Q_{1-\rho}$。本文默认取$\rho = 0.04$，即负载排名前200的卫星节点被标记为高需求节点。该阈值并非任意设定，而是基于业务压力覆盖充分性与绕行开销之间的折中考虑。若阈值过低，则只能标记少量极端高压节点，难以形成空间连续的高需求区域，导致后续航点绕行缺乏足够的区域边界信息；若阈值过高，则高需求区域会过度扩张，压缩普通边缘节点的可选范围，并增加额外路径开销。

得到高需求节点集合后，在卫星逻辑拓扑上按4邻接关系提取其连通分量，并将每个连通分量定义为一个高需求区域。该结构既能保持在线计算开销较低，又能够为后续多级航点选择提供清晰的边界信息；进一步地，将与高需求区域中任一节点的曼哈顿距离为1的普通节点定义为边缘节点，边缘节点是后续进行航点选择的主要候选目标，星间链路采用激光或毫米波通信，传播时延极低[19]，所以后续路由策略采用逻辑坐标近似评估路径方向与跳数，以降低大规模拓扑下的在线计算复杂度。图3为5 000节点星座业务压力驱动的高需求区域划分结果。其中，黑色星号标记为高需求卫星节点，与深灰色方形节点共同组成高需求区域，高需求区域及其边缘节点共同构成后续结构化分段路径构造的输入。由此可见，本文高需求区域划分并不以获得复杂区域形状为目标，而是面向后续分段路由构造提取具有路由意义的高压区域边界。边缘节点作为高需求区域与普通区域之间的过渡节点，可为源侧快速退出、目的侧定向进入以及中间路径局部绕行提供候选航点，从而使区域划分结果能够直接服务于结构化分段路径构造。

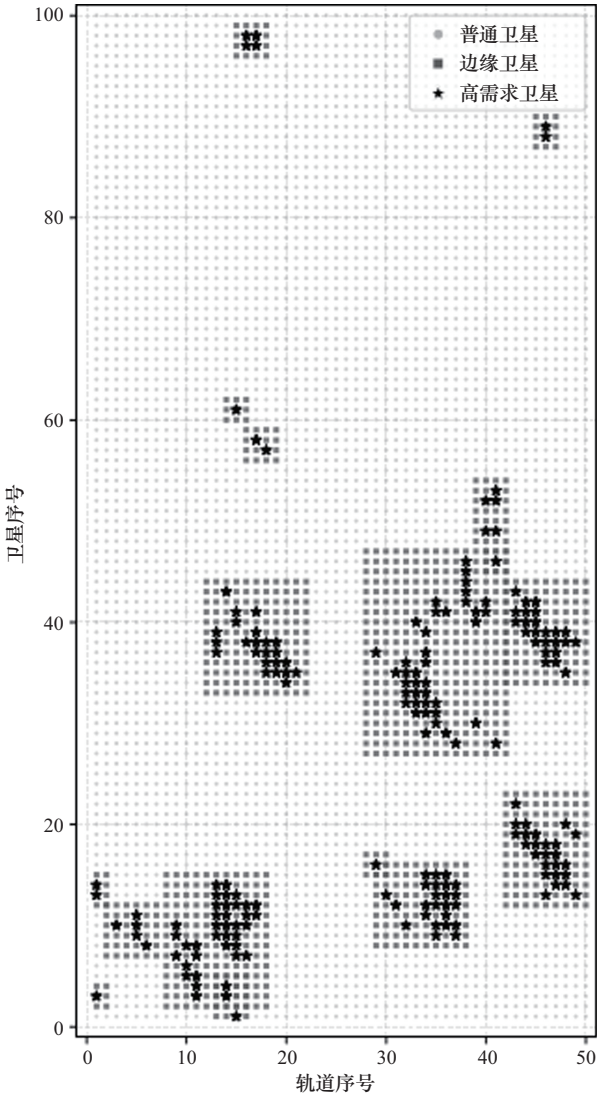


图3　5 000节点星座业务压力驱动的高需求区域划分结果

3　结构化分段路由和航点选择算法设计

3.1　分段路由标识设计

以业务压力驱动高需求区域划分模型输出的高需求区域集合以及节点类型为输入，为不同区域卫星分配SID。图4为本文的SID字段划分方案。

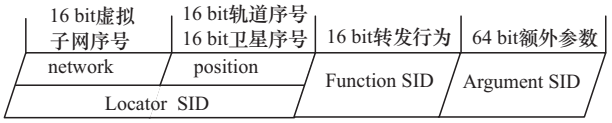


图4　SID字段划分方案

Locator SID：可分为逻辑网络network字段和地理位置position字段，network区分不同业务类型的虚拟子网，实现不同业务的网络切片，position用于标识节点的网络位置，同一颗卫星的Locator SID是固定的，由卫星轨道平面和编号决定。

Function SID：定义转发行为，每个节点接收到Locator SID指向自己的报文后，根据Function SID执行对应操作。针对不同类型的节点，具有不同的本地SID表，如表3所示。

Argument SID：可选参数字段，可携带服务质量需求（QoS）、业务优先级等信息。

3.2　一级航点选择算法

若源节点到目的节点的最短路径与高需求区域存在交叠，则业务流可能持续穿越该区域内部的链路和节点，进一步加剧区域内的链路竞争与节点拥塞。为此，本文在高需求区域划分的基础上，将源宿业务的分段路由过程表述为高需求区域协同的结构化分段路径构造问题，旨在保证源宿总体前进方向和端到端可达性的前提下，优先借助航点完成源侧快速退出、中间局部绕行以及目的侧定向进入，以降低业务流在高需求区域内部的停留和转发压力。

假设软件定义网络（SDN）控制器能实时获取网络节点和链路的状态，且源节点s_{src}和目的节点s_{dst}分别表示已完成终端接入关联和业务落地选择后的星间转发起止节点。节点已知自身所处地理坐标和逻辑坐标。控制器基于前文算法划分高需求区域后，为所有卫星节点创建SID本地表格，分配不同功能SID，为节点递交上来的数据包制定分段路由策略，选择恰当的航点，将报文按选定的航点路径传输到目的节点。原则是尽量绕开高需求区域，减少占用高需求区域卫星和链路的资源。由于大规模星座拓扑较为复杂，本文设计了分段路由多级航点选择算法，具体如下。

一级航点的作用是在保持路径朝向目的节点总体前进的前提下，以尽可能少的逻辑跳数离开高需求区域或进入目标高需求区域。由于本文后续路由以卫星逻辑坐标进行计算，因此任意两颗卫星s_i和s_j之间的逻辑距离定义为：

$$h\left(s_i,s_j\right)=\left|x_i-x_j\right|+\left|y_i-y_j\right| \tag{9}$$

对源节点所在高需求区域边界上的任一边缘节点s_e，记其逻辑坐标为$\left\langle x_e,y_e\right\rangle$。为了保证所选一级航点位于源节点指向目的节点的前向方向上，定义源侧方向一致性判据为：

$$\begin{aligned}g_{\mathrm{src}}\left(s_e\right)=&\left(x_e-x_{\mathrm{src}}\right)\left(x_{\mathrm{dst}}-x_{\mathrm{src}}\right)+\\&\left(y_e-y_{\mathrm{src}}\right)\left(y_{\mathrm{dst}}-y_{\mathrm{src}}\right)\end{aligned} \tag{10}$$

当$g_{\mathrm{src}}\left(s_e\right)>0$时，说明边缘节点$s_e$位于源节点指向目的节点的前向方向内，可作为离开当前高需

表3　Function SID解释

节点类型	Function SID	名字	描述
通用	::100	End	目的节点
	::1000 ::2000 ::3000 ::4000	End.X	从指定链路转发
边缘节点	::1	重建分段路由拓展报文头（rebuild SRH, RSRH）	作为一级航点时，判断是否需要添加二级航点，插入分段路由扩展报文头（SRH）中
	::2	段处理转发（SPF）	计算到下一个二级航点的路径
故障/拥塞节点	::3	重路由（reroute, RER）	重路由

求区域的候选一级航点。因此，源侧最近边缘节点定义为满足方向一致性条件的候选边缘节点中逻辑距离最小者，即$w_{\mathrm{src}}^{*}=\operatorname{argmin}\left(h\left(s_{\mathrm{src}},s_e\right)\right)$。

若源节点本身就是边缘节点，则直接取$w_{\mathrm{src}}^{*}=s_{\mathrm{src}}$。

同理，对目的节点所在高需求区域边界上的任一边缘节点s_e，定义目的侧方向一致性判据为：

$$g_{\mathrm{dst}}\left(s_e\right)=\left(x_e-x_{\mathrm{dst}}\right)\left(x_{\mathrm{src}}-x_{\mathrm{dst}}\right)+\left(y_e-y_{\mathrm{dst}}\right)\left(y_{\mathrm{src}}-y_{\mathrm{dst}}\right) \tag{11}$$

当$g_{\mathrm{dst}}\left(s_e\right)>0$时，说明该边缘节点位于目的节点朝源节点的前向方向上，可作为进入目标高需求区域的候选一级航点。因此，目的侧最近边缘节点定义为满足方向一致性条件的候选边缘节点中逻辑距离最小者，即$w_{\mathrm{dst}}^{*}=\operatorname{argmin}\left(h\left(s_{\mathrm{dst}},s_e\right)\right)$。

若目的节点本身就是边缘节点，则直接取$w_{\mathrm{dst}}^{*}=s_{\mathrm{dst}}$。

由此可见，一级航点对应两个局部边界决策：一是在源节点所在高需求区域边界上选择出口点，使业务流能够以较少的跳数尽快离开当前高需求区域；二是在目的节点所在高需求区域边界上选择入口点，使业务流能够沿总体前进方向进入目标高需求区域，二者共同构成源宿路径的外层分段框架。据此，一级航点选择如图5所示。

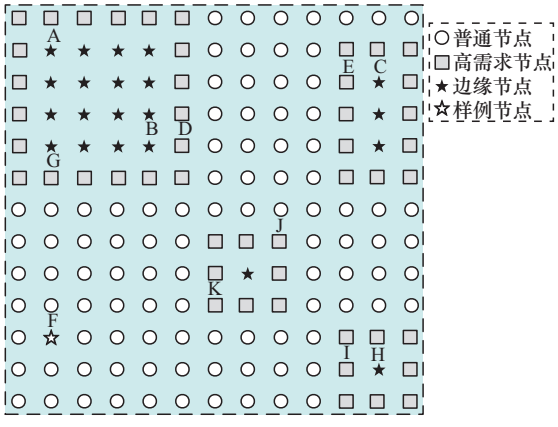


图5　一级航点选择示例

（1）若源节点和目的节点同处普通区域（A→B），或同处同一高需求区域，则不需要设置一级航点。

（2）若源节点位于高需求区域内而目的节点位于普通区域（A→F），源侧出口航点G作为w_{src}^{*}。

（3）若源节点位于普通区域而目的节点位于高需求区域内（F→H），目的侧入口航点I作为w_{dst}^{*}。

（4）若源节点和目的节点分别位于两个不同的高需求区域（B→C），则同时设置源侧出口航点D作为w_{src}^{*}和目的侧入口航点E作为w_{dst}^{*}。

一级航点的候选集合由2.3节得到的高需求区域边缘节点构成，因此一级航点选择并非在全网范围内任意搜索中间节点，而是在与当前源宿业务相关的区域边界上进行局部决策。一级航点选择算法如算法2所示，由源节点、目的节点的地址推算出各自的坐标后，遍历当前存在的所有高需求区域边界范围，若源节点或目的节点位于高需求区域内，则需要在二者互相的方向上找到距离所处高需求区域边界上最近的点，以最少的跳数走出高需求区域，避免在高需求区域过多停留造成不必要的时延和资源占用。

算法2　一级航点选择算法

初始化　段列表SegmentList，w_{src}^{*}，w_{dst}^{*}

1) 根据源节点和目的节点逻辑坐标$\langle x_{\mathrm{src}},y_{\mathrm{src}}\rangle$、$\langle x_{\mathrm{dst}},y_{\mathrm{dst}}\rangle$确定源宿转发方向；
2) 遍历高需求区域，如果源节点位于区域方框rectangle内，则执行源侧出口航点选择；
3) 遍历rectangle边界上的全部边缘节点s_i，通过式(10)计算s_i源侧方向一致性判据；
4) 通过式(9)对每个$g_{\mathrm{src}}\left(s_e\right)>0$的边缘节点计算跳数距离，取距离最小节点作为$w_{\mathrm{src}}^{*}$；
5) 结束当前节点和区域遍历；
6) 遍历高需求区域，如果目的节点位于区域rectangle内，则执行目的侧出口航点选择；
7) 遍历rectangle边界上的全部边缘节点s_i，通过式(11)计算s_i源侧方向一致性判据；
8) 通过式(9)对每个$g_{\mathrm{dst}}\left(s_e\right)>0$的边缘节点计算跳数距离，取距离最小节点作为$w_{\mathrm{dst}}^{*}$；
9) 结束当前节点和区域遍历；
10) 将w_{src}^{*}和w_{dst}^{*}对应的一级航点段加入SegmentList；
11) 将目的节点的End SID加入SegmentList尾部；
12) 返回一级航点序列SegmentList。

3.3　二级航点选择算法

一级航点主要解决源侧退出和目的侧进入的问题，但一级航点之间的中间路径仍可能与其他高需求区域相交，如图6所示。若继续沿原有路径转发，业务流仍会经过中间高需求区域，从而削弱一级航点的绕行效果。为此，经过一级航点的选择后，在SegmentList中加入目的节点END、一级航点的RSRH，在一级航点基础上进一步引入二级航点，对中间仍与高需求区域相交的路径段进行补充绕行。本文采用角点边缘节点构造方式，将二级航点的搜索范围限制在与当前路径相交的高需求区域外侧边界附近，以较小的附加逻辑跳数绕开高需求区域。因此，二级航点并不是对源宿路径任意插入的附加中间节点，而是针对一级分段后仍与高需求区域相交的局部路径段进行补充修正，其作用是进一步减少业务流穿越高需求区域内部链路的概率。

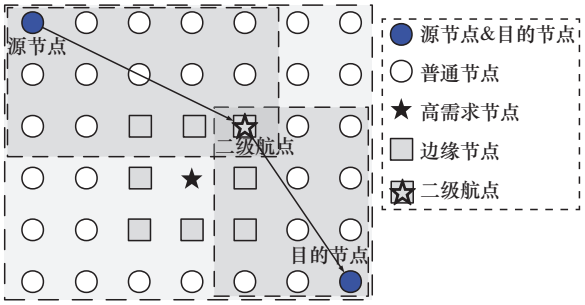


图6　二级航点选择示例

设当前节点为s_{cur}，下一目标航点为s_{nxt}，其逻辑坐标分别记为$\boldsymbol{u}_{\text{cur}}$和$\boldsymbol{u}_{\text{nxt}}$。当前路径方向向量为$\boldsymbol{v}_{\text{dir}}=\boldsymbol{u}_{\text{nxt}}-\boldsymbol{u}_{\text{cur}}$，对于第$m$个与当前路径相交的高需求区域$R_m$，记其4个角点边缘节点的逻辑坐标分别为对任一角点$\boldsymbol{u}_m^r$，定义从当前节点指向该角点的候选方向向量为$\boldsymbol{v}_m^r=\boldsymbol{u}_m^r-\boldsymbol{u}_{\text{cur}}$。则该角点相对于当前转发方向的偏转角定义为：

$$\varphi_m^r=\arccos\left[\frac{\boldsymbol{v}_{\text{dir}}\cdot\boldsymbol{v}_m^r}{\|\boldsymbol{v}_{\text{dir}}\|\cdot\|\boldsymbol{v}_m^r\|}\right] \tag{12}$$

选取偏转角最大的两个角点构成候选集合$\boldsymbol{C}_m$。设当前路径与K个高需求区域相交，从每个候选集合中选取一个角点，可形成候选序列$\boldsymbol{W}=(w_1,w_2,\cdots,w_K)$。定义序列$\boldsymbol{W}$的后续可行路径数为：

$$N_{\text{path}}(\boldsymbol{W})=\prod\left|P\left(w_m,w_{(m+1)}\right)\right| \tag{13}$$

其中，$P(a,b)$表示从节点a到节点b且不穿越高需求区域内部的可行逻辑路径集合，$\left|P(a,b)\right|$表示该集合中的路径条数。于是，二级航点序列选择为$\boldsymbol{W}^*=\text{argmax}N_{\text{path}}(\boldsymbol{W})$。

二级航点不仅要求位于有利于当前绕行的方向上，而且应当为后续路径保留更大的继续前进空间。当首个二级航点确定后，若当前节点到该二级航点的连线仍与其他高需求区域相交，则以该二级航点作为新的目标节点重复上述过程，直到当前节点到下一目标节点的连线与任何高需求区域均不相交为止。二级航点选择算法如算法3所示。

算法3　二级航点选择算法

初始化　newSegmentList, updated SL

1) 根据s_{cur}和s_{nxt}的逻辑坐标计算当前路径方向向量；
2) 判断当前线段$(s_{\text{cur}},s_{\text{nxt}})$是否与任一高需求区域相交；若不存在相交区域，则结束本次二级航点选择newSegmentList←SegmentList；
3) 按距离遍历相交的高需求区域R_m，取其4个角点边缘节点$\boldsymbol{u}_m^1$、$\boldsymbol{u}_m^2$、$\boldsymbol{u}_m^3$和$\boldsymbol{u}_m^4$；
4) 根据式(12)计算每一个角点边缘节点s_m^r相对于当前路径方向的偏转角，保留最大值的角点边缘节点，构成该区域的候选点集合C_m；
5) 继续遍历其余相交高需求区域，构成候选二级航点序列$\boldsymbol{W}$；
6) 根据式(13)计算$\boldsymbol{W}$的后续可行路径数；
7) 在全部候选序列中，选择使$N_{\text{path}}(\boldsymbol{W})$取得最大值的序列作为最优二级航点序列$\boldsymbol{W}^*$；
8) 将最优二级航点序列$\boldsymbol{W}^*$依次插入SegmentList，得到newSegmentList和updated SL；
9) 依次检查新增的相邻节点对连线是否仍与高需求区域相交；若存在相交情况，则以该相邻节点对作为新的当前段，重复步骤1)至步骤8)。

完成二级航点的选择后，将一二级航点路径依次倒序导入SRH中，剩余段数SL指向下一航点，各相邻航点间使用最短路径路由，节点通过接收到报文的SRH头部，执行对应的指令操作，即可完

成数据报文的端到端传输。接下来每到达一个一级航点，即进行一次二级航点的选择。

仿真过程中网络拓扑更新采用时间切片方案，每个时间片重新划分一次高需求区域。若在数据传输过程中，故障或拥塞节点通过以下机制被动态检测到，也将触发高需求区域重划分和航点重选。

（1）链路故障检测：相邻节点间通过周期性Hello报文监测链路状态。若节点i连续n_{miss}个Hello周期未收到邻居j的应答，则判定链路(i,j)故障，节点i向SDN控制器发送故障信令。

（2）拥塞检测：节点i持续监测其输出队列占用率q_i，若q_i连续n_{cong}个采样周期超过拥塞阈值θ_q，则节点i向SDN控制器上报拥塞状态。

（3）高需求重标记：SDN控制器收到故障或拥塞信令后，将相关节点临时标记为高需求节点，加入高需求区域H，并更新高需求区域集合。

（4）航点重选触发：高需求区域更新后，控制器检查当前在途数据包的后续航点是否落入新的高需求区域。若存在冲突，则为受影响的数据流重新执行算法2和算法3，生成新的SegmentList并通过控制信令下发至源节点或当前转发节点。

（5）恢复与退出：当故障链路恢复或拥塞缓解，队列占用率q_i连续n_{cong}个周期低于θ_q后，控制器在下一个拓扑更新时隙将临时标记的高需求节点移出高需求区域H，恢复为普通节点。

上述结构化分段路径构造机制说明，本文多级航点分段路由并不是一次性静态生成的分段路径，而是能够随高需求区域变化进行局部更新的动态转发过程。当一级或二级航点的后续路径重新落入新的高需求区域时，控制器仅需对受影响的局部路径段重新执行航点选择，而不需要对全网业务流统一进行全局最短路重计算。若当前路径段不存在满足条件的二级航点，则退化为仅使用一级航点；若一级航点也不需要设置，则继续采用最短路径转发。

以图5中节点A到节点H的传输为例，首先根据算法确认一级航点为G和I，源节点判断到一级航点G之间不需要添加二级航点，便直接通过最短路径路由转发到G，G判断到下一级航点I之间需要添加二级航点K以绕开高需求区域，故在SegmentList中插入二级航点，并经K绕行，后续节点均判断不需要添加额外航点，于是逐段递减SegmentLeft，直到到达目的节点H。高需求区域协同的结构化分段路由数据包传输流程如图7所示。

3.4 算法复杂度分析

本节对本文算法的计算复杂度进行分析，以评估其在大规模星座中的可扩展性。

算法1的计算开销来自初始分配和负载均衡迭代。初始化分配需要遍历所有高需求卫星N及其波束K，因此时间复杂度为$O(N\cdot K)$。负载均衡优化需要遍历所有高需求区域$|H|$并进行迭代调整。由于$|H|$远小于N，且优化过程通常快速收敛，算法1的整体时间复杂度约为$O(N\cdot K+|H|)\approx O(N)$，与高需求卫星节点数量呈线性关系。该算法通常在管控中心周期性地执行，其计算开销是可接受的。

算法2需要遍历所有高需求区域$|H|$，其时间复杂度$O(|H|)$与高需求区域数量线性相关。算法3最坏情况下需要遍历所有相交区域的候选角点并评估有限组合。由于高需求区域经合并后数量有限，且候选点数量为常数，因此算法3的时间复杂度可视为$O(|H|)$或与$|H|$相关的低阶多项式。

上述分析说明，本文算法与仅依赖固定象限规则的QMSR相比，额外增加了业务压力驱动的高需求区域更新与局部航点筛选开销，但其主要计算压力集中在高需求区域及受影响路径段，而不需要像负载感知迪杰斯特拉（Dijkstra）最短路径（LA-DSP）和拥塞感知自适应路由（CAAR）那样在每个更新周期依据全网链路负载或拥塞状态重新构造加权代价并执行全局最短路计算，也不需要像优化链路状态路由算法（OLSR）那样依赖全网链路状态收敛，因而在5 000节点的大规模低轨星座场景下仍具有较好的可扩展性和在线执行可行性。本文算法并非通过无约束叠加模块来提升性能。高需求区域划分、多级航点选择和故障/拥塞重标记均具有明确触发条件，计算范围主要集中在高需求区域及受影响路径段，而不需要持续进行全网重计算。

4 实验结果与分析

为验证所提高需求区域协同的结构化分段路由策略的可行性与有效性，本文采用Ubuntu 22.04操

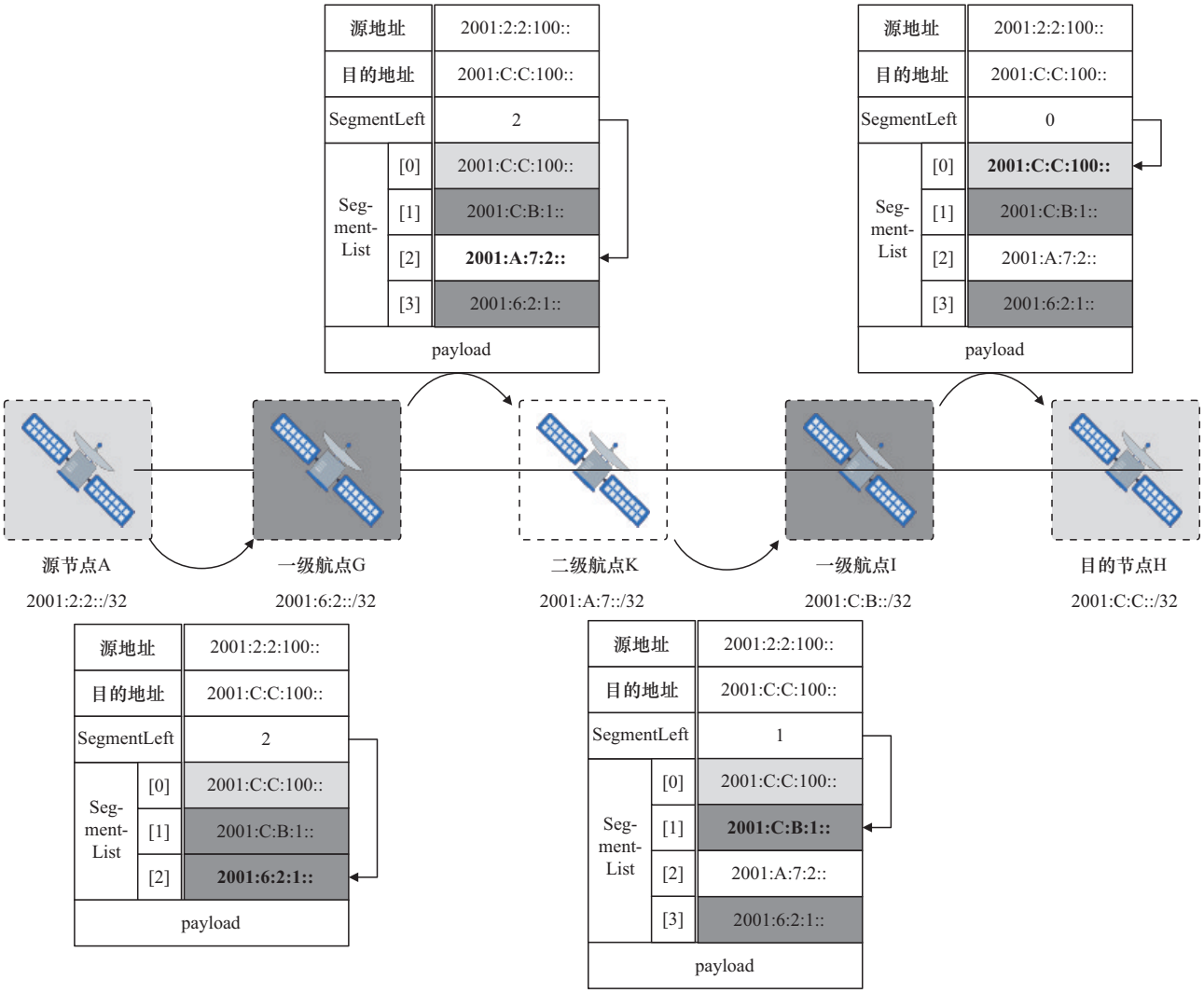


图 7　高需求区域协同的结构化分段路由数据包传输流程

作系统，在网络仿真器 NS-3[20] 平台上搭建了 Walker-Delta 星座拓扑，如图 8 所示，卫星节点数量为 5 000 个，并考虑反向缝的影响，用于构造大规模低轨星座下的高负载转发场景，以验证本文算法在大规模拓扑和非均匀业务压力叠加条件下的性能表现。每颗卫星连接一个地面节点，用于模拟其覆盖区域聚合后的业务注入量。需要说明的是，该地面节点仅表示汇聚后的流量需求，而不等价于实际终端接入过程。本文采用这种设置的目的是控制接入与落地因素影响，单独考察星间转发层绕行机制的有效性，从而验证本文算法在骨干承载层面的负载均衡能力。实验中的高需求区域业务注入由需求权重映射得到，以模拟真实场景中高需求区域因用户密集接入导致的流量聚集效应，链路拥塞将由流量竞争自然产生。

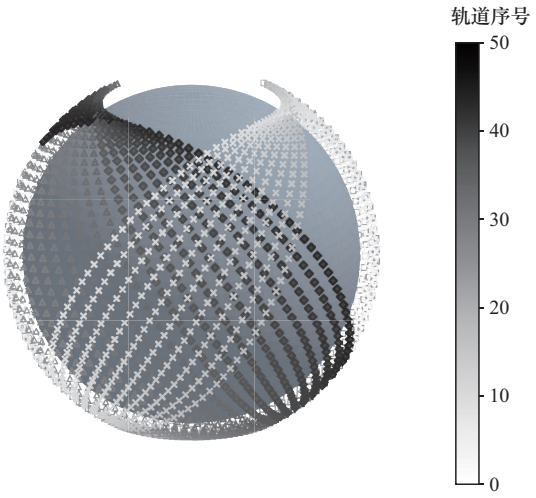


图 8　Walker-Delta 星座拓扑

表 4 为主要仿真参数，用于验证不同算法在相同空间非均匀压力背景下的负载疏导能力。在

存在初始负载的情况下，选取5类具有代表性的对比算法，具体包括QMSR[16]、LGSR[11]、OLSR[21]、LA-DSP[22]以及CAAR[23]，分别作为静态划分的分段路由、骨架图分层分段路由、链路状态路由、全局重赋权最短路由和拥塞感知自适应路由的代表。QMSR凸显动态需求感知划分相较于静态几何划分的优势；LGSR补充了与骨架图分层SR思路的对比；OLSR作为领域内经典性能基准，用于评估本文算法相对经典链路状态路由的性能差异；LA-DSP和CAAR是链路级代价调整，与本文算法具有显著差异，用于比较仅做链路代价感知与显式进行高需求区域绕行两类思路之间的差异。所有动态路由算法置于统一控制假设下，所有算法均面对相同的业务压力背景，除算法本身要求的特有参数外，其余链路带宽、缓存长度、分组大小、仿真时长等参数完全一致，使用了4个指标来评估性能：端到端时延、吞吐量、丢包率和流量分布指数（traffic distribution index, TDI）。

表4 主要仿真参数

配置项	取值
轨道数	50
单条轨道上卫星数	100
卫星轨道高度/km	550
星间/星地链路带宽/(Mbit·s^{-1})	100
数据包大小/B	1 400
缓存队列长度/packet	100
状态更新时隙/s	0.1
总仿真时长/s	60
TDI采样间隔/s	1
源宿节点对数量	50
业务流流量类型	UDP CBR
发送间隔/ms	25:5:55
高需求/普通区域注入方式	按需求权重映射
随机种子	2 025~2 034
重复实验次数	10
统计窗口/s	5-55

4.1 平均端到端时延

不同路由算法的平均端到端时延比较如图9所示。从图9可以看出，随着数据包发送间隔缩小，网络负载逐渐升高，各算法的端到端时延均有不同程度的增加。LA-DSP引入链路负载感知代价，能够在一定程度上缓解路径集中转发造成的局部拥塞；CAAR利用链路拥塞状态进行自适应调整，整体时延低于多数基线算法。相比之下，本文算法在各负载区间内保持较低端到端时延。这是因为，LA-DSP和CAAR主要依赖链路代价调整来被动规避拥塞，通常在拥塞形成后再改变路径；本文算法能够先显式识别高需求区域，再通过结构化分段绕行减少数据流在高需求区域内部的停留时间，因此在大规模星座和非均匀业务压力叠加的场景下具有更稳定的时延优势。

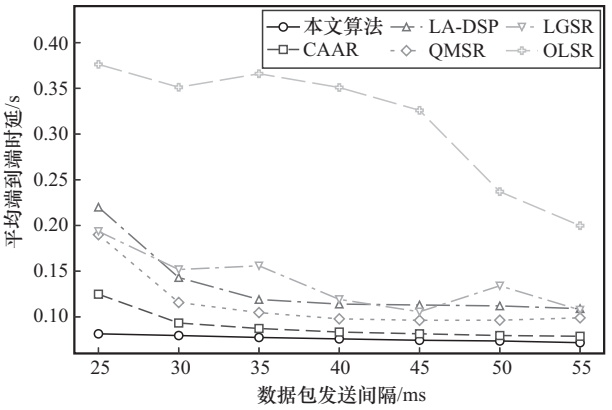


图9 不同路由算法的平均端到端时延比较

4.2 吞吐量和丢包率

不同路由算法的吞吐量和丢包率比较分别如图10和图11所示。随着数据包发送间隔缩小，网络负载持续升高，各算法的丢包率普遍上升，而吞吐量在高负载区间逐步趋于饱和。LA-DSP相较于OLSR具有更高的吞吐量和更低的丢包率，说明基于链路负载信息进行全局路径重赋权能够改善高需求链路上的竞争压力。CAAR则利用拥塞状态反馈实现自适应路径调整，其吞吐量和丢包率表现总体优于LA-DSP，并在中高负载区间逼近QMSR和LGSR。本文算法保持了最高吞吐量和最低丢包率，表明其不仅能够减少高需求链路上的排队竞争，还能够通过引导机制使流量更早退出高压区域，从而降低局部拥塞扩散带来的性能损失。

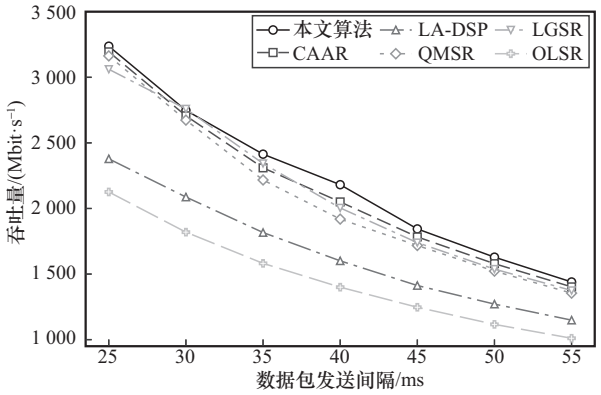


图10　不同路由算法的吞吐量比较

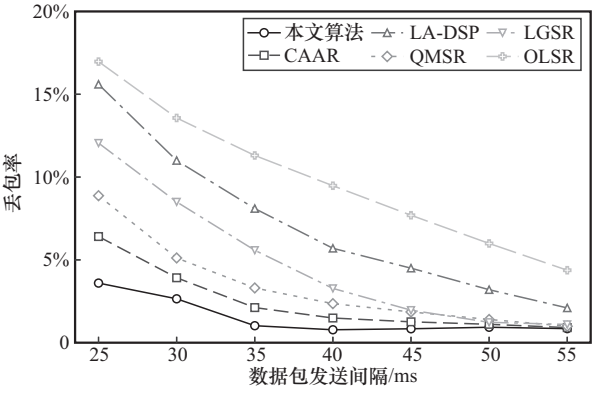


图11　不同路由算法的丢包率比较

4.3　流量分布指数

为了具体量化网络中流量的分布情况，引入流量分布指数。

$$\mathrm{TDI}=\frac{\left(\sum_{i=1}^{l}p_i\right)^2}{m\left(\sum_{i=1}^{l}p_i^2\right)} \tag{14}$$

其中，l表示仿真中所有建立的星间链路（ISL）总数，p_i表示第i条链路上统计周期内转发的数据包个数。TDI取值范围为0~1，越接近1表示网络中流量越均匀，路由负载均衡效果越好。

不同路由算法的流量分布指数比较如图12所示。在不同网络负载程度下，LA-DSP和CAAR均取得了较高的TDI，说明负载感知最短路和拥塞感知自适应路由均可在一定程度上提升网络流量分布的均衡性。其中，CAAR整体优于LA-DSP，表明拥塞反馈机制有助于抑制局部链路过载。但这类算法主要基于链路级代价调整，难以显式处理空间上连续分布的高需求区域。本文算法能够从区域层面识别高需求区域，通过结构化分段路径主动绕开高需求区域，因此在全网流量疏导和负载均衡方面更具优势。

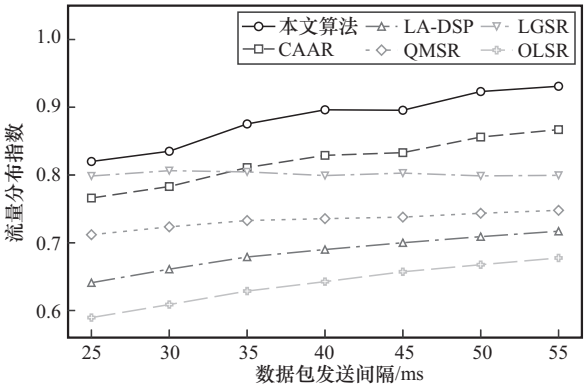


图12　不同路由算法的流量分布指数比较

由于建模所依据的地面流量需求本身具有地理不均衡性，且在给定的星座拓扑与区域划分约束下，路径选择的自由度有限。因此，本仿真场景下的TDI未能达到极限值1，但仍能说明本文算法在给定拓扑与需求约束下取得了更优的流量均衡效果。

4.4　抗毁性能分析

由于太空中干扰因素众多，卫星节点在运行过程中时常发生故障[24]。为了验证本文算法的抗毁性能，在网络动态运行过程中引入了不同程度的故障场景进行测试。单链路故障：在随机选取的10组源–目的节点对的数据传输路径上，在仿真稳定阶段切断一条关键链路；多链路并发故障：在相同条件下，同时切断数据路径上多条不相邻的链路，模拟局部严重干扰。对所有动态算法采用相同的故障触发时刻、故障位置和状态同步流程，并比较端到端时延变化和路由恢复时间。

在链路断开的情况下不同路由算法对端到端时延影响的比较如图13所示。在固定源宿节点和其他仿真参数的条件下，仿真过程中在路由稳定阶段切断数据传输路径上的一条链路。图13曲线中断区间表示故障触发后的业务接收中断。QMSR出现较长的业务中断，说明其依赖周期性路径更新，在拓扑突变场景下恢复较慢；OLSR通过邻居失效判定和链路状态重新收敛完成路径切换，时延峰值明显升高且恢复过程较长；LGSR在故障后完成路径重选，恢复表现整体处于本文算法与传统基线算法之间；LA-DSP通过负载加权后的全局最短路径重选实现故障后恢复，但故障后仍表现出较明显的恢复滞后和时延波动；CAAR根据链路拥塞状态动态调整转发代价，故障后的恢复速度和稳定性均优于LA-DSP和多数传统基线算法，但仍弱于本文算

法。本文算法在连续丢失Hello报文并触发故障检测后，将相关链路邻接区域临时标记为高需求区域，为受影响数据流重选航点并绕行故障区域，表现出更好的抗毁性能。

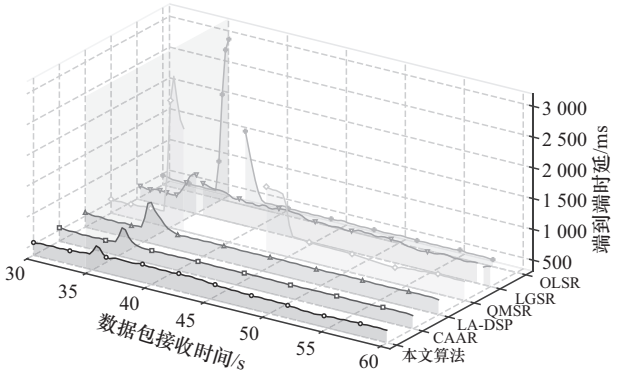


图13　在链路断开的情况下对端到端时延影响的比较

为进一步评估算法在更严苛故障下的性能，在多链路故障场景下路由恢复时间的比较如图14所示。在多链路故障场景下，随着并发故障链路数增加，各算法的平均路由恢复时间均有所增大，但增幅存在明显差异。本文算法在不同故障规模下始终保持最短恢复时间；CAAR次之；LA-DSP在少量故障场景下优于部分传统基线，但随着并发故障规模增加，其恢复时间增长更快；LGSR和OLSR恢复较慢；QMSR由于依赖周期性更新，在多链路故障场景下恢复时间显著偏大。这说明本文算法通过动态区域划分快速识别故障链路影响区域，并为后续数据包重新选择绕行路径，因此在故障范围扩大时仍具有较好的快速重路由能力。

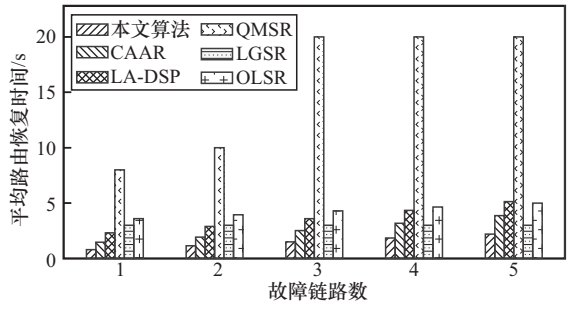


图14　在多链路故障场景下路由恢复时间的比较

综上所述，本文算法在不同严重程度的网络故障下，均能实现快速的路由收敛，说明其在动态故障环境中具有更强的快速重路由能力，表现出良好的抗毁性与鲁棒性。

4.5　算法运行开销分析

为验证本文算法在大规模星座场景下的算法侧可执行性，对各个算法的关键运行开销进行了统计与比较。定义如下4项指标：①控制面预处理耗时指一次完整的拓扑预处理或分区计算所需时间；②单流转发决策耗时指在预处理完成后，为单个源–目的节点对计算一次转发决策或查表所需时间；③平均SID栈深度指SegmentList中平均SID条目数；④平均SRH头部开销指每包因SRH引入的平均附加字节数。不同算法运行开销对比如表5所示。

表5　不同算法运行开销对比

算法	控制面预处理耗时/ms	单流转发决策耗时/μs	平均SID栈深度	平均SRH头部开销/B
本文算法	10.4	50.5	2.0	39.7
QMSR	0.9	0.4	1.7	35.8
LGSR	1.2	95.1	4.6	82.0
OLSR	12.3	0.3	1.0	0
LA-DSP	5.5	5.5×10^3	1.0	0
CAAR	7.1	5.7×10^3	1.0	0

从表5可以看出，在控制面预处理方面，本文算法耗时与OLSR处于同一量级，虽高于QMSR与LGSR以及LA-DSP、CAAR，但该过程只需在SDN控制器集中执行一次即可完成全网区域划分，且耗时远小于仿真时隙100 ms，可在单周期内完成更新。单流转发决策方面，本文算法决策耗时高于QMSR和OLSR的逐跳查表，但明显低于需要依据链路负载或拥塞状态重算全局加权最短路的LA-DSP和CAAR，并低于LGSR。与此同时，本文算法平均SID栈深度仅略高于QMSR，显著低于LGSR，说明在完成高需求区域绕行时仅需插入少量附加段；对应的平均SRH头部开销与QMSR接近，明显低于LGSR。综上，本文算法在预处理开销可控、单流决策复杂度适中、头部开销较低的条件下，具备大规模星座场景下的算法可执行性。

进一步地，从多模块组合后的工程可实现性角度看，本文算法的额外开销主要集中在控制面预处理阶段，数据面仅引入少量SID封装开销，并未显著增加逐包转发负担。结合上述运行开销结果可知，本文算法在性能提升与系统开销之间保持了相对合理的权衡，具备大规模星座场景下算法的可执行性和一定的工程可实现性。

4.6　算法消融实验

为更清晰地分析本文算法中高需求区域提取与结构化分段路由框架的作用，设计如下对照与消融实验。其中，A0 表示静态均匀分区和一级航点选择，其中仅利用静态均匀分区边界构造一级航点候选集，而一级航点的选择准则与本文算法保持一致；M1 表示业务压力驱动的动态高需求区域划分（算法 1）；M2 表示多级航点选择（算法 2 和算法 3）；M3 表示故障后高需求重标记机制（3.3 节）。实验以不引入负载感知的最短路径算法（DSP）为基础，叠加不同模块，观察不同区域结构和路由机制对整体性能的影响。

算法消融实验结果如图 15 所示。在 DSP 基础上引入 A0 后，各指标仅有小幅改善，其中端到端时延小幅下降，吞吐量和 TDI 波动不大。这说明在不引入业务压力驱动信息时，仅依赖静态均匀分区边界所形成的一级航点机制只能提供有限的路径分散能力，对高需求路径的引导作用较弱。将静态均匀分区替换为业务压力驱动的高需求区域划分后，端到端时延和 TDI 明显改善，说明需求感知的区域划分能够有效识别负载高需求并为后续路由提供有价值的拓扑信息；在 M1 基础上叠加二级航点选择，端到端时延和丢包率进一步下降，表明主动绕行高需求区域的航点机制对路径质量有显著改善；加入故障/拥塞触发的高需求重标记与重路由机制后，各指标均有进一步提升，说明动态故障感知机制能够在拓扑变化时及时调整路由策略。各模块均表现出正向增益，组合使用时整体性能最优，验证了本文算法设计的合理性。

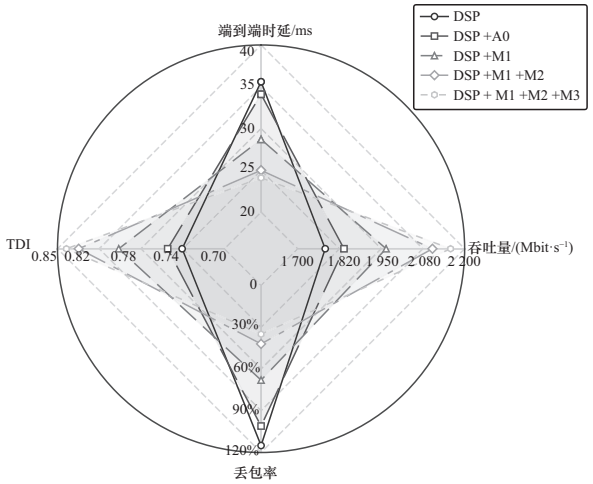


图 15　算法消融实验结果

为进一步说明不同模块组合带来的运行开销，本文对消融实验中各模块组合的控制面预处理耗时、单流转发决策耗时、平均 SID 栈深度和平均 SRH 头部开销进行了统计，结果如表 6 所示。

表 6　不同模块组合下的运行开销对比

模块组合	控制面预处理耗时/ms	单流转发决策耗时/μs	平均 SID 栈深度	平均 SRH 头部开销/B
DSP	0	0.7	1.0	0
DSP+M1	10.4	7.0	1.2	27.9
DSP+M1+M2	10.4	50.5	2.0	39.7
DSP+M1+M2+M3	10.4	50.5	2.0	39.7

由表 6 可知，引入 M1 后，控制面预处理耗时增至 10.4 ms，主要来自高需求区域划分和边界节点提取；在 M1 基础上加入 M2 后，单流转发决策耗时增至 50.5 μs，主要来自多级航点选择；加入 M3 后，各项开销与 M1+M2 组合基本一致，表明故障/拥塞触发重标记机制主要在异常事件发生时执行局部更新，不会在常规转发过程中持续增加数据面封装开销。完整方案的控制面预处理耗时低于 100 ms 仿真时隙，单流转发决策耗时保持在微秒级。上述结果表明，消融实验中的性能提升并非依赖无约束增加模块获得，而是在开销可控的条件下实现的。

4.7　阈值敏感性分析

为说明高需求节点阈值设置的合理性，本文进一步对高需求比例阈值 ρ 进行了敏感性分析。

阈值敏感性分析实验结果如图 16 所示。随着 ρ 由 2% 增大至 4%，网络吞吐量和 TDI 都有所提升，说明适当增大高需求节点比例有助于更充分地识别高负载区域，并为后续结构化分段路径构造提供更有效的区域边界信息。与此同时，平均额外跳数增幅相对平缓，表明该阶段由绕行带来的路径代价仍处于可接受范围内。当 ρ 继续增大至 5% 和 6% 时，吞吐量和 TDI 均有所回落，而平均额外跳数则进一步上升。这表明当阈值过大时，高需求区域会出现过度扩张，压缩普通边缘节点的可选范围，从而增加不必要的绕行开销并削弱负载均衡收益。这是由于 ρ 过小时只能标记少量极端高负载节点，难以绕行星座中必要的高需求区域；ρ 过大又会使高需求区域占比过高，降低分段绕行的灵活性。综合来看，ρ 取 4% 时在高需求节点识别充分性与路径绕行代价之间取

得了较好的平衡，因此本文在5 000节点星座中选取前200个节点作为高需求节点是合理的。

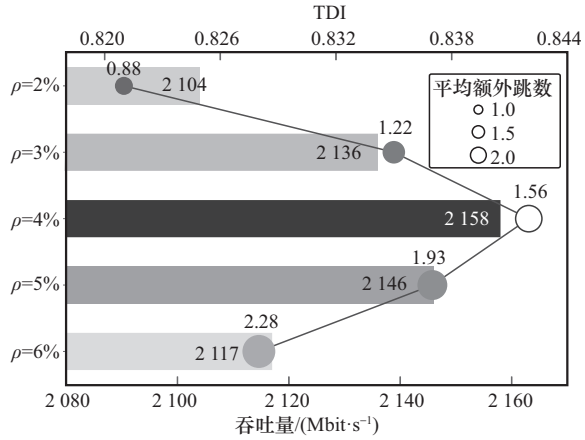


图16　阈值敏感性分析实验结果

需要说明的是，4%阈值是本文5 000节点星座和当前业务压力背景下的场景化取值，并不表示该比例在所有星座规模和业务分布下均为最优。实际系统中，可根据节点规模、业务负载、链路状态和控制面资源对高需求比例阈值进行动态调整。

5　结束语

本文面向星地接入关系与业务落地关系已知的星间转发层场景，提出了一种高需求区域划分与结构化分段路由算法。工作的重点不在于求解完整的端到端低轨卫星路由问题，而在于验证在动态拓扑和非均匀业务压力下，通过高需求区域识别与结构化分段路径构造改善星间转发层负载均衡与抗毁性能。

首先构建业务压力驱动的高需求区域划分模型，将空间异构业务压力映射为卫星节点负载，并据此划分高需求区域和普通区域。根据分配结果对每个卫星分配不同转发行为的SID，设计了SID框架和分段路由实现方法，最后将源宿业务的分段路由过程表述为高需求区域协同的结构化分段路径构造问题，并在此基础上完成分层航点决策。仿真结果显示，在相同条件下与传统路由算法相比，本方案在端到端时延、吞吐量、丢包率和流量均衡分布方面具有优势，在应对网络出现的突发故障时，均能实现更快的路由切换与恢复，说明其在动态故障环境中具有更强的快速重路由能力。

未来工作可以从以下几个方面展开：一是研究接入卫星选择、落地卫星/信关站选择与星间转发的跨层联合优化机制，将本文的转发层绕行方法扩展为端到端的负载均衡框架；二是引入真实业务观测数据，对需求模型进行校准；三是在多层星座、连续级联故障和不完美状态信息条件下进一步验证算法性能；四是研究复杂度约束下的自适应模块启用、分布式或增量式高需求区域更新和轻量化部署策略，以进一步降低超大规模星座下的控制面开销。

参考文献：

[1] Chen C, Sun C X, Li H M, et al. ST-GAGCN-LEO: a spatiotemporal graph attention and gated convolutional network for LEO satellite traffic prediction[J]. IEEE Transactions on Aerospace and Electronic Systems, 2025, 61(4): 9669-9685.

[2] Ju Y, Song J X, Li W J, et al. Dynamic load-balancing routing strategy for LEO satellite networks based on spatio-temporal traffic prediction[J]. IEEE Transactions on Aerospace and Electronic Systems, 2025, 61(5): 11954-11970.

[3] Du J B, Kong Z W, Sun A J, et al. MADDPG-based joint service placement and task offloading in MEC empowered air–ground integrated networks[J]. IEEE Internet of Things Journal, 2024, 11(6): 10600-10615.

[4] 赵鑫. 低轨卫星航点路由与用户切换技术研究[D]. 南京: 南京邮电大学, 2023.
Zhao X. Research on waypoint routing and user handover technologies for low-earth-orbit satellites[D]. Nanjing: Nanjing University of Posts and Telecommunications, 2023.

[5] International Telecommunication Union. Measuring digital development: facts and figures 2024[M]. Geneva: International Telecommunication Union, 2024.

[6] 廖新悦, 张然, 黄正璇, 等. 空间卫星网络组网与管控技术综述[J]. 天地一体化信息网络, 2023, 4(3): 48-58.
Liao X Y, Zhang R, Huang Z X, et al. Review on networking and control technologies of space satellite network[J]. Space-Integrated-Ground Information Networks, 2023, 4(3): 48-58.

[7] Bai W G, Yang H M, Tong J C, et al. Vector segment routing for large-scale multilayer satellite network[J]. Journal of Communications and Information Networks, 2023, 8(1): 24-36.

[8] 徐晓帆, 章跃跃, 曹馨悦, 等. 面向多层多域巨型星座的卫星网络拓扑控制[J]. 通信学报, 2024, 45(7): 1-9.
Xu X F, Zhang Y Y, Cao X Y, et al. Topology control of satellite networks for multi-layer and multi-domain mega-constellation[J]. Journal on Communications, 2024, 45(7): 1-9.

[9] 罗树欣, 张超, 肖勇, 等. 面向巨型星座的智能负载均衡算法[J]. 天地一体化信息网络, 2023, 4(4): 49-60.
Luo S X, Zhang C, Xiao Y, et al. Intelligent load balancing algorithm of mega constellation[J]. Space-Integrated-Ground Information Networks, 2023, 4(4): 49-60.

[10] Li R, Zhang J X, Zheng S, et al. LEO mega-constellations routing algorithm based on area segmentation[C]//Proceedings of the 2023 IEEE Wireless Communications and Networking Conference (WCNC). Piscataway: IEEE Press, 2023: 1-6.

[11] Hu M L, Wang C X, Cao B, et al. A lightweight and scalable design of segment routing in broadband LEO constellations using landmark-based skeleton graphs[PP]. V1. (2024-11-29) [2026-04-30]. arXiv: arXiv.2411.19679.

[12] Chen C, Liao Z, Ju Y, et al. Hierarchical domain-based multicontroller deployment strategy in SDN-enabled space–air–ground integrated network[J]. IEEE Transactions on Aerospace and Electronic Systems, 2022, 58(6): 4864-4879.

[13] Mao B M, Zhou X M, Liu J J, et al. On an intelligent hierarchical routing strategy for ultra-dense free space optical low earth orbit satellite networks[J]. IEEE Journal on Selected Areas in Communications, 2024, 42(5): 1219-1230.

[14] 段洁, 李朝江, 刘浩, 等. 以信息为中心的天地一体化网络分域路由机制研究[J]. 重庆邮电大学学报(自然科学版), 2024, 36(5): 847-858.
Duan J, Li C J, Liu H, et al. Area-divided-based inter-satellite routing mechanism for space-ground integrated network based on ICN[J]. Journal of Chongqing University of Posts and Telecommunications (Natural Science Edition), 2024, 36(5): 847-858.

[15] Zhao Y T, Wu W H, Ning X Y, et al. STSR: a satellite-tailored segment routing method for efficient space communication[C]//Proceedings of the 2024 IEEE Wireless Communications and Networking Conference (WCNC). Piscataway: IEEE Press, 2024: 1-6.

[16] Tu J H, Dang J C, Wu K K, et al. Segment routing algorithm for mega LEO constellations[C]//Proceedings of the 2024 4th International Conference on Electronic Information Engineering and Computer (EIECT). Piscataway: IEEE Press, 2024: 892-896.

[17] 戴天, 冯旭, 虞志刚, 等. 基于SRv6的空间网络路由技术研究[J]. 中国电子科学研究院学报, 2021, 16(6): 539-547.
Dai T, Feng X, Yu Z G, et al. Research on SRv6 based spatial network routing technology[J]. Journal of China Academy of Electronics and Information Technology, 2021, 16(6): 539-547.

[18] 燕锋, 聂浩俭, 夏玮玮, 等. 面向大规模低地球轨道卫星网络的星间链路连接策略[J]. 通信学报, 2024, 45(8): 100-109.
Yan F, Nie H J, Xia W W, et al. Inter-satellite link connection strategy for large-scale low earth orbit satellite networks[J]. Journal on Communications, 2024, 45(8): 100-109.

[19] Yang J Y, Li B, Fan K X, et al. Analysis of laser intersatellite links and topology design for mega-constellation networks[J]. IEEE Internet of Things Journal, 2024, 11(21): 34554-34566.

[20] Riley G F, Henderson T R. The ns-3 network simulator[C]// Modeling and Tools for Network Simulation. Berlin: Springer, 2010: 15-34.

[21] Clausen T, Jacquet P. Optimized link state routing protocol (OLSR)[R]. 2003.

[22] Choi I, Kim S, Choi J P. A simplified algorithm for online satellite network slice planning with trade-off between load-balancing and minimum-hop routing[C]//Proceedings of the 2024 15th International Conference on Information and Communication Technology Convergence (ICTC). Piscataway: IEEE Press, 2024: 128-133.

[23] Zhou Y H, Chen H M, Dou Z B. MOLM: alleviating congestion through multi-objective simulated annealing-based load balancing routing in LEO satellite networks[J]. Future Internet, 2024, 16(4): 109.

[24] 王超, 安建平, 邢成文, 等. 面向空间信息网络的隐蔽通信技术综述[J]. 中国科学(信息科学), 2024, 54(6): 1319-1349.
Wang C, An J P, Xing C W, et al. A review of covert communication technologies for space information networks[J]. Scientia Sinica (Informationis), 2024, 54(6): 1319-1349.

[作者简介]



郑恺填（2002– ），男，西安电子科技大学博士生，主要研究方向为大规模低轨卫星通信网络、分层分级网络管理、卫星通信与星间路由。



陈晨（1977– ），男，博士，西安电子科技大学教授，主要研究方向为卫星通信网络、边缘智能、车辆自组网、智能交通和物联网。



杜平（1978– ），男，博士，上海卫星互联网研究院有限公司研究员，主要研究方向为卫星互联网网络架构、承载网和核心网技术体制标准。



俱莹（1986– ），女，博士，西安电子科技大学副教授，主要研究方向为空天地一体化网络、卫星路由、卫星网络仿真、5G/6G无线通信网络。



徐宁（1983– ），男，博士，重庆卫星网络系统研究院有限公司研究员，主要研究方向为卫星混合组网。