

# 基于确定性网络的网端一体流量交换机制

王晨阳, 张亚生

(中国电子科技集团公司 第五十四研究所, 石家庄 050081)

**摘要:** 低轨卫星通信系统不受地理条件限制, 相比于地面通信有覆盖范围广、运维成本低、部署周期短等优点, 因此逐渐受到人们的关注; 但对于低轨星座来说, 拥有众多卫星节点的无线多跳网络, 总传输和处理时延明显增大; 确定性网络是近几年工业界新提出的一种网络技术, 可以解决传统网络有关丢包和时延相关的需求, 这与低轨卫星网络的需求相同; 研究了地面网络确定性网络的关键技术和卫星网络的相关特征, 将确定性网络技术应用到低轨星座, 将 ATS 异步流量整形算法应用到星地间流量控制上, 提出一种星载交换机制, 优化了星载交换机的资源使用和丢包; 借助 NS3 软件仿真了相关场景, 并对结果进行分析, 在多个仿真场景中均优化了星上资源利用, 缓解了拥塞问题, 在最优参数场景下实现了 0 丢包, 吞吐量有 5% 以上的提升。

**关键词:** 低轨卫星通信; 确定性网络; 流量调度; 异步流量整形; 星上交换; NS3

## Network Terminal Integrated Traffic Switch Mechanism Based on Deterministic Network

WANG Chenyang, ZHANG Yasheng

(China Electronics Technology Group Corporation No. 54 Research Institute, Shijiazhuang 050081, China)

**Abstract:** Low earth orbit (LEO) satellite communication system is not restricted by geographical conditions, and compared to ground communication, there are the characteristics of wide coverage, low operation and maintenance costs, and short deployment period, therefore, it is gradually paying attention to of people. But as for LEO satellite communication system, total transmission and processing latency increase significantly due to wireless mesh network with many satellite nodes. A new network technology named deterministic networking is mentioned by industrial circle in recent years. The deterministic networking can solve the requirements of traditional packet loss and delay, which is matching the requirements of the LEO satellite networks. The key technologies of the deterministic network in ground network and the relevant characteristics of satellite network are studied. The deterministic network technology is applied to the LEO constellation, and the ATS asynchronous traffic shaping algorithm is applied to the inter satellite ground flow control, then an on-board switch mechanism is proposed to optimize the resource usage and packet loss of the on-board switch. The relevant scenarios were simulated by using the Network Simulator 3 software, and the results were analyzed. In many simulation scenarios, the utilization of on-board resources was optimized, which alleviated the network congestion problem. Under the optimal parameter scenario, zero packet loss was achieved, and throughput was improved by more than 5%.

**Keywords:** LEO satellite communication; deterministic network; traffic scheduling; asynchronous traffic shaping; on board switch; network simulator 3

## 0 引言

卫星通信就是以卫星作为“基站”来中继转发信号的通信方式, 可以在多个地面站之间通信, 卫星通信可以对地面的实现近乎无缝的覆盖, 由于地面基站在沙漠、极地、海洋等地形环境下难以建造, 因此地面网在全球范围内的通信都受到了限制<sup>[1]</sup>, 而卫星工作于几百甚至上万公里的太空轨道上, 因此覆盖范围相比于传统的移动通信系统要大得多, 解决地面网的问题。

低轨卫星通信是卫星通信的重要组成部分, 各种类型的低轨通信卫星组合在一起形成了低轨星座, 其传输路径

包括卫星之间的星间链路以及卫星和地面站、用户间的星地链路, 低轨卫星系统可以接入高轨通信网和地面核心网, 实现地网互联等, 构成空天一体的信息传输处理体系。

低轨卫星的主要优势在于以下几点。首先, 由于低轨卫星轨道高度较低<sup>[2]</sup>, 与地面的通信链路和星间链路距离较短, 因此信号的传播路径相较于中高轨卫星较短, 传播时延和传播损耗均减小; 其次, 数颗低轨卫星构成的低轨卫星网络可以完整的连续的实现对包括高纬度区域在内的全球覆盖; 此外, 低轨卫星大多体积较小, 因此发射成本更低, 可以大量发射, 这样即便个别卫星节点出现问题,

收稿日期: 2022-11-13; 修回日期: 2022-11-18。

作者简介: 王晨阳(1997-), 男, 河北石家庄人, 在读硕士研究生, 主要从事卫星网络方向的研究。

通讯作者: 张亚生(1969-), 男, 甘肃金昌人, 硕士, 研究员, 主要从事卫星 IP 网方向的研究。

引用格式: 王晨阳, 张亚生. 基于确定性网络的网端一体流量交换机制[J]. 计算机测量与控制, 2023, 31(1): 270-275.

仍能保证通信系统的运转, 并且成本低的小型化低轨卫星更加便于卫星功能重构, 达到更加灵活和部署和功能性。

但低轨卫星系统仍然存在一些缺陷和问题。卫星网络相比较于地面网络来说, 有较大的卫星通信距离, 并且拥有众多卫星节点的无线多跳网络使得传播时延远大于地面网络。另外, 低轨卫星具有高动态性的特点, 星间链路和星地链路高频切换, 拓扑结构容易变化<sup>[3]</sup>, 因此传播时延变化也大, 时间同步也难以利用。另外由于传输距离和链路信道会受到各种干扰, 传输数据的误码率也大于地面网络。

本文针对低轨卫星的问题, 提出研究内容的需求, 即为追求卫星通信系统的低时延、确定性和高可靠性。针对该需求, 研究地面确定性网络中的关键技术和相关技术标准<sup>[4]</sup>, 分析其优缺点和适应性, 设计了一种流量交换机制, 聚焦在 ATS 流量整形优先级和队列选择, 着重解决确定性传输中队列拥塞和丢包的问题。最终借助 NS3 软件仿真, 测试结果在本文提出的流量控制机制下, 星上交换的流量丢包率有明显下降, 且资源利用率得到了提高。

## 1 网端一体的流量控制机制

### 1.1 低轨卫星通信系统的组成

低轨卫星系统的主要组成一般是三部分, 即为用户段、空间段以及地面段<sup>[5]</sup>。这三段各自的组成个体也不相同, 并且有各自的功能。

1) 用户段: 主要组成部分是各种用户, 包括各种类型的终端设备, 如手机、车载终端等, 终端用户是整个系统的最边缘部分, 主要就是向卫星网络提起各种业务请求, 只要在卫星某一天线的波束覆盖内, 就可以于卫星之间形成一条通信链路实现信息交互和传输<sup>[6]</sup>。

2) 空间段: 主要组成部分是低轨通信卫星, 由一定数量的卫星组成了卫星星座, 每颗卫星是组成整个网络各自的节点。各位卫星之间也是固定格式的卫星数据帧在星间链路中传播。低轨卫星的轨道高度一般是 500~2 000 公里。

3) 地面段: 主要组成部分就是地面的信关站。信关站是地面网和卫星网之间互联的桥梁, 是实现地面用户信息向卫星传输和接受自卫星的信息的重要连接点。当信关站在卫星的可视范围内时, 卫星和地面信关站之间就可以实现包括各种信令和业务信息等数据。当信关站和卫星间信号强度较弱时, 就会以一定的方案选择切换至其他卫星。

### 1.2 地面确定性网络到卫星网络

在地面网络中, 许多新兴的服务和应用对网络的性能提出了很高的要求, 智能制造、汽车自动驾驶、远程医疗等, 这些应用对网络传输的要求已经不仅是传输速率和带宽的问题, 能够实现端到端的确定性服务同样至关重要, 这其中包括带宽、时延、抖动和丢包等指标要求, 因此确定性网络应运而生, 它是网络技术的一种新型服务质量保障技术。

卫星网络相较于地面网络更加复杂, 特别是低轨星座的星间拓扑频繁变化导致数据传输容易震荡, 卫星相对于

地面的高速移动则带来客户端变化频繁、网内流量需求大规模变化的问题, 因此想要在低轨星座间实现确定性传输更加困难。因此在卫星网络中, 假如单独依靠低时延星载交换技术等方法, 无法完全保证传输质量, 因为地面终端设备如果不对自己传输行为进行控制, 很容易造成星上交换机的拥塞。

网络传输保证服务质量的通常手段包括扩容、流量整形、队列调度和拥塞控制等。本文提出的问题场景在于解决丢包的问题, 因此提出一种流量反馈控制机制, 借助地面设备终端的行为控制, 并和星上交换设备一体化处理, 形成星地融合、网端一体的网络传输架构, 解决星上交换拥塞和丢包的问题。

### 1.3 流量队列反馈控制机制

星载交换机上可用的存储资源有限, 因此当网络流量过大或存在突发流量时很容易造成队列的拥塞从而造成丢包, 为了支撑确定性的端到端传输, 实现低丢包甚至零丢包, 必须要进行动态、高效的网络资源分配, 因此有必要设计星地一体、网端协同的传输架构体系。本文采用星上反馈地面设备的流量交换控制机制<sup>[7]</sup>, 基于图 1 所示的队列模型, 这种机制的主要工作就是将星上缓存队列状态信息反馈给地面设备, 地面设备收到广播信息根据队列调度算法调整相应队列的带宽分配, 通过这种方式缓解流量过大或突发流量带来的星上缓存拥塞问题<sup>[8]</sup>。

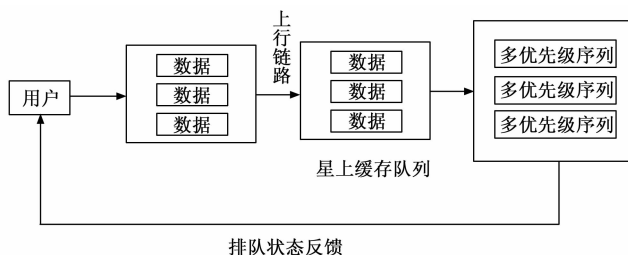


图 1 队列模型和流量控制机制

地面设备调整流量队列借助一定的流量调度算法, 对发送到星上的链路帧进行编排, 按照一定的规则分配合适的逻辑队列<sup>[9]</sup>, 不同的逻辑队列发送到星上交换设备后, 转发至交换设备输出端口的顺序和优先级是不同的, 当地面设备收到星上的广播以后, 修改链路帧具体的字段和参数, 以此来控制发送到星上后属于不同的虚拟输出队列, 从而达到应对突发流量, 缓解星上交换部分短时间拥塞的问题, 提高系统的吞吐量, 减少丢包。

反馈信息有两类, 一是通知给地面设备降低较高次序逻辑队列的发送带宽, 提高低次序逻辑队列的发送带宽, 从而缓解高次序队列的拥塞问题; 二是通知给地面设备提高较高次序逻辑队列的发送带宽, 相应的低次序带宽就会提高。具体两种反馈信息的判定依据是在星上的缓存队列设置两个阈值, 最大阈值  $Q_{max}$  和最小阈值  $Q_{min}$ , 当星上的较高次序缓存队列到达最大阈值, 即发送反馈信息一, 降低高次序队列资源分配; 当星上较高次序缓存队列到达最小

阈值，即发送反馈信息二，提高高次序队列资源分配。

### 1.4 ATS 异步流量整形调度算法

卫星通信网络的数据流量很大<sup>[10]</sup>，并且由于异轨道星间距离处在不断变化之中，导致基于时间同步的转发机制性能受到影响，所以采用异步的流量整形机制，即不需要全网时钟同步，选择 ATS 异步流量整形算法为地面设备分配带宽，对传输的链路帧进行控制，实现提出的反馈控制机制。即如果去往某个输出端口的业务量较大时，采取网端一体的流量控制机制，通过星地广播流控帧，通知地面设备减少流入的业务量，特别是去往某个端口的业务量，减少星上分组丢弃。

IEEE 802.1Qcr 标准提出了基于 UBS 的 ATS 调度机制。IEEE 802.1Qcr 标准中指出实现 ATS 调度机制的交换设备都需要对以下组件进行支持：流过滤器、服务数据单元过滤器、流门控、整形器、整形器组和传输选择算法<sup>[11]</sup>，其整体架构如图 2 所示。

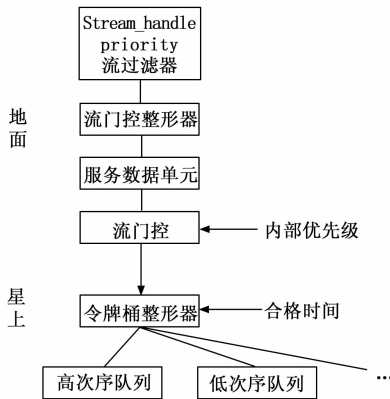


图 2 ATS 调度机制的整体架构

本文将 ATS 整形调度算法应用于星地间的流量交换控制机制，在本文提出的流量控制机制中，整个流程在从流门控输出以前的部分都设置于地面设备，首先当数据帧从地面设备发送前通过 stream\_handle (IEEE Std 802.1CB 中流识别功能的一个子参数) 和 priority (VLAN 标签<sup>[12]</sup>中的优先级字段)，这两个字段对数据流进行识别和匹配，根据这一步的处理决定后续进入合适的流过滤器，不同的流过滤器后是相对应的流门控整形器进程处理，随后进入数据处理单元，数据处理单元的主要任务是过滤控制一定大小的数据帧，不符合要求的数据帧在数据处理单元处就会被丢弃；符合数据单元大小限制的数据帧会按照之前流过滤器匹配的指定，进入相应的流门控；在指定的流门控内，数据帧会被分配一个内部优先级该优先级就用于星上的队列控制。通过地面设备的这些部分，各个数据帧被分析处理并分配了合适的内部优先级，经处理后的数据帧随后将会被传输到星上，在星上将会进行后续的处理。

星上设备接收到带有内部优先级的数据帧后，应用基于令牌桶的交织整形算法，整形后的数据帧会被分配一个合格时间，合格时间用于后续计算传输选择算法，并分配

不同级别的转发队列。在整形器之后，数据帧就会按照合格时间的映射进入不通过次序的队列排队发送，排队发送的基于严格的优先级调度，高次序队列传输先于低次序队列，相对高次序队列当前首个数据帧合格时间小于当前时间则发送，并重新开始执行从高次序到低次序的传输算法，假如出现合格时间判定异常，即为收到地面设备的处理后的地面设备的控制，则转而发送下一较低次序的传输队列中的数据帧<sup>[13]</sup>，按照这样的排队序列，可以保证从地面设备发出的数据帧，按照发出时数据帧所要求的传输次序，在星上转发时，按照要求的次序传输到星上交换输出口，从而达到缓解星上缓存队列拥塞丢包的问题。

## 2 仿真系统设计

### 2.1 NS3 仿真软件

NS3 作为一个离散事件网络模拟器，有很强的可扩展性与易用性，提供各种用于网络仿真的 API，可以在脚本中调用这些 API，来构建自己的仿真模型<sup>[14]</sup>。NS3 的核心架构和各种网络通信模型是用 C++ 实现的，并且用 Python 语言绑定，这样就可以用 Python 语言编写脚本，NS3 的组织结构如图 3 所示，内核和 Network 模块可以为网络仿真提供一般性服务，上层模块可以根据不同的仿真场景进行选择搭配<sup>[15]</sup>。

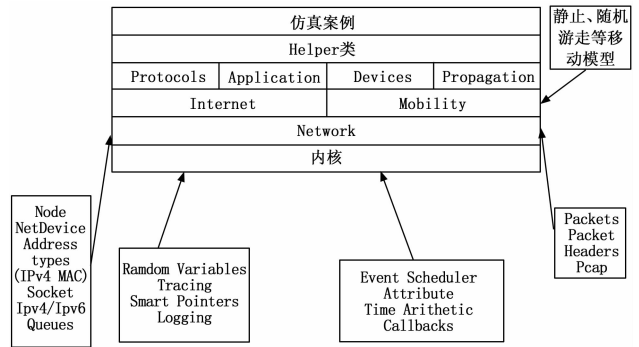


图 3 NS3 的组织结构

NS3 内核定义了其核心功能<sup>[16]</sup>，包括随机变量 (random variables) 的生成、追踪 (tracing)、智能指针 (smart pointers)、日志 (logging)、事件调度 (event scheduler)、属性 (attribute)、回调 (callbacks) 和时间记录 (time arithmetic)。

NS3 中的基本网络组件可分为以下四类。节点：各类通信设备的抽象化描述，可以通过添加移动模型设置初始位置与移动特征，能够添加应用、协议、外部接口等；网卡及其驱动：能实现网络连接的硬件设备和软件驱动的抽象化描述，有各种不同类型的网卡：PointToPointNetDevice、WifiNetDevice 等；信道：节点通过相应的信道和其它节点通信，包括 CsmaChannel、WifiChannel 等；应用程序：提供了管理用户应用并设置数据收发格式的各种方法，包括 UdpEchoClientApplication、UdpServerApplication 等。

借助 Helper 类，输入参数来节点个设置、网络设备添

加、协议栈安装、应用程序设置、信道配置等操作, 构成网络通信的基本模型, 如图 4 所示。

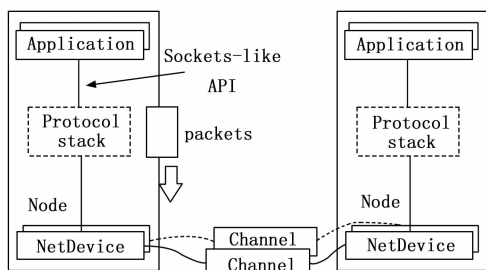


图 4 NS3 中网络通信的基本模型

利用 NS3 平台实现网络仿真, 有五个基本步骤。第一步是设计仿真场景、仿真参数, 例如节点个数、开始时间和仿真时长等。第二步是创建网络拓扑, 借助 Helper 类在节点上添加网络设备, 安装协议栈和应用、配置移动模型等。第三步是在调度器中添加事件, 收集仿真相关参数。第四步是启动调度器, 在仿真时间内处理离散事件, 完成待定任务。最后进行结果分析, 可以通过 Tracing 收集网络数据<sup>[17]</sup>, 例如数据传输速率、丢包率等, 借助统计模块等进行分析, 也可使用 Wireshark 等工具进行网络分析。

NS3 具有良好的可扩展性, 可以方便地自己开发需要的模块或者构建自己的通信场景, 比如本文仿真的卫星场景, 就要首先设计一个卫星通信的场景, 之后要将地面移动通信体制进行修改, 使其适应于卫星通信系统。

## 2.2 卫星仿真系统设计

卫星星座的星地链路间存在传输时延与自由空间损耗, 其中影响星地时延的主要是地面用户和卫星的距离, 而星地损耗由信号频率、传播距离决定, 最终的接收功率还与发射功率、多波束天线的增益、用户接收增益有关<sup>[18]</sup>。

整个仿真系统, 借助 NS3 的基础模型, 模拟整个卫星通信系统, 星地链路模型以传播时延模型 PropagationDelayModel 和自由空间传输损耗模型 FreeSpaceLossModel 为基础, 卫星模块主要结构分 model、helper、examples、doc 子部分, 这几部分主要是卫星模型底层源码、实现特定功能的 Helper 类、测试与仿真、文档说明, 模型实现星历文件解析、轨迹获取、波束设计等功能。Satellite-beam 类完成波束数量、波束增益、波束增益最大指向位置等的设置, SatHelper 提供创建卫星场景的接口, 主程序传入仿真时间与用户数目等可变参数, 过程需要解析星历文件, 初始化卫星相关参数, 计算卫星运动轨迹、判断是否闲置卫星等功能<sup>[19]</sup>, 场景设计的不同在于卫星数量与用户规模, 本文测试的场景是小范围场景, 且仿真时间短, 只需要数量很少的地面用户, 同时卫星只需要加载在仿真时间内的可见卫星。建立卫星、波束节点、用户节点后, 安装相应的移动模型与波束天线模型后, 可通过 NetDevice 类来设置频率、切换算法等参数, 卫星场景将按既定规则运行。用户初始位置与移动模型可以通过 MobilityHelper 这个帮助类

进行设置, 将地表某个区域随机分布的用户三维坐标依次添加到链式位置分配器 (ListPositionAllocator) 中, 用户运动模型可选择固定位置、固定速度、固定加速度、随机游走等模型<sup>[20]</sup>。

## 2.3 卫星仿真系统参数设置

在 NS3 中构建网络仿真环境, 仿真参数: 仿真时间  $t = 10$  s; 本文为了测试该流量控制机制的性能, 设置仿真场景较简单, 星上交换设备输入输出端口数,  $N_{in} = 1$ ,  $N_{out} = 1$ ; 地面设备发送带宽  $W = 300$  Mbps; 包长度  $L_{data} = 1\ 000$  Byte; 地面设备为去往星上交换机基本带宽均为 6%; 源节点发送数据包的起始时间  $t_0$  设为随机数,  $t_0$  服从均匀分布。

为了测试该流量控制机制的性能, 设计了四个应用场景, 在场景一中, 设置星上缓存队列长度  $L$  为 200 (packet); 星上缓存队列阈值  $Q_{max}$  和  $Q_{min}$  分别为 170、100。

场景二改变缓存队列阈值: 最大阈值  $Q_{max}$  和最小阈值  $Q_{min}$  分别为 150、50, 该场景是为了测试当最大阈值和最小阈值都减小时, 星上缓存队列的变化。

场景三设置星上缓存队列阈值  $Q_{max}$  和  $Q_{min}$  分别为 170、130, 该场景主要测试当最小阈值较大时, 对星上缓存队列的影响。

场景四测试缓存队列长度变化对该控制机制的性能有何影响, 因此星上缓存队列长度设置为 150, 星上缓存队列阈值  $Q_{max}$  和  $Q_{min}$  分别为 120、80。

## 3 实验结果与分析

地面设备向上行链路发送数据链路帧, 观察星上输入端口的虚拟输出队列长度  $L_q$  变化, 随着时间增加, 星上输入端口中高次序虚拟输出队列  $L_{q1}$  长度增大。星上输入端口中高次序队列  $L_{q1}$  和低次序队列  $L_{q2}$  实时队列长度变化情况如图 5 所示, 从图中可以看出, 高次序队列在持续接受地面设备发来的链路帧一段时间后, 达到星上缓存队列的设置长度 200 后, 会逐渐出现丢包情况。

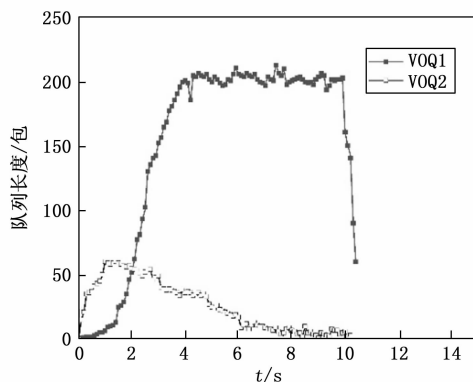


图 5 反馈前输入端口 0 中  $L_{q1}$ 、 $L_{q2}$  实时队列长度

之后测试当流量控制子模块程序启动后, 星上实时队列的长度。如图 6 所示, 可以观察到场景 1 到场景 3 在反馈后输入端口队列  $L_{q1}$  实时长度。

首先分析场景一, 当  $L_{q1}$  到达最大阈值  $Q_{max}$  后, 根据流量控制机制的流程, 星载交换设备将会广播队列状态信息,

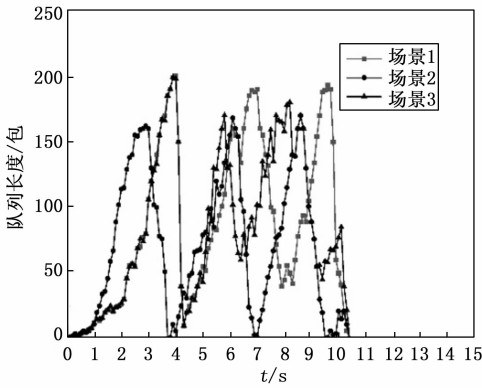


图 6 三个场景星上输入端口中  $L_{q1}$  实时队列长度

地面设备接收星上的反馈信息，高次序分配的带宽资源减少， $L_{q1}$ 的链路帧数量明显降低；随着链路帧的持续减少，当  $L_{q1}$ 到达最小阈值  $Q_{min}$ 后，与之前方式相同，地面设备接收到反馈信息，增加高次序队列的带宽资源分配，随后  $L_{q1}$ 的实时队列长度开始增加。

对于场景二来来说，与场景一对比起来，场景二减小了最小阈值的大小，相比于场景一，队列长度增加、减少的时间间隔较小，并且最大的实时队列长度也较小，说明减小最大阈值确实限制了实时队列长度，但是场景二最小阈值偏小，在调整至最低点的前后一小段时间内，星上实时缓存队列长度都保持较低的状态，资源利用率较低。

场景三相较于场景一大幅提高了最小阈值的值，最大阈值和最小阈值相差较小，因此经反馈后星上  $L_{q1}$ 队列长度变化范围较小，说明最小阈值的设置限制了缓存队列中链路帧的不断减小，队列长度不断在小范围内波动变化，相较于大范围内波动来说，总吞吐量要更加大，因此资源利用率也较高，但是星载交换机需要不断的向地面设备广播缓存队列状态信息。

反馈后场景一、二、三中星上输入端口  $L_{q2}$ 的实时队列长度变化情况如图 7 所示。

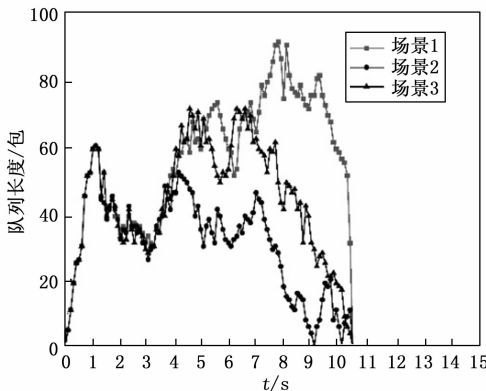


图 7 三个场景中星上输入端口到  $L_{q2}$  实时队列长度

为了具体衡量资源利用情况，对三个场景在流量控制机制使用前后的两个队列的吞吐量和总吞吐量进行计算比较，结果如表 1 所示。（吞吐量 (bps) = 包数 \* 包长 \* 8 /

仿真时间)

表 1 反馈前后吞吐量对比

bps

	$L_{q1}$ 吞吐量	$L_{q2}$ 吞吐量	总吞吐量
反馈前	18 635 485	19 037 408	292 454 113
场景一反馈后	18 631 685	19 059 758	293 084 146
场景二反馈后	18 550 248	19 048 866	292 516 592
场景三反馈后	18 632 379	19 059 310	292 592 403

在流量控制机制的作用下，反馈前高次序队列  $L_{q1}$ 的丢包数为 440 包，低次序队列  $L_{q2}$ 的丢包数为 0；反馈后三个场景中两个队列的丢包率均为 0，因此该机制对解决丢包的问题效果显著。由于地面设备受到反馈信息的控制，因此高次序队列  $L_{q1}$ 实时队列的长度总是处在最大阈值和最小阈值之间，几乎不存在队列长度数值稳定的情况，因此相较于反馈前，反馈后  $L_{q1}$ 的总包数一定会减少，因此吞吐量也会减少，而由于  $L_{q1}$ 的资源减少使得  $L_{q2}$ 得到的资源分配增加，因此相较于反馈前，反馈后到低次序队列  $L_{q2}$ 的总包数有所增加；经计算后总吞吐量相较于反馈前也有所增加，这验证了该流量控制机制对于提高星上资源的利用有效。对于高次序队列  $L_{q1}$ 来说，场景三设置的最大最小阈值间隔较小的情况下，高次序队列  $L_{q1}$ 的吞吐量相较于场景一来说更有优势，但是低次序队列  $L_{q2}$ 的吞吐量和总吞吐量相较于场景一却不再有优势。

场景四在场景一的基础上减小了星上缓存队列长度为 150，如图 8 所示是场景四反馈前后，星上输入端口中  $L_{q1}$ 的实时队列长度变化。

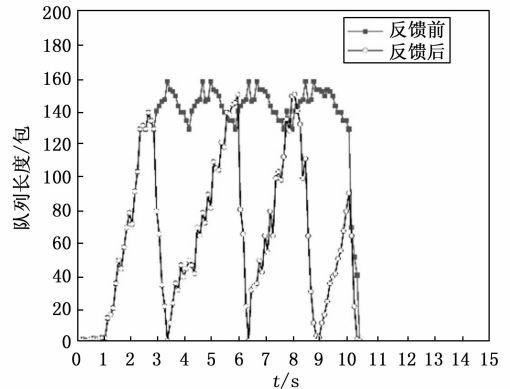


图 8 场景四反馈前后星上输入端口  $L_{q1}$  实时队列长度

从图中可以看出，与场景三类似，场景四中减少最大最小阈值间隔，因此  $L_{q1}$ 队列长度不断在小范围内波动变化，作为对比，反馈前后，星上输入端口中  $L_{q2}$ 的实时队列长度变化情况如图 9。

与其他场景类似，在流量控制机制的影响下，高次序队列  $L_{q1}$ 的带宽分配降低就会带来低次序队列  $L_{q2}$ 的带宽资源增加。

场景四中，反馈前后上行链路发送至两个输出端口的两个队列的包数及上行链路总吞吐量对比如表 2 所示。

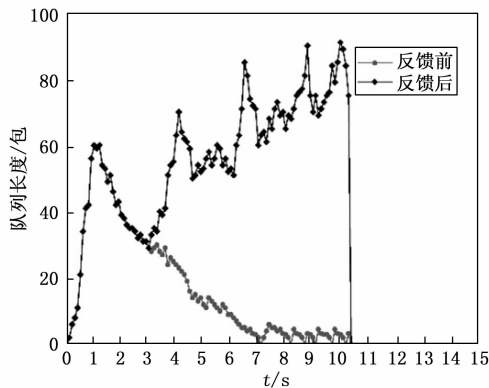


图 9 场景四反馈前后星上输入  $L_{q2}$  实时队列长度

表 2 场景四反馈前后吞吐量对比 bps

	$L_{q1}$ 吞吐量	$L_{q2}$ 吞吐量	总吞吐量
反馈前	18 633 920	19 037 406	292 701 855
反馈后	18 617 491	19 060 178	292 930 044

场景四反馈前队列丢包数为 400 (packet), 队列丢包数为 0; 反馈后队列丢包数为 15, 队列的丢包数为 0。与场景一相比, 场景四中反馈前后上行链路发往高次序队列的吞吐量和反馈后上行链路总吞吐量都略有降低。值得关注的是, 由于场景四中星上缓存空间有限, 因此虽然反馈后, 队列的丢包问题得到改善, 但是链路帧仍然有一定的溢出, 这说明了星上缓存空间相比较于传输带宽的极限仍较小时, 就无法避免丢包的问题。

通过四个场景对流量控制机制进行仿真分析, 首先本文提出的星地流量交换控制机制对解决星上交换的流量拥塞丢包问题有一定的优化效果, 但是参数设置会在一定程度上影响流量控制机制的性能。其中, 星上缓存队列的最大最小阈值影响很大。当最大阈值设置很大, 逼近甚至超越星上缓存队列时, 流量控制机制就在一定程度上失效, 星上的缓存队列仍然解决不了丢包的问题; 假如像场景二中, 最大阈值设置过小, 则队列长时间处于一个空闲的状态, 因此资源利用率较低; 假如像场景三中, 最大阈值和最小阈值之间差距较小, 则缓存队列的长度会快速波动, 这样会产生一定不必要的开销, 同时对于整个链路的吞吐量增大没用帮助; 另外像场景四中测试分析的, 星上初始的缓存队列长度选择也十分重要, 应当接近传输带宽的极限, 假如星上缓存队列长度相较于传输带宽的极限仍然较小时, 即便最大阈值设置合理, 那么实际的队列长度仍然可能逼近甚至超过星上设置的缓存队列长度, 因此会造成链路帧的溢出, 产生丢包的问题。

#### 4 结束语

为了节约宝贵的卫星带宽资源, 提高卫星网络流量转发的效率, 减少丢包。本文深入研究了确定性网络的相关技术标准, 并结合低轨卫星网络 and 具体特点, 提出一种适用于低轨卫星的星地流量交换机制, 借助 ATS 异步流量整

形算法, 提出了一种星地之间交互、网端一体的流量控制机制, 并利 NS3 仿真, 测试了该机制的性能并作出了相关分析。测试结果表明, 在应用了该流量控制机制后, 星上转发效率得到提高, 星上缓存资源得到充分利用, 缓解了队列拥塞的问题, 解决了大流量下丢包的问题。

#### 参考文献:

- [1] 王子剑, 杜欣军, 尹家伟, 等. 低轨卫星互联网发展与展望 [J]. 电子技术应用, 2020, 46 (7): 49-52.
- [2] 李新桐, 张亚生. 一种适用于低轨卫星的 SDN 网络人工智能路由方法 [J]. 电子测量技术, 2020, 43 (22): 109-114.
- [3] 吴奇龙, 龙 坤, 朱启超. 低轨卫星通信网络领域国际竞争: 态势、动因及参与策略 [J]. 世界科技研究与发展, 2020, 42 (6): 587-597.
- [4] 黄 韬, 汪 硕, 黄玉栋, 等. 确定性网络研究综述 [J]. 通信学报, 2019, 40 (6): 160-176.
- [5] 张晓娜, 张亚生, 何 辞. 基于低轨星座的星地路由技术研究 [J]. 计算机测量与控制, 2019, 27 (10): 264-267.
- [6] 包少彬, 骆 乐. 浅谈星地通信中的传输协议 [J]. 数字通信世界, 2018 (3): 17, 29.
- [7] 于 倩, 陈庆华, 乔庐峰, 等. 星载高速标签交换机星地流控机制设计与仿真 [J]. 军事通信技术, 2015, 36 (2): 13-17.
- [8] SPECHT J, SAMII S. Urgency-based scheduler for time-sensitive switched ethernet networks [C] //2016 28th Euromicro Conference on Real-Time Systems (ECRTS), IEEE, 2016: 75-85.
- [9] 张晓雯. 高速星载路由器交换单元的设计与实现 [D]. 西安: 西安电子科技大学, 2021.
- [10] 汪宏武, 张更新, 余金培. 低轨卫星星座通信系统的分析与发展建议 [J]. 卫星应用, 2015 (7): 38-39, 42-44.
- [11] 岳家瑞. 基于异步流量整形的时延保障机制的研究与实现 [D]. 北京: 北京交通大学, 2021.
- [12] 李春红. VLAN 交换技术在路由器上的实现及应用 [D]. 长春: 吉林大学, 2006.
- [13] 刘依男. 基于流量整形的时间敏感路由研究与实现 [D]. 北京: 北京交通大学, 2021.
- [14] 林晓辉, 苏恭超. NS3 仿真工具在计算机网络课程教学中的应用 [J]. 智能计算机与应用, 2019, 9 (3): 187-189.
- [15] 陈运海. 基于 NS3 的无线自组网仿真实现 [J]. 电脑知识与技术, 2014 (12X): 8365-8367.
- [16] 茹新宇, 刘 渊. 网络仿真器 NS3 的剖析与探究 [J]. 计算机技术与发展, 2018, 28 (3): 72-77, 82.
- [17] 蒋凌云, 卢 凯, 黄海平, 等. 基于 NS3 的 BCube 数据中心网络仿真 [J]. 软件导刊, 2020, 19 (5): 190-194.
- [18] 刘瑞军. 基于 NS3 的多波束卫星通信仿真系统设计与实现 [D]. 西安: 西安电子科技大学, 2021.
- [19] 王 程, 徐 毗, 张素兵, 等. 基于 Linux 系统的 LEO 卫星动态路由协议研究与实现 [J]. 北京邮电大学学报, 2020 (2): 94-102.
- [20] 时晨皓. 车联网通信中的协同优化机制研究及仿真评估 [D]. 开封: 河南大学, 2018.