计算机研究与发展
Journal of Computer Research and Development
DOI: 10.7544/issn1000-1239.202110957
60(3): 606−618, 2023

面向 SD-DCN 的 OpenFlow 分组转发能效联合优化模型

罗 可[1] 曾 鹏[1] 熊 兵[1] 赵锦元[2]
[1](长沙理工大学计算机与通信工程学院 长沙 410114)
[2](长沙师范学院信息科学与工程学院 长沙 410199)
(luok@csust.edu.cn)

Joint Optimization Model of Energy Consumption and Efficiency Regarding OpenFlow-Based Packet Forwarding in SD-DCN

Luo Ke[1], Zeng Peng[1], Xiong Bing[1], and Zhao Jinyuan[2]
[1](*School of Computer Science and Communication Engineering*, *Changsha University of Science & Technology*, *Changsha* 410114)
[2](*School of Information Science and Engineering*, *Changsha Normal University*, *Changsha* 410199)

Abstract In software-defined networking (SDN), OpenFlow switches typically utilize ternary content addressable memory (TCAM) to store flow tables for fast wildcarding lookups. In order to promote packet forwarding performance, it usually requires enlarging TCAM capacity to store more entries. However, TCAM performs lookups in parallel matching, which brings about high energy consumption. Therefore, it is necessary to choose the appropriate TCAM capacity to balance the delay and energy consumption of packet forwarding. For the typical scenario of software-defined data center network (SD-DCN), we characterize the packet processing of an OpenFlow switch as a multi-priority M/G/1 queueing model, and build an OpenFlow-based packet forwarding delay model. Meanwhile, we establish a hit rate model of TCAM flow tables based on flow distribution characteristics, to solve the relational expression between packet forwarding delay and TCAM capacity. Considering the energy consumption of TCAM lookups, we establish a joint optimization model of energy consumption and efficiency regarding packet forwarding, and design an optimization algorithm to solve the optimal TCAM capacity. The experimental results indicate that our proposed delay model can more accurately characterize OpenFlow-based packet forwarding delay than existing models do. Meanwhile, we leverage the optimization algorithm to solve the optimal TCAM capacity with different parameter configurations, which provides a guideline for actual SD-DCN deployments.

Key words software-defined data center network (SD-DCN); joint optimization model; TCAM energy consumption; packet forwarding delay; optimal TCAM capacity

摘 要 在软件定义网络(software-defined networking, SDN)中,OpenFlow 交换机通常采用三态内容可寻址存储器(ternary content addressable memory, TCAM)存储流表,以支持快速通配查找.然而,TCAM 采用并行查找方式,查找能耗高,因此有必要为 OpenFlow 交换机选择合适的 TCAM 容量,以平衡分组转发时延和能耗.针对软件定义数据中心网络(software-defined data center network, SD-DCN)这一典型应用场景,利用多优先级 M/G/1 排队模型刻画 OpenFlow 交换机的分组处理过程,进而建立 OpenFlow 分组转发时延模型.同时,基于网络流分布特性,建立 TCAM 流表命中率模型,以求解 OpenFlow 分组转发时延与 TCAM 容量的关系式.在此基础上,结合 TCAM 查找能耗,建立 OpenFlow 分组转发能效联合优化模型,并设计优化算

收稿日期: 2021-09-23; **修回日期**: 2022-06-10
基金项目: 国家自然科学基金项目 (11671125, 61972057,61502056)
This work was supported by the National Natural Science Foundation of China (11671125, 61972057,61502056).
通信作者: 熊兵 (xiongbing@csust.edu.cn)

法求解 TCAM 最优容量. 实验结果表明：所提时延模型比现有模型更能准确刻画 OpenFlow 分组转发时延. 同时，利用优化算法求解不同参数配置下的 TCAM 最优容量，为 SD-DCN 实际部署提供参考依据.

关键词 软件定义数据中心网络；联合优化模型；TCAM 能耗；分组转发时延；TCAM 最优容量

中图法分类号 TP393

软件定义网络（software-defined networking, SDN）作为一种新兴网络架构，将网络控制功能从数据交换设备中解耦出来，形成逻辑上集中的控制平面. SDN 控制平面负责构建并维护全局网络视图，根据网络拓扑结构制定流规则，并通过以 OpenFlow 为代表的南向接口协议下发到数据交换设备中，从而实现灵活高效的数据传输. 基于 OpenFlow 的 SDN 技术有力地打破了传统网络的封闭和僵化问题，大大提升了网络的灵活性、可管控性和可编程能力，被普遍认为是未来网络最有发展前景的方向之一[1-2]. 经过十来年的不断发展与演进，SDN 技术已广泛应用于各种网络场景，尤其是数据中心网络. 软件定义数据中心网络（software-defined data center network, SD-DCN）显著简化了网络功能管理，降低了部署成本，提高了数据传输效率，优化了网络应用性能，为数据中心的优化部署提供了新的技术方案[3-4].

数据中心作为承载海量数据处理的重要基础设施，广泛应用于在线购物、网络电视、短视频分享等数据密集型产业，目前已进入井喷式的高速建设和发展时期[5]. 然而，在数据中心规模飞速扩张的同时，能耗问题已成为制约其可持续发展的瓶颈. 在数据中心能耗中，网络设备产生的能耗占比可达 50% 以上[6]，主要由提供高速数据传输服务的交换机产生. 在 SD-DCN 网络中，OpenFlow 交换机通常采用三态内容可寻址存储器（ternary content addressable memory, TCAM）存储流表以支持快速通配查找，其查找能耗高（15~30 W/Mbit）[7-8]，约为静态存储器的 50 倍[9]. 同时，SD-DCN 网络采用等价多路径路由机制，将产生不少额外的流规则并存储到 TCAM 中，导致能耗问题更加凸显. 因此，如何设置 OpenFlow 交换机的 TCAM 容量，以平衡分组转发时延和 TCAM 查找能耗，是 SD-DCN 实际部署需要解决的一个关键问题.

目前已有不少研究工作关注 OpenFlow 交换机的 TCAM 能耗问题. 为降低 TCAM 查找能耗，部分研究人员采用内容可寻址存储器（CAM）缓存流表中的活跃流[10-11]，进而直接转发大多数分组，以大幅度减少 TCAM 流表查找操作. 然而，CAM 同样采用并行查找方式，查找能耗仍高. 也有研究者利用过滤器预测流表查找失败情形[9]，减少不必要的 TCAM 查找操作，但只能过滤每条流的首个分组，节能效果极为有限. 还有研究人员关注 OpenFlow 交换机的 TCAM 流表优化模型[12-13]，但却主要关注流超时设置、流规则放置等问题，缺乏对其最优容量的考量. 此外，许多工作关注 OpenFlow 交换机的分组转发时延，利用排队论构建 OpenFlow 分组转发性能模型[14-15]，但却同样忽略了 TCAM 容量对分组转发时延的影响.

针对上述问题，本文面向 SD-DCN 网络场景，拟提出一种 OpenFlow 分组转发能效联合优化模型，以求解 TCAM 最优容量. 为此，本文首先描述了一个典型的 SD-DCN 网络部署场景，分析其分组转发过程和排队特性，构建 OpenFlow 分组转发时延模型. 然后，根据数据中心网络中的流分布特性，建立 TCAM 命中率模型，进而求解 OpenFlow 分组转发时延与 TCAM 容量的关系式. 进一步，结合 TCAM 查找能耗，建立 OpenFlow 分组转发能效联合优化模型，并设计对应的优化算法求解 TCAM 最优容量. 最后，通过模拟实验评估本文所提 OpenFlow 分组转发时延模型，并利用优化算法求解不同参数配置下的 TCAM 最优容量.

本文的主要贡献有 4 个方面：

1）针对 SD-DCN 网络典型部署场景，在分析其分组到达和处理过程的基础上，为 OpenFlow 交换机构建了多优先级 M/G/1 排队模型，进而建立了一种更准确的 OpenFlow 分组转发时延模型；

2）基于 SD-DCN 网络中的流量分布特性，为 OpenFlow 交换机建立了 TCAM 命中率模型，以求解 OpenFlow 分组转发时延与 TCAM 容量的关系式；

3）以分组转发时延和能耗为优化目标，建立 OpenFlow 分组转发能效联合优化模型；

4）证明了优化目标函数的凸性质，进而设计了优化算法求解 TCAM 最优容量，为 SD-DCN 实际部署提供有效指导.

1 相关工作

针对 OpenFlow 交换机的 TCAM 能耗问题，部分研究人员设计了 TCAM 流表节能查找方案. Congdon

等人[10]根据网络流量局部性, 为交换机的每个端口设置 CAM 缓存, 存储包签名与流关键字之间的映射关系, 以预测包分类结果, 使大部分分组绕过 TCAM 查找过程. 然而, CAM 存储器同样采用并行查找方式, 查找能耗仍高. 针对 OpenFlow 多流表的流水线查找模式, Wang 等人[11]利用马尔可夫模型选取每个流表中的活跃表项, 并集中存放到流水线前的 Pop 表中, 使大部分分组查找命中 Pop 表, 以避免复杂的多流表查找过程. Kao 等人[9]提出了基于布鲁姆过滤器的流表查找方案 TSA-BF, 通过优化设计布鲁姆过滤器以预测流表查找失败情形, 使新流分组绕过 TCAM 失败查找操作. 但该方案只能过滤每条流的首个分组, 节能效果极为有限.

同时, 许多研究工作利用排队论建立 OpenFlow 分组转发时延模型. 针对 OpenFlow 交换机的分组处理过程和 SDN 控制器的 Packet-in 消息处理过程, Xiong 等人[14]将其分别建模为 M^X/M/1 和 M/G/1 排队模型, Abbou 等人[15]则分别建模成 M/H_2/1 排队模型和 M/M/1 排队模型, Chilwan 和 Jiang[16]分别建模成 M/M/1/∞和 M/M/1/K 排队模型, 进而推导 OpenFlow 平均分组转发时延. 针对多控制器部署场景, Zhao 等人[17]为 SDN 控制器集群和 OpenFlow 交换机分别建立 M/M/n 和 M/G/1 排队模型, 分析平均分组转发时延, 并结合 SDN 控制器集群的部署成本, 求解最优控制器数量. 然而, 文献 [14–17] 所述模型未考虑不同类型网络分组的优先级差异. 对此, Rahouti 等人[18]将 SDN 网络建模成带反馈机制的双队列排队系统, 将分组划分成多个优先级队列, 以提供差异化的 QoS 服务. Li 等人[19]为 OpenFlow 虚拟交换机的分组转发过程建立排队系统, 以分析丢包率、流表查找失败概率、分组转发时延等关键性能指标, 进而利用多线程处理、优先制队列设置、分组调度策略和内部缓存区设置等多种方法优化分组转发性能. 然而, 文献 [18–19] 所述模型均未考虑 TCAM 容量对 OpenFlow 分组转发时延的影响.

进一步, 已有部分研究工作关注 OpenFlow 交换机的 TCAM 流表优化模型. Metter 等人[20-21]建立基于 M/M/∞排队系统的流表解析模型, 分析不同网络流量特性下流超时时长对 Packet-in 发送消息速率和流表占用率的影响. 进一步, AlGhadhban 等人[12]为流安装过程建立基于类生灭过程的解析模型, 分析流表匹配概率的影响因素, 进而推导不同流超时时长下的流表容量. Zhang 等人[13]采用 M/G/c/c 排队系统分析流超时时长对流规则的截断时间、冗余时间和安装失败率的影响, 进而提出自适应流超时算法, 以提高 TCAM 流表资源利用率. 然而, 文献 [12–13, 20–21] 所述模型未考虑 TCAM 容量优化设置问题. 对此, Shen 等人[22]将流表项生命周期划分为 Packet-in 消息发送、控制器处理和流表项超时 3 个阶段, 并分别建立 M/M/1, M/M/1 和 M/G/c/c 排队模型, 进而提出流表空间估计模型, 求解流安装失败概率约束下的 TCAM 最小容量. 然而, 该工作仅关注 TCAM 容量对流安装成功率的影响, 却没有考虑流表命中率和分组转发时延等关键性能指标. 本文则考虑 TCAM 容量对分组转发时延和能耗的影响, 建立 OpenFlow 分组转发能效联合优化模型, 求解 TCAM 最优容量, 为 SD-DCN 实际部署提供参考依据.

2 面向 SD-DCN 的 OpenFlow 分组转发时延模型

本节在描述 SD-DCN 网络典型部署场景的基础上, 为 OpenFlow 交换机的分组处理过程构建多优先级 M/G/1 排队模型, 进而建立 OpenFlow 分组转发时延模型.

2.1 SD-DCN

随着在线购物、网络电视、视频分享、搜索引擎等数据密集型应用的日益盛行, 数据中心作为承载海量数据处理的重要基础设施, 其规模不断扩大, 网络通信量正快速增长[23]. 传统数据中心网络因扩展性差、缺乏灵活的管理, 无法满足数据处理业务中日益增长的网络需求. 基于 OpenFlow 的 SDN 技术将控制逻辑与数据转发相解耦, 进而对网络设备进行逻辑上集中的管理和控制, 并为上层应用提供统一的编程接口, 大大提升了网络的灵活性、开放性和可管控能力, 为构建高性能数据中心网络提供了新的解决思路. SD-DCN 具备动态路由控制、服务质量管理、安全智能连接等技术优势, 降低了数据中心网络的部署成本, 有助于构建更加灵活高效的数据中心, 已成为一种新的发展趋势[24].

图 1 描述了一种典型的 SD-DCN 网络架构, 分为基础设施平面、数据平面、控制平面和应用平面. 基础设施平面包含众多服务器, 提供强大的存储和计算能力. 数据平面大多采用 fat-tree 拓扑结构组网[25-26], 交换机自下而上分为边缘交换机、汇聚交换机和核心交换机, 为众多服务器提供高性能的网络互联和数据传输服务. 在数据平面中, 每个交换机根据控制平面下发的流规则, 快速转发网络分组. 控制平面根据上层应用需求, 基于全局网络视图制定流规则, 并

通过以 OpenFlow 为代表的南向接口协议下发到各个交换机中，指导分组转发行为. 应用平面基于控制平面提供的北向接口，实现服务编排、安全控制、QoS 管理、资源调度等功能.

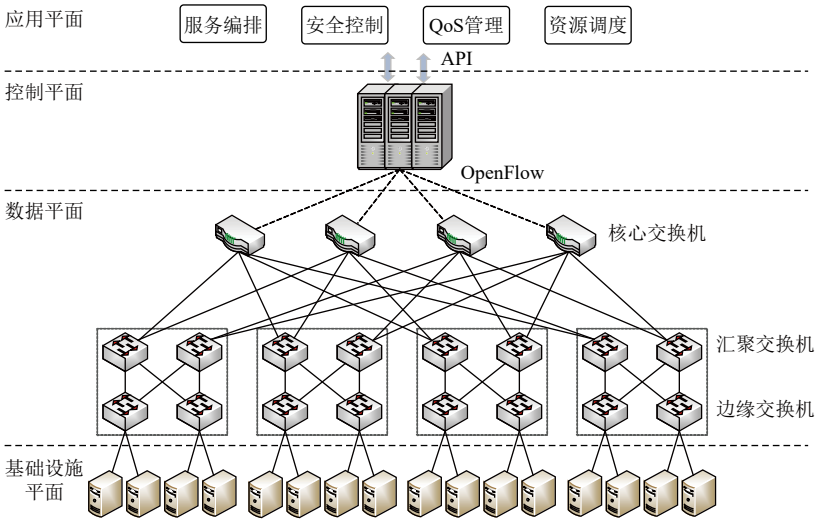

Fig. 1 Network architecture of SD-DCN

图 1 SD-DCN 网络架构

在图 1 所示的 SD-DCN 网络场景中，OpenFlow 交换机根据 SDN 控制器制定的流规则转发分组. 对于每个到达的分组，交换机从分组首部提取匹配字段，进而查找流表. 若查找成功，则依据流表项中给定的动作集转发分组. 否则，交换机判定该分组属于新流，将其首部信息或整个分组封装成流安装请求发送到控制器. 控制器根据全局网络视图生成流规则，并下发到流传输路径上的各个交换机中. 交换机将流规则安装至流表，并据此转发该流后续到达的分组.

2.2 OpenFlow 交换机排队模型

在 SD-DCN 数据平面中，来自服务器的大量分组汇聚到 OpenFlow 交换机，形成队列等待处理. OpenFlow 交换机的分组排队和处理过程如图 2 所示. 对于到达的每个分组，交换机首先将其缓存到入端口队列，然后逐个解析分组首部信息，提取关键字段，以计算流标识符. 再根据流标识符查找 OpenFlow 流表，以定位对应的流表项. 若查找成功，交换机在 ACL 应用、计数器更新等一系列相互独立的操作后，将分组发送到出端口等待转发；若查找失败，交换机发送 Packet-in 消息给控制器，待收到相应的 Flow-mod 消息后，将其中的流规则安装到 TCAM 流表，并据此转发该流后续到达的分组. 控制器下发的 Flow-mod 消息同样以分组的形式到达交换机，但优先级高于数据分组，以保证分组转发行为的一致性和高效性.

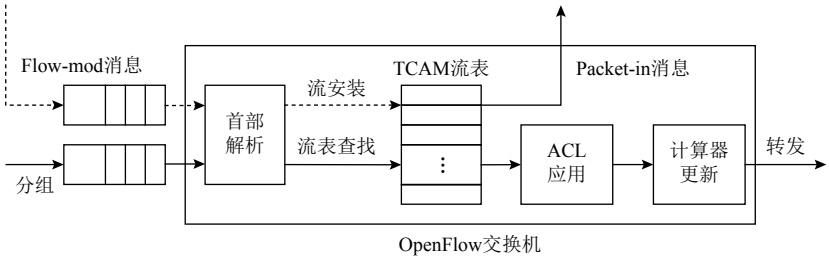

Fig. 2 The queueing and processing of packet in the OpenFlow switch

图 2 OpenFlow 交换机的分组排队和处理过程

网络测量结果表明：在数据中心等大规模网络场景下，流量汇聚程度高，并发流数量庞大，趋于相互独立[27-28]. 因此，OpenFlow 交换机的分组到达过程和流到达过程均可视为泊松过程[17,22]. 在 OpenFlow 分组转发过程中，所有新流的首个分组在 OpenFlow 交换机中的处理步骤相同，且处理过程相互独立. 根据 Burke 定理[29]，交换机发送的 Packet-in 消息流仍为泊松流. 假设每台交换机发送的 Packet-in 消息流相互独立，根据泊松流的可加性，汇聚到控制器的 Packet-in 消息流可叠加为泊松流. 控制器为每个

Packet-in 消息独立生成流规则后, 以 Flow-mod 消息的形式下发至流路径上的各个交换机. 因此, OpenFlow 交换机的 Flow-mod 消息到达过程同样为泊松过程. 假定控制器收到交换机 S_k 发送的 Packet-in 消息后, 生成流规则并下发 Flow-mod 消息给交换机 S_i 的概率为 δ_{ik}, 则有 δ_{ii}=1. 若交换机 S_k 的 Packet-in 消息发送速率为$\lambda_k^{(\mathrm{in})}$, 控制器共管理 K 台交换机, 则交换机 S_i 的 Flow-mod 消息到达速率如式(1)所示:

$$\lambda_i^{(\mathrm{m})}=\sum_{k=1}^{K}\delta_{ik}\lambda_k^{(\mathrm{in})}. \tag{1}$$

OpenFlow 交换机的分组处理过程可分解为首部解析、流表查找、计数器更新等相互独立的步骤, 不妨假设其共有 M 步. 假设交换机 S_i 中第 $j(1\leqslant j\leqslant M)$ 步的处理时间 T_{ij} 服从速率为 μ_{ij} 的负指数分布, 则其均值 $E(T_{ij})$=1/μ_{ij}, 方差 $D(T_{ij})$=1/μ_{ij}^2. 由于每个步骤相互独立, 因此交换机 S_i 的分组处理时间 T_i 服从一般分布, 其均值 $E(T_i)$ 如式(2)所示:

$$E(T_i)=\sum_{j=1}^{M}E(T_{ij})=\sum_{j=1}^{M}\frac{1}{\mu_{ij}}. \tag{2}$$

因此, 交换机 S_i 的平均分组处理速率$\mu_i^{(\mathrm{s})}$和分组处理时间方差$\sigma_i^{(\mathrm{s})^2}$分别如式(3)(4)所示:

$$\mu_i^{(\mathrm{s})}=\frac{1}{E(T_i)}=\frac{1}{\sum\limits_{j=1}^{M}1/\mu_{ij}}, \tag{3}$$

$$\sigma_i^{(\mathrm{s})^2}=D(T_i)=\sum_{j=1}^{M}D(T_{ij})=\sum_{j=1}^{M}\frac{1}{\mu_{ij}^2}. \tag{4}$$

基于以上排队分析, 本文将 OpenFlow 交换机的分组处理过程建模为多优先级 M/G/1 排队模型: 1)交换机 S_i 的 Flow-mod 消息和分组到达过程为泊松过程, 到达速率分别为$\lambda_i^{(\mathrm{m})}$和$\lambda_i^{(\mathrm{p})}$; 2) OpenFlow 交换机的入端口有 Flow-mod 消息高优先级队列和分组低优先级队列, 按非抢占式多优先级调度策略依次处理; 3)交换机 S_i 的分组处理时间服从一般分布, 分组处理速率和分组处理时间方差, 分别如式(3)(4)所示. 根据排队论, 可计算出 Flow-mod 消息和分组在交换机中的平均逗留时间为$W_i^{(\mathrm{m})}$和$W_i^{(\mathrm{p})}$, 分别如式(5)(6)所示:

$$W_i^{(\mathrm{m})}=\frac{\rho_i^{(\mathrm{s})}\bar{R}_i^{(\mathrm{s})}}{1-\rho_i^{(\mathrm{m})}}+\frac{1}{\mu_i^{(\mathrm{s})}}, \tag{5}$$

$$W_i^{(\mathrm{p})}=\frac{\rho_i^{(\mathrm{s})}\bar{R}_i^{(\mathrm{s})}}{\left(1-\rho_i^{(\mathrm{m})}\right)\left(1-\rho_i^{(\mathrm{m})}-\rho_i^{(\mathrm{p})}\right)}+\frac{1}{\mu_i^{(\mathrm{s})}}. \tag{6}$$

其中$\rho_i^{(\mathrm{s})}=(\lambda_i^{(\mathrm{m})}+\lambda_i^{(\mathrm{p})})/\mu_i^{(\mathrm{s})}$, $\rho_i^{(\mathrm{m})}=\lambda_i^{(\mathrm{m})}/\mu_i^{(\mathrm{s})}$, $\rho_i^{(\mathrm{p})}=\lambda_i^{(\mathrm{p})}/\mu_i^{(\mathrm{s})}$, $\bar{R}_i^{(\mathrm{s})}$为交换机处理分组时的剩余服务时间均值, 如式(7)所示:

$$\bar{R}_i^{(\mathrm{s})}=\frac{E(T_i^2)}{2E(T_i)}=\frac{\sigma_i^{(\mathrm{s})^2}+1/\mu_i^{(\mathrm{s})^2}}{2/\mu_i^{(\mathrm{s})}}. \tag{7}$$

2.3 OpenFlow 分组转发时延模型

根据 SD-DCN 网络中的分组转发流程, 结合 2.2 节所述的 OpenFlow 交换机排队模型, 可建立 OpenFlow 分组转发排队系统如图 3 所示. 在图 3 中, 网络分组以速率$\lambda_i^{(\mathrm{p})}$到达 OpenFlow 交换机 S_i, 在入端口处排队等待处理. 交换机 S_i 依据 TCAM 流表以速率$\mu_i^{(\mathrm{s})}$逐个处理分组. 假定 TCAM 流表命中率为 h_i, 则交换机 S_i 以速率$\lambda_i^{(\mathrm{in})}=\lambda_i^{(\mathrm{p})}(1-h_i)$发送 Packet-in 消息到 SDN 控制器. 控制器接收其管理的所有交换机发送的 Packet-in 消息流, 并统一存入队列等候处理. 控制器的 Packet-in 消息到达速率为$\lambda^{(\mathrm{c})}=\sum\limits_{i=1}^{K}\lambda_i^{(\mathrm{p})}(1-h_i)$, 处理速率为$\mu^{(\mathrm{c})}$. 控制器为每个 Packet-in 消息生成流规则后, 以速率$\lambda_i^{(\mathrm{m})}=\sum\limits_{k=1}^{K}\delta_{ik}\lambda_k^{(\mathrm{p})}(1-h_k)$下发 Flow-mod 消息到交换机 S_i. 交换机将所有流规则依次安装到流表, 进而转发新流的分组.

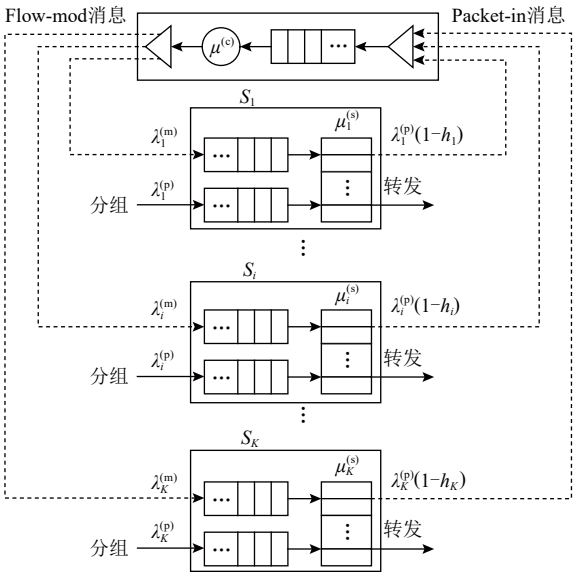

Fig. 3 OpenFlow-based packet forwarding queueing system

图 3 OpenFlow 分组转发排队系统

在上述 OpenFlow 分组转发排队系统中, 到达控制器的 Packet-in 消息流相互独立, 其到达过程为泊松过程, 且控制器对每个 Packet-in 消息的处理过程相互独立, 处理时间可视为服从负指数分布[30-31]. 因此, 可将控制器的 Packet-in 消息处理过程建模为 M/M/1 排队模型, 进而可知 Packet-in 消息在控制器中

的平均逗留时间 $W^{(c)}$ 如式(8)所示：

$$W^{(c)}=\frac{1}{\mu^{(c)}-\lambda^{(c)}}. \tag{8}$$

根据上述 OpenFlow 交换机的分组处理排队模型和控制器的 Packet-in 消息处理排队模型，可推导出交换机 S_i 的平均分组转发时延. 根据 OpenFlow 分组转发过程可知，交换机 S_i 中的分组转发过程可分为 2 种情况：直接转发和请求控制器安装流规则的间接转发. 分组直接转发时延即为分组在交换机中的逗留时间 $W_i^{(p)}$. 分组间接转发时延包含分组在交换机中的逗留时间 $W_i^{(p)}$、Packet-in 消息在控制器中的逗留时间 $W^{(c)}$、Flow-mod 消息在交换机中的逗留时间 $W_i^{(m)}$，以及 Packet-in 消息和 Flow-mod 消息在交换机到控制器之间的总传输时延 $W_i^{(t)}$. 因此，交换机 S_i 的平均分组转发时延可表达如式(9)所示：

$$D_i=\begin{cases} W_i^{(p)}, & \text{概率为}h_i. \\ W_i^{(p)}+W_i^{(m)}+W_i^{(t)}+W^{(c)}, & \text{概率为}(1-h_i). \end{cases} \tag{9}$$

将式(5)(6)(8)代入式(9)可得，交换机 S_i 的平均分组转发时延 D_i 如式(10)所示：

$$\begin{aligned} D_i=&h_iW_i^{(p)}+(1-h_i)\left(W_i^{(m)}+W_i^{(p)}+W_i^{(t)}+W^{(c)}\right)= \\ &\frac{\rho_i^{(s)}\bar{R}_i^{(s)}}{\left(1-\rho_i^{(m)}\right)\left(1-\rho_i^{(m)}-\rho_i^{(p)}\right)}+\frac{1}{\mu_i^{(s)}}+ \\ &(1-h_i)\left(\frac{\rho_i^{(s)}\bar{R}_i^{(s)}}{1-\rho_i^{(m)}}+\frac{1}{\mu_i^{(s)}}+\frac{1}{\mu^{(c)}-\lambda^{(c)}}+W_i^{(t)}\right). \end{aligned} \tag{10}$$

3 面向 SD-DCN 的 OpenFlow 分组转发能效联合优化模型

本节根据 SD-DCN 中的网络流分布特性建立 TCAM 命中率模型，进而结合 2.3 节所述的 OpenFlow 分组转发时延模型，建立 OpenFlow 分组转发能效联合优化模型，以求解 TCAM 最优容量.

3.1 TCAM 命中率模型

在分组交换网络中，网络流量存在明显的局部性特点，大部分分组集中分布在少数流中[32]. 以数据中心网络为例，众多测量研究结果表明：20% 的 top 流占据分组总数的 80% 以上[33]. 根据网络流量局部性，众多研究利用 Zipf 分布刻画网络流中的分组数量分布特性[34-37]. 假设网络中有 N 条流，则可按照流大小即流的分组数量依次递减排序为 $(f_1, f_2, \cdots, f_N)$. 根据 Zipf 分布可知，流 f_r 的分组数量 $Q(r)$ 与其大小排名 r(r=1, 2, …, N)存在式(11)所示的幂律关系.

$$Q(r)=\frac{C}{r^{\alpha}}. \tag{11}$$

其中 C 和 α 均为大于 0 的常数，α 表示分组在网络流中分布的倾斜程度. 假设 OpenFlow 交换机的 TCAM 容量为 n 条流表项，且存储所有网络流中排名靠前的 n 条流，则 TCAM 命中率 $h(n)$ 如式(12)所示：

$$h(n)=\frac{\sum_{r=1}^{n}Q(r)}{\sum_{r=1}^{N}Q(r)}=\frac{\sum_{r=1}^{n}r^{-\alpha}}{\Gamma_{\alpha}(N)}. \tag{12}$$

其中 $\Gamma_{\alpha}(N)=\sum_{r=1}^{N}r^{-\alpha}$. 假定网络流总数 N=20 000，对于不同的网络流分布倾斜度 α，根据式(12)可得 TCAM 命中率与 TCAM 容量的估计关系如图 4 所示.

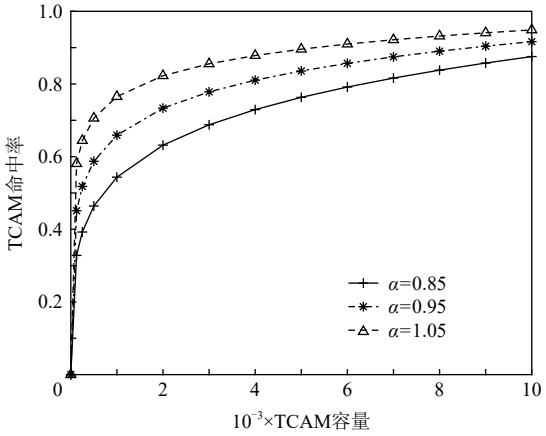

Fig. 4 Estimated relationship between TCAM hit rate and TCAM capacity

图 4 TCAM 命中率与 TCAM 容量的估计关系

从图 4 中可看出：α 越大，相同容量下的 TCAM 命中率越高，即相同数量 top 流所占据的分组比例越高，网络流量局部性越明显. 当 TCAM 容量为 4 000 条流表项，即可存储前 20% 的流时，若 α=0.95，则 TCAM 命中率可达 81.05%，与数据中心网络中的流量测量结果基本一致.

3.2 OpenFlow 分组转发能效联合优化模型

在 OpenFlow 分组转发过程中，每个到达交换机的分组都需要查找 TCAM 流表，进而实现分组转发处理. 根据 2.3 节所述的 OpenFlow 分组转发时延模型和 3.1 节所述的 TCAM 命中率模型可知：TCAM 容量越大，存储的流规则越多，TCAM 命中率越高，即 TCAM 流表查找成功的分组占比越大，平均分组转发时延越小. 因此，平均分组转发时延与 TCAM 容量呈负相关关系. 由于 TCAM 采用并行匹配方式查找整个流表，查找能耗基本上与其容量成正比，假定分

组转发过程中的其他能耗固定,则分组转发能耗可视为与 TCAM 容量呈正线性相关关系. 因此,可将 TCAM 容量作为决策变量,以分组转发时延和能耗为优化目标,建立能效联合优化模型求解 TCAM 最优容量.

对于 TCAM 容量为 n 的 OpenFlow 交换机,根据式(12)所示的 TCAM 命中率,定义 TCAM 流表命中失败概率 $q(n)=1-h(n)$. 进而结合式(10),可求出交换机的平均分组转发时延 $D(n)$:

$$D(n)=W^{(\mathrm{p})}(n)+q(n)\left[W^{(\mathrm{m})}(n)+W^{(\mathrm{c})}(n)+W^{(\mathrm{t})}\right]=\frac{\bar{R}^{(\mathrm{s})}\left[a_1q(n)+\rho^{(\mathrm{p})}\right]}{\left[1-a_1q(n)\right]\left[1-a_1q(n)-\rho^{(\mathrm{p})}\right]}+\frac{1}{\mu^{(\mathrm{s})}}+q(n)\left\{\frac{\bar{R}^{(\mathrm{s})}\left[a_1q(n)+\rho^{(\mathrm{p})}\right]}{1-a_1q(n)}+\frac{1}{\mu^{(\mathrm{s})}}+\frac{1}{\mu^{(\mathrm{c})}-a_0q(n)}+W^{(\mathrm{t})}\right\},\tag{13}$$

其中

$$a_0=\sum_{i=1}^{K}\lambda_i^{(\mathrm{p})},\ a_1=\frac{\sum_{k=1}^{K}\delta_k\lambda_k^{(\mathrm{p})}}{\mu^{(s)}},\bar{R}^{(\mathrm{s})}=\frac{\sigma^{(\mathrm{s})^2}+1/\mu^{(\mathrm{s})^2}}{2/\mu^{(\mathrm{s})}}.\tag{14}$$

同时,根据式(12)给出的 TCAM 命中率,可建立 OpenFlow 交换机的分组转发能耗模型. 假定每条 TCAM 流表项的平均查找能耗为 e_1,由于 TCAM 容量为 n 条流表项,且采用并行查找方式,因此每个分组的 TCAM 查找能耗为 ne_1. 若分组转发过程中的其他处理能耗为 e_0,则交换机的平均分组转发能耗 $E(n)$ 如式(15)所示:

$$E(n)=ne_1+e_0.\tag{15}$$

以式(13)和式(15)分别给出的平均分组转发时延和能耗为优化目标,可建立式(16)所示的 OpenFlow 分组转发能效联合优化模型,求解 TCAM 最优容量.

$$\arg\min_{n}(D(n),E(n)).\tag{16}$$

其中约束条件为

$$\begin{aligned}&D(n)<D_{\max},\\&E(n)<E_{\max},\\&n\in\mathbb{N}_+.\end{aligned}\tag{17}$$

式(16)优化模型包含 3 个约束条件:1) 平均分组转发时延 $D(n)$ 不能超过 QoS 规定的最大时延 $D_{\max}$;2) 平均分组转发能耗 $E(n)$ 不能超过上限值 $E_{\max}$;3) TCAM 容量 n 为正整数.

3.3 优化模型求解

OpenFlow 分组转发能效联合优化模型是一个不等式约束下的多目标优化模型. 在该模型中,以 TCAM 容量 n 为决策变量,平均分组转发时延 $D(n)$ 可通过定理 1 证明具有单调递减性,平均分组转发能耗 $E(n)$ 显然单调递增. 而约束条件限定了 $D(n)$ 和 $E(n)$ 的最大值,即决定了 TCAM 容量的最小值 $n_{\min}$ 和最大值 $n_{\max}$,进而可求得 $D(n)$ 和 $E(n)$ 的最小值分别为 $D_{\min}=D(n_{\max})$ 和 $E_{\min}=E(n_{\min})$.

定理 1. 对于式(13)中的平均分组转发时延 $D(n)$,若其参数均为正,且 $\rho^{(\mathrm{m})}+\rho^{(\mathrm{p})}=\rho^{(\mathrm{s})}<1$, $\rho^{(\mathrm{c})}=\lambda^{(\mathrm{c})}/\mu^{(\mathrm{c})}<1$,则 $D(n)$ 具有单调递减性,即 $D'(n)<0$.

证明. 对于式(13),不妨假设 n 为连续自变量,利用复合函数求导法,可得 $D(n)$ 的一阶导数:

$$D'(n)=D'(q(n))q'(n).\tag{18}$$

其中 $D'(q(n))$ 如式(19)所示:

$$D'(q(n))=\frac{\bar{R}^{(\mathrm{s})}a_1}{g(q(n))}+\frac{-g'(q(n))\bar{R}^{(\mathrm{s})}\left[a_1q(n)+\rho^{(\mathrm{p})}\right]}{g(q(n))^2}+\frac{\bar{R}^{(\mathrm{s})}\left[2a_1q(n)+\rho^{(\mathrm{p})}\right]}{1-a_1q(n)}+\frac{\bar{R}^{(\mathrm{s})}a_1\left[a_1q(n)+\rho^{(\mathrm{p})}\right]q(n)}{\left[1-a_1q(n)\right]^2}+\frac{1}{\mu_i^{(\mathrm{s})}}+\frac{1}{\mu^{(\mathrm{c})}-a_0q(n)}+\frac{a_0q(n)}{\left[\mu^{(\mathrm{c})}-a_0q(n)\right]^2}+W^{(\mathrm{t})}.\tag{19}$$

其中

$$g(q(n))=\left[1-a_1q(n)-\rho^{(\mathrm{p})}\right]\left[1-a_1q(n)\right],\tag{20}$$

$$g'(q(n))=a_1\left[2a_1q(n)+\rho^{(\mathrm{p})}-2\right].\tag{21}$$

在式(20)中,有

$$a_1q(n)=\frac{\sum_{k=1}^{K}\delta_k\lambda_k^{(\mathrm{p})}q(n)}{\mu^{(\mathrm{s})}}=\rho^{(\mathrm{m})}<1.\tag{22}$$

进而可知 $1-a_1q(n)>0$. 由于 $\rho^{(\mathrm{m})}+\rho^{(\mathrm{p})}=\rho^{(\mathrm{s})}<1$,则有 $1-a_1q(n)-\rho^{(\mathrm{p})}=1-\rho^{(\mathrm{s})}>0$. 带入式(20)(21)中可得

$$g(q(n))>0,\tag{23}$$

$$g'(q(n))=a_1\left[\rho^{(\mathrm{s})}-1+\rho^{(\mathrm{m})}-1\right]<0.\tag{24}$$

同时 $a_0q(n)=\sum_{i=1}^{K}\lambda_i^{(\mathrm{p})}q(n)=\lambda^{(\mathrm{c})}$,而 $\mu^{(\mathrm{c})}>\lambda^{(\mathrm{c})}$,因此可得

$$\mu^{(\mathrm{c})}-a_0q(n)=\mu^{(\mathrm{c})}-\lambda^{(\mathrm{c})}>0.\tag{25}$$

加之各项参数为正,因此式(19)等号右边的每项均为正,进而可得 $D'(q(n))>0$. 对 $q(n)$ 求导有

$$q'(n)=-\frac{n^{-\alpha}}{\Gamma_\alpha(n)}<0.\tag{26}$$

将上述结论带入式(18)可知:$D'(n)<0$.

证毕.

此时,可将优化目标 $D(n)$ 和 $E(n)$ 分别进行归一化处理,进而利用线性加权法将式(16)中的多目标优化函数转换成单目标优化函数 $f(n)$:

$$f(n)=\omega\frac{D(n)-D_{\min}}{D_{\max}-D_{\min}}+(1-\omega)\frac{E(n)-E_{\min}}{E_{\max}-E_{\min}},\tag{27}$$

其中 ω 为 OpenFlow 平均分组转发时延所占的权重. 该函数具有凸性质, 如定理 2 所证.

定理 2. 对于式(27)所示的目标函数, 若其参数均为正, 且 $\rho^{(s)}<1$, $\rho^{(c)}<1$, 则该函数具有凸性质, 即 $f''(n)>0$.

证明. 对 $f(n)$ 二阶求导有

$$f''(n)=\frac{\omega}{D_{\max}-D_{\min}}\left[D''(q(n))\frac{n^{-2\alpha}}{\Gamma_\alpha(N)^2}+D'(q(n))\frac{\alpha}{\Gamma_\alpha(N)}n^{-\alpha-1}\right]. \tag{28}$$

其中

$$\begin{aligned}D''(q(n))=&\frac{-2\bar{R}^{(s)}a_1g'(q(n))}{g(q(n))^2}+\\&\frac{2\bar{R}^{(s)}\left[a_1q(n)+\rho^{(p)}\right]\left[g'(q(n))^2-{a_1}^2g(q(n))\right]}{g(q(n))^3}+\\&\frac{2\bar{R}^{(s)}a_1}{1-a_1q(n)}+\frac{2\bar{R}^{(s)}a_1\left[2a_1q(n)+\rho^{(p)}\right]}{\left[1-a_1q(n)\right]^2}+\\&\frac{2\bar{R}^{(s)}{a_1}^2\left[a_1q(n)+\rho^{(p)}\right]q(n)}{\left[1-a_1q(n)\right]^3}+\\&\frac{a_0}{\left[\mu^{(c)}-a_0q(n)\right]^2}+\frac{2a_0q(n)}{\left[\mu^{(c)}-a_0q(n)\right]^3}.\end{aligned} \tag{29}$$

其中 $g(q(n))$ 和 $g'(q(n))$ 分别如式(20)和式(21)所示. 根据定理 1 的证明过程可知

$$g'(q(n))=a_1\left[\rho^{(s)}-1+\rho^{(m)}-1\right]<a_1\left[\rho^{(m)}-1\right]<0, \tag{30}$$

因此有

$$g'(q(n))^2={a_1}^2\left[\rho^{(s)}-1+\rho^{(m)}-1\right]^2>{a_1}^2\left[a_1q(n)-1\right]^2. \tag{31}$$

而

$$\begin{aligned}{a_1}^2g(q(n))=&{a_1}^2\left[1-a_1q(n)-\rho^{(p)}\right]\left[1-a_1q(n)\right]<\\&{a_1}^2\left[1-a_1q(n)\right]^2,\end{aligned} \tag{32}$$

因此, 在式(29)中, 有

$$g'(q(n))^2-{a_1}^2g(q(n))>0. \tag{33}$$

根据定理 1 的证明过程可知 $1-a_1q(n)>0$, $\mu^{(c)}-a_0q(n)>0$, $g(q(n))>0$, $g'(q(n))<0$. 加之各项参数均为正, 则式(29)等号右边的每项均为正, 进而可知 $D''(q(n))>0$. 同时, 根据定理 1 的证明过程可知 $D'(q(n))>0$. 代入式(28)可得 $f''(n)>0$.

证毕.

由于目标函数 $f(n)$ 具有凸性质, 因此可利用二分法在整数范围 $[n_{\min}, n_{\max}]$ 内搜索 TCAM 最优容量 n_{opt}, 使 $f(n)$ 取最小值. 算法 1 给出了 OpenFlow 分组转发能效联合优化模型的求解算法.

算法 1. OpenFlow 分组转发能效联合优化算法.

输入: 1)网络拓扑信息 $G(V,E)$; 2) OpenFlow 交换机的分组到达速率 $\lambda^{(p)}$, 每个步骤 j 的处理速率 $\lambda_j^{(s)}$, 控制器的 Packet-in 处理速率 $\mu^{(c)}$; 3) 平均每条 TCAM 流表项的查找能耗 e_1, 分组转发过程中其他处理步骤的能耗之和 e_0, 分组转发能耗上限 $E_{\max}$; 4) QoS 允许的分组转发时延上限 $D_{\max}$.

输出: 交换机的 TCAM 最优容量 n_{opt}.

① 计算交换机的分组处理速率 $\mu^{(s)}$ 和分组处理时间方差 $\sigma^{(s)^2}$;

② 计算控制器收到交换机 S_k 发送的 Packet-in 消息后, 生成流规则并下发 Flow-mod 消息给交换机 S_i 的概率为 δ_{ik};

③ if $\lambda^{(p)}>\mu^{(s)}$

④ exit;

⑤ end if

⑥ $n_{\max}=\lfloor(E_{\max}-e_0)/e_1\rfloor$;

⑦ $n_{\min}\leftarrow D_{\max}$;

⑧ while $n_{\min}\neq n_{\max}$ do

⑨ $n_{\mathrm{mid}}=\lfloor(n_{\min}+n_{\max})/2\rfloor$;

⑩ 计算目标函数 $f(n_{\mathrm{mid}})$, $f(n_{\mathrm{mid}}+1)$;

⑪ if $f(n_{\mathrm{mid}})>f(n_{\mathrm{mid}}+1)$

⑫ $n_{\min}=n_{\mathrm{mid}}+1$;

⑬ else

⑭ $n_{\max}=n_{\mathrm{mid}}$;

⑮ end if

⑯ end while

⑰ return $n_{opt}=n_{\min}$.

算法 1 分为 3 个步骤: 1)计算中间参数, 包括交换机处理速率 $\mu^{(s)}$(行①), 交换机 S_k 发送的 Packet-in 消息触发控制器下发 Flow-mod 消息到交换机 S_i 的概率 δ_{ik}(行②); 2)确定 TCAM 容量 n 的整数范围 $[n_{\min}, n_{\max}]$(行⑥⑦); 3)二分查找 TCAM 容量的最优值 n_{opt}(行⑧~⑰).

4 实　　验

本节首先通过模拟实验评估本文所提 OpenFlow 分组转发时延模型的准确性, 然后采用数值分析方法分析不同因素对分组转发时延的影响, 进而求解不同参数下的 TCAM 最优容量.

4.1 时延模型对比

实验采用 Mininet 平台模拟典型的 Fat-tree 网络拓扑结构[22], SDN 控制器共管理 10 台 OpenFlow 交换机, 包含 2 台核心交换机、4 台汇聚交换机、4 台边缘交换机. 其中, SDN 控制器采用 OpenDaylight, Open-

Flow 交换机采用 Open vSwitch v2.13. 在模拟实验中，利用 OFsuite 性能测试工具测得控制器的 Packet-in 消息处理速率为 21 kmsg/s，每台交换机的分组处理速率为 20 kpkt/s. 同时，将交换机的流表容量设置为 8 000 条流表项，流超时间隔为 10 s. 实验利用 Iperf 工具为每台交换机模拟产生不同速率的网络流量，其中新流分组占比约为 10%，进而测得平均分组转发时延如图 5 所示. 同时，将上述参数代入本文所提的 OpenFlow 分组转发时延模型和现有模型，其中控制器均采用 M/M/1 模型，OpenFlow 交换机分别采用多优先级 M/G/1 模型、M/G/1 模型[17]、M^k/M/1 模型[14]，进而计算出平均分组转发时延的估计值如图 5 所示.

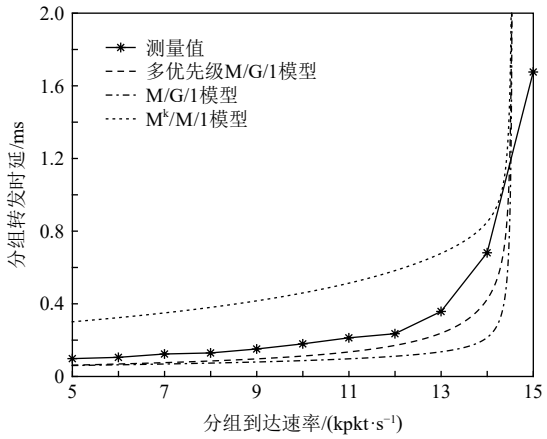

Fig. 5 Comparison of delay models with different packet arrival rates

图 5 不同分组到达速率下的时延模型对比

从图 5 中可以看出：与现有模型相比，本文所提基于多优先级 M/G/1 的 OpenFlow 分组转发时延模型具有更接近于测量值的估计时延. OpenFlow 交换机的分组处理过程包含多个相互独立的步骤，而 M^k/M/1 模型将分组处理时间简单地看作服从泊松分布，故其估计时延与测量值相差较大. M/G/1 模型可较为准确地估计 OpenFlow 交换机的分组处理时间，但在分组到达速率较大时，其估计时延明显较小. 这是因为该模型只考虑到达 OpenFlow 交换机的数据分组，而忽略了控制器下发的消息分组. 多优先级 M/G/1 模型则着重考虑了 Flow-mod 消息的分组处理时延，因而其分组转发时延估计值更接近于测量值.

实验采用 4.1 节所述参数，将 OpenFlow 交换机的分组到达速率 $\lambda_i^{(p)}$ 设为 10 kpkt/s，并不断调整 OpenFlow 交换机的流表容量值，可测得平均分组转发时延如图 6所示. 同时，将上述参数代入本文所提的 OpenFlow 分组转发时延模型和现有模型，计算出平均分组转发时延的估计值如图 6 所示.

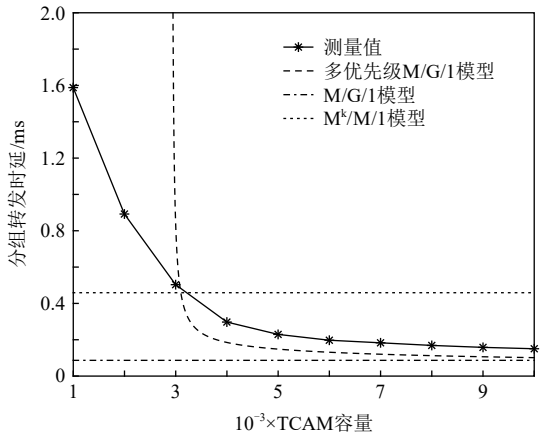

Fig. 6 Comparison of delay models with different flow table capacities

图 6 不同流表容量下的时延模型对比

从图 6 中可以看出：本文所提 OpenFlow 分组转发时延模型的估计时延比现有模型更接近于测量时延. 现有模型由于忽略了流表容量对分组转发时延的影响，其分组转发时延始终保持不变，估计不准确. 与此形成对照的是，随着流表容量的不断增大，本文所提模型的分组转发时延估计值逐步降低，并趋于稳定，与测量值接近. 这是因为交换机的流表命中率随流表容量的增大而升高，进而导致发送给控制器的 Packet-in 消息逐渐减少，其估计时延逐步接近于 M/G/1 模型. 特别地，当交换机的流表容量小于 3 000 条流表项时，模拟实验中的控制器负载过大，将会丢弃部分 Packet-in 消息，导致分组转发时延测量值急剧升高. 此时，对于 OpenFlow 分组转发时延模型，由于流表命中率过低，控制器的 Packet-in 消息到达速率高于其处理速率，导致模型失效.

4.2 分组转发时延

进一步，实验采用数值分析方法研究平均分组转发时延的主要影响因素. 实验参数设定为：假定分组在网络流中的分布倾斜程度 $\alpha = 0.95$，每台 OpenFlow 交换机的分组到达速率 $\lambda^{(p)} = 10$ kpkt/s，分组处理过程分为 10 个步骤，且每个步骤的处理速率相同，分组处理速率 $\mu^{(s)} = 20$ kpkt/s. 同时，SDN 控制器的 Packet-in 消息处理速率 $\mu^{(c)} = 21$ kmsg/s，每台控制器管理 10 台 OpenFlow 交换机.

实验设置不同的 TCAM 容量，可得到平均分组转发时延与分组到达速率之间的关系如图 7 所示. 从图 7 可看出：当 TCAM 容量一定时，分组到达速率越高，交换机和控制器的负载越大，平均分组转发时延越高. 同时，当分组到达速率一定时，交换机的 TCAM 容量越大，流表命中率越高，Packet-in 消息的发送速

率越低，其在控制器中的逗留时间越短，平均分组转发时延越低. 此外，TCAM 容量越大，允许的分组到达速率越高. 具体而言，当交换机的 TCAM 容量 n 分别为 4 000, 6 000, 8 000, 10 000 条流表项时，平均分组转发时延在分组到达速率 $\lambda^{(p)}$ 分别超过 10 kpkt/s, 11.6 kpkt/s, 12.8 kpkt/s, 13.8 kpkt/s 时急剧上升，且允许的最大分组到达速率分别为 10.7 kpkt/s, 12.6 kpkt/s, 14.3 kpkt/s, 15.3 kpkt/s.

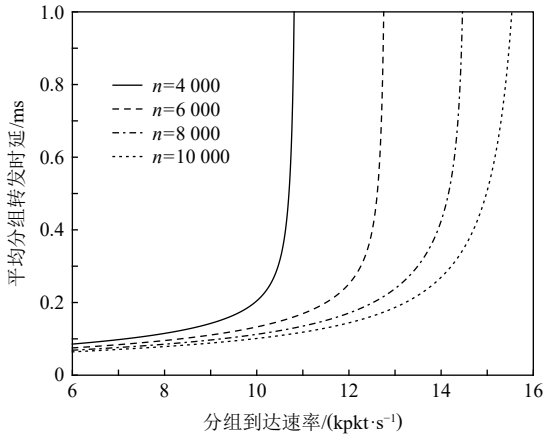

Fig. 7 Relationship between average packet forwarding delay and packet arrival rate

图 7 平均分组转发时延与分组到达速率的关系

实验设置不同的 TCAM 容量，可得到平均分组转发时延与分组处理速率之间的关系，如图 8 所示. 从图 8 可看出：当 TCAM 容量一定时，分组处理速率越高，分组在交换机中的逗留时间越短，平均分组转发时延越低；同时，当分组处理速率一定时，交换机的 TCAM 容量越大，流表命中率越高，发送给控制器的 Packet-in 消息越少，相应收到的 Flow-mod 消息也越少，平均分组转发时延越低. 此外，TCAM 容量越大，交换机的分组处理速率需求越低. 具体而言，当交换机的 TCAM 容量 n 分别为 4 000, 6 000, 8 000, 10 000 条流表项时，平均分组转发时延在分组处理速率 $\mu^{(s)}$ 分别小于 16 kpkt/s, 14.5 kpkt/s, 13.7 kpkt/s, 13.2 kpkt/s 时急剧上升，且交换机允许的最小分组处理速率分别为 14.3 kpkt/s, 13.4 kpkt/s, 12.7 kpkt/s, 12.1 kpkt/s.

实验设置不同的 TCAM 容量，可得到平均分组转发时延与 Packet-in 消息处理速率之间的关系如图 9 所示. 从图 9 中可看出：当 TCAM 容量一定时，Packet-in 消息处理速率越高，其在控制器中的逗留时间越短，平均分组转发时延越低. 同时，当 Packet-in 消息处理速率一定时，交换机的 TCAM 容量越大，流表容量越高，Packet-in 消息的发送速率越小，其在控制器中的逗留时间越短，平均分组转发时延越低. 此外，交换机的 TCAM 容量越大，控制器允许的 Packet-in 消息处理速率越低. 具体而言，当交换机的 TCAM 容量 n 分别为 4 000, 6 000, 8 000, 10 000 条流表项时，平均分组转发时延在 Packet-in 消息处理速率 $\mu^{(c)}$ 分别低于 18 kmsg/s, 15 kmsg/s, 12 kmsg/s, 8 kmsg/s 时急剧上升，且控制器允许的最低 Packet-in 消息处理速率分别为 18.9 kmsg/s, 14.1 kmsg/s, 10.6 kmsg/s, 7.9 kmsg/s.

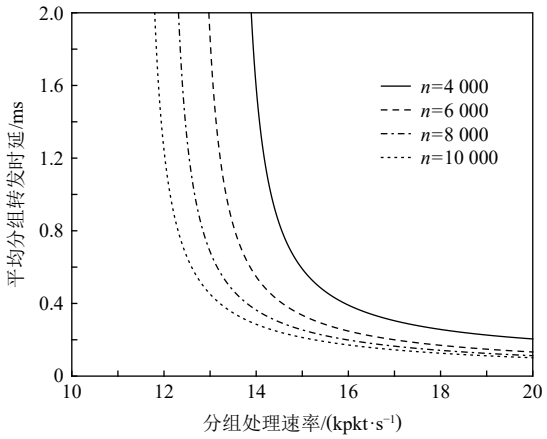

Fig. 8 Relationship between average packet forwarding delay and packet processing rate

图 8 平均分组转发时延与分组处理速率的关系

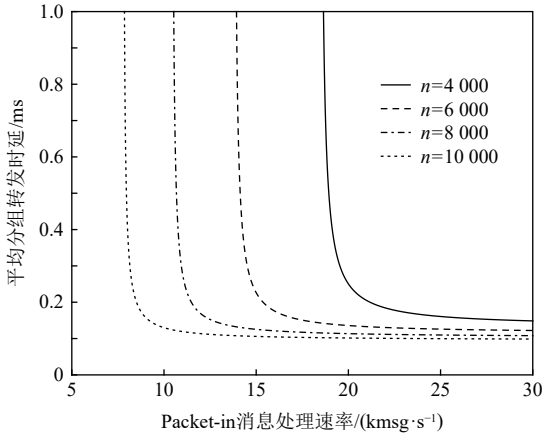

Fig. 9 Relationship between average packet forwarding delay and Packet-in message processing rate

图 9 平均分组转发时延与 Packet-in 消息处理速率的关系

4.3 TCAM 最优容量

对于 OpenFlow 分组转发能效联合优化模型，不妨假设每条 TCAM 流表项的平均查找能耗 e_1 为 1 个单位，其他处理步骤的能耗 e_0 为 3 000 个单位，分组转发能耗上限为 E_{max} 为 15 000 个单位，即最大 TCAM 容量为 12 000 条流表项. 同时，QoS 要求的最大分组转发时延 D_{max}=1.5 ms. 基于上述参数配置，将单目标优化函数 $f(n)$ 中的权重 ω 设置不同值，进而实现

OpenFlow 分组转发能效联合优化算法, 求解不同分组到达速率下的 TCAM 最优容量如图 10 所示.

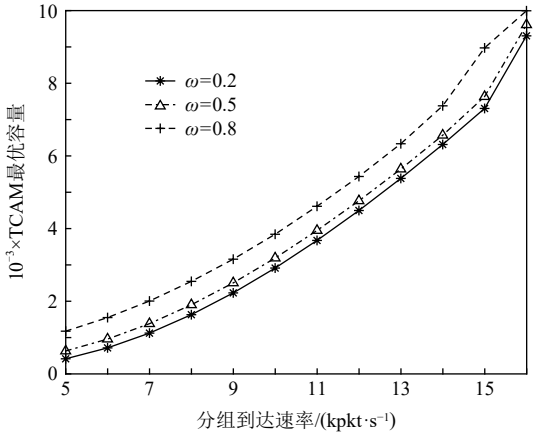

Fig. 10 Optimal TCAM capacity with different packet arrival rates

图 10 不同分组到达速率下的 TCAM 最优容量

从图 10 中可看出: 当每个交换机的分组到达速率增加时, TCAM 最优容量将随之增大, 以提高 TCAM 命中率, 保证 Packet-in 消息发送速率不会过高, 从而防止控制器过载, Packet-in 消息逗留时间过大. 具体而言, 当交换机的分组到达速率 $\lambda^{(p)}$ 分别为 5 kpkt/s, 10 kpkt/s, 15 kpkt/s, 且权重 $\omega = 0.8$ 时, TCAM 最优容量 n_{opt} 分别为 1 400, 4 600, 10 800 条流表项. 当分组到达速率超过 16 kpkt/s 时, 交换机需要继续增大 TCAM 容量, 以保证分组转发时延, 但分组转发能耗将会超出上限, 因而无解. 此外, 当分组到达速率一定时, 分组转发时延的权重越高, TCAM 最优容量越大, 以使 TCAM 命中率越高, 平均分组转发时延越小.

采用上述同样的参数配置, 并将交换机的分组到达速率 $\lambda^{(p)}$ 设为 10 kpkt/s, 进而求解不同分组处理速率下的 TCAM 最优容量, 如图 11 所示. 从图 11 中可看出: 当交换机的分组处理速率升高时, TCAM 最优容量将随之减小, 并趋于稳定. 这是因为在保证分组转发时延的情况下, 交换机的分组处理速率越高, 需要的 TCAM 容量越小. 具体而言, 当交换机的分组处理速率 $\mu^{(s)}$ 分别为 14 kpkt/s, 16 kpkt/s, 18 kpkt/s, 且权重 $\omega = 0.8$ 时, TCAM 最优容量 n_{opt} 分别为 8 900, 5 800, 4 900 条流表项. 此外, 当交换机的分组处理速率低于 12 kpkt/s 时, 由于逐步接近于分组到达速率, 平均分组转发时延将超出 QoS 规定的上限, 因而无解.

采用上述同样的参数配置, 并将交换机的分组处理速率 $\mu^{(s)}$ 设为 20 kpkt/s, 进而求解不同 Packet-in 消息处理速率下的 TCAM 最优容量, 如图 12 所示.

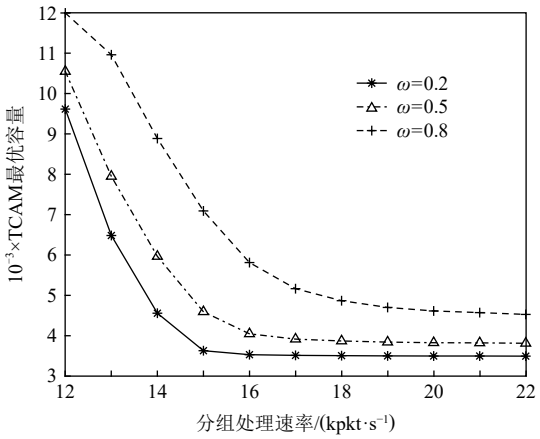

Fig. 11 Optimal TCAM capacity with different packet processing rates

图 11 不同分组处理速率下的 TCAM 最优容量

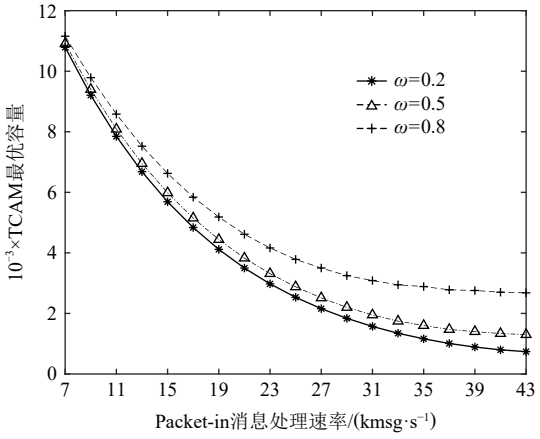

Fig. 12 Optimal TCAM capacity with different Packet-in message processing rates

图 12 不同 Packet-in 消息处理速率下的 TCAM 最优容量

从图 12 中可看出: 当控制器的 Packet-in 消息处理速率升高时, TCAM 最优容量将随之降低, 并趋于稳定. 这是因为控制器的 Packet-in 消息处理速率越高, 其允许的 Packet-in 消息到达速率越高, 进而交换机所需的 TCAM 容量越少. 具体而言, 当 Packet-in 消息处理速率 $\mu^{(c)}$ 分别为 15 kmsg/s, 25 kmsg/s, 35 kmsg/s, 且权重 $\omega = 0.8$ 时, TCAM 最优容量 n_{opt} 分别为 6 600, 3 800, 2 900 条流表项. 此外, 当 Packet-in 消息处理速率小于 7 kmsg/s 时, 交换机需要继续增加 TCAM 容量, 以提高 TCAM 命中率, 保证平均分组转发时延, 但分组转发能耗将会超出上限, 因而无解.

5 结 论

本文针对 SD-DCN 网络场景, 利用多优先级 M/G/1 排队模型刻画 OpenFlow 交换机的分组处理过

程，进而构建 OpenFlow 分组转发时延模型. 进一步，基于网络流分布特性建立 TCAM 命中率模型，求解 OpenFlow 分组转发时延与 TCAM 容量的关系式. 在此基础上，结合 TCAM 查找能耗，建立 OpenFlow 分组转发能效联合优化模型，以求解 TCAM 最优容量. 实验结果表明：与已有排队模型相比，本文所提时延模型更能准确估计 SD-DCN 网络场景下的 OpenFlow 分组转发性能. 同时，数值分析结果表明：交换机的 TCAM 容量和控制器的 Packet-in 消息处理速率对 OpenFlow 分组转发时延有着关键性的影响，而交换机的分组到达速率和分组处理速率影响较弱. 最后，通过 OpenFlow 分组转发能效联合优化算法，求解出不同参数配置下的 TCAM 最优容量，为 SD-DCN 网络的实际部署提供参考依据.

作者贡献声明：罗可提出研究思路，并设计研究方案，以及修订论文最终版本；曾鹏完善论文创新点，并完成模型的建立和推导，以及撰写论文初稿主要部分；熊兵设计了实验思路，以及审查和修改润色论文；赵锦元协助创新点推导和论文修改；所有作者都参与了实验分析和手稿撰写.

参 考 文 献

[1] Kreutz D, Ramos F M V, Verissimo P E, et al. Software-defined networking: A comprehensive survey[J]. Proceedings of the IEEE, 2014, 103(1): 14−76

[2] Hakiri A, Gokhale A, Berthou P, et al. Software-defined networking: Challenges and research opportunities for future Internet[J]. Computer Networks, 2014, 75: 453−471

[3] Cui Laizhong, Yu F R, Yan Qiao. When big data meets software-defined networking: SDN for big data and big data for SDN[J]. IEEE Network, 2016, 30(1): 58−65

[4] Li Dan, Chen Guihai, Ren Fengyuan et al. Data center network research progress and trends[J]. Chinese Journal of Computers, 2014, 37(2): 259−274 (in Chinese)
(李丹, 陈贵海, 任丰原, 等. 数据中心网络的研究进展与趋势[J]. 计算机学报, 2014, 37(2): 259−274)

[5] Xie Kun, Huang Xiaohong, Hao Shuai, et al. E^{3}MC: Improving energy efficiency via elastic multi-controller SDN in data center networks[J]. IEEE Access, 2016, 4: 6780−6791

[6] Yao Hong, Li Hui, Liu Chao, et al. Joint optimization of VM placement and rule placement towards energy efficient software-defined data centers[C] //Proc of IEEE Int Conf on Computer and Information Technology. Piscataway, NJ: IEEE, 2016: 204−209

[7] Kannan K, Banerjee S. Compact TCAM: Flow entry compaction in TCAM for power aware SDN[C] //Proc of Int Conf on Distributed Computing and Networking. Berlin: Springer, 2013: 439−444

[8] Jia Xuya, Li Qing, Jiang Yong, et al. A low overhead flow-holding algorithm in software-defined networks[J]. Computer Networks, 2017, 124: 170−180

[9] Kao Shengchun, Lee Dingyuan, Chen Tingsheng, et al. Dynamically updatable ternary segmented aging Bloom filter for OpenFlow-compliant low-power packet processing[J]. IEEE/ACM Transactions on Networking, 2018, 26(2): 1004−1017

[10] Congdon P T, Mohapatra P, Farrens M, et al. Simultaneously reducing latency and power consumption in OpenFlow switches[J]. IEEE/ACM Transactions on Networking, 2013, 22(3): 1007−1020

[11] Wang Cheng, Kim K T, Youn H Y. PopFlow: A novel flow management scheme for SDN switch of multiple flow tables based on flow popularity[J/OL]. Frontiers of Computer Science, 2020, 14(6) [2021-08-16].https://link.springer.com/article/10.1007/s11704-019-8417-5

[12] AlGhadhban A, Shihada B. Delay analysis of new-flow setup time in software defined networks[C/OL] //Proc of IEEE/IFIP Network Operations and Management Symp. Piscataway, NJ: IEEE, 2018 [2021-08-16].https://ieeexplore.ieee.org/abstract/document/8406231

[13] Zhang Linlian, Lin Rongping, Xu Shizhong, et al. AHTM: Achieving efficient flow table utilization in software defined networks[C] //Proc of IEEE Global Communications Conf. Piscataway, NJ: IEEE, 2014: 1897−1902

[14] Xiong Bing, Yang Kun, Zhao Jingyuan, et al. Performance evaluation of OpenFlow-based software-defined networks based on queueing model[J]. Computer Networks, 2016, 102: 172−185

[15] Abbou A N, Taleb T, Song J S. Towards SDN-based deterministic networking: Deterministic E2E delay case[C/OL] //Proc of IEEE Global Communications Conf. Piscataway, NJ: IEEE, 2021 [2021-08-16].https://ieeexplore.ieee.org/document/9685656

[16] Chilwan A, Jiang Y. Modeling and delay analysis for SDN-based 5G edge clouds[C/OL] //Proc of IEEE Wireless Communications and Networking Conf. Piscataway, NJ: IEEE, 2020 [2021-08-16]. https://ieeexplore.ieee.org/abstract/document/9120849

[17] Zhao Jinyuan, Hu Zhigang, Xiong Bing, et al. Modeling and optimization of packet forwarding performance in software-defined WAN[J]. Future Generation Computer Systems, 2020, 106: 412−425

[18] Rahouti M, Xiong Kaiqi, Xin Yufeng, et al. QoSP: A priority-based queueing mechanism in software-defined networking environments[C/OL] //Proc of IEEE Int Performance, Computing, and Communications Conf. Piscataway, NJ: IEEE, 2021 [2021-08-16]. https://ieeexplore.ieee.org/document/9679409

[19] Li Fuliang, Zheng Naigong, Zhang Yuchao, et al. Queueing theory over OpenvSwitch: Performance analysis and optimization[C] //Proc of Int Conf on Web Services. Berlin: Springer, 2021: 46−62

[20] Metter C, Seufert M, Wamser F, et al. Analytic model for SDN controller traffic and switch table occupancy[C] //Proc of the 12th Int Conf on Network and Service Management. Piscataway, NJ: IEEE,

2016: 109−117

[21] Metter C, Seufert M, Wamser F, et al. Analytical model for SDN signaling traffic and flow table occupancy and its application for various types of traffic[J]. IEEE Transactions on Network and Service Management, 2017, 14(3): 603−615

[22] Shen Gengbiao, Li Qing, Ai Shuo, et al. How powerful switches should be deployed: A precise estimation based on queuing theory[C]//Proc of IEEE Conf on Computer Communications. Piscataway, NJ: IEEE, 2019: 811−819

[23] Cai Yueping, Wang Changping. Software defined data center network with hybrid routing[J]. Journal on Communications, 2016, 37(4): 44−52 (in Chinese)
(蔡岳平, 王昌平. 软件定义数据中心网络混合路由机制[J]. 通信学报, 2016, 37(4): 44−52)

[24] Xu Guan, Dai Bin, Huang Benxiong, et al. Bandwidth-aware energy efficient routing with SDN in data center networks[C]//Proc of the 17th IEEE Int Conf on High Performance Computing and Communications, the 7th IEEE Int Symp on Cyberspace Safety and Security, and the 12th IEEE Int Conf on Embedded Software and System. Piscataway, NJ: IEEE, 2015: 766−771

[25] Pang Junjie, Xu Gaochao, Fu Xiaodong. SDN-based data center networking with collaboration of multipath TCP and segment routing[J]. IEEE Access, 2017, 5: 9764−9773

[26] Rocha A L B, Verdi F L. EFM: Improving DCNs throughput using the transmission rates of elephant flows[C]//Proc of IEEE Symp on Computers and Communications. Piscataway, NJ: IEEE, 2018: 155−157

[27] Karagiannis T, Molle M, Faloutsos M, et al. A nonstationary Poisson view of Internet traffic[C]//Proc of IEEE Conf on Computer Communications. Piscataway, NJ: IEEE, 2004: 1558−1569

[28] Arfeen M A, Pawlikowski K, Willig A, et al. Internet traffic modelling: From superposition to scaling[J]. IET Networks, 2014, 3(1): 30−40

[29] O'Connell N, Yor M. Brownian analogues of Burke's theorem[J]. Stochastic Processes and Their Applications, 2001, 96(2): 285−304

[30] Wang Guodong, Li Jun, Chang Xiangqing. Modeling and performance analysis of the multiple controllers' approach in software defined networking[C]//Proc of the 23rd IEEE Int Symp on Quality of Service. Piscataway, NJ: IEEE, 2015: 73−74

[31] Huang Xinli, Li Fanshuo, Cao Kun, et al. Queueing theoretic approach for performance-aware modeling of sustainable SDN control planes[J]. IEEE Transactions on Sustainable Computing, 2018, 5(1): 121−133

[32] Xiong Bing, Wu Rengeng, Zhao Jinyuan, et al. DAFT: A differentiated storage and accelerated lookup architecture for large-scale flow tables in OpenFlow networks[J]. Chinese Journal of Computers, 2020, 43(3): 453−470 (in Chinese)
(熊兵, 邬仁庚, 赵锦元, 等. DAFT: 一种OpenFlow大规模流表区分存储与加速查找架构[J]. 计算机学报, 2020, 43(3): 453−470)

[33] Benson T, Akella A, Maltz D A. Network traffic characteristics of data centers in the wild[C]//Proc of the 10th ACM SIGCOMM Conf on Internet Measurement. New York: ACM, 2010: 267−280

[34] Shi Weiguang, MacGregor M H, Gburzynski P. Load balancing for parallel forwarding[J]. IEEE/ACM Transactions on Networking, 2005, 13(4): 790−801

[35] Wallerich J, Feldmann A. Capturing the variability of Internet flows across time[C/OL]//Proc of the 25th IEEE Int Conf on Computer Communications. Piscataway, NJ: IEEE, 2006 [2021-08-16]. https://ieeexplore.ieee.org/abstract/document/4146690

[36] Basat R B, Einziger G, Friedman R, et al. Randomized admission policy for efficient top-k and frequency estimation[C/OL]//Proc of IEEE Conf on Computer Communications. Piscataway, NJ: IEEE, 2017 [2021-08-16].https://ieeexplore.ieee.org/document/8057215

[37] Basat R B, Chen Xiaoqi, Einziger G, et al. Randomized admission policy for efficient top-k, frequency, and volume estimation[J]. IEEE/ACM Transactions on Networking, 2019, 27(4): 1432−1445

Luo Ke, born in 1961. PhD, professor, master supervisor. His main research interests include data mining, computer applications.

罗　可, 1961 年生. 博士, 教授, 硕士生导师. 主要研究方向为数据挖掘、计算机应用.

Zeng Peng, born in 1997. Master. His main research interests include future networks, network modeling and optimization, computer applications.

曾　鹏, 1997 年生. 硕士. 主要研究方向为未来网络、网络建模与优化、计算机应用.

Xiong Bing, born in 1981. PhD, associate professor, master supervisor. His main research interests include future networks, network security, network modeling and optimization.

熊　兵, 1981 年生. 博士, 副教授, 硕士生导师. 主要研究方向为未来网络、网络安全、网络建模与优化.

Zhao Jinyuan, born in 1980. PhD, lecturer. Her main research interests include future networks, cloud computing.

赵锦元, 1980 年生. 博士, 讲师. 主要研究方向为未来网络、云计算.