

基于 KDN 的智能路由系统设计与实现

周洪利¹ 关 春²
ZHOU Hongli GUAN Chun

摘 要

随着新兴互联网业务的发展,网络流量激增,传统网络路由方案难以满足要求。知识定义网络作为新范式,将知识融入网络决策,为解决问题提供新思路,成网络领域热门研究方向。基于此,文章在研究了知识定义网络的核心技术,包括软件定义网络、机器学习和网络分析等基础上,提出了一种适应于智能路由的简化模型,并对模型进行了系统实现,不仅为后续智能路由领域提供了可复用、可验证的理论模型,也为网络技术的工程化实践提供了重要参考,对推动知识定义网络在实际场景中的应用具有重要意义。

关键词

知识定义网络; 软件定义网络; 机器学习; 网络路由

doi: 10.3969/j.issn.1672-9528.2025.06.021

0 引言

在数字化进程加速的当下,互联网业务呈现出如高清视频流、实时在线游戏、大规模云计算服务、工业互联网应用等变革形态,进而对网络的带宽、延迟、服务质量及可定制化服务等各方面都提出了更高的要求。然而,在传统网络中,各种网络设备的功能与硬件深度耦合,使得网络复杂和僵化^[1]。为此,软件定义网络^[2](software-defined networking, SDN)提出数控分离的架构思想,并通过标准 API 进行通信。目前传统网络正不断向软件定义的架构转变,实现着网络的软件化和可编程。在人工智能的新时代,机器学习已广泛应用在自然语言处理、计算机视觉、自动驾驶汽车等领域,然而,在网络领域,尽管已有部分研究和实践探索,但距离在网络中全面、广泛地应用机器学习技术,仍面临诸多技术挑战。为此,知识定义网络^[3](knowledge-defined network, KDN)提出了统一 SDN、网络分析和人工智能的全新架构,以更高效和智能化的方式运营网络,之后逐渐成为网络智能化研究的优选框架^[4-5]。

网络路由作为网络工程的基础问题,在网络数据传输与通信过程中起着关键作用。本文对 KDN 架构展开了深入剖析,探讨了 SDN、网络分析以及机器学习的技术创新成果与研究进展;随后,基于对这些前沿技术的综合考量,提出了一种适应于智能路由的简化模型。并对所提模型进行了实现,为后续智能路由的研究提供了可验证的模型与工程实践参考。

1 知识定义网络

知识定义网络探究了将人工智能技术应用于网络控制和操作的新范式,指出当前在网络中部署人工智能技术受限的主要原因是分布式网络,同时提出软件定义网络和网络分析的发展有助于人工智能技术在网络中的应用和发展。其主要思想是通过数据平面实时或准实时采集网络运行中的各类数据,这些数据随即被传输至管理平面,由管理平面进行统一收集与整合;之后知识平面运用机器学习技术对管理平面汇聚的数据展开深度分析,从而提炼出具有价值的知识;最终控制平面依据这些知识,借助自身的集中管控能力,将经过分析得出的转发逻辑精准地下发至数据平面。图 1 展示了知识定义网络控制的基本步骤。

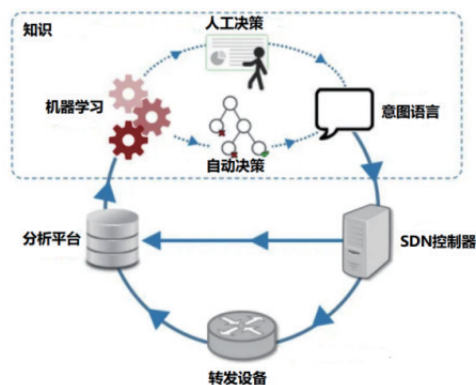


图 1 KDN 控制循环

(1) 数据采集: 网络分析平台可以直接从数据平面网元采集数据,也可以向 SDN 控制器查询来采集数据。

(2) 机器学习: 采集的数据作为机器学习的输入,借助机器学习算法学习网络行为,生成知识或网络模型。

(3) 决策制定: 学习到的知识或网络模型可通过意图

1. 江苏省未来网络创新研究院 江苏南京 211100
2. 江苏省食品药品监督信息中心 江苏南京 210012

驱动语言做出决策，也可通过人工做出决策。

(4) 意图转换：SDN 控制器将接收到的决策信息转换为命令式控制动作，通过北向南向接口与数据平面设备通信，实现对网络设备的控制。

本质上 KDN 借助 SDN 的集中控制和网络分析的实时数据，为机器学习技术在网络中的部署创造了条件，使这种部署成为可能。

2 软件定义网络

SDN 作为一种新兴的网络架构范式，将网络的控制平面和数据平面分离，具备集中控制和开放可编程等特点，使得 SDN 能够收集网络中所有节点和链路的信息，进而获取全局网络视图，实现传统网络无法满足的业务创新。

在 SDN 发展的早期阶段，研究重点是实现控制平面的集中化和可编程性，其本质是网络操作系统，并且已有大量案例的实现，如 OpenDaylight、ONOS、Ryu、Floodlight 等。这些网络操作系统提供具有高级网络抽象的北向 API，以供上层应用程序调用；向下通过南向接口 OpenFlow 协议对数据平面设备进行灵活控制。但 OpenFlow 协议在数据平面受制于功能固定的包处理硬件，无法进行数据平面可编程的创新。

数据平面可编程是指网络设备的数据处理逻辑可以通过编程的方式进行定义和修改。可编程转发芯片是数据平面可编程的硬件基础，并且其快速转发能力也至关重要。网络处理器（network processing unit, NPU）凭借专为网络处理设计的架构与指令集，现场可编程门阵列（field-programmable gate array, FPGA）通过可重构逻辑与并行处理，二者可提供每秒数百 GB 的数据包处理速度。而内置协议无关交换架构的 PISA 芯片更加强大，可实现每秒 TB^[6] 级别的数据包处理速度。数据平面编程语言是实现数据平面可编程的软件工具。packetC^[7] 是一种用于 NPU 的可编程语言。PX^[8] 是一种专门设计用于 FPGA 的可编程数据包处理语言。而 P4 则是由 Nick 等人^[9] 提出的可编程协议无关报文处理语言，允许对网络设备的数据平面进行编程，定义数据包的处理流程，包括如何解析包头、如何进行匹配和转发等操作。

目前，SDN 已实现应用平面、控制平面和数据平面的全可编程。在全可编程技术的加持下，网络业务的创新和服务能力持续增强，云网络的创新尤为突出，使云服务提供商能够快速调整网络架构和功能，同时也在服务客户的过程中不断进行技术革新，产生了一系列优秀成果。Pan 等人^[10] 提出的 Sailfish，基于数据平面可编程硬件和 x86 软件实现了软硬件协同的云网关系统。Lyu 等人^[11] 提出的 Poseidon，面对超大规模、超高弹性适配等需求，设计实现了具有极致性能的虚拟云网络控制器。Pan 等人^[12] 提出的 LuoShen，针对边缘云中因大带宽和大单流而导致网关水位突破上限和大单流将单核打爆的问题，实现了高性能超融合边缘网关。

3 网络分析平台

网络分析平台从转发平面或 SDN 控制器接收数据，并向机器学习提供数据。数据的质量和完整性直接影响到后续机器学习的效果。传统的数据采集技术有 NETCONF、NetFlow 和 IPFIX 等。其中，NETCONF 侧重于网络设备的配置管理和状态信息获取，在设备配置层面提供数据支持。NetFlow 和 IPFIX 侧重于网络流量的统计和分析，从宏观流量角度提供数据。

在 SDN 架构下，则可使用 OpenFlow 和 P4 Telemetry^[13]。OpenFlow 提供支持在转发设备中维护流级流量测量（如流量大小、流持续时间）。P4 Telemetry 则填补了传统数据采集技术在实时性和数据平面可编程性上的空白，能提供纳秒级延迟、队列深度等实时数据，以进行精细化和定制化的数据采集。P4 Telemetry 的一种实现方式是带内网络遥测（in-band network telemetry, INT）。INT 的核心特点是允许通过数据平面收集和报告网络状态，且不需要控制平面的干预。在 INT 架构模型中，数据包包含被网络设备解释为“遥测指令”的包头字段，这些指令指导支持 INT 功能的设备在数据包转发时收集和写入某些状态信息，比如可以收集链路的延迟、带宽利用率等信息。其重点在于利用数据包携带的指令在数据平面实现高效的网络状态信息采集，并通过特定的机制将这些信息反馈给相关的监测或管理系统。

4 机器学习

机器学习使计算机系统能够从数据或过往经验中自主学习，其重要的技术领域有深度学习和强化学习。深度学习依托深层神经网络架构，具备自动提取复杂特征的能力；强化学习以智能体与环境的交互为基础，智能体依据环境反馈的奖励信号对自身行为策略进行优化，在机器人控制、博弈游戏等场景中有着广泛的应用；而深度强化学习融合了深度学习的特征提取优势与强化学习的策略优化能力，在处理复杂高维任务时展现出独特的适用性。

最常见的深度学习模型是深度神经网络（deep neural network, DNN）。Hinton 等人^[14] 以深度神经网络（DNN）为基础，创新性地提出了深度置信网络（deep belief network, DBN）。Mao 等人^[15] 针对路由决策问题，构建了一种基于深度置信网络（DBN）的解决方案。图神经网络（graph neural network, GNN）作为一种能够高效处理拓扑信息提取任务的新型神经网络架构^[16]，被 Geyer 等人^[17] 应用于分布式智能路由算法的设计中。

Q-Learning 作为一种经典的强化学习模型，被 Boyan 等人^[18] 应用于通信网络的路由决策过程，提出了 Q-routing 算法，开启了基于强化学习的路由算法研究领域。Google DeepMind 团队将深度学习模型引入强化学习框架，提出深度 Q 值学习（deep q-learning, DQN）^[19]。此外，研究人员通过有机结合 Q 值学习与策略梯度方法，构建出演员 - 评价

者 (actor-vritic, AC) 框架。此后, Schulman 等人^[20]提出置信域策略优化算法 (trust region policy optimization, TRPO), OpenAI 与 DeepMind 提出近端策略优化算法^[21] (proximal policy optimization, PPO)。PPO 算法现已成为当前主流的强化学习算法。

5 智能路由系统

5.1 业务需求

在网络发展进程中, 提升网络效率与性能始终是关键任务, 而实现智能路由的自动化在其中起着举足轻重的作用。自动化智能路由控制可以通过动态自适应优化机制, 实时采集流量负载、链路质量等关键信息, 对网络配置与路由策略进行快速且精准的调整。此外, 也能克服人工操作的弊端, 降低因人工疏忽、知识局限等因素导致的配置误差。KDN 自提出以来得到了广泛关注, 但尚未有实际部署。因此, 基于 KDN 实现智能路由的自动化控制显得极为重要。

5.2 系统设计

根据智能路由自动化控制的业务需求, 对 KDN 范式的控制循环重新定义, 构建并提出一种具有简化特性的实现架构。该架构由数据平面、控制平面与知识平面有机组成, 形成了闭环的自动化控制循环, 其基本的流程包含初始化、数据采集与处理、算法运算及路径规划、流表规则下发与反馈等多个关键环节。如图 2 所示。

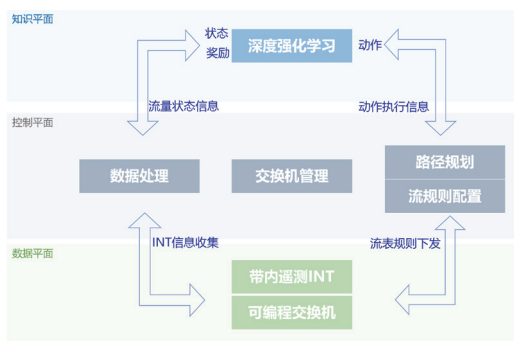


图 2 智能路由自动化控制框架

数据平面借助基于 P4 语言实现的带内遥测技术达成对网络状态的实时监测。具体而言, 该技术会按照预先设定的周期, 向流经数据平面的数据包中嵌入 INT 遥测指令。当数据包在网络中传输时, 会依据遥测指令, 收集交换机队列拥塞程度、排队时延及链路负载 3 类数据。控制平面初始化后与交换机建立连接, 并下发默认流表。对于数据平面上传的遥测数据, 控制平面进行归一化等处理; 结合知识平面的动作执行信息, 经计算生成路由, 进而生成流表信息并下发至交换机。知识平面采用深度强化学习的近端策略优化算法, 以控制平面处理后的流量状态为输入, 输出所有链路权值。算法以链路利用率为性能指标, 以最小化最大链路利用率为目标, 将奖励函数设为全局链路最大链路负载与带宽比值的负数。

5.3 原型实现

系统原型采用模块化分层设计, 并且各层可独立演进, 如图 3 所示, 自下而上分为数据层、控制层与知识层, 各层通过标准化接口与协议实现高效协同, 具体架构如下:

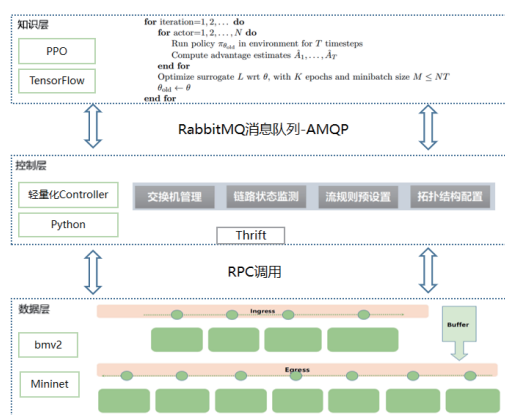


图 3 技术架构

(1) 数据层基于 Mininet 仿真平台与可编程交换机 bmv2 构建, 采用 P4 语言实现数据包处理流水线。系统设计了自定义路由元数据信息 (metadata routing information, MRI) 头部, 用于记录转发路径标识符及链路状态信息; 定义了元数据总线存储临时处理变量, 并实现有状态对象, 统计各端口转发数据量。数据包处理采用两级流水线模型: Ingress 流水线负责插入 MRI 头部、匹配流表设置转发端口, 并周期性复制数据包至控制层; Egress 流水线则在 MRI 头部追加当前链路状态信息, 并于目的交换机剥离 MRI 头部, 确保终端设备可解析原始数据包。同时, 实现了 Thrift 接口服务, 提供标准化的流表操作接口, 以方便控制层对数据层的细粒度控制。

(2) 控制层没有借助传统的 OpenDaylight 或 ONOS 控制器, 而是基于 Python 自研轻量化控制器, 实现路由计算引擎、策略优化器与流表下发等模块。并通过 Thrift RPC 框架实现与数据层的交互; 同时, 基于 AMQP 协议构建 RabbitMQ 异步消息通道, 接受知识层下发的链路权值矩阵。

(3) 知识平面的核心组件是近端策略优化算法, 采用 TensorFlow 框架和 Python 语言进行了实现, 接收控制层实时聚合的网络状态数据, 不断进行数据的学习, 计算出链路的权值, 通过 RabbitMQ 传递给控制层。

6 结语

本文研究了 KDN 范式, 并对其控制循环进行了重新定义, 构建了一种具有简化特性且能够自动化控制的智能路由系统。该系统深度融合了数据平面可编程技术与近端策略优化算法等前沿技术手段。其中, 数据平面可编程技术赋予了系统对网络数据处理和转发的高度灵活性, 使其能够依据不同的网络场景和业务需求, 动态调整数据处理规则; 近端策略优化算法则凭借其强大的策略优化能力, 能够高效地做出智能路由决策, 从而显著提升网络性能。展望未来研究方向, 将对该系统进行

模块化设计。通过模块化架构,系统能够更便捷地集成和适配更多不同类型的算法,为后续的算法验证和功能拓展提供有力支撑,进而推动智能路由自动化控制技术的持续创新与发展,以更好地满足日益增长的网络智能化需求。

参考文献:

- [1] BALAKRISHNAN H, BANERJEE S, CIDON I, et al. Revitalizing the public internet by making it extensible[J]. ACM SIGCOMM computer communication review, 2021, 51(2): 18-24.
- [2] CASADO M, FREEDMAN M J, PETTIT J, et al. Ethane: taking control of the enterprise[J]. ACM SIGCOMM computer communication review, 2007, 37(4), 1-12.
- [3] MESTRES A, RODRIGUEZ-NATAL A, CARNER J, et al. Knowledge-defined networking[J]. ACM SIGCOMM computer communication review, 2017, 47(3): 2-10.
- [4] HYUN J, TU N V, WON-KI HONG J. Towards knowledge-defined networking using in-band network telemetry[C/OL]//2018 IEEE/IFIP Network Operations and Management Symposium. Piscataway: IEEE, 2018[2024-06-12]. <https://ieeexplore.ieee.org/document/8406169>. DOI: 10.1109/NOMS.2018.8406169.
- [5] STAMPA G, ARIAS M, SANCHEZ-CHARLES D, et al. A deep-reinforcement learning approach for software-defined networking routing optimization[EB/OL]. (2017-09-20)[2024-05-17]. <https://doi.org/10.48550/arXiv.1709.07080>.
- [6] BOSSHART P, GIBB G, KIM H-S, et al. Forwarding metamorphosis: fast programmable match-action processing in hardware for SDN[J]. ACM SIGCOMM computer communication review, 2013, 43(4): 99-110.
- [7] DUNCAN R, JUNGCK P. PacketC language for high performance packet processing[C/OL]//2009 11th IEEE International Conference on High Performance Computing and Communications. Piscataway: IEEE, 2009: 450-457.
- [8] BREBNER G, JIANG W R. High-speed packet processing using reconfigurable computing[J]. IEEE micro, 2014, 34 (1): 8-18.
- [9] 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.
- [10] PAN T, YU N B, JIA C H, et al. Sailfish: accelerating cloud-scale multi-tenant multi-service gateways with programmable switches[C]//SIGCOMM'21: Proceedings of the 2021 ACM SIGCOMM 2021 Conference. New York: ACM, 2021: 194-206.
- [11] LÜ B, SONG E, PAN T, et al. POSEIDON: a consolidated virtual network controller that manages millions of tenants via config tree [EB/OL] (2025-05-06)[2025-06-02]. <https://openreview.net/forum?id=XoTDbmHMxb>.
- [12] PAN T, LIU K, WEI X, et al. LuoShen: a hyper-converged programmable gateway for multi-tenant multi-service edge clouds [C/OL]//Proceedings of the Networked Systems Design and Implementation Symposium (NSDI 2024). (2025-05-06)[2025-06-02]. <https://ennanzhai.github.io/pub/nsdi24fall-luoshen.pdf>.
- [13] KIM C, SIVARAMAN A, KATTA N, et al. In-band network telemetry via programmable dataplanes[EB/OL]. [2024-06-19]. <https://xueshu.baidu.com/usercenter/paper/show?paperid=e5be28025bc7a480e3009a7fdd88e4a8>.
- [14] HINTON G E, OSINDERO S, TEH Y W. A fast learning algorithm for deep belief nets [J]. Neural computation, 2006, 18(7): 1527-1554.
- [15] MAO B M, FADLULLAH Z M, TANG F X, et al. Routing or computing? the paradigm shift towards intelligent computer network packet transmission based on deep learning [J]. IEEE transactions on computers, 2017, 66(11): 1946-1960.
- [16] ZHOU J, CUI G Q, ZHANG Z Y, et al. Graph neural networks: a review of methods and applications [J]. AI open, 2020, 1: 57-81.
- [17] GEYER F, CARLE G. Learning and generating distributed routing protocols using graph - based deep learning [C] //Big-DAMA '18: Proceedings of the 2018 Workshop on Big Data Analytics and Machine Learning for Data Communication Networks. New York: ACM, 2018: 40-45.
- [18] BOYAN J A, LITTMAN M L. Packet routing in dynamically changing networks: a reinforcement learning approach [C]//NIPS'93: Proceedings of the 7th International Conference on Neural Information Processing Systems. New York: ACM, 1993: 671-678.
- [19] MNIH V, KAVUKCUOGLU K, SILVER D, et al. Playing atari with deep reinforcement learning [EB/OL]. (2013-12-19) [2024-09-09]. <https://doi.org/10.48550/arXiv.1312.5602>.
- [20] SCHULMAN J, LEVINE S, MORITZ P, et al. Trust region policy optimization [C] //ICML'15: Proceedings of the 32nd International Conference on International Conference on Machine Learning. New York: ACM, 2015: 1889-1897.
- [21] SCHULMAN J, WOLSKI F, DHARIWAL P, et al. Proximal policy optimization algorithms [EB/OL]. (2017-08-28)[2024-06-12]. <https://doi.org/10.48550/arXiv.1707.06347>.

【作者简介】

周洪利(1981—),男,山东临沂人,硕士,高级工程师,研究方向:软件定义网络、云数据中心、智能网络。

(收稿日期: 2025-02-23 修回日期: 2025-06-05)