



研究与开发

流数据驱动的多模态网络转发

吴剑松¹, 谭小彬¹, 袁莘智¹, 琚可欣¹, 骆汉光²

(1. 中国科学技术大学信息科学技术学院, 安徽 合肥 230026;

2. 之江实验室, 浙江 杭州 311121)

摘要: 近年来通信技术发展迅速, 传统传输控制协议/因特网互联协议 (transmission control protocol/Internet protocol, TCP/IP) 网络瘦腰结构的弊端凸显, 新网络架构协议相继被提出, 多种模态网络将长期共存。传统的网络转发方案难以适应多模态网络的动态变化特性, 且存在时延高、成本高等问题。基于此, 提出了一种流数据驱动的多模态网络转发方案和基于强化学习的多模态网络转发算法。针对多模态网络动态变化的特点, 设计了基于可编程数据平面的多模态网络转发方案。该方案包含负责网络信息采集的带内网络遥测 (in-band network telemetry, INT) 采集模块、解析和过滤网络流的网络流处理模块、识别和分流多模态网络流的模态处理模块、提供网络功能的服务功能模块、负责转发的数据转发模块、用于数据缓存的缓存模块, 以及生成和下发转发策略的节点控制模块。利用编写协议无关的包处理器 (programming protocol-independent packet processor, P4) 编程语言协议无关的特性, 该方案可以支持多个网络模态的共生共存, 且通过可编程交换机硬件可以有效提高转发效率。同时, 针对可编程交换机通过表的方式进行数据更新的特点, 提出了基于强化学习的多模态网络转发算法, 并针对可编程硬件限制进行适配, 根据多模态网络流的状态, 为流选择合适的转发端口, 以实现多模态网络的高效转发。在未来网络试验设施 (China Environment for Network Innovation, CENI) 中完成了转发算法的实现和测试, 验证了所提算法的有效性。

关键词: 多模态网络; 网络转发; 可编程数据平面

中图分类号: TP393

文献标志码: A

doi: 10.11959/j.issn.1000-0801.2025047

Streaming data-driven forwarding in polymorphic network

WU Jiansong¹, TAN Xiaobin¹, YUAN Shenchi¹, JU Kexin¹, LUO Hanguang²

1. School of Information Science and Technology, University of Science and

Technology of China, Hefei 230026, China

2. Zhejiang Lab, Hangzhou 311121, China

Abstract: In recent years, rapid advancements in communication technology have highlighted the shortcomings of the traditional transmission control protocol/Internet protocol (TCP/IP) network architecture. New network architecture protocols have been proposed in response to the evolving landscape, where multiple modal networks are expected

收稿日期: 2024-11-14; 修回日期: 2025-01-05

基金项目: 国家重点研发计划项目 (No.2022YFB2901401)

Foundation Item: The National Key Research and Development Program of China (No.2022YFB2901401)



to coexist for an extended period. Traditional network forwarding solutions struggle to adapt to the dynamic nature of polymorphic network, leading to issues such as high latency and costs. Based on this, a flow data-driven polymorphic network forwarding scheme and a reinforcement learning-based algorithm tailored for polymorphic network were proposed. Addressing the dynamic nature of polymorphic network, a polymorphic network forwarding scheme based on programmable data planes was proposed. This scheme comprised the in-band network telemetry (INT) collection module responsible for network information gathering, the network flow processing module for parsing and filtering network flows, the modal processing module for identifying and routing polymorphic network flows, the service function module for providing network functions, the data forwarding module for routing data, the cache module for data storage, and the node control module for generating and deploying forwarding strategies. Leveraging the protocol-independent nature of the programming protocol-independent packet processors (P4) programming language, this scheme facilitated the coexistence of multiple network modes and enhances forwarding efficiency through programmable switch hardware. Furthermore, considering the data update mechanism of programmable switches using tables, a reinforcement learning-based algorithm for polymorphic network forwarding was proposed. Adapting to the constraints of programmable hardware, this algorithm selects appropriate forwarding ports based on the state of polymorphic network flows to achieve efficient forwarding. Finally, the implementation and testing of the forwarding algorithm was conducted in China Environment for Network Innovation (CENI), validating the effectiveness of the algorithm.

Key words: polymorphic network, network forwarding, programable data plane

0 引言

近年来通信技术发展迅速,以传输控制协议/因特网互联协议(transmission control protocol/Internet protocol, TCP/IP)为基石的互联网技术对经济生活的方方面面产生了深远影响。随着互联网与经济社会深度融合发展,互联网+、工业4.0等新兴产业逐渐成为国民经济的新支柱^[1]。互联网在当前社会中扮演的角色日益增多,这使得用户对网络的专业化、个性化需求不断提升。面对互联网与经济社会深度融合发展所带来的多元化业务承载需求,传统TCP/IP网络的“瘦腰”结构在安全保障、扩展、服务质量、效能、移动等方面遭受了前所未有的挑战,在应对大带宽、低时延、高可靠等更高的网络服务能力时显得力不从心。此外,随着网络应用的不断拓展,互联网技术内涵的发展却未能提供充分的支持,基于IP的寻址路由能力是有限且确定的,这与丰富多样的上下层功能之间存在巨大的反差,无法满足多元化和专业化所需高效、高质量服务需求^[2]。

为打破上述网络发展困境,创新网络技术,世界各国均已在新型网络领域开展了基础研究和关键技术攻研布局,研究人员先后提出了内容标识^[3]、地理空间标识^[4]、身份标识^[5]等多种新网络技术。邬江兴院士团队提出了“多模态网络”的概念^[6],将多种网络体制通过可编程技术以网络模态的形式融合到网络中,力求整合多种网络优势,提升网络对多元业务的承载能力。

然而,多模态网络是一个崭新的研究课题,许多问题还有待解决。其中,多模态网络转发是一个关键的问题。多模态网络转发需要在多种模态的数据流到达交换机时,交换机要能够识别这些网络流的模态,为它们选择合适的转发端口,高效地转发到目的节点。同时,多模态转发应不修改模态原有的转发方式,并支持多种模态数据流的动态转发。由于网络中的模态是不断变化的,当有新的网络模态流到达时就需要有应对方式。多模态网络转发主要负责协调各种网络模态之间的各种资源分配,通过合理分配带宽、优化路由选择和动态调整转发策略,实现多模态网络

的高效转发。

目前,针对多模态网络转发方面的研究相对较少,一般都是针对面向特定数量的网络模态转发进行研究。文献[7]提出了一种Trosky网络架构,基于在网络层之上构建3.5层的overlay思想,力求使IP网络更好地兼容以内容标识为中心的网络;文献[8]通过操作系统和网络虚拟化技术,在TCP/IP和基于命名数据网络(named data networking, NDN)的架构上同时支持视频流;文献[9]为了实现高效的NDN流媒体分发,提出了一种NDN/IP混合网络过渡架构Centaur,兼顾了NDN的智能性及可扩展性和IP的传输效率及部署可行性;文献[10]在混合信息中心网络(hybrid information-centric networking, hICN)中采用了一种翻译策略,将ICN/NDN分组集成在IP协议栈中,hICN重用由IP头和端口头的组合派生出ICN/NDN前缀名的报文格式;文献[11]使用网关进行翻译,网关由4个组件组成:网关守护进程接受数据包、网关配置文件命名和配置数据包、流量处理器转发数据包、命令行接口(command line interface, CLI)监控,然而,每一个IP包转换过程都基于一个完全匹配的包过滤器,这需要额外的正则表达式计算作为开销,此外,该策略通过读取数据负载构造前缀名并绑定数据包,存在侵犯数据隐私等安全隐患。综上,现有的多模态网络转发研究存在以下问题:(1)大部分考虑了固定数量的网络模态,不支持多种模态的动态加载,不适用于多模态网络,不具备可扩展性;(2)这些方案依然在借助IP网络进行传输,当大规模部署时,网络层的转发压力仍然存在,无法体现新网络协议的优势;(3)通过使用软件或者定制硬件的方式完成多模态网络转发都存在一定的弊端,使用软件的方式处理数据时,会产生新的时延,降低数据传输的效率,而使用定制的硬件实现多模态网络的转发会产生高昂的成本。总体而言,当前的网络转发方案不适用于

多模态网络。

针对以上问题,本文设计了基于可编程数据平面^[12]的多模态网络转发架构。该架构引入了支持可编程数据平面的交换机作为多模态网络的转发设备,它是一种高度灵活的网络设备,可以在不损失性能的前提下,自定义逻辑实现各种功能,这使得构建一个适应不同网络模态的高效转发架构成为可能。为了保留各个网络模态原生的转发特性,支持不同网络模态“即插即用”,本文将多模态网络划分成多个单一模态的虚拟网络切片,虚拟网络切片内保持模态原有的转发方式,再通过虚拟端口和物理端口映射来完成多种模态网络的转发,解决了依赖IP进行转发产生的问题,充分发挥了新网络协议的优势。此外,多个模态间会出现资源争用,为了缩短多模态网络流的流传输时间,减少链路拥塞发生,针对可编程数据平面的特性,本文设计并实现了在可编程数据平面运行的强化学习转发算法,采用可编程硬件辅助实现,通过累积奖励来指导智能体的学习和决策过程。

综上所述,本文的主要贡献如下。

(1)提出了针对多模态网络的转发方案。本文基于可编程数据平面设计了针对多模态网络的转发方案,支持多模态网络的各种需求,通过编写协议无关的包处理器(programming protocol-independent packet processor, P4)程序对网络模态进行识别,再使用转发算法根据网络流的需求进行转发,且借助可编程硬件可以有效降低转发时延,最终实现多模态网络的高效转发。

(2)在可编程数据平面设计并实现了基于强化学习的多模态网络转发算法。本文利用可编程交换机硬件采用表的方式进行数据更新的特性,设计了基于Q-learning强化学习算法的多模态网络转发算法。这一方案通过在交换机的M/A表中存储Q值,根据多模态网络流到达交换机的状态,为数据流智能地选择合适的转发端口。



(3) 本文基于未来网络试验设施 (China Environment for Network Innovation, CENI), 使用 Ubuntu 虚拟机、bmv2 软件交换机和 Tofino 硬件可编程交换机搭建测试平台, 使用 iperf3 模拟多种模态的网络流量进行实验测试, 采用转发时延、平均队列长度和吞吐量作为评估指标, 通过对实验结果的分析验证了架构和算法的可行性。

1 多模态网络转发架构

本节首先介绍了多模态网络转发的设计思路, 然后提出了流驱动的多模态网络转发架构, 介绍了架构中各个模块的功能, 并描述了架构的整体工作流程。

1.1 多模态网络转发设计思路

多模态网络转发的目标是支持多种网络模态数据流的高效转发。首先, 多模态网络转发并不意味着修改原有的传统网络转发方式。相反, 它是在传统转发方式的基础上进行操作和扩展。传统网络转发方式在网络设备中已经得到了广泛应用和验证, 具有一定的稳定性和成熟度, 而多模态网络中网络模态是动态变化的, 改变所有模态的转发方式以支持整体架构并不合理。因此, 多模态网络转发不会放弃网络模态原有的转发方式, 而是在其基础上引入新的机制, 以实现多种网络模态共存的高效转发。多模态网络转发需要

协调各种网络模态之间的资源分配, 根据实时网络条件和网络流需求, 以及数据量、时延、可靠性和带宽等因素, 选择最适合的端口来进行传输, 以获得最佳的传输效果。

为了实现以上目标, 本文提出了以下思路, 根据节点部署的网络模态将多模态网络从逻辑上切分为多个单一模态的网络切片。多模态网络划分成虚拟网络切片示意图如图1所示, 其中A、B、C表示1号节点的转发端口号, 1、2表示IP虚拟网络中的转发端口号。每个模态的虚拟网络运行模态原有的转发算法, 通过端口映射将模态虚拟网络与真实物理网络对应, 模态网络转发通过为网络流分配转发端口提高转发效率。将多模态网络切片为单一模态的虚拟网络, 可以针对每个模态应用其原有的转发算法, 以确保最佳的网络性能和效率。每个切片都可以根据其特定的模态需求进行优化, 并运行相应的转发算法, 以满足特定模态的传输要求。同时, 借助端口映射技术, 实现了虚拟网络与物理网络之间的无缝连接, 从而确保数据的顺利传输和转发。此外, 为了提高转发效率, 从网络流可用的物理端口中进行选择, 通过合理分配转发端口, 可以有效地管理网络流的转发过程, 避免冲突和混乱, 提高网络的整体效率和吞吐量。

1.2 流驱动的多模态网络转发架构

本文根据多模态网络需求, 构建网络模态无

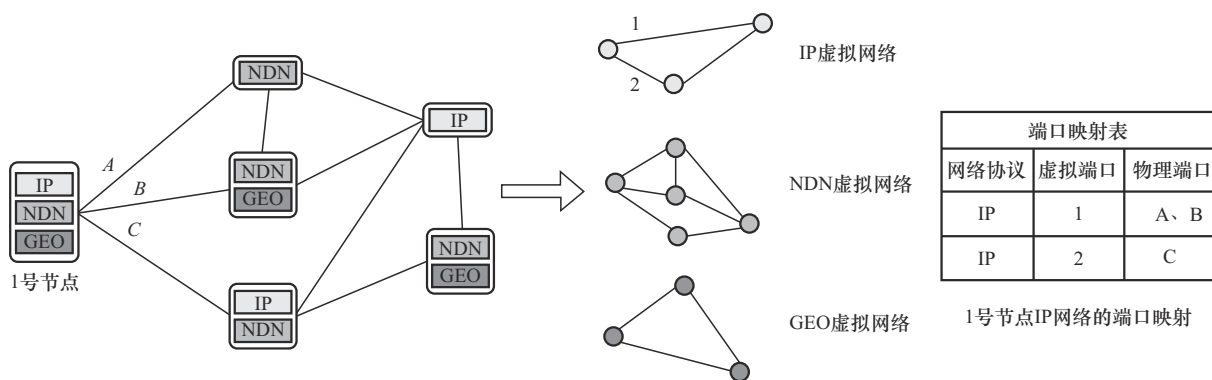


图1 多模态网络划分成虚拟网络切片示意图

关、数据类型无关的数据平面转发架构，实现在同一物理网络域中支持多种网络模态。在模态隔离方面，该架构突破传统的单一模态固定流水线数据包处理模式，以可编程的方式灵活地配置数据面网络设备内的解析器、流水线，从而实现不同模态的识别与分流。该架构设计了面向数据流的调度部署模块，自适应地为多模态网络数据的处理做出调整，实现网络节点内资源的充分利用。流数据驱动的多模态网络转发架构如图 2 所示。

多模态网络转发架构分为节点控制平面和节点数据平面，其中节点控制平面由节点控制模块组成，节点数据平面由网络流处理模块、模态处理模块、数据转发模块、服务功能模块和缓存模块

块组成。各个模块的主要功能介绍如下。

(1) 带内网络遥测 (in-band network telemetry, INT) 数据采集模块：主要是在数据平面周期性采集交换机的状态信息，如队列排队长度、吞吐量等网络状态信息，这部分状态信息后续会发送到控制平面做进一步分析和应用。

(2) 网络流处理模块：主要是解析、过滤和接收网络流。网络流处理模块会对传入的数据进行分析，提取关键信息，并通过对这些信息的解析获取当前网络中不同网络流的占比信息，负责将网络流信息传输给调度部署模块。除了解析功能，网络流处理模块还能执行过滤操作。它可以根据预先设定的规则，筛选和过滤特定类型的网络流量。通过设置合适的过滤规则，网络流处理

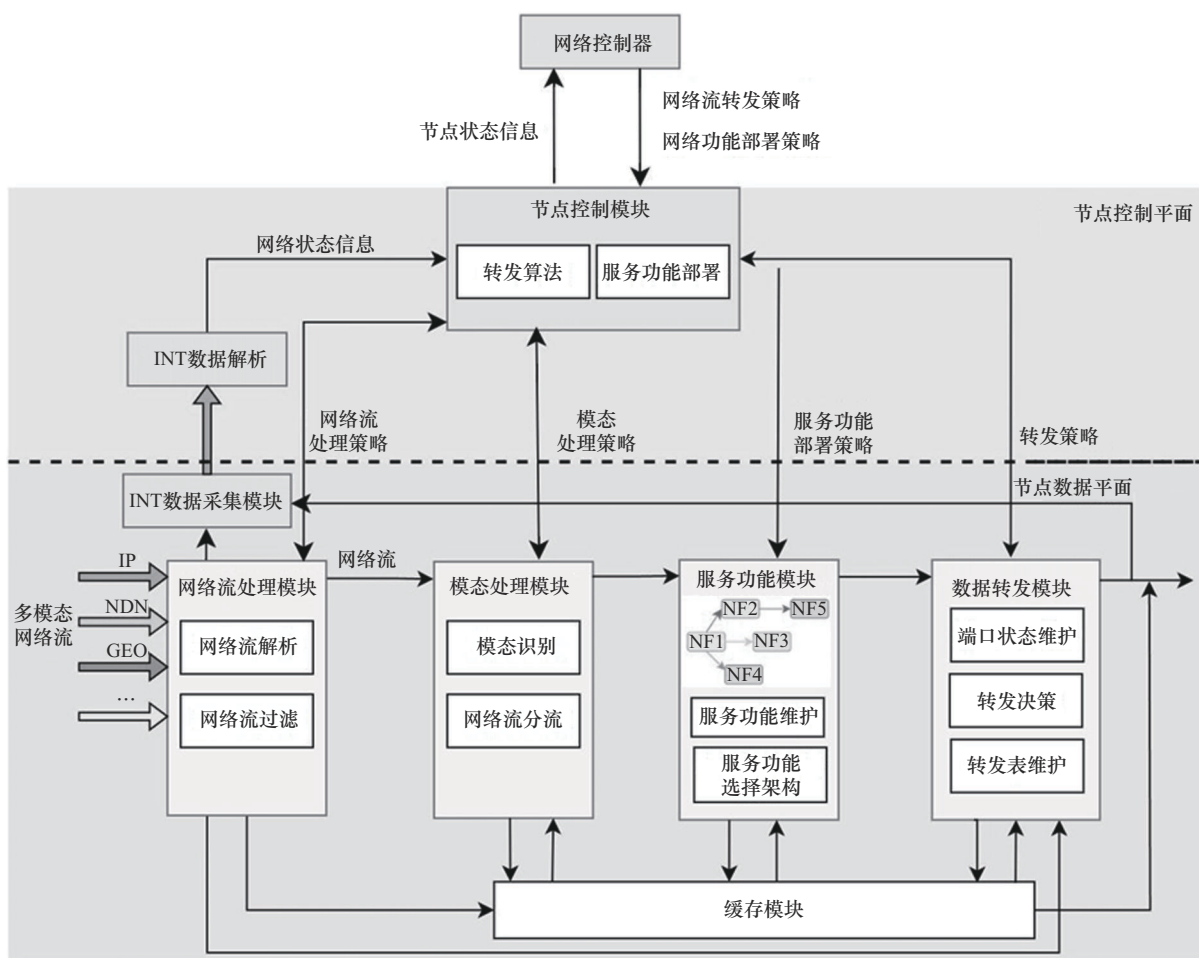


图2 流数据驱动的多模态网络转发架构



模块能够提高网络安全性,并且确保只有符合要求的流量被接收和处理。最后,作为一个网络流处理模块,它还负责接收和接管网络流量。一旦解析和过滤操作完成,该模块将负责将合格的流量传递给下一个处理环节。

(3) 模态处理模块:主要负责识别网络流模态并对网络流进行分流。在这个过程中,该模块运用P4编程语言对数据包进行解析,通过提取关键字段来确定流的模态,即确定其所属的特定网络协议。然后,利用布隆过滤器,通过多个哈希函数计算包头的关键字,根据得到的哈希值映射到位阵列上的位置,确定该流所需的网络功能。这一过程为后续流处理任务提供了重要的基础,有助于实现更高效的资源利用和流程管理。对于可编程数据平面而言,寄存器资源较为稀缺,使用布隆过滤器可以有效节省存储资源,而且布隆过滤器查询元素的时间复杂度为 $O(K)$,其中 K 为哈希函数的个数(通常较小),这意味着查询速度与数据量大小无关,因此查询速度较快。当布隆过滤器出现误判时,可以将数据包转发给控制平面进行重传。

(4) 服务功能模块:这是一个由多个服务功能链^[13](service function chain, SFC)组成的模块,其主要任务是处理各种网络流,将多模态网络流原有的转发机制进行转发,从而获取网络流转发的虚拟端口,还包括但不限于数据包过滤、负载均衡、安全检测、流量优化等功能。通过对这些功能链的有序组合,服务功能模块能够实现对网络流的深度分析和处理,以确保网络的安全性、可靠性和性能优化。无论是在企业网络还是云计算环境中,服务功能模块都扮演着关键的角色,它为网络流提供高效的处理,以满足用户的需求并提升网络的整体效能。

(5) 数据转发模块:主要负责维护转发表和转发数据。转发表基于Q-learning训练所得,通过指令台下发到可编程交换机上。每次有网

络流到达时,转发模块就会根据网络流的信息查询转发表,为网络流选择收益最大的转发端口,完成转发后等待新的网络流到达。该模块为网络转发系统的正常运行提供了基础支持。

(6) 缓存模块:主要用于存储被当前模块处理完成的网络流,由于链路带宽等问题来不及传输到下一模块,可以先存储到缓存中。这样可以保证整个网络架构的高效率运行。

(7) 节点控制模块:这是一个关键的组件,其主要职责是根据当前流信息生成有效的调度方案,并下发相应的转发策略给数据转发模块,同时也提供内生安全相关功能。该模块可以在网络流转发效果不佳时,重新部署网络功能,以实现更好的性能和效果。通过不断监测和分析网络流量的特征和负载情况,节点控制模块能够评估当前网络功能的工作状态,并根据需要进行调整。这包括在网络节点上重新部署功能实体、优化网络流的路径选择,以及实施负载均衡等操作。通过灵活的调度方案和策略下发,节点控制模块能够及时应对流量变化和 network 需求的变动,确保网络的稳定性和性能的最佳化。

多模态网络转发架构的工作流程如下。

(1) 交换机会根据历史信息部署好网络功能(network function, NF),并将NF的部署信息传递给节点控制模块,这些信息包括整个网络节点的拓扑关系以及每个节点上部署的网络模态功能。不同的网络模态需要特定的网络功能进行处理。控制器会根据网络模态的部署信息,将整个网络拓扑抽象成多个网络模态的虚拟网络切片,虚拟网络中节点的虚拟端口会与其物理端口形成映射关系。控制器会通过指令以表的方式将节点虚拟端口和物理端口的映射关系下发到各个网络节点。当网络拓扑发生变化时,也要按照这个流程对端口映射表进行更新。

(2) 当网络流进入网络流处理模块后,首先

会被过滤掉垃圾数据, 然后由网络流解析功能进行解析, 以判断当前网络流中各种模态网络的组成。INT 数据采集模块会将收集到的网络状态信息发送给控制平面的节点控制模块。此时, 节点控制模块会以表的形式将这些网络信息下发到数据平面。

(3) 网络流通过网络流处理模块转发到模态处理模块, 该模块会对到达的网络流进行分流, 通过解析数据包来确定该网络流所属的网络模态, 并使用布隆过滤器根据数据包携带的信息来确定网络流所需的网络功能, 再将网络流转发到服务功能模块。

(4) 服务功能模块会对网络流进行处理, 它会根据虚拟网络 and 对应网络模态原有的转发机制确定该网络流的虚拟转发端口, 并按照需求提供对应的网络功能。若该节点没有对应模态的服务功能, 则直接进行二层转发。

(5) 数据转发模块会根据虚拟端口和物理端口的映射表, 确定该网络流可用的物理端口。结合网络流的状态信息和当前物理端口的信息, 数据转发模块会查询控制平面下发的转发表, 为网络流选择转发收益最大的端口进行转发, 如果无法确定合适的端口, 则会丢弃该数据。

2 基于强化学习的多模态网络转发算法

本节首先介绍了系统建模, 然后描述了该问题的马尔可夫决策过程 (Markov decision process, MDP) 建模、优化目标以及约束条件, 最后提出了适用于可编程数据平面的强化学习的转发算法。

2.1 系统建模

用无向图 $G=(V,E)$ 来抽象化网络模型, 其中 V 表示网络中的节点集, E 表示网络链路的集合, 多模态网络集合为 $W=\{w_1, w_2, \dots, w_N\}$ 。多模态网络流到达网络节点后, 经过网络识别模块解析网

络流的模态, 再交给对应模态的服务功能进行处理, 以确定该网络流转发的虚拟端口, 最终进入队列, 等待网络节点转发。

节点上的多模态网络流集合表示为:

$$F = \left\{ (f_1^1, f_1^2, f_1^3, \dots), (f_2^1, f_2^2, f_2^3, \dots), \dots, (f_N^1, f_N^2, f_N^3, \dots) \right\} \quad (1)$$

其中, N 表示网络模态类 W_N 。

虚拟端口集合表示为:

$$M = \left\{ (m_1^1, m_1^2, \dots), (m_2^1, m_2^2, \dots), \dots, (m_N^1, m_N^2, \dots) \right\} \quad (2)$$

物理端口集合表示为:

$$P = \{p_1, p_2, p_3, \dots\} \quad (3)$$

使用变量 y 表示端口之间的映射关系, 即:

$$y_{m_i, p_k} =$$

$$\begin{cases} 1, \text{虚拟端口 } m_i^j \text{ 和物理端口 } p_k \text{ 存在映射关系} \\ 0, \text{虚拟端口 } m_i^j \text{ 和物理端口 } p_k \text{ 不存在映射关系} \end{cases} \quad (4)$$

使用变量 η 表示流 f 与物理端口 p 之间的关系:

$$\eta_f^p = \begin{cases} 1, \text{流 } f \text{ 通过物理端口 } p \text{ 转发} \\ 0, \text{流 } f \text{ 不通过物理端口 } p \text{ 转发} \end{cases} \quad (5)$$

2.2 MDP 建模和问题描述

MDP 建模中的状态空间 S 、动作空间 A 和奖励函数 R 映射如下。

(1) 状态空间 S : $s_t \in S$, 用 $(B_t, C_t, W_t, I_t, Q_t)$ 表示当前状态 s_t , 其中 B_t 表示当前节点上端口的剩余带宽集合, C_t 表示网络流的类型 (大象流或老鼠流), W_t 表示当前流的模态, I_t 表示转发的虚拟端口, Q_t 表示当前节点各个端口的队列长度集合。

(2) 动作空间 A : 当前节点针对当前状态下流指定的虚拟端口和当前网络流的模态, 对照端口映射表确定可选择的物理端口, 这些物理端口的合集 P_t 为动作空间。



(3) 奖赏函数 R : $R=r_1b-r_2q$, b 为所选端口的剩余带宽, q 表示所选端口的队列长度, $r_1, r_2 \in [0, 1]$ 表示各个优化目标的权重, 可根据不同的网络特性进行调整。

在该场景下, 每个节点的控制器运行一个程序, 用于探测各个端口的网络情况, 并定时刷新数据, 用 b 表示端口的剩余带宽, 如 b_{p_1} 表示物理端口 p_1 的剩余带宽。用 d 表示节点之间的时延, 如 $d_{v_1, v_2}^{p_1}$ 表示 v_1 节点通过 p_1 端口到 v_2 节点的时延。每个流在进入交换机和离开交换机时在数据包的包头记录下时间, 到达地点后发送给控制器, 流 f_i^j 开始传输时间记为 T_{in}^{ij} , 结束时间记为 T_{out}^{ij} 。

以下是约束条件。

网络流的带宽需求不能超过所选端口的剩余带宽:

$$b_f \leq b_{p_f}, \forall f \in F \quad (6)$$

网络流选择的物理端口要符合虚拟端口的映射:

$$y_{m_f, p_f} = 1, \forall f \in F \quad (7)$$

网络流只能选择一个端口:

$$\sum_{p \in P} \eta_f^p = 1, \forall f \in F \quad (8)$$

最终的优化目标为最小化转发时延, 可表示为:

$$\min \sum_{i=1}^N \sum_{f \in F_i} (T_{out}^{ij} - T_{in}^{ij}) \quad (9)$$

2.3 基于强化学习的转发算法

可编程交换机的匹配/动作 (match/action, M/A) 表是一种用于指导数据包处理的关键数据结构^[14]。在 M/A 表中, 每个条目都包含一个匹配条件和一个对应的动作。当数据包到达可编程交换机时, 交换机会根据数据包的头部字段匹配 M/A 表中的条件, 并执行相应的动作。由于可编程交换机可以通过控制平面对表进行数据更新, 且不

需要重启交换机, 多模态网络转发算法可以选择 Q -learning 算法为基础。 Q -learning 是一种简单而高效的强化学习算法, 尤其适用于解决离散动作空间的问题。它不需要对环境进行建模, 只需要通过不断的学习和更新 Q 值来实现智能体的决策。 Q -learning 与多模态网络转发的场景高度匹配, 每次转发的决策都是一次动作, 做出决策后环境就会发生变化, 下次转发时端口的剩余带宽和排队情况都会发生变化, 这是一个收益累积的过程, 强化学习的奖励机制对于需要长期累积收益的场景非常适合。

在数据平面实现强化学习算法的主要难点在于如何更新 Q 表, 包括如何存储 Q 表和计算 Q 值^[15]。考虑可编程交换机数据平面的计算能力和资源有限, 本文将复杂的计算操作和 Q 表更新操作转移到控制平面, Q 表则存储在数据平面的 M/A 表中。强化学习转发算法流程如图 3 所示。首先输入一个网络流和当前网络流的状态 s' , 以及上一步的状态 s 和动作 a ; 然后, 交换机通过 M/A 表匹配获取本次的动作 a' , 完成网络流的转发; 再将更新 Q 值表需要的信息发送给控制平面; 最后, 控制平面计算 Q 值, 更新数据平面中的表项, 将实际场景中收集的数据进行离线训练, 训练完成后生成 Q 表。

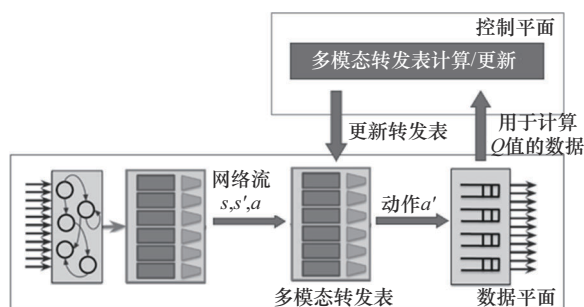


图3 强化学习转发算法流程

具体的转发流程为: 首先将生成的 Q 表下发到可编程交换机中, 当有网络流到达转发模块时, 获取该网络流的模态、需要转发的虚拟端

口、网络流的类型（判断是大象流还是老鼠流），以及当前各个物理端口的剩余带宽和队列长度等状态信息；查询 Q 表，选择在当前状态下所对应的动作中 Q 值最大的动作 a ；执行动作 a ，按照选择的端口转发网络流，持续监控网络是否有新的网络流出现。

基于 Q -learning 的多模态网络转发算法伪代码如下。

输入 S, F

输出 P

初始化状态空间 S ，奖励值 R 和 Q 值表

For episode in MAX_EPISODE:

For f in F :

初始化动作空间 A

查看动作空间

根据贪婪策略从动作空间 A 中选择动作

读取转发端口的队列长度

计算奖励值 R

记录下一个状态 s'

更新 Q 值表， $Q'(s, a) \leftarrow Q(s, a) + \alpha[r + \gamma \max_a Q(s', a') - Q(s, a)]$

//等待下一个网络流

End

End

3 实验与结果

3.1 实验环境

本文在 CENI 上开展仿真实验，具体实验环境及参数设置如下。

- (1) Ubuntu20.04 LTS。
- (2) P4 编译器 P4c^[16]，软件交换机 Bmv2。
- (3) Tofino 可编程交换机。
- (4) 使用 Mininet 搭建网络拓扑。实验采用 6 个主机、1 个交换机和 3 个路由器来验证算法，构

建了如图 4 所示的网络拓扑，其中 P4 SW 为 Bmv2 软件交换机，负责运行多模态网络转发算法，RT1、RT2、RT3 为普通路由器， $h1$ 、 $h2$ 、 $h3$ 、 $h4$ 、 $h5$ 和 $h6$ 为用户端。链路带宽设置为 30 Mbit/s，时延为 10 ms。

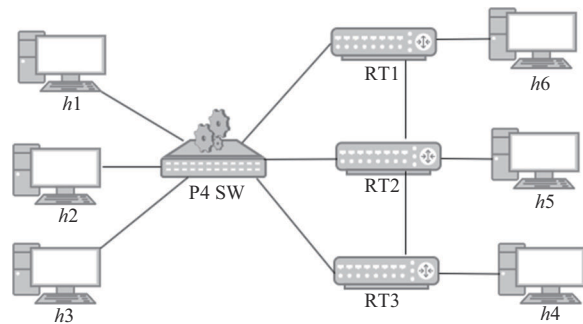


图4 网络拓扑

由于目前没有多模态网络流的流量数据集，本实验使用 iperf3 工具向交换机发送不同模态的数据流来模拟多模态网络环境，目前选择的是 3 种模态，每种模态的网络流占比相同，每秒到达的流量符合泊松分布。在很多研究中，大象流通常占总流量的 10%~20%^[17]，但可能占用 80%~90% 的带宽，老鼠流则占据绝大多数流量，可能达到 80%~90%，因此，本文设置网络流中大象流和老鼠流的比例为 1:5。

为了验证硬件交换机的性能优势，本文使用转发时延作为评估指标，并将其与软件交换机进行对比。为了评估算法的有效性，本文采用平均队列长度和吞吐量作为评估指标，选择最短路径优先算法和最大带宽路径算法进行比较。最短路径优先算法因其具有简单性、适用性广泛、计算效率高以及资源消耗低的特点，适合作为对比对象。最大带宽路径算法能够优化网络带宽利用，避免网络拥塞，提升带宽利用率，特别是在需要关注数据传输质量的网络环境中，最大带宽路径算法是一个重要的对比选项。所有的仿真数据结果均基于多次重复实验的平均值得出，以保证实验数据的客观性和准确性。



具体训练过程如下：在软件交换机 Bmv2 上启动对应的 P4 程序，使用 iperf3 工具从 $h1$ 、 $h2$ 和 $h3$ 客户端向 Bmv2 发送多模态网络流。交换机根据转发表确定转发端口，每次转发时读取转发端口的队列长度计算奖励值，对多模态转发表进行更新。多次迭代直到训练结果趋于稳定。

3.2 实验结果与分析

3.2.1 转发时延

使用软件和硬件方式分别测试 100 次转发时延，并绘制转发时延的累积分布函数 (cumulative distribution function, CDF)，如图 5 所示。由图 5 可知，使用硬件方式转发数据的时延明显低于软件方式，使用软件方式的时延平均值为 $2.57 \mu\text{s}$ 。在 Tofino 硬件交换机上平均转发时延为 $0.98 \mu\text{s}$ ，相比于软件方式提升了 61.87%。通过计算时延数据的方差，得到软件交换机的方差为 0.261，硬件交换机为 0.124，显然硬件交换机的转发时延更加稳定，而软件交换机容易受服务器上其他程序的影响，导致转发时延不稳定。Tofino 交换机转发时延低，主要是因为它使用专用集成电路 (application specific integrated circuit, ASIC) 硬件架构，能够快速处理数据包并实现并行处理，显著降低了处理时间。软件交换机则在用户空间运行，面临更高的上下文切换和调度时延。综合以上因素，Tofino 硬件交换机在转发时延方面显著优于软件。

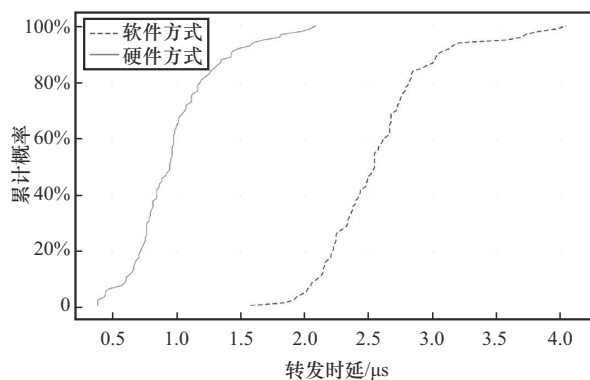


图5 转发时延的累积分布函数

3.2.2 平均队列长度

将训练得到的转发表部署到交换机上，实验过程中不对多模态网络转发表进行更新。使用与训练过程一样的流量，分别测试 3 种算法的平均队列长度和吞吐量。平均队列长度指标对比如图 6 所示，其中多模态网络转发算法的平均队列长度最短，相较于最大带宽路径算法和最短路径优先算法，其队列长度分别降低了 2.1% 和 7.9%。最短路径优先算法通常选择单一路径作为数据包的转发路径，这可能导致所有流量集中在同一条链路上，从而造成该链路的拥塞和队列长度增加。最大带宽路径算法在链路发生拥塞的情况下，容易选择不当的路径，导致队列加长。基于强化学习的转发算法能够根据网络环境的动态变化进行自我调整，提高链路的利用率，减少平均队列长度。

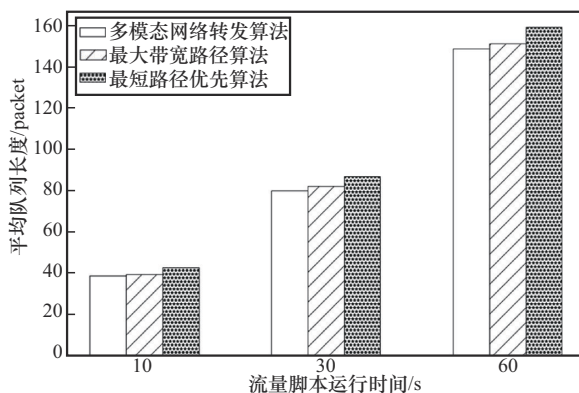


图6 平均队列长度指标对比

3.2.3 吞吐量

吞吐量指标对比如图 7 所示。由图 7 可知，多模态网络转发算法的吞吐量相比另外两种算法有一定的提升，其中，较最大带宽路径算法平均提升了 6.97%，较最短路径优先算法平均提升了 14.32%。最短路径优先算法容易出现多个网络流选择同一路径而造成链路拥塞，最终影响转发效率。最大带宽路径算法在为网络流选择路径时，没有考虑链路时延问题，带宽更大路径的流完成

时间可能更长。本文提出的多模态网络转发算法,通过多次迭代学习积累经验,能够降低上述问题发生的概率,以上实验结果充分说明了该算法具有良好的效果。

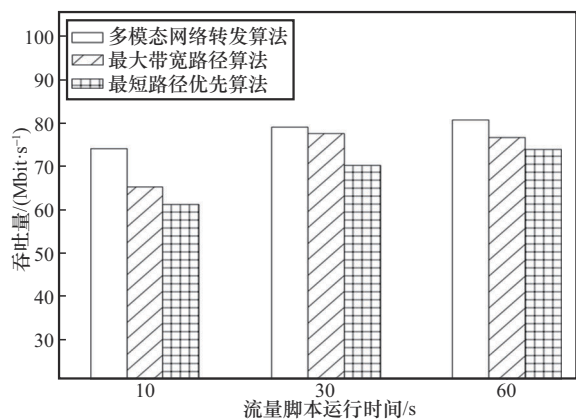


图7 吞吐量指标对比

综上所述,本文通过对比软件方式和硬件方式的转发时延,验证了使用硬件交换机能够有效降低转发时延。对方案中提出的转发算法进行仿真分析,对比了多种算法的平均队列长度和吞吐量,结果表明该算法具有更好的性能,进一步验证了本文所提多模态网络转发方案的有效性。

4 结束语

本文提出了一种流数据驱动的多模态网络转发方案以及基于强化学习的多模态网络转发算法。该方案设计了基于可编程数据平面的多模态网络转发方案,包含多个关键模块:INT数据采集、网络流处理、模态识别与分流、服务功能、数据转发、数据缓存以及节点控制。利用P4编程语言的协议无关特性,该方案实现了多个网络模态的共生共存,并通过可编程交换机硬件有效提高了转发效率。此外,针对可编程数据平面的特性,本文提出了基于强化学习的多模态网络转发算法,根据网络流状态智能选择转发端口,实现多模态网络的高效转发,

并在CENI中验证了该算法的有效性。本文旨在针对传统TCP/IP网络结构的不足,结合多模态网络提出一种网络转发方案,为多模态网络的研究提供了新的思路和方法。未来可以考虑在多交换机等更复杂的场景下,提升多模态网络的稳定性和效率。

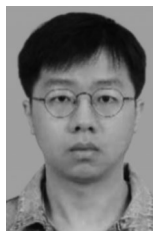
参考文献:

- [1] 郭江兴, 兰巨龙, 程东年. 新型网络体系结构[M]. 北京: 人民邮电出版社, 2014.
WU J X, LAN J L, CHENG D N. New network architecture [M]. Beijing: Posts and Telecom Press, 2014.
- [2] 胡宇翔, 伊鹏, 孙鹏浩, 等. 全维可定义的多模态网络体系研究[J]. 通信学报, 2019, 40(8): 1-12.
HU Y X, YI P, SUN P H, et al. Research on the full-dimensional defined polymorphic smart network[J]. Journal on Communications, 2019, 40(8): 1-12.
- [3] HLGREN B, DANNEWITZ C, IMBRENDA C, et al. A survey of information-centric networking[J]. IEEE Communications Magazine, 2012, 50(7): 26-36.
- [4] DONG F, CHENG C Q, GUO S D. Design and research on GeoIP[C]//Proceedings of 2010 14th International Conference on Computer Supported Cooperative Work in Design. Piscataway: IEEE Press, 2010: 13-17.
- [5] ZHANG H K, QUAN W, CHAO H C, et al. Smart identifier network: a collaborative architecture for the future Internet[J]. IEEE Network, 2016, 30(3): 46-51.
- [6] 李军飞, 胡宇翔, 伊鹏, 等. 面向2035的多模态网络技术发展路线图[J]. 中国工程科学, 2020, 22(3): 141-147.
LI J F, HU Y X, YI P, et al. Development roadmap of polymorphic intelligence network technology toward 2035[J]. Strategic Study of CAE, 2020, 22(3): 141-147.
- [7] MCCAULEY J, HARCHOL Y, PANDA A, et al. Enabling a permanent revolution in Internet architecture[C]//Proceedings of the ACM Special Interest Group on Data Communication. Piscataway: IEEE Press, 2019: 1-14.
- [8] DASGUPTA S, SHANNIGRAHI S, ZINK M. A hybrid NDN-IP architecture for live video streaming: from host-based to content-based delivery to improve QoE[J]. International Jour-



- nal of Semantic Computing, 2022(2):163-187.
- [9] ZHANG X, GONG L, XUN Y, et al. Centaur: a evolutionary design of hybrid NDN/IP transport architecture for streaming application[C]//2016 IEEE 7th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON). Piscataway: IEEE Press, 2016: 1-7.
- [10] CAROFIGLIO G, MUSCARIELLO L, JORDAN AUGÉ, et al. Enabling ICN in the Internet protocol: analysis and evaluation of the hybrid-ICN architecture[C]//Proceedings of the 6th ACM Conference on Information-Centric Networking. Piscataway: IEEE Press, 2019: 55 - 66.
- [11] REFAEI T, MA J, HA S, et al. Integrating IP and NDN through an extensible IP-NDN gateway[C]//Proceedings of the 4th ACM Conference on Information-Centric Networking. Piscataway: IEEE Press, 2017: 224 - 225.
- [12] OLIVER M, ROBERTO B, GÁBOR R, and STEFAN S. The Programmable data plane: abstractions, architectures, algorithms, and applications[J]. ACM Computing Surveys, 2021,54(4): 1-36.
- [13] BRADATSCHE L, KARGL F. Integration of security service functions into network-level access control[J]. IEEE Access, 2024(12): 197783-197815.
- [14] WU Q Q, LIU Q, JIA Z Q, et al. P4SQA: a P4 switch-based QoS assurance mechanism for SDN[J]. IEEE Transactions on Network and Service Management, 2023, 20(4): 4875-4886.
- [15] ZHENG C G, HONG X P, DING D M, et al. In-network machine learning using programmable network devices: a survey[J]. IEEE Communications Surveys & Tutorials, 2024, 26(2): 1171-1200.
- [16] BOSSHART P, DALY D, GIBB G, et al. P4: programming protocol-independent packet processors[J]. ACM SIGCOMM Computer Communication Review, 2014, 44(3): 87-95.
- [17] ESTAN C, VARGHESE G. New directions in traffic measurement and accounting: focusing on the elephants, ignoring the mice[J]. ACM Transactions on Computer Systems, 2003, 21(3): 270-313.

[作者简介]



吴剑松 (1997-), 男, 中国科学技术大学信息科学技术学院硕士生, 主要研究方向为计算机网络。



谭小彬 (1973-), 男, 博士, 中国科学技术大学信息科学技术学院副教授, 主要研究方向为未来网络架构、智能网络和多媒体通信。



袁莘智 (1998-), 男, 中国科学技术大学信息科学技术学院博士生, 主要研究方向为计算机网络。



据可欣 (2002-), 女, 中国科学技术大学信息科学技术学院硕士生, 主要研究方向为计算机网络。



骆汉光 (1988-), 男, 博士, 之江实验室副研究员, 主要研究方向为新型网络、网络安全等。