信息通信领域产学研合作特色期刊　十佳皖刊
第三届国家期刊奖百种重点期刊　中国科技核心期刊

ISSN 1009-6868
CN 34-1228/TN

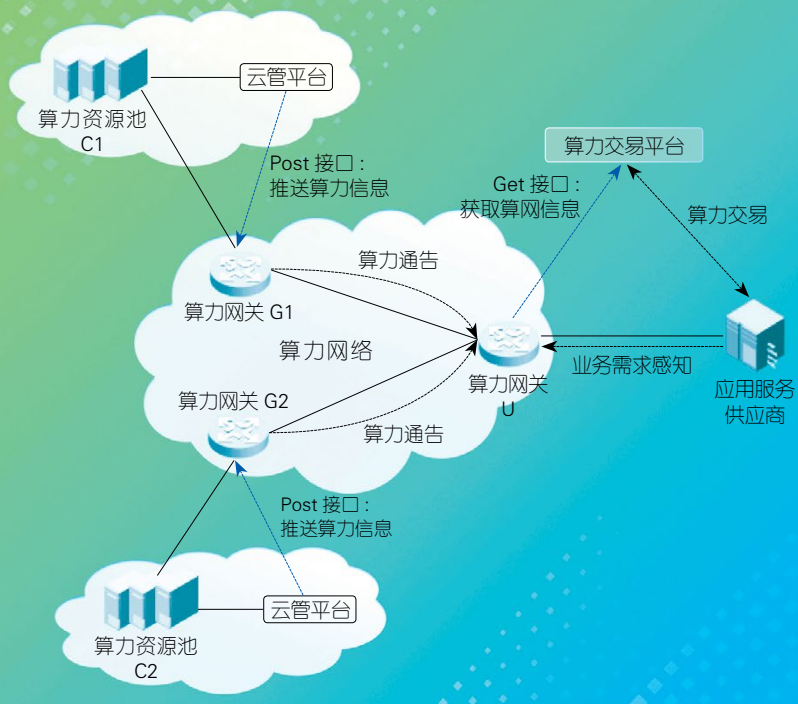
中兴通讯技术

ZTE TECHNOLOGY JOURNAL

http://tech.zte.com.cn

2023 年 8 月 · 第 4 期

专题：算力网络和东数西算



（封面图片详解见 P04）

ISSN 1009-6868
9 771009 686236 08>

《中兴通讯技术》第 9 届编辑委员会成员名单

目　次

中兴通讯技术（ZHONGXING TONGXUN JISHU）
总第 171 期　第 29 卷　第 4 期　2023 年 8 月

信息通信领域产学研合作特色期刊　第三届国家期刊奖百种重点期刊　中国科技核心期刊　工信部优秀科技期刊　十佳皖刊　中国五大文献数据库收录期刊　1995 年创刊

热点专题 ▶　算力网络和东数西算

01　专题导读 …… 赵慧玲
02　东数西算场景下的算力网关研发及应用 …… 马思聪，孙吉斌，孙一豪
08　算力网络四面三级算力度量技术体系 …… 杜宗鹏，李志强，陆璐
14　东数西算下面向业务的路由策略分析与探索 …… 魏汝翔，刘琦，赵广，曹畅，唐雄燕
19　面向算力网络的多路径时敏优先调度机制 …… 夏华屹，权伟，张宏科
26　算力网络资源协同调度探索与应用 …… 彭开来，王旭，唐琴琴
32　面向算力网络的云边端协同调度技术 …… 周旭，李琢
38　一种面向服务的算网路由架构方案 …… 黄光平，谭斌，吉晓威
43　通用在网计算系统架构及协议设计 …… 姚柯翰，陆璐，徐世萍
49　大规模语言模型的跨云联合训练关键技术 …… 潘囿丞，侯永帅，杨卿，余跃，相洋

专家论坛 ▶　57　面向新型智能计算中心的全调度以太网技术 …… 段晓东，程伟强，王瑞雪，王雯萱

企业视界 ▶　64　数据管理系统发展趋势与挑战 …… 韩银俊，牛家浩，屠要峰

技术广角 ▶　72　基于 5G 连接的集中式 PLC 新型工业组网架构 …… 邓伟，于天意，侯庆东
78　基于数字子载波和概率整形的相干光通信系统设计及应用 …… 陆源，牛文林，王永奔，胡子荷

综合信息 ▶　31　新增编委介绍

《中兴通讯技术》2023 年热点专题名称及策划人

1. 面向云网安全的新型防护技术
中国电信研究院教授级高工　解冲锋
北京邮电大学教授　杨义先

2. 语义通信
中国科学院院士　陆建华
清华大学教授　陶晓明

3. 数字孪生技术
重庆邮电大学教授　陈前斌

4. 算力网络和东数西算
工业和信息化部信息通信科技委常委　赵慧玲

5. 6G 网络技术
北京邮电大学教授　王文东

6. 面向双碳的新一代无线通信网络
华中科技大学教授　葛晓虎
西安电子科技大学教授　李建东

MAIN CONTENTS

ZTE TECHNOLOGY JOURNAL
Vol. 29 No. 4 Aug. 2023

Special Topic ▶

Computing Power Network and East-Data-West-Computing

01 Editorial ····· ZHAO Huiling

02 Research and Application of Computing Power Gateway in East-Data-West-Computing Project ····· MA Sicong, SUN Jibin, SUN Yihao

08 Three-Level and Four-Aspect Computing Measurement System in Computing Force Network ····· DU Zongpeng, LI Zhiqiang, LU Lu

14 Analysis and Exploration of Service Oriented Routing Strategies in East-Data-West-Computing Requirement Transfer ····· WEI Ruxiang, LIU Qi, ZHAO Guang, CAO Chang, TANG Xiongyan

19 A Multipath Time Sensitive Priority Scheduling Mechanism for Computing Power Network ····· XIA Huayi, QUAN Wei, ZHANG Hongke

26 Collaborative Scheduling of Computing Power Network Resources: Exploration and Application ····· PENG Kailai, WANG Xu, TANG Qinqin

32 Cloud-Edge-End Collaborative Scheduling Technology for Computing Power Network ····· ZHOU Xu, LI Zhuo

38 An Architecture Solution of Service-Oriented Routing for Computing and Networking ····· HUANG Guangping, TAN Bin, JI Xiaowei

43 System Architecture and Protocol Design for Generic In-Network Computing ····· YAO Kehan, LU Lu, XU Shiping

49 Key Technologies for Cross-Cloud Joint Training of Large-Scale Language Models ····· PAN Youcheng, HOU Yongshuai, YANG Qing, YU Yue, XIANG Yang

Expert Forum ▶

57 Global Scheduling Ethernet for New Intelligent Computing Center ····· DUAN Xiaodong, CHENG Weiqiang, WANG Ruixue, WANG Wenxuan

Enterprise View ▶

64 Development Trends and Challenges of Data Management Systems ····· HAN Yinjun, NIU Jiahao, TU Yaofeng

Reseach Paper ▶

72 5G-Based Centralized PLC New Industrial Networking Architecture ····· DENG Wei, YU Tianyi, HOU Qingdong

78 Coherent Optical Communication System Based on Digital Subcarrier and Probabilistic Shaping: Design and Application ····· LU Yuan, NIU Wenlin, WANG Yongben, HU Zihe

期刊基本参数：CN 34-1228/TN*1995*b*16*82*zh*P*¥20.00*6500*14*2023-08

算力网络和东数西算专题导读

专题策划人

赵慧玲，工业和信息化部信息通信科技委常委、中国通信学会理事、中国通信学会北京通信学会副理事长、中国通信标准化协会网络与业务能力技术工作委员会主席、鹏城实验室高级专家、中国电信科技委常委；长期从事电信网络领域技术和标准工作；曾获多个国家及省部级科技进步奖项；发表技术文章百余篇，出版技术专著12部。

随着中国“东数西算”工程的不断发展，由中国的电信运营商提出的算力网络技术逐渐成熟，其行业标准也持续成为产业界和学术界的研究热点。

算力网络有哪些规划和实践？算力度量的研究进展如何？算力网络资源调度策略如何实施？算力网络路由策略应该如何考量和实践？算力网络的云边端协同调度技术研究进展如何？算力网络智能计算中心的关键技术有什么主要挑战？本期专题邀请了算力网络领域的专家进行撰稿，从不同的角度论述该领域的研究进展及相关成果。

《东数西算场景下的算力网关研发及应用》提出了以算力网关相关技术为基础的算力网络形态，给出了算力网关组网的技术方案和路由方案，实现算力资源最优调度。文章特别阐述了运营商骨干和省网两个层面的具体建设实施方案，通过现网实践验证了算力网关技术方案的有效性。《算力网络四面三级算力度量技术体系》指出算力度量是算力网络的关键要素之一，算力度量要求量化异构算力资源，并使业务需求变得多样化，从而建立统一的描述语言。文章提出四面三级算力度量技术体系，同时还探索了在算力路由决策中网络按需进行算力信息传递等关键技术。《东数西算下面向业务的路由策略分析与探索》介绍了算力业务的需求指标和分类，分析了现网流量对东数西算业务路由策略的影响，研究了算力业务的路由策略，并提出了基于业务属性与算力资源分布的混合式路由策略技术方案。针对算力网络的低时延传输需求，《面向算力网络的多路径时敏优先调度机制》提出了多路径时敏优先调度机制，设计了基于强化学习的多路径低时延转发调度算法，并在转发出端口设计了等级与队列自适应映射算法，以减少低时延应用的排队时延。《算力网络资源协同调度探索与应用》详细介绍了一种算力网络资源调度的技术架构和系统功能，并针对用户差异化需求下多层次算力资源的弹性灵活调度问题，设计了一个算力网络资源协同调度平台。《面向算力网络的云边端协同调度技术》介绍了分布式云边端算力的发展趋势，探讨了融合云边端的协同网络技术架构和关键技术，给出了基于分布式强化学习的云边端协同网络中的流量调度模型，并通过仿真实验验证了所提协同流量调度方案的有效性。《一种面向服务的算网路由架构方案》分析了算力路由在IP分组网络面临的主要问题，提出了一种基于服务标识的算力路由技术，其核心思路是引入独立于IP主机地址的服务标识，并构建用户与算网系统之间、网络与业务之间、网络与算力系统之间的简明高效互通接口。《通用在网计算系统架构及协议设计》论述了通用在网计算的架构和主要应用场景，介绍了SRv6协议的实现流程，指出在网计算技术在不断成熟发展的同时，仍存在一些问题和挑战。《大规模语言模型的跨云联合训练关键技术》介绍了大规模语言模型跨云训练的主要挑战和关键技术，通过采用模型分割、拆分学习、跨云协同、压缩通信和模型复用等关键技术，能够有效解决跨云训练过程中可能出现的算力和数据不足的问题，并提高训练速度和效率。ChatGPT的出现表明智能算力占比将会大幅度提升，因此本期特别邀请了智能计算中心网络的专家撰写《面向新型智能计算中心的全调度以太网技术》一文，探讨这一领域的技术发展及挑战。

算力网络目前还处于发展阶段，仍需要不断探索和实践。本期文章汇聚了各位作者现阶段的研究思路及成果，希望能给读者带来有益的收获与参考。在此，对各位作者的积极支持和辛勤付出表示衷心感谢！

DOI：10.12142/ZTETJ.202304001
收稿日期：2023-07-15

东数西算场景下的算力网关研发及应用

Research and Application of Computing Power Gateway in East-Data-West-Computing Project

马思聪/MA Sicong，孙吉斌/SUN Jibin，孙一豪/SUN Yihao
（中国电信股份有限公司研究院，中国 北京 102209）
(China Telecom Corporation Research Institute, Beijing 102209, China)

DOI：10.12142/ZTETJ.202304002
网络出版地址：http://kns.cnki.net/kcms/detail/34.1228.TN.20230724.1524.004.html
网络出版日期：2023-07-25
收稿日期：2023-06-15

摘要：提出了一种面向算力网络场景的新型网络设备——算力网关。认为算力网关是实现算力网络一体化调度的基础，通过感知业务应用需求，结合当前的算力状况和网络状况，生成路由信息并发布到网络，将计算任务报文路由到合适的计算节点，以实现算力资源的最优调度。现网实践验证了算力网关技术方案的有效性。

关键词：算力网关；算力感知；算力路由；东数西算

Abstract: The computing power gateway, a new network device, is proposed for computing power network scenarios, which is the foundation for achieving integrated scheduling of computing power networks. By perceiving the information of services and combining the current computing power and network performance, the routing information will be published to the network to route computing task packets to suitable computing nodes, in order to achieve optimal scheduling of computing resources. Finally, the effectiveness of the computing power gateway technology solution is verified through practical verification in the current network.

Keywords: computing power gateway; perception of computing power; routing of computing power; east-data-west-computing

引用格式：马思聪，孙吉斌，孙一豪 . 东数西算场景下的算力网关研发及应用 [J]. 中兴通讯技术，2023, 29(4): 2-7. DOI: 10.12142/ZTETJ.202304002

Citation: MA S C, SUN J B, SUN Y H. Research and application of computing power gateway in east-data-west-computing project [J]. ZTE technology journal, 2023, 29(4): 2-7. DOI: 10.12142/ZTETJ.202304002

随着数字经济进入新发展阶段，业务数字化、技术融合化和数据价值化等加速演进，开启数字经济引领高质量发展新征程。在此发展过程中，算力作为数字时代核心资源的作用日益突出，以算力为核心的数字信息基础设施建设被提到前所未有的高度[1]。中国相继出台一系列围绕算力基础设施的政策文件，如《全国一体化大数据中心协同创新体系算力枢纽实施方案》《新型数据中心发展三年行动计划》等[2]，并加快实施“新基建”“东数西算”等工程，加强区域协同联动，推进热点区域与中西部地区、一线城市与周边地区的数据中心协调发展，引导算力的集群化发展。

为了实现算力像电力、热力、水一样，由统一的社会基础设施进行供应，真正地服务于社会经济的各行各业，需要在算力基础设施的供给模式方面进行创新，算力网络应运而生。算力网络是通过网络分发算力节点的计算、存储、算法等资源信息，并结合网络信息和用户需求，提供最佳的计算、存储、网络等资源的分发、关联、交易与调配，从而实现各类资源最优化配置使用的新型网络技术。作为算力网络中的核心网元设备，算力网关以算力度量、算力标识为依据，通过算力路由、算力感知等核心功能，传输发布相关算力策略与数据转发，是实现算力网络一体化调度的基础。

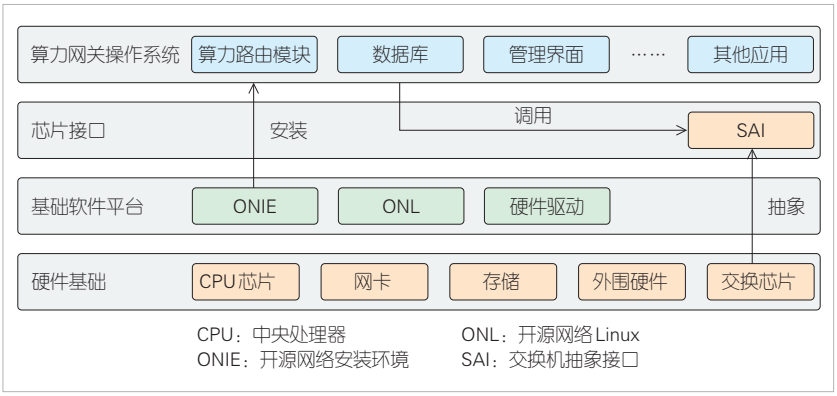
1 算力网关架构及组网方案

1.1 总体架构

算力网关基于开放的白盒网络设备架构，将网络中的物理硬件和节点操作系统（NOS）进行解耦，使标准化的硬件

基金项目：国家科技重大专项（2022YFB2901400）

配置与算力网络相关协议进行组合匹配，具有灵活、高效、可编程等特点，有助于算力网络相关协议的制定。算力网关整体架构如图1所示，主要分为硬件基础、基础软件平台、芯片接口和操作系统4个部分[3]：



▲图1 算力网关整体架构

1）硬件基础

硬件是算力网关系统运行的物理基础，主要由CPU、交换芯片、网卡、存储和外围硬件等构成。其中，CPU是对计算机的所有硬件资源（如存储器、输入输出单元）进行控制调配并执行通用运算的核心硬件单元，主要管控系统运作；交换芯片主要提供高性能和低延时的交换能力，是算力网关的核心芯片；网卡分为用于设备管理的管理网卡和用于网关与网络中其他设备通信的业务网卡，业务网卡与交换芯片共同决定了算力网关的转发性能；存储主要包括内存和硬盘，用于设备应用数据的存储和保存；外围硬件主要包括风扇、电源等用于维持设备正常运行的其他基础硬件。

2）基础软件平台

基础软件平台由开源网络安装环境（ONIE）、开源网络Linux（ONL）以及硬件驱动构成。其中，ONIE为算力网关提供一个开放的安装环境，可实现网关硬件和网络操作系统系统的解耦，支持在不同厂商的硬件上引导启动算力网关操作系统；ONL建立在开放网络硬件上，向网关系统提供基础操作系统，为交换硬件提供管理接口，使用ONIE来安装到板载闪存中。

3）芯片接口

交换机抽象接口（SAI）是一种标准化的应用程序编程接口（API），可以看作是一个用户级的驱动。在不同的专用集成电路（ASIC）芯片上，SAI为上层应用提供了统一的API。SAI的具体实现由不同ASIC芯片提供商负责，使用者不需要关心网络硬件供应商的硬件体系结构的开发和革新，通过始终一致的编程接口就可以很容易地应用最新、最好的硬件。

SAI本质就是在各ASIC的软件开发套件（SDK）之上抽象出统一接口。芯片厂商研发的ASIC的SDK需要与这层抽象适配，使得转发应用能够在不同的ASIC上运行。SAI向上为操作系统提供统一的API，向下可以对接不同的ASIC。

4）操作系统

算力网关操作系统基于社区版本的云开发网络软件（SONiC）开发，通过拓展协议和网关接口等能力实现了算力网络所需的相应功能。

算力网关操作系统由多个功能模块组成，这些模块通过集中式和可扩展的基础架构相互交互。本系统模块间交互依赖于redis数据库引擎（一个键值数据库，提供独立于语言的接口，可以在所有子系统之间进行数据持久化、复制和多进程通信）。通过依赖redis引擎基础架构提供的发布者/订阅者消息传递模型，应用程序可以仅订阅它们需要的数据，并避免与其功能无关的实现细节[4]。

算力网关操作系统将每个模块放置在独立的docker（容器）中，以保持语义相似组件之间的高内聚性，同时减少不相关组件之间的耦合。每个组件都被设计得相对独立，摆脱了平台和底层交互的限制。当前，算力网关操作系统主要包含以下几个dockers：Bgp、Web、Database、SwSS、Syncd、Teamd、Pmon、DHCP-relay等。

1.2 组网方案

算力网络目前在技术路线上可以分为集中式、分布式和混合式3种。在算力网关应用部署中，我们主要考虑混合式和分布式两种组网方案[5]。

1）混合式方案

在混合式的方案中，算网编排系统依靠云/算管控模块通过算力网关收集来自每个资源池的算力信息，通过网络管控模块收集网络拓扑信息。算网编排系统确定最优算力资源节点和网络路径。云/算管控模块与网络管控模块分别下发算力资源分配指令和路由策略，如图2所示。

在此架构下，算力网关主要功能包括：获取算力节点的算力信息及链路信息，接收网络管控模块下发的路径策略信息等。

2）分布式方案

在分布式方案中，算力网关需要实现算力资源感知、算力路由分发、资源表项生成、策略定制等全部功能，如图3所示。除支持算力信息发布和通告外，分布式方案还需通过算力和路径计算生成路由策略，并依据用户和应用感知对路由策略进行绑定，进而实现对算力资源和网络资源的信息同步与统一调度。

2 算力网关核心技术实现

算力网关通过感知算力和网络信息，将当前的计算能力状况和网络状况作为路由信息发布到网络，并将计算任务报文路由到合适的计算节点，以实现整体系统最优和用户体验最优。其中，算力感知和算力路由是算力网关的两大核心技术能力。

2.1 算力感知能力

算力感知是对算力资源的性能、实时负载、网络状况以及业务需求的全面感知，主要是需要明确网络中有多少算力资源，用户有怎样的算力需求。算力感知包括算力信息感知、网络状况感知、业务需求感知。

1）算力信息感知

算力信息的感知通常包括算力资源池的IP地址、计算能力、存储能力等内容。

如图4所示，云资源池一般由云管平台集中纳管，算力网关可以与云管平台通过RESTful API等接口交互，获取算力资源池的IP地址、计算能力、存储能力等，并最终把感知的算力信息上报到算力交易平台。

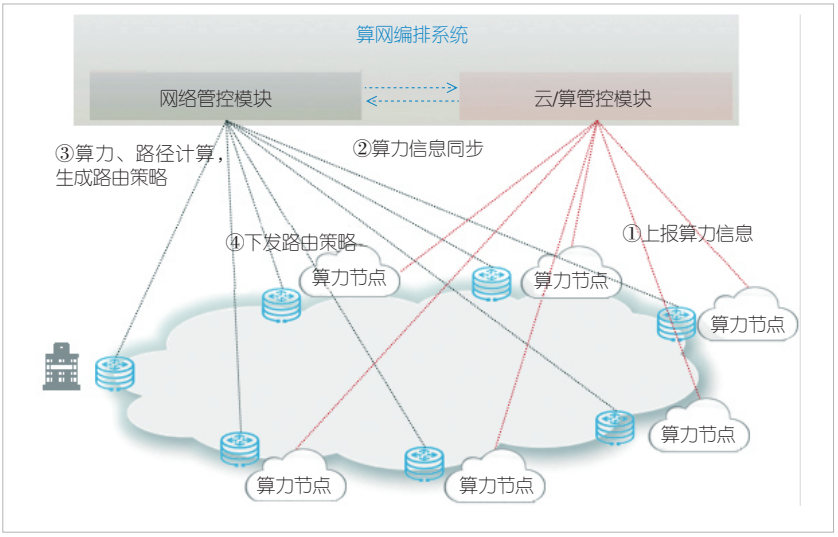
2）网络状态感知

网络信息的感知通常包括时延、带宽、丢包率、抖动等内容。以网络时延为例，由于算力资源池分布在不同位置，用户到资源池的网络路径也会根据网络拥塞状态发生变化，因此需要探测用户与各个算力资源之间的时延信息。算力网关的时延探测分为两部分：一是算力网关到算力资源池之间的时延探测；二是算力网关与算力网关之间的时延探测，如图5所示。

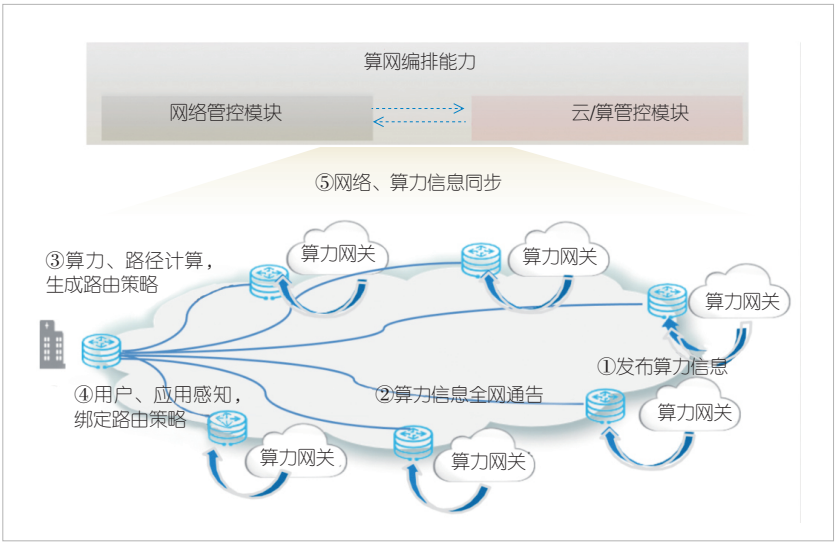
3）业务需求感知

除了对算力资源和网络状况的感知外，算力感知还应具备感知用户业务需求的能力，以实现更优的算力调度。

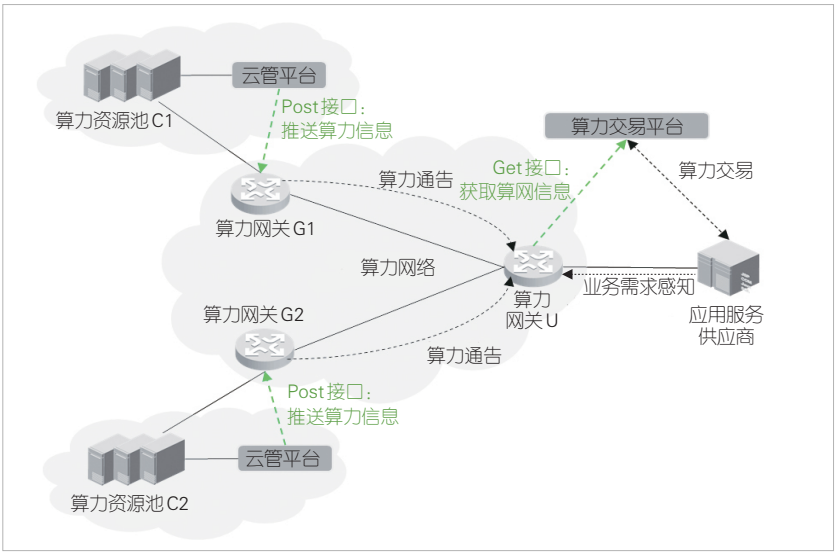
业务需求感知可以在用户入口的算力网关接收业务请求并感知业务需求，包括网络需求（时延、抖动等）和算力需求（算力请求类型、算力需求参数等），依据算力度量标准和特定的算法匹配可用算力。这样不仅能够精确匹配具体应用的业务需求，还能动态



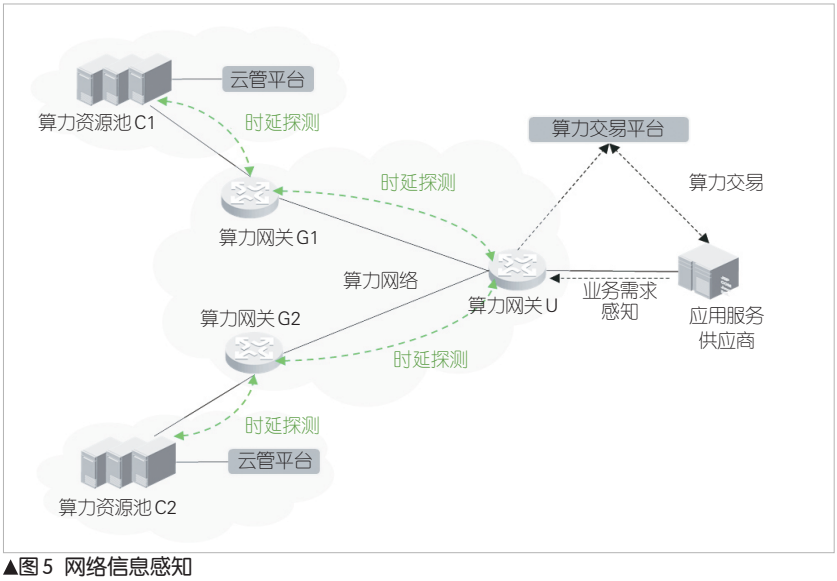
▲图2 算力网关混合式组网



▲图3 算力网关分布式组网



▲图4 算力信息感知



▲图5 网络信息感知

和实时地对算网进行调度，达到算力和网络的最优化。

业务应用需求可通过特定的协议或字段来与算力网关交互，从而实现算力网络对用户业务需求的感知。IPv6协议增加了扩展头部，具有很强的扩展性，可以在用户侧数据包头采用IPv6标准头+目的选项报头（DOH）扩展头的方式，利用扩展的字段携带应用的需求信息，包括带宽需求、时延需求、抖动需求、丢包率需求、计算和存储需求等[7]。

2.2 算力路由能力

算力路由是将算力信息引入路由域，通过对用户的业务需求、算力资源和网络资源的信息感知，动态选择满足业务需求的“转发路径+目的服务节点”，将业务沿指定路径调度至服务节点，从而实现算力和网络资源的全局优化。

算力路由技术可分为算力路由控制技术和算力路由转发技术。根据实现方式不同，算力路由控制技术又可以分为集中式控制和分布式控制。算力路由转发技术需要支持根据算力路由生成的“转发路径+目的节点”来指导业务转发，并且能够根据算力资源和网络状况的变化，动态调整控制面信息。

1）集中式算力路由控制

集中式的算力路由控制主要依托于上层算力交易平台及软件定义网络（SDN）控制器协同：先通过SDN控制器采集算力网络拓扑，再根据算力平台的算力匹配结果向算力网关下发SRv6 Policy，通过网络编排的方式形成算力与用户之间的路由控制，如图6所示。

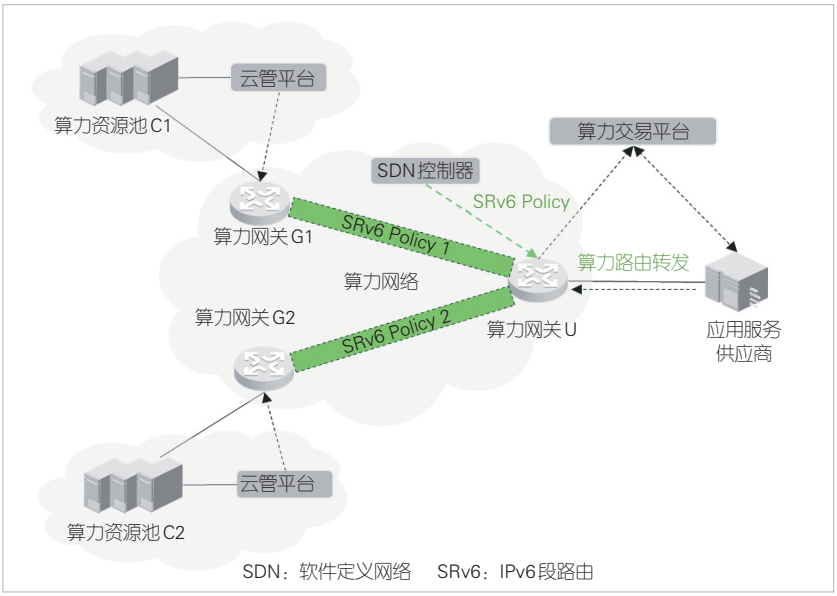
在SRv6网络里，业务需求可以被翻译成有序的指令列表，由沿途的网络设备去执行，以实现网络业务的灵活编排和按需定制。SRv6网络主要有SRv6 BE（指用最短路径算法计算得到的最优SRv6路径）和SRv6 Policy两种引流技术。SRv6 BE通过内部网关协议（IGP）收敛得出最短路径，业务无法按照指定的路径转发。SRv6 Policy可以在网络中任意节点之间规划路径。因此，使用SRv6 Policy不仅能够满足用户网络在时延、带宽、抖动和可靠性等各方面的差异化需求，还能够基于确定性路径的精细化控制来提高网络带宽的利用率[6]。

在集中式路由控制场景中，通过SRv6 Policy技术既能够实现算力网络的编排，保障算力资源与用户之间的路径确定性，又可以根据算力的实时变化实现算力的控制与调度。

2）分布式算力路由控制

分布式控制需要算力网关将感知的算力信息和网络信息进行通告，并且在用户入口的网关生成算力路由表项，形成用户业务需求与算力资源的协商和映射。而这种机制需要依靠特定的算力路由协议。

算力路由协议是实现算力路由控制和调度的关键技术。算力路由协议需要支持将感知的算力信息和网络信息在算力网关之间通告，并且在用户入口网关支持算力路由表的生成与更新，即基于通告的算力节点信息生成算力状态拓扑，进

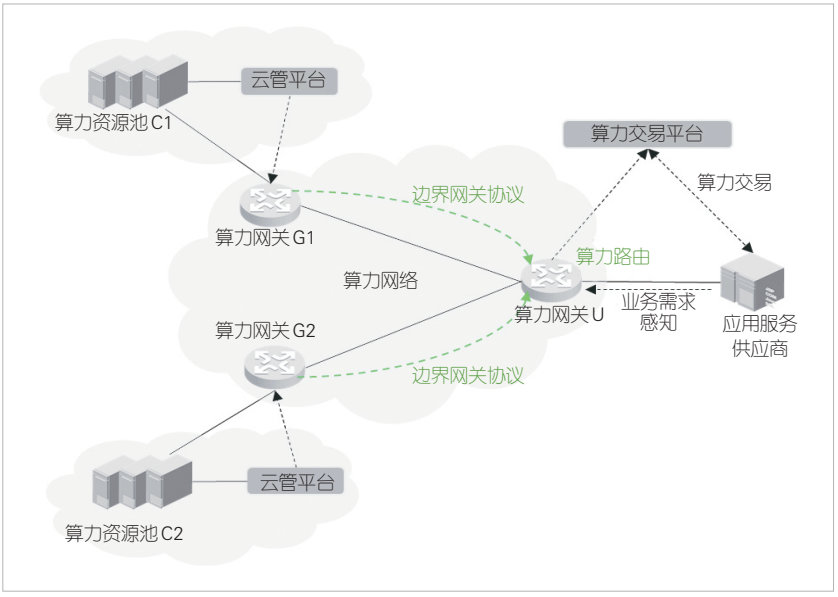


▲图6 支持SRv6的集中式算力路由控制

一步生成算力感知的新型路由表，用于支持后续业务转发。算力路由协议可以通过扩展基础路由协议来实现算网信息的通告。

以边界网关协议（BGP）为例，基于BGP的多协议扩展可利用BGP update消息中的路径属性预留字段TLV（一种可变格式）来扩展传递算力信息和网络信息。这种扩展的BGP协议就是算力边界网关协议（CP-BGP）。使用CP-BGP协议的算力路由控制如图7所示。

在算力网络中，算力资源池侧的算力网关可以感知算力节点的算力信息和网络信息，将相应信息填充到扩展的BGP update报文中，并通告用户侧的算力网关。用户侧算力网关可以接收扩展的BGP协议报文，解析算力信息和网络信息并生成BGP算力路由表。



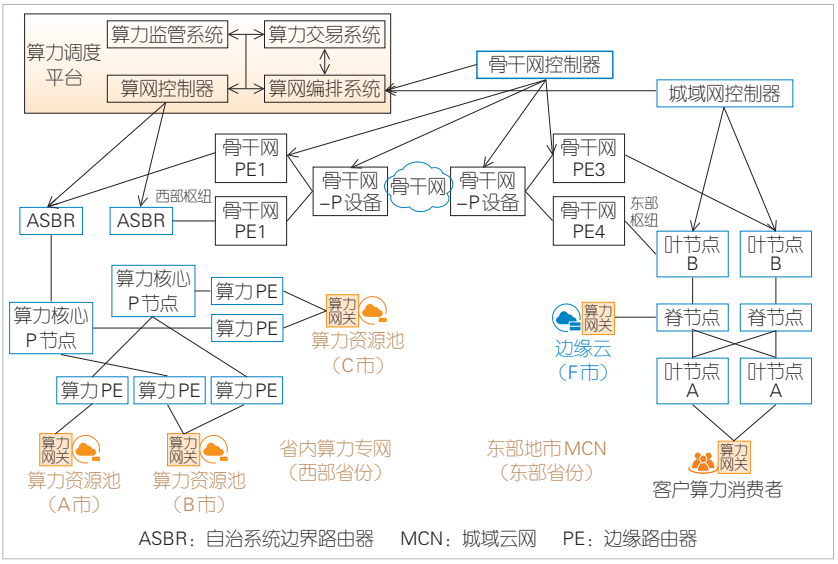
▲图7 支持算力边界网关协议的分布式算力路由控制

3 面向东数西算场景的实践

算力网关设备目前已经在“东数西算”业务场景成功落地应用，可以将东部算力需求有序引导到西部，促进东西部协同联动。

3.1 建设实施方案

算力网关的实践方案主要包括网络层面、管控层面两大部分，如图8所示。其中，网络层面包括西部的省内算力调度专网、运营商骨干网络以及东部的城域网络，管控层面包括西部省内的算网调度平台、骨干网控制器以及城域网控制器。各资源域网络控制器对接算网调度平台中的算网编排系统，同时基于部署在各资源池节点的算力网关设备，获取算力感知信息和算力路由信息，实现对云网资源的全局统一管控和调度。



▲图8 面向东数西算场景的算力网关实践方案

网络方案设计采用核心层和接入层两层架构，全路由器组网，如图9所示。

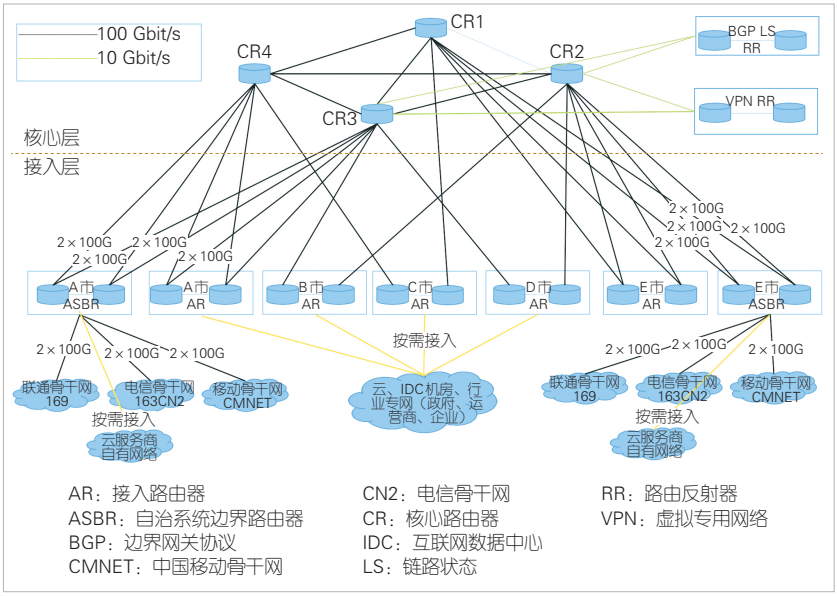
核心层核心路由器（CR）互联各市的接入路由器（AR）节点，虚拟专用网络（VPN）路由反射器（RR）负责VPN业务路由反射，BGP LS RR负责上送SR-TE信息。接入层每个地市部署2台AR，对接行业专网、IDC网络，并互联各云资源池。A地市和B地市各部署2台ASBR，对接各运营商骨干网络及云服务商自有网络。

核心层CR路由器之间采用Full Mesh互联。VPN RR和BGP-LS RR接入CR2和CR3。对于A地市和B地市的AR及ASBR，每组通过8条100GE线路交叉互联至属地CR路由器。对于其他地市AR，每组通过2条100 GE线路上联至核心层路由器。其中，一条上联A地市，另一条上联B地市。

路由协议设计采用公有AS号，并配有相应的IPv4/IPv6地址。网络架构采用SRv6技术路线，通过以太网虚拟专用网络（EVPN）统一业务面协议，并部署SRv6流量工程（SRv6-TE）。所有设备通过OpenAPI接口与控制器对接，并通过Telemetry上送网络运行数据。

3.2 实践成效

本次实践基于算力网关和算力调度平台，通过中国电信主导的西部多云协同算力调度专网、东部城域云网、骨干网



▲图9 面向东数西算场景的算力网络架构设计

以及各资源域网络控制器，对接统一云网编排系统，实现东西部之间的三维空间重构、实时云渲染等业务场景的全局可视调度。

本次算力网关的落地实践具有重大意义：一方面，中国电信在东数西算领域开展了创新尝试。省内算力专网及算力资源调度的实践充分验证了算力网关落地的可行性。另一方面，借助算力调度平台能够实现西部省份算力资源的统筹调度，打造全栈算力服务，全面提升信息技术（IT）资源利用率，助力产业数字化及数字产业化的发展[8]。

本次实践表明，算力网络和算力网关有助于实现算力设施由东向西布局，未来将带动相关产业有效转移，促进东西部数据流通、价值传递，延展东部发展空间，推进西部大开发形成新格局，提升国家整体算力水平。

4 结束语

算力网关通过网络控制面分发服务节点的计算能力、存储、算法等资源信息，力图打破传统网络的界限，将网络传送能力与IT的计算、存储等基础能力更好地结合起来，实现整网资源的最优化配置和使用，推动网络从“泛在连接能力平台”向“融合资源供给平台”升级演进。

算力网关的落地应用有助于实现算网一体化服务，有效提升资源利用率，减少网络资源和计算资源的浪费，降低整体能耗，助力东数西算战略落地[9]。

致谢

本文相关技术应用由中国电信股份有限公司、中电万维信息技术有限责任公司、中兴通讯股份有限公司、英特尔（中国）有限公司等单位共同完成。解云鹏、高守纪、乔建、田毅、何秀文、段敏等人承担了大量研发和试验工作。在此，向他们表示感谢！

参考文献

[1] 李正茂, 雷波, 孙震强, 等. 云网融合: 算力时代的数字信息基础设施 [M]. 北京: 中信出版集团, 2022
[2] 中国工信产业网. “四力”汇聚，算力网络发展迈入快车道 [EB/OL]. [2023-06-10]. https://www.cnii.com.cn/tx/202303/t20230320_455966.html
[3] 网络通信与安全紫金山实验室. 白盒交换机技术白皮书 [R]. 2021
[4] Github. SONiC system architecture [EB/OL]. [2023-06-10]. https://github.com/sonic-net/SONiC/wiki/Architecture
[5] 赵倩颖, 邢文娟, 雷波, 等. 一种基于域名解析机制的算力网络实现方案 [J]. 电信科学, 2021, 37(10): 86-92. DOI: 10.11959/j.issn.1000-0801.2021233
[6] 黄光平, 史伟强, 谭斌. 基于SRv6的算力网络资源和服务编排调度 [J]. 中兴通讯技术, 2021, 27(3): 23-28. DOI: 10.12142/ZTETJ.202103006
[7] DEERING S, HINDEN R. Internet protocol, version 6 (IPv6) specification [J]. RFC, 1995, 2460: 1-39. DOI: 10.17487/rfc8200
[8] 解云鹏, 马思聪, 田毅, 等. 从“东数西算”甘肃节点 看中国电信的算力调度探索与实践 [J]. 通信世界, 2022(22): 34-37
[9] 国家发展改革委, 中央网信办, 工业和信息化部, 等. 关于加快构建全国一体化大数据中心协同创新体系的指导意见 [EB/OL]. [2023-06-10]. https://www.gov.cn/zhengce/zhengceku/2020-12/28/content_5574288.htm

作 者 简 介

马思聪，中国电信股份有限公司研究院高级工程师；主要研究领域为未来网络、云网融合下的算力网络技术；主持和参与算力网关试点验证与研发工作；发表论文3篇。

孙吉斌，中国电信股份有限公司研究院研发工程师；主要研究领域为未来网络关键技术、算力路由协议等；先后参与“东数西算”场景下的算力网关试点部署与研发工作，以及算力路由协议标准的制定工作。

孙一豪，中国电信股份有限公司研究院研发工程师；主要研究领域为算力网络、IPv6标准和关键技术等。

算力网络四面三级算力度量技术体系

Three-Level and Four-Aspect Computing Measurement System in Computing Force Network*

杜宗鹏/DU Zongpeng，李志强/LI Zhiqiang，陆璐/LU Lu
（中国移动通信有限公司研究院，中国 北京 100053）
(China Mobile Research Institute, Beijing 100053, China)

DOI：10.12142/ZTETJ.202304003
网络出版地址：http://kns.cnki.net/kcms/detail/34.1228.TN.20230717.1807.006.html
网络出版日期：2023-07-18
收稿日期：2023-05-26

摘要：算力网络要实现广泛的算力互联，就离不开统一的算力度量和算力建模机制。算力度量技术主要解决算力描述的标准化，提供方便的跨厂商互通能力、算力资源协同管控能力。对算力网络中算力度量需求进行分析，并提出四面三级算力度量技术体系，同时还探索了在算力路由决策中，网络按需进行算力信息通告等关键技术，为后续的算力度量研究提供参考。

关键词：算力网络；算力度量；算力建模；算网融合；算力感知流量调度

Abstract: Unified measurement and modeling mechanisms are the guarantee for the computing force network to realize extensive computing interconnection. The computing measurement technology mainly solves the standardization of computing information description and provides convenient cross-vendor communication ability and collaborative control ability of computing resources. The three-level and four-aspect measurement system is proposed, and the key technologies such as on-demand computing information transmission for the computing aware routing decision are also explored, to form a reference for future computing measurement research.

Keywords: computing force network; computing measurement; computing modeling; computing and network convergence; computing-aware traffic steering

引用格式：杜宗鹏，李志强，陆璐．算力网络四面三级算力度量技术体系 [J]. 中兴通讯技术，2023，29(4)：8-13. DOI：10.12142/ZTETJ.202304003
Citation: DU Z P, LI Z Q, LU L. Three-level and four-aspect computing measurement system in computing force network [J]. ZTE technology journal, 2023, 29(4): 8-13. DOI: 10.12142/ZTETJ.202304003

随着互联网技术的不断进步，当今社会的发展逐渐呈现数字化、智能化的趋势。智能运算需要大量的算力来完成数据处理。为了缓解云数据中心的计算压力，获得更快的业务响应速度，算力逐渐从中心走向边缘，形成网络中泛在的算力资源。为了支持分散算力资源的统一感知、统一决策和统一管理，算力网络的概念被提出[1-5]。算力网络将提供算力和网络的一体化服务，支持算力资源和网络资源的联合优化，从而充分利用有限的算网资源，为用户提供高品质的算力服务。

不同算力业务的算力需求有所不同，不同算力节点的服务能力也各不相同。要实现灵活高效的业务调配和服务映射，算力网络首先需要对算力进行感知，而算力度量就是算力感知的基础之一。正如热力学温标的提出者开尔文勋爵曾提到的："如果你无法测量它，那么你就无法改进它"。

传统运营商主要提供网络资源，在用户需求、网络能力度量方面也有相对成熟的方案。网络度量常见的度量方式包括带宽和流量等。运营商围绕带宽和流量提供网络服务，进行网络运营，同时也可以基于时长、网络质量，与用户进行签约，并提供网络服务。

算力网络提供的是算力和网络的综合服务，这时需要一种方便对算力进行度量的机制。该机制一方面可以支持用户对算力需求的描述，另一方面也可以支持运营商、服务提供商或第三方对所提供的算力能力的描述。

基金项目：北京邮电大学-中国移动研究院联合创新中心基金资助（CMYJY-202000332）、东南大学-中国移动研究院联合创新中心基金资助（CMYJY-202100163）

* 作者确认算力网络译为computing force network

1 算力度量的需求架构

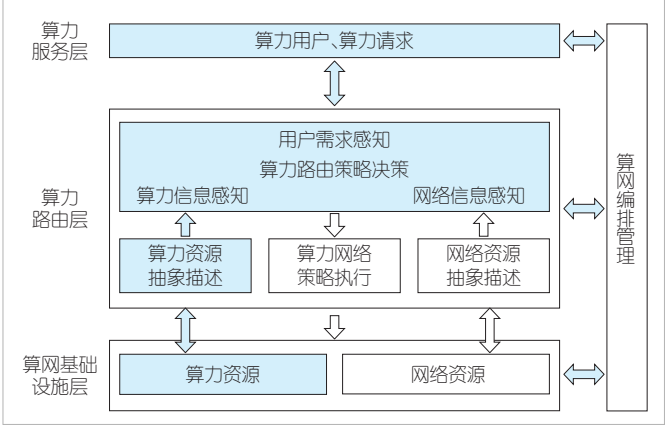
在算力网络中，算力度量的目标是关联和整合网络中的异构计算资源，使能多维资源的统一协同管理，从而通过统一的算力度量体系和异构计算资源的服务映射机制，实现算网资源的合理分配和高效调用。

目前，中国通信标准化协会（CCSA）TC3已经布局了算力网络的相关标准，包括算力网络总体技术要求、算力路由、算网编排、算力交易等。根据相关的工作[2, 6]，算力网络包括算力服务层、算力路由层、算网基础设施层和算网编排管理层。图1展示了算力网络总体架构中与算力度量相关的功能模块。在算力网络中，算力用户需求和算网资源的映射决策可能发生在算力路由层的策略决策模块或是算网编排管理模块。

目前，针对算力网络系统中网络资源的感知和通告，我们可以使用已有的网络操作维护管理（OAM）机制进行测量和收集。本文中，我们主要关注算力资源的感知和通告，以及算网信息的联合决策，尤其是算力度量信息在这些功能或流程中的需求和作用。

算力资源的有效管理，首先需要一个统一的算力资源模型，即对异构的算力资源进行建模。算力建模的目标是构建异构算力资源的统一描述方式。算力建模的具体信息包括数值型和非数值型。算力度量是算力建模的一部分，而算力资源的统一建模是算力度量的前提。算力度量中的算力评估值，可以更为方便地作为算力决策模块的输入参数，从而影响算力服务的部署和调度。算力度量和建模的相关需求，主要包括以下3个方面：

1）支持算力资源的度量和建模，对运营商、服务提供商及第三方算力资源节点所提供的算力进行可量化的能力描述；



▲图1 算力网络中算力度量需求

2）支持算力用户需求的度量和建模，对用户算力业务进行可分类分级的需求描述；

3）算力路由策略决策模块或算网编排管理模块可基于用户需求，合理分配算力节点来完成计算任务。

其中，前2个方面主要涉及算力度量和建模信息的抽象和通告，即把算力信息或算力需求信息按照约定的格式发送至算力路由策略决策模块或算网编排管理模块；第3个方面主要涉及算力度量信息的使用，即算力路由策略决策模块或算网编排管理模块如何利用算力度量和建模信息进行决策和管理，执行算网感知调度。

2 算力度量信息的使用

算力路由策略决策模块和算网编排管理模块都需要收集算力度量和建模信息，并据此进行策略决策。算力路由策略决策模块主要生成路由相关的策略，可根据感知到的算力服务的总体状态信息（尤其是算力服务节点当前的算力业务会话接入能力等动态参数），进行快速决策。算网编排管理模块支持更复杂的处理逻辑，例如与人工智能（AI）分析平台互通，根据更全面的算网信息进行网络和算力资源的全局优化。

算力路由策略决策模块可根据收集到的算力信息和网络信息，生成策略并执行，引导算力业务流量的高效转发。算力业务的流量指的是用户请求算力业务时所使用的连接建立报文，以及所需传给服务节点的数据报文，例如需要被分析的图片数据信息等。同时，算力路由层也支持算力业务的用户需求感知，以按需提供算网资源。

算力路由层的功能可以在网络控制器/网络转发设备的控制面上实现。算力路由层的策略决策模块可将收集到算力信息保存在一个或者多个专用的表中，并且支持按需更新。网络转发设备中默认的路由转发架构可以保持不变。这时，算力网络的路由策略优先级更高，即优先按照算力网络策略对算力业务流量进行转发。

算网编排管理模块支持算力调度策略的生成，甚至可与算力路由决策模块进行策略交互。算网编排管理模块感知到的信息会更加全面，因此能够支持更加复杂的决策逻辑，例如利用AI机制进行训练和推理，优化全网的算力业务。同时，算网编排管理模块需要根据感知到的算力资源情况，选择合适的算力节点，进行算力服务实例的部署。

2.1 算力服务部署

算力网络服务在部署时，需要对云、边、端各级架构中泛在异构设备进行纳管，还需要对中央处理器（CPU）、图

形处理器（GPU）、内存、磁盘、网络等多维算力资源进行感知和维护。当有服务部署需求时，算力网络系统需要基于维护的资源视图，通过适配算法为用户找到合适的算力资源以完成服务部署。

算力网络中各种异构的算力资源散落各地，用户不同的部署服务对各维度算力资源也有着不同的需求。算力服务部署要解决的问题是在掌握资源状态信息的情况下，如何合理调度资源，部署服务。算力服务的部署是算力网络实现低时延、高可靠优质服务调度的前提。

在云计算或边缘计算的领域，算力数据模型相关的多项工作都可以作为算力网络中算力度量和建模的参考。例如，用于云服务器带外管理的Redfish项目从不同维度定义了大量算力资源相关的模型；创建于2021年的Anuket项目支持云原生和虚拟网络功能、基础设施和服务的快速部署，致力于整合OpenStack和Kubernetes（K8s）等知名工具的不同架构。

云原生中流行的编排管理工具K8s支持集群内部的服务部署和资源调度。一个K8s集群包含至少一个主节点和若干从节点，主节点为集群管理节点，能根据各节点的资源状态完成容器到物理节点的分配[7]。K8s中的最小调度单位为Pod，一个Pod可以包含多个容器。每个Pod都可以在部署时指定其CPU和内存的数量。K8s的资源调度是指将Pod任务映射到物理资源的过程。在该过程中，主节点中的调度器Kube-scheduler负责决策一个Pod应该被调度到集群中的哪一个节点上。在服务节点的具体选择时，可以指定不同的节点优选策略，例如优选算力资源占用比较小的节点，或者是优选CPU和内存使用率接近的节点。

资源的感知和监控是资源调配的前提。有效的资源感知能够精确地传递资源状态信息，从而为后续的资源调度、服务迁移等业务奠定基础。例如，作为一款基于时序数据库的监控告警系统[8]，开源软件Prometheus支持多维数据模型，收集由度量名和键值对组成的时间序列数据，被广泛应用于云原生中的资源监控中。

算力网络的服务部署和资源调度模式可以参考上述云原生中的编排管理工具，在面向异构立体泛在的算力场景方面进行增强。通过了解各个算力节点的算力资源和剩余可用资源的情况，并根据算力业务需求，按需制定合适的部署策略，灵活地调度资源，从而完成算力服务部署，提升算力服务提供能力。

2.2 算力服务选择

在完成算力服务部署后，算力网络需要为算力用户找到并接入一个合适的算力服务节点。该节点需要离用户较近，满足用户的时延需求，还需要有充足的算力资源，支持算力业务的快速完成。这个过程被称为算力服务选择，或算力服务节点的负载均衡。算力服务节点的状态是实时变化的，因此在网络中实时同步这些信息存在一定的挑战。

在算力网络服务场景中，不同网络位置能够提供相同的算力服务，然而，此时不同服务节点的计算资源通常是不同的，计算资源的负载情况也在变化。尤其是在多接入边缘计算（MEC）的场景中，计算资源相对有限，因此MEC之间的协同就显得更加重要。

算力感知网络（CAN）[3, 9]为MEC的协同提供了一种基于网络的负载均衡机制。在相关场景中，整个网络就像一个虚拟的负载均衡设备一样运作。在CAN的机制中，策略决策点是一个网络设备，位于网络转发节点或是网络管控节点上。然而，对于一个网络设备而言，了解算力的状态还是一个新的技术领域。

3 四面三级算力度量技术体系

网络所提供的服务主要聚焦于转发，例如IP网络的核心就是按照目的地址，把报文转发到目标网络。计算所对接的服务会更加复杂，影响计算性能的因素也更多，因此较难给出单维度的度量单位。网络中的算力资源分散，异构算力资源种类繁多，是算力统一度量所面临的主要困难。

算力芯片的种类包括CPU、GPU、嵌入式神经网络处理器（NPU）、张量处理器（TPU）等。这些芯片都有各自的性能指标和应用场景，可以针对特定业务进行计算优化。例如，通用算力以CPU承载为主，主要面向通用软件应用，执行逻辑运算；人工智能算力以NPU/TPU/GPU承载为主，主要面向AI应用，逻辑简单，但计算密集、并发程度高；超算算力以CPU/GPU承载为主，主要面向科学计算、工业仿真等场景，计算复杂且对计算精度要求高。

在计算领域中，常见的一个度量单位是FLOPS，即每秒执行的浮点运算次数。但计算业务的执行效率和能力，并不仅仅取决于计算节点的浮点运算能力，其影响因素还包括输入/输出（I/O）效能、内存架构、缓存速率等。因此，在算力通告和服务调度中，需要综合考虑如算力节点的负载等的多种因素。

算力服务的性能受到芯片、存储、网络、平台软件等多维因素的影响，因此我们需要对算力服务节点进行综合评估。算力的度量和建模不仅可以参考制造商提供的规格算力值，例如FLOPS，还可以参考一些可以全面反映服务支持能力的综合指标值。为了描述算力网络中服务节点的算力，我

们提出了“四面三级”算力度量技术体系，从节点的计算、通信、内存和存储能力4个方面，按照三级指标的方式来建模和描述算力网络中异构立体泛在的算力，如图2所示。“四面”指标具体如下：

1）计算。节点资源的计算能力评估主要是指对CPU、GPU等计算资源运算能力的评估。根据计算程序的不同特点和需求，计算性能可分为整数计算性能、浮点计算性能以及哈希计算性能。

2）通信。节点资源的通信能力评估主要是指对节点连接到外部网络的通信能力的评估。从单个算力节点的角度考虑，节点的通信能力指标主要参考节点连接到外部网络的带宽大小。

3）内存。节点的内存（缓存）能力评估主要是对节点的内存（缓存）单元的性能评估。节点的内存（缓存）指标主要涉及节点内存容量和内存带宽。

4）存储。节点的存储能力评估主要是指对节点的外存储器（如硬盘）的性能评估。节点的存储指标主要涉及节点的存储容量、存储带宽和每秒读写操作数（IOPS）。

“三级”指标具体如下：

1）一级指标代表异构硬件算力度量，包括整数计算速率、浮点计算速率等；

2）二级指标代表节点服务能力度量，包括节点可提供的计算、通信、内存、存储4方面的能力等；

3）三级指标代表节点对业务的支撑能力度量，包括节点业务处理能力度量等，可以按照不同的服务类型进行描述。

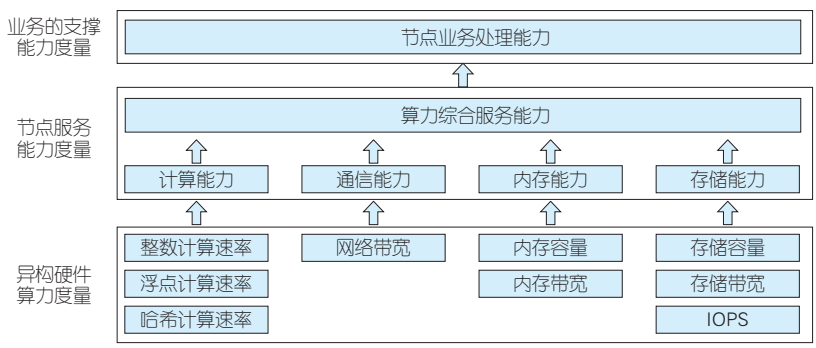
上述度量体系中的各种指标可以根据业务需求的不同，按需提供给其他的算力网络节点，如决策节点。下文中，我们将简要介绍上述相关指标的获取方式，以及一些示例性的指标参数。

1）一级指标可以通过不同的评估方法来获得，如通过基准测试程序测试、直接从厂商获取、公式计算等方式。该级别的指标可以是厂家提供的性能参数，例如CPU的型号、主频、内核数，GPU的型号、单精度浮点性能等，也可以是常用的基准测试程序的测试值。厂商提供的性能参数也可以包含动态的资源状态，例如目前可用的CPU/虚拟CPU（vCPU）核数等信息；基准测试程序的测试可分为整数计算性能测试、浮点计算性能测试等。

2）二级指标由一级指标抽象得到，主要用于对算力的综合评价，可提供节点某方面的或是综合的服务能力。节点某方面的能力可以通过执行对应方面的多次测试，并且基于输出结果进行运算后得到；综合服务能力可以基于多个方面的评估值，再次进行运算处理得到。文献[10]提供了一种基于PageRank的节点性能评价算法。在相关流程中，首先对每个节点使用主流基准测试进行评价，然后采用PageRank算法处理每个基准测试的执行结果，从而得到节点某方面的性能指标。

3）三级指标与节点上部署的业务相关，围绕业务的维度进行描述。三级指标提供的主要是与业务相关的评价指标，例如针对特定的算力业务，节点能够提供的实际处理能力等。文献[11]提供了一种“有效算力”的计算方法，用真实业务软件在一定规模的信息通信技术（ICT）基础设施系统上（含计算、存储、网络、软件中间件等）进行性能测量，之后通过几何加权平均，对多个真实业务性能测试结果进行整合。例如，在人工智能领域，对特定训练作业的有效计算能力是单位时间内训练过程消耗的样本数量，或称为业务吞吐量。在视觉类测试中，业务吞吐量单位是图片数每秒；在自然语言处理类测试中，单位为句数每秒。

上述度量体系中各个维度的信息，可以用于网络算力服务的调度和算力业务的转发决策，从而提高算力资源和网络资源的利用率。同时，上述评估机制反应了算力节点对计算任务的支撑能力，牵引最终用户从使用效果的角度，进行算力节点的规划和建设。



▲图2 算力资源抽象描述的四面三级度量体系

算力度量的指标包括静态指标和动态指标。前者通常与分配给服务的算力资源有关，后者通常与服务器在运行时能够提供的可用资源有关，例如，服务器当前的可用会话数或CPU利用率。在算力服务的部署和节点选择中，上述的动态指标更具参考性。例如，在部署算力服务时，需要了解当前算力节点可用的CPU/vCPU核数，并将其视为一种较为动态的一级指标；在选择算力服务节点时，业务相关的动态指

标包括当前服务节点状态等，可将其视为动态的三级指标。

这些动态指标可以与上文所述的规格算力信息、综合节点算力信息、业务处理能力信息综合考量，从而影响算力服务的决策。这些动态指标可以指定不同的更新频率与算力网络的决策点进行通信，更新相关的动态算力信息。

4 算力度量的关键技术

算力网络的定位是面向未来网络中泛在的算力资源，支持网络和算力的融合统一，提供算网一体化服务的新型信息基础设施。算力网络的愿景是推动算力像水电一样，逐步成为“一点接入、即取即用”的社会级服务[1]。算力度量在算力网络中的作用类似于水网、电网中水表和电表的作用，是算力网络的重要基础技术之一。

为了实现异构算力资源的统一度量和统一建模，支持跨厂商算力网络节点互通，方便算力网络的管理和运营，算力度量和建模机制标准化工作已经被提上日程。目前，中国通信标准化协会（CCSA）已经有了一些正在进行中的算力度量相关的标准化项目。同时，国际互联网工程任务组（IETF）的CATS工作组也在推进该领域的工作。

算力信息的格式设计与使用场景密切相关，不同的使用场景有着不同的算力度量指标的组合和选择。本文中，我们围绕“四面三级”算力度量技术体系，对算力信息如何在网络中传递也进行了探索。其核心的两个问题为：1）算力信息的格式，即应该传递什么样的算力信息到网络中；2）算力信息的通告方式，即在网络中以什么样的方式通告算力信息。

4.1 算力度量信息的格式设计

算网编排管理模块可以通过定义算力相关的YANG模型来传递算力信息；算力路由模块可以通过扩展目前的路由协议来传递算力信息。本节将主要探索后者的格式，并给出一种示例性的描述方式。路由协议携带的算力信息会更加简单，一方面这样可以减少网络中的信息传递，避免不必要的信息暴露；另一方面，可以使路由决策流程快速完成。

不同的业务可能有不同的计算需求，需要使用不同的算力路由决策依据。因此，算力度量的指标体系应该可扩展、可编程，支持灵活自定义，以便能够反映不同业务甚至未来业务需求。对于特定的业务，决策点应能订阅想了解的算力度量指标，以及指定指标的相关更新频率。也就是说，决策点可以按需选择感兴趣的各级细节指标进行了解，并作为决策的依据。

作为参考，在K8s的Pod资源调度中，调度器会执行初选和优选两个步骤[7]。初选时，调度器会过滤掉不满足Pod资源需求的节点；优选时，调度器会根据设定的策略对初选后的节点进行评分，并据此决定调度节点。在算力网络中，对于一个选择服务节点的业务而言，首先要确定的是目标服务节点是否还能够接入新的会话，其次需要按照业务特定的策略，优选综合性能更高的算力节点来提供服务，例如有些业务有较快的业务完成时间的需求。在文献[12]中，建议使用动态的参数来反应服务状态，使用相对静态的参数来反应节点执行该服务的能力信息。

本节中，我们提出了一种示例性的算力度量信息的格式设计思路：首先要携带算力节点的服务状态信息；其次，需要按照业务需求或优化目标，携带一个或者几个算力信息相关的类型–长度–值（TLV）。这里不同的业务类型可以根据业务需求，自定义不同的TLV组合。

在服务节点选择时，算力路由的决策点首先要感知备选的算力节点是否还能接入新的算力用户。例如，文献[9]提到了一种算力节点忙闲状态的通告方式。该方式设定了3种状态：红色代表繁忙状态，绿色代表空闲状态，黄色代表即将繁忙的状态。我们可以在服务节点上设定阈值，触发这些状态的改变，例如接入的用户数达到阈值，或某种资源（如CPU、内存等）的利用率达到阈值。

如果多个算力节点都标记了绿色的空闲状态，同时算力业务又有较快的业务完成时间需求，那么也可以进一步地按照算力节点的综合性能来决策接入哪个节点。这些性能的综合评估也是算力度量体系的一部分，一般体现为三级指标或二级指标，或多级指标的组合等。

如果面向的业务较为明确，并且存在相关业务的有效算力测试结果，那么可以优先考虑使用三级指标中的有效算力来评估算力节点的计算能力。对于相同的业务，如果算力用户之间的计算需求差异较大，仅使用三级指标并不能很好地反映节点的计算能力；或是由于面向新的业务，没有可参考的有效算力结果评估，那么可以考虑二级指标中的综合服务能力指标。在该指标中，针对不同的业务，在计算、通信、内存、存储4个方面有不同的系数，即影响因子。例如，目标业务如果是计算密集型的，就可以适当调高计算维度的影响因子。

如果业务对于能耗比较敏感，除了算力相关的信息外，也可以要求算力节点提供能耗相关的信息。这时需要对算力节点的能耗进行统一建模，从而将其作为算力服务调度的参考因素之一，即在一定程度上优先使用较为节能的算力节点来完成计算任务。

4.2 算力度量信息的通告

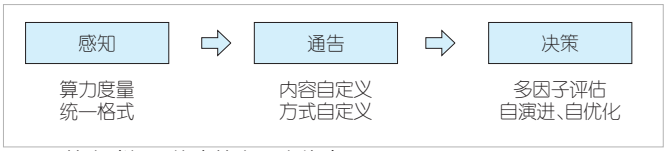
CAN是算力网络中的关键技术之一。图3为CAN算力通告流程：首先，对于泛在的算力节点，需要统一度量值的描述方式；其次，通过某种方法，将这些节点的算力能力通告到网络中；最后，算力网络的决策点可以参考收到的算力信息来进行算力服务调度。上一节主要关注的是第一步，即描述算力能力和服务状态的方法。一个标准化的算力能力的描述方法能够在很大程度上提升算力网络不同节点之间的协同能力。本节中，我们主要关注第二步，这其中也会涉及相关的标准化工作。

通常，算力节点和CAN并不在同一个网络域中，因此算力信息的通告经常会考虑使用边界网关协议（BGP）。BGP可以用于网络中边缘路由器（PE）节点之间的路由信息交流，是计算机网络实现广泛互联的基础协议之一。针对特定的算力服务，可以考虑扩展BGP协议，如在通告该服务的路由信息时，BGP消息中携带节点的忙闲信息，以及节点的业务执行能力TLV。

在路由协议的相关扩展中，网络中的多个算力节点会共享同一个Anycast（任播）地址，并同时发布自身相关算力信息。一方面，多点部署的Anycast地址本身就不支持按照前缀汇聚；另一方面，不同的算力节点的算力情况不同，在BGP路由优选时，需要对BGP协议进行扩展以支持对同一个Anycast地址存在不同的Nexthop（下一跳）以及算力信息的情况。

5 结束语

算力度量是算力网络技术的重要基础之一。统一的度量标准、描述方式，有助于算力网络的各个网元之间进行高效协作。目前，算力网络业务以及针对相关业务的度量还处于发展的初期阶段。未来，随着算力网络业务的普及，统一的算力度量机制将促进算网融合共生，避免“算力孤岛”，实现全网算力的高效协同，从而使整个算力网络能够像一台虚拟的计算机一样统一调度、统一管理，支撑千行百业的数智化转型。



▲图3 算力感知网络中算力通告的流程

参考文献

[1] 中国移动. 算力网络白皮书 [R]. 2021
[2] CCSA. 算力网络 总体技术要求:CCSA TC3 WG3 [S]. 2021
[3] 姚惠娟, 陆璐, 段晓东. 算力感知网络架构与关键技术 [J]. 中兴通讯技术, 2021, 27(3): 7-11. DOI: 10.12142/ZTETJ.202103003
[4] 雷波，赵倩颖，赵慧玲．边缘计算与算力网络综述 [J]. 中兴通讯技术，2021,27(3):
[5] 唐雄燕，张帅，曹畅．夯实云网融合 迈向算网一体 [J]. 中兴通讯技术，2021,27(3):
[6] CCSA. 面向算网融合的算力度量与算力建模研究 [R]. 2021
[7] ZHANG W G, MA X L, ZHANG J Z. Research on kubernetes´ resource scheduling scheme [C]//Proceedings of the 8th International Conference on Communication and Network Security. ACM, 2018: 144-148. DOI: 10.1145/3290480.3290507
[8] LEE S, SON S, HAN J, et al. Refining micro services placement over multiple kubernetes-orchestrated clusters employing resource monitoring [C]//Proceedings of 2020 IEEE 40th International Conference on Distributed Computing Systems (ICDCS). IEEE, 2021: 1328-1332. DOI: 10.1109/ICDCS47774.2020.00173
[9] DU Z P, LI Z Q, DUAN X D, et al. Service information informing in computing aware networking [C]//Proceedings of 2022 International Conference on Service Science (ICSS). IEEE, 2022: 125-130. DOI: 10.1109/ICSS55994.2022.00027
[10] SONG D, CHEN X S, RUI L L, et al. Resource allocation algorithm based on modeling of ubiquitous network node capability [J]. Journal of physics: conference series, 2022, 2224(1): 012089. DOI: 10.1088/1742-6596/2224/1/012089
[11] 中国电子技术标准化研究院. 计算中心有效算力评测体系白皮书 [R]. 2022
[12] DU Z P, FU Y X, LI C, et al. Computing information description in computing-aware traffic steering: draft-du-cats-computing-modeling-description-00 [S]. 2023

作 者 简 介

杜宗鹏，中国移动通信有限公司研究院基础网络技术研究所研究员；研究方向为算力网络、未来IP网络、确定性网络等；发表论文10余篇。

李志强，中国移动通信有限公司研究院基础网络技术研究所研究员；长期从事未来IP网络演进、标准和技术研究工作，主要涉及下一代IP网络、算力网络、云网融合、SDN/NFV、5G核心网等。

陆璐，中国移动通信有限公司研究院基础网络技术研究所副所长、中国通信标准化协会TC5核心网组组长；长期从事移动核心网策略、演进、标准和技术研究工作，主要涉及未来网络架构、智能管道、边缘计算、算力网络等领域。

东数西算下面向业务的路由策略分析与探索

Analysis and Exploration of Service Oriented Routing Strategies in East-Data-West-Computing Requirement Transfer

魏汝翔/WEI Ruxiang，刘琦/LIU Qi，赵广/ZHAO Guang，曹畅/CAO Chang，唐雄燕/TANG Xiongyan

（中国联合网络通信有限公司研究院，中国 北京100048）
(The Research Institute of China Unicom, Beijing 100048, China)

DOI：10.12142/ZTETJ.202304004
网络出版地址：http://kns.cnki.net/kcms/detail/34.1228.TN.20230717.1807.006.html
网络出版日期：2023-07-18
收稿日期：2023-06-12

摘要：基于各类业务对算力与网络指标的需求，结合当前网络流量现状与未来网络流量发展趋势，分析了包含算力业务的路由策略与实现机制，探索在东数西算场景下，面向业务和算力分布的路由策略。提出了一种基于业务属性与算力资源分布的混合式路由策略，将全国分成若干“算力区域”，在不同算力区域之间采用集中式路由决策，在算力区域内部采用分布式路由决策。该策略能够提供一种安全、灵活、高效的全局路由方法。

关键词：东数西算；算力网络；算力业务；路由策略

Abstract: Based on the demand of various services for computing power and network indicators, combined with the current status of network traffic and the future trends of network traffic development, the routing strategy and implementation mechanism including computing power business are analyzed, and the routing strategy oriented to business and computing power distribution under the scenario of east-data-west-computing is explored. A hybrid routing strategy based on service attributes and computing power resource distribution is proposed, which divides the whole region into several "computing power regions". The centralized routing strategy is adopted between different computing power regions, and the distributed routing strategy is adopted within the computing power regions. The hybrid routing strategy can provide a secure, flexible, and efficient overall routing strategy.

Keywords: east-data-west-computing; computing power network; computing power service; routing strategy

引用格式：魏汝翔, 刘琦, 赵广, 等. 东数西算下面向业务的路由策略分析与探索 [J]. 中兴通讯技术, 2023, 29(4): 14-18. DOI: 10.12142/ZTETJ.202304004

Citation: WEI R X, LIU Q, ZHAO G, et al. Analysis and exploration of service oriented routing strategies in the east-data-west-computing requirement transfer [J]. ZTE technology journal, 2023, 29(4): 14-18. DOI: 10.12142/ZTETJ.202304004

1 算力业务的需求指标和分类

在数字经济时代，算力如同农业时代的水利、工业时代的电力，既是国民经济发展的重要基础，也是科技竞争的新焦点。算力是设备或平台为完成某种业务所具备的处理业务信息的关键核心能力，包括逻辑运算能力、并行计算能力、神经网络加速能力等[1-2]。

算力业务除了对上述算力资源有要求外，还会根据具体的业务类型，对网络的上下行带宽、时延、抖动、存储等方面有不同的要求。算力业务从应用的角度可以分为计算类、存储类、互动类，这3类业务会互相重叠、交叉[3]，一种算力业务经常同时属于上述多种分类。不同业务对算力与网络指标的要求不尽相同。总的来说，对于计算类与存储类等对网络时延与抖动要求不高的算力业务，适用于“东数西算”场景，可将海量数据传输到西部进行计算与存储；而对于互动类业务，不适用于东数西算，应合理利用边缘算力，就近提供服务。

东数西算对传统基于距离的路由策略提出了新的需求。在路由决策时，不再将所有业务都分配到距离（包括跳数、开销、时延等）最近的节点，而是综合考虑业务属性与算力资源的分布，在已知各类业务对网络指标、算力与存储需求的情况下，通过路由策略对各类业务进行分配：将对实时性要求较低的冷数据分配到西部算力枢纽节点，将对实时性有

一定要求的温数据分配到东部算力枢纽节点，将对实时性与可靠性要求较高的热数据分配到距离最近的边缘算力节点。

2 现网流量对东数西算业务路由策略的影响

当前电信运营商的网络流量主要以视频业务为主，预计未来网络流量中视频流量的占比将进一步增加。据Omdia网络流量报告预测[4]，2019年视频流量占全球总流量的76%，到2024年，视频流量将占到总流量的80%。据爱立信移动市场报告预测[5]，到2025年，移动视频流量在移动数据总流量中的占比将从2019年的略高于60%增长到近75%。从用户角度来看，视频业务主要以下行流量为主。无论是当前主流的短视频、社交媒体视频等业务，还是不断发展的4K、8K等超高清视频点播类业务，均需要占用大量的网络下行带宽。此外，虚拟现实（VR）、混合现实（MR）等新兴业务也对网络的下行带宽提出了很高的要求。

当前中国电信运营商城域网流量主要以下行流量为主。以中国联通为例，在城域网家庭宽带流量中，上行流量与下行流量比值大约为1∶3；在移动互联网流量中，上行流量与下行流量比值大约为1∶7。据中国信息通信研究院的监测报告[6]，2022年第1季度，中国5G平均用户下载速率为304.8 Mbit/s、平均用户上传速率为49.6 Mbit/s；4G平均用户下载速率为27.8 Mbit/s、平均用户上传速率为2.9 Mbit/s；Wi-Fi平均用户下载速率为179.7 Mbit/s、平均用户上传速率为36.3 Mbit/s。当前主流的固网宽带业务、移动网络4G或5G业务、移动终端通过Wi-Fi上网等业务的下载速率远远高于上传速率。随着未来视频业务占比的不断增加，网络下行带宽与上行带宽仍然会存在利用率不均衡的问题。

对于计算类与存储类的算力业务，东数西算可以将东部的海量数据从用户侧通过网络传输到西部算力枢纽节点，这需要占用大量的网络上行带宽。对于以下行流量为主的电信运营商网络而言，东数西算刚好能够复用当前电信运营商的城域网带宽。此外，远程医疗、高清直播等业务对网络带宽的需求也主要集中在上行带宽。如果在路由决策时能够分开考虑上下行带宽利用率，电信运营商的网络带宽将会得到更充分的利用。因此，在路由决策时除了要满足业务对算力、时延、带宽等业务属性的需求外，还要充分、有效地利用电信运营商的网络资源以及国家算力枢纽节点的资源。

3 算力业务的路由策略分析

软件定义网络（SDN）的出现，给多业务场景中的路由策略制定提供了新的思路[7]。SDN与传统网络的区别体现在转发平面和控制平面的解耦。相较于传统网络，SDN架构下的控制层可以利用编程的方法对数据层面的转发行为进行控制，从而更加灵活地管理网络。

基于IPv6段路由（SRv6）可以实现报文转发路径的可编程。同时，采用SRv6技术可以简化网络结构，实现网络之间的无缝衔接。这使得路由能力不再割裂[8]，并可以结合SDN控制器实现业务的路由决策。基于SRv6算力路由技术的相关行业标准已完成初稿的编制，正在意见征求中。

算力业务的集中式路由决策正是利用SDN与SRv6的思想，首先通过组合使用边界网关协议段路由连接状态（BGP-LS）、随流检测（iFIT）、Telemetry等技术，实现网络上下行带宽利用率、时延、抖动、丢包率等网络指标的实时检测[9]；然后通过算力感知等技术获取全网的算力及存储信息；再根据不同业务对算力及网络性能的需求，通过控制器依次为各种业务分配最优路径。目前来看，算力业务的集中式路由决策的标准化及设备性能等都较为成熟[1]。

集中式路由决策的方式需要控制系统掌握全网实时的详细信息，根据不同的业务类型以及用户对业务服务级别协议（SLA）的不同需求，将业务分配到能够满足用户对算力、存储及网络能力需要的节点。全网设备及链路众多，算力与网络信息的及时更新，对控制层的计算、存储都提出了较高的要求。例如，当因业务的分配或完成而导致网络指标或算力资源发生变化时，或因新设备上线、原设备掉线以及链路通断状态的改变引起网络拓扑结构变化时，都需要控制层及时更新网络信息，重新完成路径规划再将信息下发至网络头节点。这一过程所需的时延会极大地影响用户体验，甚至影响业务的正常运行。

与集中式路由决策相对的是分布式路由决策。分布式路由决策由网络中的路由器设备完成路由决策与转发。这其中的关键在于对算力的感知与标识，以及如何将计算能力与网络状态信息发布到全网。为了解决上述问题，计算优先网络（CFN）的概念被提出[10-11]。

CFN延续了传统分布式路由协议的设计思路，通过对网络架构和协议的改进，将计算能力与网络资源作为路由信息发布到网络，并路由到相应的计算节点，从而实现计算能力与网络资源的优化和高效利用。CFN的核心功能在于算力资源感知和算力任务调度。

由电信运营商与部分业内主流厂商提出的《算力网络 算力路由协议技术要求：OSPF协议扩展》当前已在中国通信标准化协会（CCSA）立项。该标准定义了开放式最短路径优先（OSPF）相关协议的扩展选项，如OSPFv2/OSPFv3协议。这些协议能够携带算力信息和网络信息，推动了分布式算力路由决策的标准化工作。

分布式路由决策具有良好的扩展性，但实现起来比较复杂。目前，算力编排与网络控制等方面的标准化工作还未完成[12]，距离应用还很远。此外，将算力服务标识、路径规划、路由决策等控制权交由网络中分散的节点设备，容易导致业务流量被篡改、攻击。两种路由决策的对比如表1所示。

目前，3家电信运营商联合高校、企业正在起草行业标准——《算力网络算力路由协议技术要求》。该标准定义了算力信息和网络信息的感知和通告方法，给出集中式与分布式两种通告方式的优点及其应用场景，并重点针对算力网络集中式控制器提出相关的技术要求。

随着算力网络国家枢纽节点的建设，算力资源将向三大区域集中：西部算力枢纽节点、东部算力枢纽节点、各地市的边缘算力节点。各区域的地理位置及它们之间的网络连接基本不变，但区域内部的算力资源与这些资源之间的网络连接将随着算力网络的发展而不断变化。若采用集中式路由决策，那么上述算力资源与网络连接的频繁改变会严重影响控制器的信息更新与路径规划；若采用分布式路由决策，则无法通过全局统一的管控与路径规划来实现业务路径的全局最优。

4 东数西算下面向业务的路由策略探索

为了解决上述问题，本文中我们结合两种现有的路由策略，在东数西算场景下探索出一种基于业务属性与算力资源分布的混合式路由决策方式。我们将国家算力枢纽节点、区域边界节点等重要算力区域中的关键设备，以及电信运营商各城域网、骨干网中的核心设备集中管理。基于自身的分布式路由决策能力，这些关键设备能够感知所在的算力区域中各种网元设备的算力及相互网络指标等信息，然后将这些信息上报给控制层，再由控制层统一进行集中式路由决策。这样可以将业务调度到全局最优的算力区域，再由以上的这些关键设备通过分布式路由决策以局部最优的方式调度到自身所在算力区域的具体网元中。例如，如果人们要去某个商业中心的餐厅用餐，首先用手机软件（相当于控制层）通过全局最佳路径导航到该商业中心（相当于集中式路由决策），再通过查看商业中心的内部楼层介绍或询问等方式选择局部最佳路径，从而到达餐厅（相当于分布式路由决策）。

当区域内部算力或网络信息变化不大时，区域关键设备无须将这些信息实时上报给控制器，因此不会对全局集中式路由策略造成影响。通过设置区域资源告警，在区域内部算力资源不足或网络状态劣化达到一定程度后，区域关键设备将算力或网络信息上报给控制器，并且在后续的集中式路由决策时不再将新业务调度到该算力区域。这样既可以大幅简化控制层所需要的全网设备及链路数量，又能在部分网络指标、拓扑结构或算力资源发生微小变化时，降低区域关键设备对控制层实时更新网络信息以及重新完成路径规划所造成的时延影响。

▼表1 集中式路由决策与分布式路由决策对比

对比指标	集中式路由决策	分布式路由决策
算法思想及关键技术	SDN、SRv6	CFN、BGP
路由策略的标准化程度	较高	较低
对设备能力的要求	较高	较低
网络与设备变动对路由决策的影响	较大	较小
路径重新计算所需时间	较长	较短
路径调整灵活性	较复杂	较灵活
网络管理灵活性	较灵活	较复杂
业务安全性	较好	较差
路径规划效果	全局最优	局部最优

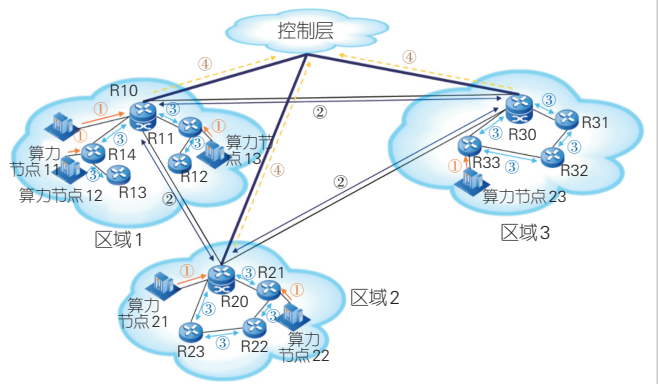
BGP：边界网关协议　　SDN：软件定义网络
CFN：计算优先网络　　SRv6：基于IPv6段路由

基于业务属性与算力资源分布的混合式路由决策的算力与网络信息通告流程如图1所示。

1）区域1、2、3中的所有路由器分别完成算力资源信息与网络指标信息的采集。需采集的信息包括各算力资源节点的逻辑运算能力、并行计算能力、神经网络加速能力等算力资源信息和其存储能力，以及路由器之间的链路上下行带宽及利用率、网络时延、丢包率、抖动等网络指标信息。

2）区域1、2、3中的关键设备（即路由器R10、R20、R30）分别通过BGP扩展协议等方式互相通告它们之间的网络连接状态信息。

3）区域1、2、3中的所有路由器分别在各自区域内部将采集到的上述信息进行域内通告。待网络收敛后，所有路由器均生成了各自区域内部的算力服务路由表项。



注：①②③④代表算力网络信息通告流程

▲图1 混合式路由决策的算力与网络信息通告示意图

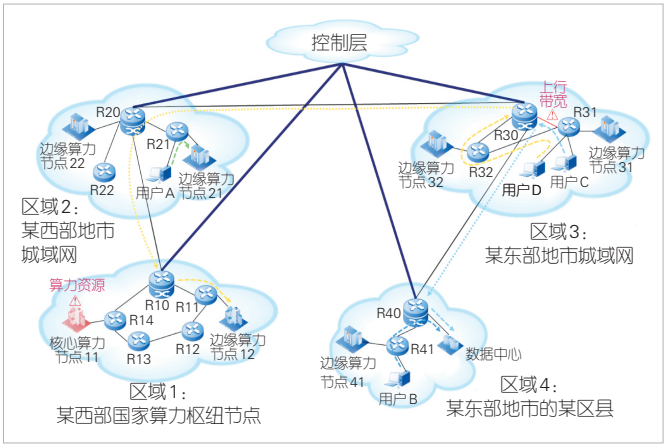
4）区域1、2、3中的关键设备将它们之间的连接信息与各自所在区域的算力资源信息上报给控制层。

在控制层进行集中式路由决策时，仅需考虑将业务分配到哪个区域。待业务到达目标区域的关键设备时，关键设备通过分布式路由决策，逐步将业务转发到区域内部最终的算力资源节点。当区域内部算力资源与网络状态的改变程度大于告警域值时（如内部大型算力节点故障或某条核心网络链路中断），该区域的关键设备将重新收敛后的算力与网络资源信息上报给控制层，从而使控制层能够根据最新的区域信息重新进行集中式路由决策；而当区域内部算力资源与网络状态的改变程度小于告警域值时（如内部小型算力节点故障或某条非核心网络链路中断），该区域的关键设备则无须将收敛后的算力与网络资源信息上报给控制层，这样不会对控制层进行集中式路由决策造成影响，仅对业务到达目标区域后的分布式路由决策产生影响。

我们通过以下4种典型的场景，介绍在东数西算背景下，基于业务属性与算力资源分布的混合式路由决策是如何将用户的业务分配到合适节点的，具体如图2所示。

场景1：位于某西部地市的用户A希望使用车联网来实现自动驾驶业务。客户端则根据该用户选择的业务类型，判断出此业务对时延非常敏感，需要将此业务分配到离其物理位置最近的、时延最小的边缘算力节点，因此客户端决定采用分布式路由决策在区域内部进行分配，用户所属的路由器R21将该业务分配到边缘算力节点21。

场景2：位于某东部地市某区县的用户B希望观看某高清视频。客户端根据该用户选择的业务类型，判断出该用户所选业务对网络下行带宽要求很高，且用户希望观看的高清视频所在数据中心的位置与用户位于同一区域。因此客户端决定采用分布式路由决策将该业务分配到存储对应资源的数据中心，用户所属的路由器R41将该业务通过路由器R40分配到数据中心。



▲图2 4种场景中基于业务属性与算力资源分布的混合式路由决策示意图

场景3：位于某东部地市的用户C希望观看某高清视频。与上一场景类似，客户端判断出用户希望观看的高清视频所在数据中心与用户位于不同区域，因此决定首先采用分布式路由决策将业务通过路由器R31派往区域内的关键路由器R30，然后采用集中式路由决策将业务派往区域4的关键路由器R40。当业务到达R40后，再通过分布式路由决策，将其分配给存储对应资源的数据中心。

场景4：位于某东部地市的用户D希望备份大量的医疗影像数据并进行智能影像分析。客户端根据该用户选择的业务类型及服务级别，判断出该用户所选业务对网络的上行带宽要求很高，且需要一定的逻辑运算能力、并行计算能力以及存储能力，对网络时延无要求，因此决定将其分配给西部国家算力枢纽节点，实现东数西算。控制层根据当时掌握的全网算力与网络资源信息，以及用户所在的位置信息，通过集中式路由决策规划出业务大方向，即经过区域3的关键路由器R30到区域2的关键路由器R20，再到区域1的关键路由器R10。在区域3中，由用户所属的路由器R31到关键路由器R30的路径由分布式路由决策进行规划。因用户所属的路由器R31检测到其至路由器R30的直连链路上联带宽利用率很高，而通过路由器R32到达路由器R30的链路上联带宽利用率很低。因此，路由器R31将业务通过路由器R32发送给区域3的关键路由器R30。核心算力节点11的并行计算能力与存储能力均不足以满足业务要求，而核心算力节点12的逻辑运算能力、并行计算能力、存储能力等性能均能满足业务要求。因此，当业务到达某西部国家算力枢纽节点所在区域1的关键路由器R10时，根据分布式路由决策，关键路由器R10将业务通过路由器R11分配到核心算力节点12。

5 混合式路由决策的应用

当前，由3家电信运营商联合行业知名企业共同起草的《算力网络混合式组网技术要求》已在CCSA立项。该标准综合集合式算力路由与分布式算力路由的优点，提出了混合式算力路由技术方案：由上层控制器/编排层完成算力信息的收集与分发，将算力状态通过上层系统向网络节点设备进行扩散与通告，再由网络节点结合计算与网络状态进行路由决策及算力服务选择，同时结合操作维护管理（OAM）技术来实现算力服务可用性状态的实时通告。该方案主要针对车联网等时延要求高的场景，通过混合式算力路由技术实现路由的快速收敛，并形成可快速落地应用的现网部署技术方案。

由于算力网络的快速发展及东数西算政策的加速实施，

算力业务的数量逐步增加，面向算力网络路由决策技术的落地需求也相应增加。当前，CCSA《算力网络 混合式组网技术要求》的项目组正在开展面向车联网场景下基于该方案的原型研发，以混合式算网一体路由调度为核心，立足具体应用场景，构建端到端的算网服务解决方案，赋能车联网等需求场景。目前该方案已具备三大核心技术能力：综合算+网因子进行服务选择及路径计算，为应用端提供最优算力服务；为终端的低时延应用提供快速调优的算力路由能力；提供网络层与应用层的移动业务连续性解决方案，实现无感知服务切换。后续我们将结合具体的实验环境进行应用部署与测试，以推动基于算力业务的算力网络混合式路由决策方案在现网场景中落地。

6 结束语

在中国东数西算政策支持下，通过路由决策将不同业务高效地分配到合适的位置，将成为算力网络发展的主要方向之一。本文中，我们首先梳理出各类业务对算力与网络指标的需求，分析、研究了包含算力业务的路由策略与实现机制，在集中式路由决策与分布式路由决策的基础上，结合当前网络流量现状与未来网络流量发展趋势，探索出一种在东数西算场景下，基于业务属性与算力资源分布的混合式路由策略。该方法结合集中式与分布式两种路由策略的优势，适用于各类传统非算力业务与新型算力业务，能够提供一种安全、灵活、高效的全局路由策略。随着未来技术标准的不断完善及商业模式的逐步明确，算力网络的路由策略及其应用实现将得到进一步的研究和落地。

致谢

作者在本文写作中得到了中国联通研究院易昕昕、庞冉的帮助，谨致谢意！

参考文献

[1] 易昕昕, 马贺荣, 曹畅, 等. 算力网络可编程服务路由策略的分析与探讨 [J]. 数据与计算发展前沿, 2022(5): 23–32
[2] 李建飞, 曹畅, 李奥, 等. 算力网络中面向业务体验的算力建模 [J]. 中兴通讯技术, 2020, 26(5): 34–38, 52. DOI: 10.12142/ZTETJ.202005007
[3] 邵杭青, 马蕴颖, 梁汗臣. 算力调度关键要素及路径分析 [J]. 江苏通信, 2022, 38(6): 76–80. DOI: 10.3969/j.issn.1007–9513.2022.06.018
[4] Omdia. Network traffic forecast report: 2019–24 [EB/OL]. (2020–07–09) [2023–05–20]. https://xueqiu.com/3861190056/153497200
[5] 爱立信. 爱立信移动市场报告 [EB/OL]. (2022–06–30)[2023–06–10]. https://www.ericsson.com/en/mobility–report/reports,2022
[6] 中国信息通信研究院. 2022年第一季度5G云测平台监测报告 [EB/OL].(2022–05–06) [2023–06–10]. https://www. cnii. com. cn/gxxww/ssgx/202205/W020220510338779858440.pdf
[7] 李然. SDN环境下基于链路状态感知的路径优化方法研究 [D]. 北京: 北京邮电大学
[8] 张帅, 曹畅, 唐雄燕. 基于SRv6的算力网络技术体系研究 [J]. 中兴通讯技术, 2022, 28(1): 11–15. DOI: 10.12142/ZTETJ.202201005
[9] 陈佳明, 方遒铿, 罗家尧, 等. 面向云网融合的IP承载网时延选路及保障解决方案 [J]. 邮电设计技术, 2022(4): 43–46. DOI: 10.12045/j. issn. 1007–3043.2022.04.008
[10] IETF. Compute first networking (CFN) scenarios and requirement: draft–geng–rtgwg–cfn–req–00 [S]. 2020
[11] IETF. Framework of compute first networking (CFN): draft–li–rtgwgcfn–framework–00 [S]. 2021
[12] 曹云飞, 霍龙社, 何涛. 基于SRv6的可编排计算优先网络实现方法 [J]. 邮电设计技术, 2022(4): 4–9. DOI: 10.12045/j.issn.1007–3043.2022.04.002

作 者 简 介

魏汝翔，中国联通研究院网络与信息化规划研究中心工程师；主要研究方向为IP网络演进、新一代城域网等。

刘琦，中国联通研究院网络与信息化规划研究中心高级工程师；主要研究方向为网络长期演进及承载网规划，长期从事通信网络规划咨询、研究设计工作。

赵广，中国联通研究院网络与信息化规划研究中心高级工程师；主要研究方向为IP网络演进、新一代城域网、IPTV的CDN等。

曹畅，中国联通研究院未来网络研究部总监、第七届中国通信学会信息通信网络技术委员会委员、中国通信标准化协会“网络5.0技术标准推进委员会”架构组副组长、边缘计算网络基础设施联合工作组（ECNI）技术规范组组长；主要研究方向为IP网络宽带通信、SDN/NFV、新一代网络编排技术等；获中国联通科技进步奖2项；已发表论文30余篇，获授权专利10余项。

唐雄燕，中国联通研究院副院长、首席科学家，“新世纪百千万人才工程”国家级人选，北京邮电大学兼职教授、博士生导师，工业和信息化部通信科技委委员兼传送与接入专家咨询组副组长，北京通信学会副理事长，中国通信学会理事兼信息通信网络技术委员会副主任，中国光学工程学会常务理事兼光通信与信息网络专家委员会主任，国际开放网络基金会ONF董事；拥有20余年电信新技术新业务研发与技术管理经验，主要专业领域为宽带通信、光纤传输、互联网/物联网、SDN/NFV与新一代网络等。

面向算力网络的多路径时敏优先调度机制

A Multipath Time Sensitive Priority Scheduling Mechanism for Computing Power Network

夏华屹/XIA Huayi，权伟/QUAN Wei，张宏科/ZHANG Hongke
（北京交通大学，中国 北京 100044）
(Beijing Jiaotong University, Beijing 100044, China)

DOI：10.12142/ZTETJ.202304005
网络出版地址：http://kns.cnki.net/kcms/detail/34.1228.TN.20230717.1741.002.html
网络出版日期：2023-07-18
收稿日期：2023-05-18

摘要：研究了一种面向算力网络的多路径低时延转发调度算法。该算法可以根据网络状态变化动态更新路径价值量生成路径转发决策，并在一定置信概率内以多路冗余发包的方式进行多路备份传输，降低路径传播时延。还提出了一种等级与队列映射算法，利用网络可编程技术改进设备转发逻辑，利用有限数量严格优先队列保障数据包近似按等级出队，降低数据排队时延。仿真实验结果表明，所提出方法可以降低数据传输时延及抖动，为算力网络业务提供稳定吞吐量。

关键词：算力网络；低时延；多路径；优先队列

Abstract: A multipath low latency forwarding scheduling algorithm for computing power networks is studied, which dynamically updates the path value to generate path forwarding decisions based on network state changes. Within a certain confidence probability, multiple backup transmissions are carried out in the form of multiple redundant contracts to reduce path propagation delay. In addition, a rank and queue mapping algorithm is proposed, which uses network programmable technology to improve the device forwarding logic and uses a limited number of strict priority queues to ensure that packets are approximately queued according to the rank, reducing the data queue delay. Simulation results show that the proposed method can reduce data transmission delay and jitter, and provide stable throughput for computing network services.

Keywords: computing power network; low latency; multipath; priority queue

引用格式：夏华屹，权伟，张宏科．面向算力网络的多路径时敏优先调度机制 [J]. 中兴通讯技术，2023, 29(4): 19-25. DOI: 10.12142/ZTETJ.202304005
Citation: XIA H Y, QUAN W, ZHANG H K. A multipath time sensitive priority scheduling mechanism for computing power network [J]. ZTE technology journal, 2023, 29(4): 19-25. DOI: 10.12142/ZTETJ.202304005

数字经济已成为中国经济发展不可或缺的驱动力，而算力作为数字经济的重要部分，在信息数据处理、智能算法优化等方面起着关键作用[1]。截至2023年3月底，中国累计建成5G基站超过264万个，算力总规模达到每秒180运算次数（EFLOPS）；算力规模快速增长，梯次优化的算力供给体系初步构建，算力规模排名全球第二，年增长率近30%。以OpenAI推出的智能对话模型ChatGPT为代表的人工智能（AI）技术的爆发让全球算力大盘中的智能算力占比提升，第三方数据分析机构IDC在《2021—2022全球计算力指数评估报告》指出：算力指数平均每增加1%，国家数字经济和国内生产总值（GDP）则分别增长3.5‰和1.8‰[2]。据国家发展改革委与工业和信息化部等部门联合实施的“东数西算”工程，中国将重点发展算力全产业链的自主可控建设，形成一体化的新型算力网络体系。该工程对国家政治、经济以及各行业的发展有着重要意义[3]。

为提升数据通信质量，边缘计算中心、高性能数据中心等算力基础设施在生活中的应用逐渐增多。但同时这也逐渐暴露出算力设施在面对多样化服务流量时，计算节点的计算任务分配机制不完善、无法合理使用算力资源等问题。这使得算力设施在使用场景中有了局限性[4]。为解决上述问题，

基金项目：中央高校基本科研业务费专项资金（2021PT202、2022JBGP002）

研发者们提出了算力网络的概念，即一种以算为中心、网为根基，网、云、数、智、安、边、端、链（ABCD-NETS）深度融合的一体化信息服务基础设施[5]。算力网络可以保证用户体验的一致性，使用户可以基本忽略基础设施资源的分布位置与调动状态，为多样化服务流量的分配与调度提供了解决思路[6]。对此，互联网研究工作组（IRTF）设立了网内计算研究组（COIN），研究算力网络新型传输架构；中国通信标准化协会网络与业务能力技术工作委员会（CCSA TC3）已完成《算力网络需求与架构》等研究。算力网络的发展已取得部分进展，但仍面临着诸多技术挑战。其中，算力网络下的低时延调度机制问题亟待解决。

面对算力网络的低时延传输需求，研究者从不同角度进行了方案与机制的研究。文献[7]设计了一种以服务器为中心、网络构造递归的数据中心模式以提供低延迟调度，支持延迟敏感数据的传输，但其部署模式不易拓展，较难根据算力调用需求进行网络规模的迭代。文献[8]设计了一种异构分布式数据中心场景下的工作流调度算法，提出了低负载、低成本与低延迟的多目标优化模型，但暂未考虑多样化服务的算力调用需求，较难满足不同用户的复杂需求。文献[9]提出了一种按需分配的算力资源的联合优化路由控制与资源分配调度模型，旨在降低算力网络的确定性时延，但这种以最短路径为指向的任务调度较难满足多样化服务流中的差异化调度优先级需求。文献[10]提出了一种基于动态三向决策的任务调度算法，为任务分配不同的权重，并结合工作模式和任务期限进行优先级规划，其调度算法能满足单一计算中心的高优先级任务调度需求，但没有考虑结合算力网络进行拓扑级的宏观调度调控。文献[11]根据算力网络不同层次的特性和各种应用的不同需求，提出一种多层次算力网络模型和计算卸载系统以降低确定性时延。该模型考虑了单一模型的任务调度场景，在面对多样化服务时，较难实现差异化调度服务需求。

综上所述，目前业界对算力网络多样化服务的低时延传输机制的研究尚不成熟。为此，本文设计面向算力网络的多路径时敏优先调度机制，通过基于强化学习的多路径转发调度以及优先级队列调度，降低算力网络中低时延需求数据包传输及排队时延，实现算力网络传输时延的相对确定性。

1 系统设计

1.1 机制总体设计

本文设计了面向算力网络的多路径时敏优先调度机制，其系统架构如图1所示。该机制保障算力网络中多样化服务的低时延通信需求：

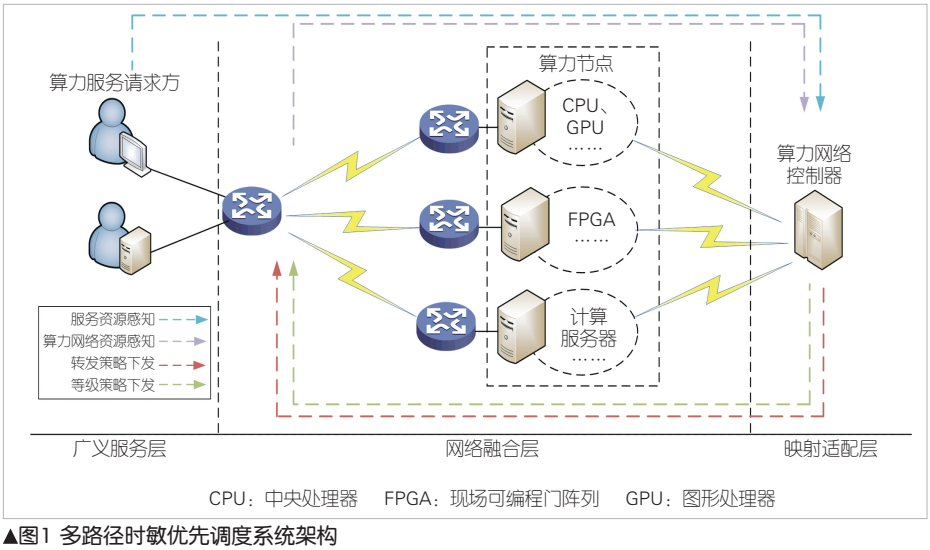
1）多路径转发调度（作用于数据包转发过程）。通过基于强化学习的路径选择算法，该机制对算力网络中各路径的传输价值进行量化，动态更新各路径的价值量，并做出路径选择；对路径时延进行随机变量数学建模，在一定置信概率内利用主从备份传输机制，实现时延性能的提升。

2）优先级队列调度（作用于算力路由设备出端口数据包排队过程）。该机制设计了包等级与优先队列映射算法：当优先队列数量小于包等级范围时，数据包将近似按等级顺序出队，拟合数据包推入先出行为，从而减少算力网络中时延敏感型网络数据包在队列缓冲区的排队时延。

1.2 多路径转发调度

本文所提的多路径转发调度指通过强化学习，探索与学习路径特征并做出相应决策。它是一种通过智能体与环境的交互来获取最优决策的方法。本模块将算力网络控制器抽象为智能体，将其所处各路径的算力网络状态定义为环境，将路径选择算法每次做出的决策定义为回合迭代，并将回合迭代中所选取的传输路径定义为动作。在选择该动作的情况下，本文所设计的多路径转发调度以所探测到的网络信息作为状态，将路径时延定义为奖赏。

由于在算力网络中存在排队、拥塞等问题，我们假设第



▲图1 多路径时敏优先调度系统架构

k路径的时延变量X_k服从对数正态分布，即$\ln(X_k)\sim N(\mu,\delta^2)$。当给定$x_k>0$时，其概率分布函数如公式（1）所示：

$$f(x_k)=\frac{1}{\sqrt{2\pi}x_k\sigma_k}e^{-\frac{(\ln x_k-\mu)^2}{2\delta_k^2}}, \tag{1}$$

其中，对数正态分布的最大似然估计如公式（2）和公式（3）所示。其中，N_k表示选择路径k的次数。

$$\hat{\mu}_k=\frac{\sum_{i=1}^{N_k}\ln(X_{k,i})}{N_k}, \tag{2}$$

$$\hat{\sigma}_k^2=\frac{[\ln(X_i)-\frac{\sum_{i=1}^{N_k}\ln(X_i)}{N_k}]^2}{N_k-1}。 \tag{3}$$

对数正态分布随机变量的均值和方差如公式（4）和公式（5）所示：

$$E(X_k)=e^{\hat{\mu}_k+\frac{\hat{\sigma}_k^2}{2}}, \tag{4}$$

$$Var(X_k)=e^{2\hat{\mu}_k+\hat{\sigma}_k^2}\cdot(e^{\hat{\sigma}_k^2}-1)。 \tag{5}$$

由于对数正态分布，其随机变量分布较为复杂。若将其长尾部分省略，则近似认为第k路径的时延变量X_k服从均值为μ_k、方差为σ_k^2的高斯随机变量，且各路径的时延变量X_k相互独立。X_k的概率密度函数如公式（6）所示：

$$f(x_k)=\frac{1}{\sqrt{2\pi}\,x_k\sigma_k}e^{-\frac{(x_k-\mu_k)^2}{2\delta_k^2}}。 \tag{6}$$

对于对数正态分布其统计意义上的期望及方差无偏估计如公式（7）和公式（8）所示：

$$\overline{X_k}=\frac{1}{N_K}\sum_{i=1}^{N_k}X_{k,i}, \tag{7}$$

$$D[X_k]=\frac{1}{N_k-1}\sum_{i=1}^{N_k}(X_{k,i}-\overline{X_k})^2。 \tag{8}$$

在路径选择算法中，每一轮次的路径选择会根据实时网络数据来更新每一条路径的价值量，并对价值量进行排序，选取最大的路径作为主传输路径。路径选择算法的价值量计算如公式（9）所示：

$$V_k=1/(a\cdot\overline{X_k}+b\cdot D[X_k]-c\sqrt{\frac{2\ln S}{N_k}}), \tag{9}$$

其中，S表示路径选择的总次数；$c\sqrt{(2\ln S)/N_k}$代表第k条路径选择次数与选择总数之间的关系，它可以表征这条路径时延的置信区间。当一条路径探索次数较其他路径较少时，可以认为该路径有较为宽泛的置信区间，即具备较大的探索价值。当一条路径反复被选取时，可以近似认为其置信区间变小，探索价值变低。我们设计的是乐观的选路算法，即在选路时认为置信区间对于选路的决策呈现正向作用。

本文中我们通过基于强化学习的路径选择算法，选取当前最大价值量路径作为主传输路径。然而，考虑路径时延抖动与路径探索等因素的影响，价值量最大的路径可能并不是算力网络中时延敏感型传输服务的最优选择，因此我们设计了一种多路主从传输机制，利用冗余发包选取备份传输路径，牺牲了部分带宽，以换取时延性能的提升。

经以上分析，我们可近似将各路径时延分布X_k作相互独立的高斯正态分布处理，并假定基于最大价值量选取的路径为i，时延分布为X_i，则$X_k-X_i\sim N(\mu_k-\mu_i,\delta_k^2+\delta_i^2),k\neq i$，即两独立路径的时延随机变量相减仍为高斯正态分布，路径k传输时延优于最大价值量路径i的概率可表示为：

$$P\{X_k-X_i\geqslant 0\}=\int_0^{\infty}\frac{1}{\sqrt{2\pi}\,x\sqrt{\delta_k^2+\delta_i^2}}e^{-\frac{[x-(\mu_k-\mu_i)]^2}{2(\delta_k^2+\delta_i^2)}}\mathrm{d}x。 \tag{10}$$

假定$P\{X_k-X_i\geqslant 0\}\geqslant\alpha$，该条路径可以作为备选次优路径。若有多条路径均符合上述概率条件，则选取置信概率最大的路径进行传输，以避免占用过多带宽资源。

多路径低时延转发调度算法的具体流程如算法1所示。

算法1：多路径低时延转发调度算法

输入：路径信息pathInformation

输出：主传输路径号pathBestNumer，备份传输路径号PathBetterNumber

```
for k = 1,2,3…M do
    根据式（7）（8）和（9）更新E[X_k]、D[X_k]和V_k
end for
for k = 1,2,3...M do
    pathBestNumer = Math.max(V_k)
end for
for k = 1,2,3...M do
    if k≠pathBestNumer
        根据公式(10)计算P{X_k − X_i ≥ 0}
        PathBetterNumber = Math.max(P{X_k − X_i ≥ 0})
    end if
end for
```

1.3 优先队列调度模块设计

当路径选择模块选定具体的传输路径后，算力网络中多种网络业务并存且需求带宽大于出口带宽时，时延敏感型算力服务会产生较大的排队时延，难以满足用户需求。因此，本文中我们在可编程交换设备出端口设计了优先级队列调度模块，设计包等级与队列自适应映射算法拟合数据包推入先出行为，减少低时延需求数据包的排队时延。我们基于P4数据平面实现的自适应队列调度机制简称为P4–APQ。

包等级与队列自适应映射算法的示意如图2所示，其作用场景为数据包等级范围大于优先队列数量。该调度算法可利用有限严格优先级队列来拟合数据包的推入先出过程，即数据包近似按等级顺序出队列（本文中，我们约定等级越小，调度优先度越高）。自适应映射算法误差定义为较大等级数据包数量小于较小等级数据包出队的数据包数量，我们将这种误差的现象称为“反转”。由于利用了严格优先级队列拟合推入先出行为即按等级顺序出队，因此当数据包等级范围大于严格优先级队列数量时，有一定概率在调度过程中会出现“反转”现象。对于本文所设计的包等级–队列映射，设计目的为在保证小等级数据包优先调度的前提下，尽可能按等级顺序出队，减少“反转”现象。

针对算力网络时敏型业务低时延需求，包等级与队列自适应映射机制设计了高优先级预留队列，避免最小等级的数据包因优先级“反转”现象而出现较高的排队时延。具体地，首先判定数据平面传入数据包的等级；若等级最小，数据包将直接进入具有最高优先级的预留队列，从而避免等级队列的动态映射带来的“反转”损失。

当数据包等级不是最小时，为保证队列调度结果近似按等级顺序出队，我们设计了自适应映射算法，动态改变各队列的边界值，以最小化拟合损失。自适应映射算法的损失函数可以描述为公式（11）：

$$L(\boldsymbol{P},\boldsymbol{q})=\sum_{p\in \mathrm{P}} cost_{q}(p), \tag{11}$$

其中，L表示为拟合损失，P表示所有入队数据包，$\boldsymbol{q}$表示一组动态变化的队列边界向量，p表示属于P的一个入队数据包。单个数据包的损失可以表示为公式（12）：

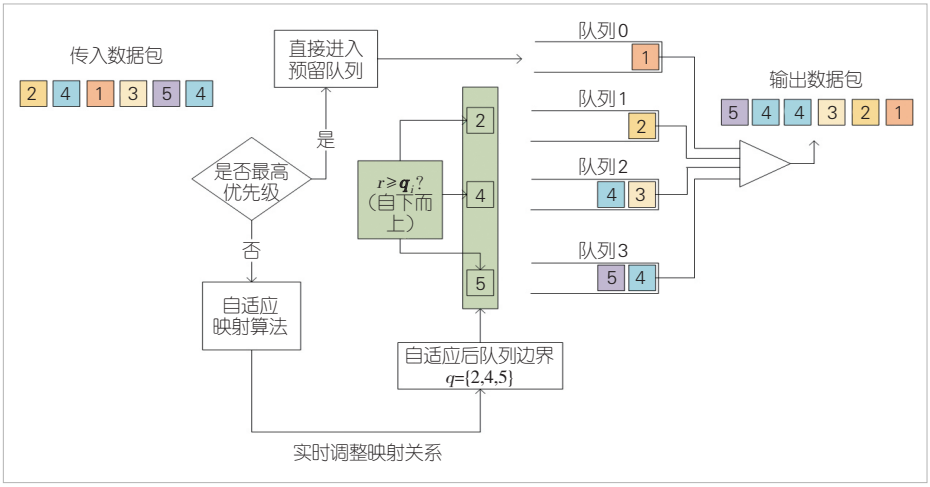
$$cost_{q}(p)=r_{p}(p,\boldsymbol{q})-r(p), \tag{12}$$

其中，$r(p)$表示为给定的数据包p的等级，$r_{p}(p,\boldsymbol{q})$表示为给定数据包所映射的队列自适应调整后的边界值，$cost$表示单个数据包产生的误差，即出现“反转”的情况。包等级与队列自适应映射算法通过动态调整数据包等级与各优先队列的映射关系，在兼顾数据平面的算法复杂度基础上，降低损失函数L。

包等级与队列自适应映射算法可分为两个阶段：

1）“上推”阶段。在该阶段，通过增加进入数据包所映射队列的边界值，减少数据包分组等级与队列边界值的差值，从而减少映射中出现的误差值。具体地，传入的数据包将从较低优先级的队列开始匹配。当数据包等级$r(p)$大于等于队列边界$\boldsymbol{q}_i$时，数据包进入该队列，同时将队列边界$\boldsymbol{q}_i$增加到等于该数据包等级$r(p)$。该过程可以尽量保证数据包实现零误差映射，并防止等级小于队列边界值的数据包映射到当前队列。以上设计仅针对非最高优先级队列。当映射过程匹配到最高优先级队列（不包括预留队列），即使$r(p)<\boldsymbol{q}_1$时，当前数据包会进入最高优先级队列。这将出现“反转”现象，并带来较大的映射损失。这时，我们将根据公式（12）来计算误差损失，并更新最高优先级队列边界值为该数据包等级。

2）“下推”阶段。由于“上推”阶段可能导致最高优先级队列出现“反转”现象，“下推”阶段将减少“上推”阶段的调度损失。当最高优先级队列出现“反转”现象时，自适应算法将根据公式（12）来计算损失成本，再根据公式（13）依次减少除最高优先级队列以外的队列边界。该阶段降低了最高优先级以外的队列边界，减少较高优先级队列中允许进入的数据包等级范围，即减少等级较大的数据包进入高优先级队列的情况，进而减少出现“反转”现象，降低调度损失。



▲图2 包等级与队列自适应映射算法示意图

$$q_i = q_i - cost_q(p)。 \quad (13)$$

本文所设计的等级与优先队列自适应映射算法的具体流程见算法2。

算法2：等级与优先队列自适应映射算法

输入：数据包等级rank

输出：入队编号enqueueNumber：

```
if meta.rank = 0 then
    输出入队编号enqueueNumber = 0
end if
if meta.rank > 0 then
    for q_i : from q_n to q_1 do
        if r ≥ q_i, or i = 1 then
            更新队列边界 q_i = r
            输出入队编号enqueueNumber = i
        end if
        if i = 1 and r < q_1 then
            计算反转损失 cost = q_1 − r
            for q_j from q_n to q_2 do
                更新队列边界 q_j = q_j − cost
            end for
        end if
    end for
end if
```

2 实验验证与性能分析

为测试多路径时敏调度机制相关功能及性能，我们于Mininet环境下搭建包含5台终端主机及5台BMv2交换机的网络拓扑。其中H1、H2、H3、H4作为算力服务请求方与算力路由设备S1相连；S1分别与算力路由设备S2、S3、S4相连作为传输的3条路径；H5作为算力节点与算力路由设备S5相连。各路径设置的实验参数如表1所示。

2.1 多路径转发调度性能测试

本文实现了多路径转发调度中置信参数α为0.05的多路径选择算法，同时实现了传统轮询方式的路径选择算法、ε值为0.1的贪心策略的路径选择算法以及基于置信区间上界（UCB）算法的路径算法以进行相关对比分析，并对每种选路算法进行了500轮次的选路决策。

▼表1 路径具体参数

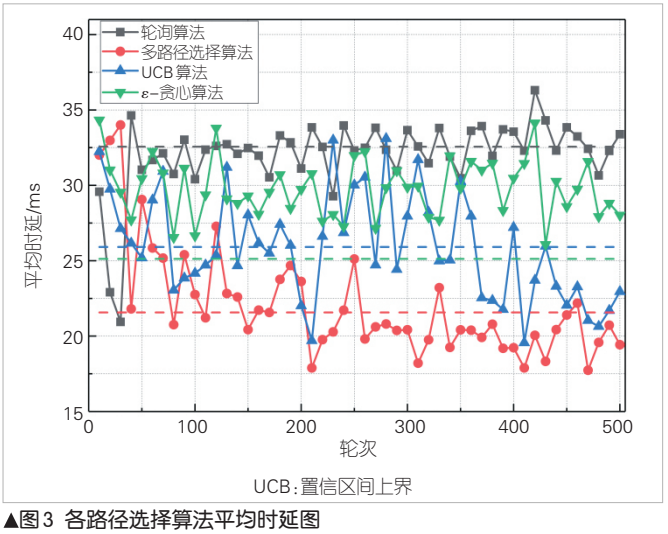
网络链路	链路带宽/（Mbit·s^{-1}）	链路时延/ms	链路抖动时延/ms
S1—S2	50	30	10
S1—S3	50	20	5
S1—S4	50	40	10

为观察各路径选择算法的时延性能，我们将每10轮计为1个记数点来计算平均时延，结果如图3所示。本文所设计的路径选择算法通过前期路径探索后，时延在50轮后有明显的降低，150轮后平均时延为20 ms左右，基本收敛于最优路径。我们将4种算法作定量分析：多路径选择算法的平均时延为21.57 ms，轮询策略的平均时延为32.56 ms，ε–贪心算法的平均时延为25.12 ms，UCB算法的平均时延为25.91 ms。本文所提的路径选择算法在进行500轮次的决策条件后，和轮询策略相比，时延均值降低33.75%；和ε–贪心算法相比，时延均值降低14.13%；和UCB算法相比，时延均值降低16.75%。

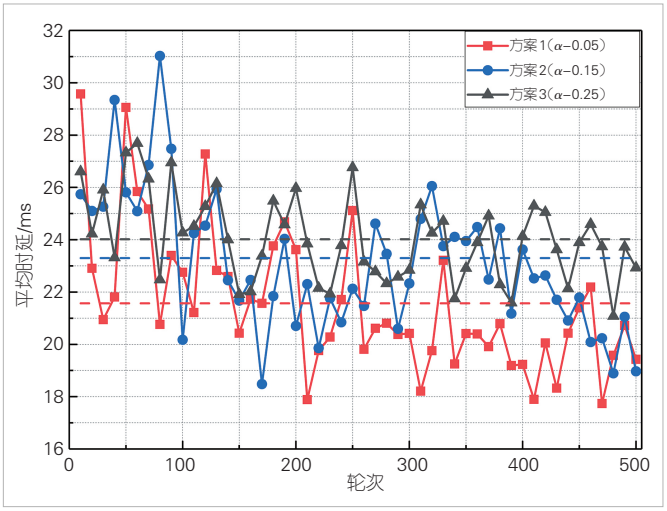
为验证不同置信参数α对多路径选择算法时延性能的影响，我们设置了3组典型参数值，分别为：α = 0.05，α = 0.15，α = 0.25，并进行了500轮次的路径决策，每10轮为1个记数点来计算平均时延，具体结果如图4所示。α = 0.05时，本文所设计的多路径选择算法平均时延为21.57 ms；α = 0.15时，平均时延为23.30 ms；α = 0.25时，平均时延为24.02 ms。实验结果表明，较小置信参数可较多地利用主从备份传输，以提升传输性能。

2.2 优先队列调度性能测试

本节中，我们将测试优先队列调度的各项性能指标，并与先进先出（FIFO）队列方案、基于严格优先级队列拟合推入先出（SP-PIFO）[12]队列调度方案进行对比。在典型场景下，我们测试队列调度算法的具体性能。假设路径选择算



▲图3 各路径选择算法平均时延图



▲图4 不同置信参数下的多路径选择算法平均时延图

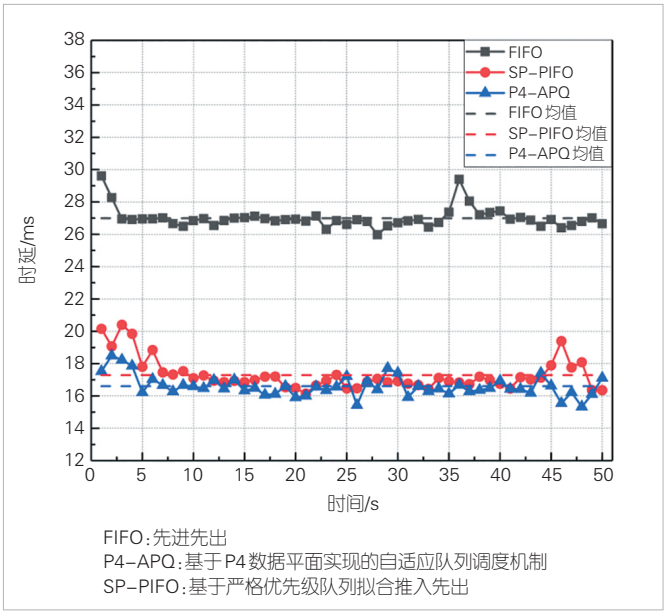
法已将S1的转发出端口路径收敛于S1–S3路径，同时设定S1与S3链路带宽为50 Mbit/s，时延为10 ms，不额外设置链路抖动。

在典型场景下，我们预设H1、H2、H3、H4的数据流等级分别为0、1、2、3，分别代表高优先级、次高优先级、中优先级以及低优先级。H1、H2、H3、H4在同一时刻采用iperf工具来生成用户数据报协议（UDP）流量，再发往终端H5（持续50 s）。其带宽分别设置为10 Mbit/s、10 Mbit/s、15 Mbit/s及20 Mbit/s，总流量大于设置链路带宽。这将造成一定程度的节点拥塞，从而验证队列调度的相关性能。

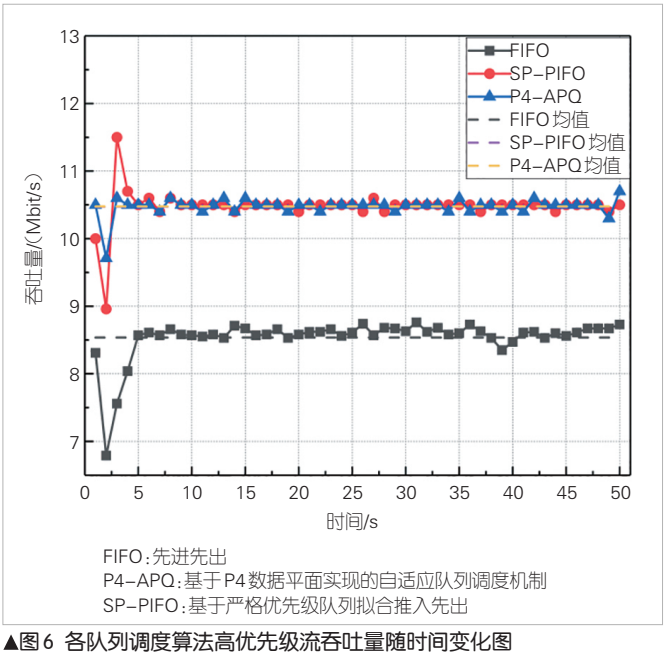
测试中，我们利用3个FIFO队列实现了P4–APQ，同时利用3个FIFO队列实现SP–PIFO。P4–APQ、SP–PIFO以及FIFO的队列数据包容量大小均设置为64。P4–APQ、SP–PIFO以及FIFO队列的高优先级流单向时延随时间的变化如图5所示。

对于高优先级流而言，本文所提的P4–APQ算法基于高优先级预留队列的设计，时延性能略优于SP–PIFO，显著优于FIFO。SP–PIFO由于没有相关预留队列的设计，在最高优先级队列中会出现“反转”的现象，因此也会出现非最高优先级的数据流量，从而造成最高优先级流时延及抖动的增加。对于FIFO队列，由于对各类流量不作区分处理，高优先级流量时延较大。高优先级流在P4–APQ算法的调度下，平均时延为16.30 ms，方差为0.39 ms^2；在SP–PIFO队列调度下中，高优先级数据流的平均时延为17.53 ms，方差为0.65 ms^2；在FIFO队列中，高优先级数据流平均时延为26.99 ms，方差为0.41 ms^2。通过定量计算可知，本文设计的P4–APQ队列调度算法针对高优先级数据流的平均时延较SP–PIFO降低7.01%，较FIFO降低39.6%；抖动较SP–PIFO降低40%，较FIFO降低4.9%。

P4–APQ、SP–PIFO以及FIFO队列的最高优先级流吞吐量随时间变化的情况如图6所示。本文所设计的P4–APQ队列调度算法高优先级流吞吐量均值为10.47 Mbit/s，SP–PIFO队列调度算法高优先级流吞吐量均值为10.41 Mbit/s；FIFO队列在高优先级流吞吐量均值为8.54 Mbit/s。由以上定量分析可以发现，本文所采用的P4–APQ通过给各流量的等级设定，并基于等级与队列映射算法，可以保障高优先级业务流量的稳定吞吐量；高优先级流量吞吐量高于FIFO队列，与



▲图5 队列调度算法高优先级流单向时延随时间变化图



▲图6 各队列调度算法高优先级流吞吐量随时间变化图

SP-PIFO吞吐量近似相等。

根据上述分析，本文所设计的P4-APQ队列调度算法针对较高优先级数据流，能够有效降低排队时延及其抖动，同时提供稳定的吞吐量。

3 结束语

算力网络作为中国率先提出的新型网络架构，是推动信息产业发展、支撑“十四五”发展规划中“网络强国”发展战略的重要基础。针对算力网络的低时延传输需求，本文提出了多路径时敏优先调度机制，设计了基于强化学习的多路径低时延转发调度算法，根据网络实时状态，动态选取低时延传输路径，并在转发出端口设计了等级与队列自适应映射算法，减少低时延应用的排队时延。经过相关实验测试及分析，本文所设计的多路径时敏优先调度机制能够在算力网络场景下提供低时延服务保障。

参考文献

[1] 张宏科, 于成骁, 权伟, 等. 融算网络体系基础研究 [J]. 电子学报, 2022, 50(12): 2928-2934. DOI: 10.12263/DZXB.20221140

[2] IDC, 浪潮信息, 清华全球产业院. 2021—2022全球计算力指数评估报告 [EB/OL]. (2022-03-17) [2023-05-11]. https://www. inspur. com/lcjtww/resource/cms/article/2734773/2734784/2022122613493315670.pdf

[3] 张宏科, 权伟, 刘康. 算力网络研究与探索 [J]. 中兴通讯技术, 2023, 29(1): 1-5. DOI: 10.12142/ZTETJ.202301001

[4] MAO Y Y, YOU C S, ZHANG J, et al. A survey on mobile edge computing: the communication perspective [J]. IEEE communications surveys & tutorials, 2017, 19(4): 2322-2358. DOI: 10.1109/COMST.2017.2745201

[5] 中国移动. 中国移动算力网络白皮书 [EB/OL]. (2021-11-02) [2023-05-15]. http://www. ecconsortium. org/Uploads/file/20211108/1636352251472904. pdf

[6] VELASCO L, SIGNORELLI M, DIOS D O G, et al. End-to-end intent-based networking [J]. IEEE communications magazine, 2021, 59(10): 106-112. DOI: 10.1109/MCOM.101.2100141

[7] NASIRIAN S, FAGHANI F. Crystal: a scalable and fault-tolerant archimedean-based server-centric cloud data center network architecture [J]. Computer communications, 2019, 147: 159-179

[8] CHATTERJEE M, SETUA S K. A multi-objective deadline-constrained task scheduling algorithm with guaranteed performance in load balancing on heterogeneous networks [J]. SN computer science, 2021, 2(5): 1-21. DOI: 10.1007/s42979-021-00609-5

[9] 孙钰坤, 张兴, 雷波. 边缘算力网络中智能算力感知路由分配策略研究 [J]. 无线电通信技术, 2022, 48(1): 60-67. DOI: 10.3969/j. issn. 1003-3114.2022.01.007

[10] LI B, TIAN L Y, CHEN D Q, et al. A task scheduling algorithm for phased-array radar based on dynamic three-way decision [J]. Sensors, 2019, 20(1): 153. DOI: 10.3390/s20010153

[11] 巩宸宇, 舒洪峰, 张昕. 多层次算力网络集中式不可分割任务调度算法 [J]. 中兴通讯技术, 2021, 27(3): 35-41. DOI: 10.12142/ZTETJ.202103008

[12] ALCOZ A G, DIETMULLER A, VANBEVER L. SP-PIFO: approximating push-in first-out behaviors using strict-priority queues [C]//NSDI. 2020: 59-76

作 者 简 介

夏华屹，北京交通大学电子信息工程学院在读硕士研究生；主要研究方向为多路径协同传输、优先队列调度等。

权伟，北京交通大学电子信息工程学院教授、博士生导师，移动专用网络国家工程研究中心新型网络系统研究所副所长，中国通信标准化协会TC1 WG4工作组副组长；主要研究方向为新型网络体系架构、高可靠网络传输关键技术；发表论文80余篇。

张宏科，中国工程院院士，现任北京交通大学电子信息工程学院教授、博导，移动专用网络国家工程研究中心主任，IEEE Fellow，曾任两期国家“973”计划首席科学家，享受国务院政府特殊津贴，是首批全国高校黄大年式教师团队带头人；长期从事专用通信网络理论与工程技术研究，建立了标识网络功能结构及解析映射机制，有效解决了复杂场景下网络高移动支持和高可靠传输难题；获国家技术发明二等奖2项、省部级一等奖4项；出版专著6部。

算力网络资源协同调度探索与应用

Collaborative Scheduling of Computing Power Network Resources: Exploration and Application

彭开来/PENG Kailai[1]，王旭/WANG Xu[2]，
唐琴琴/TANG Qinqin[2]
（1. 网络通信与安全紫金山实验室，中国 南京 211111；
2. 北京邮电大学网络与交换技术全国重点实验室，中国 北京 100876）
(1. Purple Mountain Laboratories, Nanjing 211111, China;
2. State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China)

DOI：10.12142/ZTETJ.202304006
网络出版地址：http://kns.cnki.net/kcms/detail/34.1228.TN.20230724.1549.008.html
网络出版日期：2023-07-24
收稿日期：2023-06-12

摘要：算力网络融合了多区域、多层级的算力资源。针对面向用户差异化需求下全域大范围多层次算力资源的弹性灵活调度问题，设计了一个算力网络资源协同调度平台。针对差异化业务需求，该平台可以形成多公有云与私有云之间的算力资源调度策略以及算网资源协同调度策略，实现资源的自动最优分配。同时，该平台能够自动发现、纳管算力资源，并基于新的资源池情况实现分配策略的自动调整，实现用户无感业务扩缩容，以此弹性调度算力资源。

关键词：算力网络；协同调度；微服务；动态扩缩容

Abstract: The computing power network integrates computing power resources from multiple regions and levels. In response to the flexible scheduling problem of computing power resources across a wide range and at multiple levels under differential user needs, a computing power network resource collaborative scheduling platform is designed. The platform forms a computing power resource scheduling strategy between multiple public and private clouds and a computing network resource collaborative scheduling strategy based on differentiated business needs, to achieve automated and optimal allocation of corresponding resources. At the same time, the platform can automatically discover and manage computing resources, and automatically adjust allocation strategies based on the new resource pool situation, achieving user insensitive business expansion and contraction, and thereby flexibly scheduling computing resources.

Keywords: computing power network; collaborative scheduling; micro services; dynamic scaling

引用格式：彭开来, 王旭, 唐琴琴. 算力网络资源协同调度探索与应用 [J]. 中兴通讯技术, 2023, 29(4): 26-31. DOI:10.12142/ZTETJ.202304006
Citation：PENG K L, WANG X, TANG Q Q. Collaborative scheduling of computing power network resources: exploration and application [J]. ZTE technology journal，2023, 29(4): 26-31. DOI:10.12142/ZTETJ.202304006

5G、物联网等新型网络技术的发展带来了数字经济的高速发展和数字应用场景的爆发式增加，随之而来的是数据的海量增长与算力的高度需求。根据华为发布的《计算2030》预测，2030年人类将进入尧字节（YB）数据时代，通用算力将是现有算力的10倍。在全球数字经济时代大背景下，算力相关技术及产业正成为国家推动经济发展的强大动力。2021年5月，国家发展改革委、中央网信办、工业和信息化部、国家能源局联合印发了《全国一体化大数据中心协同创新体系算力枢纽实施方案》，实施推进“东数西算”工程，进一步推进中国数字经济的发展。这对于抢占数字产业链制高点，推动建设数字强国有着极其重要的战略意义[1]。在此背景下，本文对算力网络的资源协同调度展开研究。

物联网时代下，网络中数据量呈爆发式增长，传统的云计算模式在处理海量的数据和应对实时业务需求方面存在不少弊端。在云计算模式下，所有数据需要上传到云计算中心进行处理，现有的云计算平台的计算能力难以满足日益增长的数据需求。同时，这种计算模式对终端的需求有一定的响应时间，难以满足如无人驾驶等新型业务场景下实时性较强的业务需求[2]。因此，仅靠单一云端算力难以满足所有业务需求。在此背景下，边缘计算应运而生。边缘计算通过引入边缘侧算力，使得网络中的部分数据无须上传至云计算中心，在网络边缘侧就能完成数据的分析与计算。这不仅可以高效快速地反馈需求，还分担了云计算中心的部分数据请求任务[3]。此外，随着无线接入侧能力的大幅提升，大量的终端设备逐渐接入网络中，提供了大量的端侧算力。因此，目

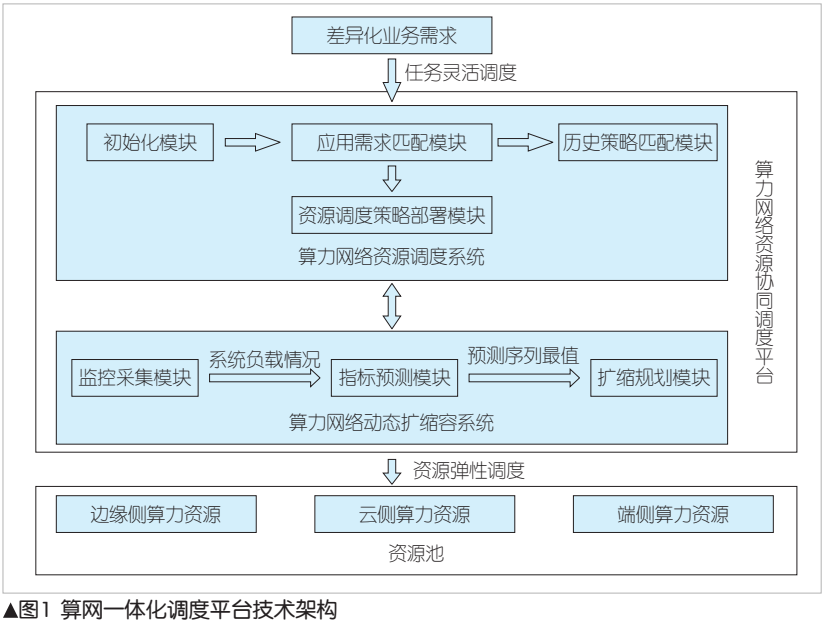
前算力资源不再集中分布在云数据中心，而是广泛分布在大量的边缘节点及海量的终端设备上。算力资源正由集中式云计算转向云边端泛在分布，算力架构逐渐演变为云、边、端三级架构形式[4]。

在这种演进的趋势下，算力资源的协同是关键。这种情况下算力网络应运而生。算力网络能够根据业务需求，在云、网、边、端之间按需调度计算资源、存储资源、网络资源（包括但不限于节点的计算、网络和存储等），旨在实现分布式计算节点的互联互通和统筹调度，通过对网络架构和协议的改进，进而实现网络和计算资源的优化和高效利用[5]。通过无处不在的网络连接，算力网络整合多级算力、存储等，将地理位置较为分散的算力资源连接起来，统筹分配和调度计算任务，针对不同的业务需求，为行业提供最佳的资源分配方案，进而实现整网资源的按需最优分配使用。

1 算力网络资源协同调度平台

算力网络融合了多区域、多层级的算力资源。面向用户差异化需求，如何实现全域大范围多层次算力资源的弹性灵活调度，是算力网络亟需解决的难点[6]。针对这一挑战，本文设计了一个算力网络资源协同调度平台，对算力资源的协同调度开展了一些探索和实践研究。

算力网络资源协同调度平台的总体技术架构如图1所示。该架构分为算力网络资源调度系统和算力网络动态扩缩容系统两个子系统。其中，算力网络资源调度系统由初始化模块、应用需求匹配模块、历史策略匹配模块、资源调度策略部署模块组成，支持算网资源的统一调度：基于各种业务场景的关键需求指标（如网络成本、能耗、带宽等），依据云平台所纳管的云资源和网络资源情况，形成满足应用需求的多公有云与私有云之间的算力资源调度策略和算网资源协同调度策略，实现相应资源的自动化最优分配。算力网络动态扩缩容系统由监控采集模块、指标预测模块、扩缩规划模块组成，实现对资源自动发现及自动纳管，并基于新的资源池情况实现分配策略的自动调整和用户无感业务扩缩容。



▲图1 算网一体化调度平台技术架构

2 算力网络资源调度系统

2.1 算力网络资源调度策略研究

算力网络资源调度系统的核心是资源调度策略。系统根据不同的优化指标提供4种算力资源调度策略：成本感知调度策略、负载感知调度策略、能效感知调度策略以及服务层协议（SLA）感知调度策略。

1）成本感知调度策略。不同云商针对计算资源、网络资源、内存资源等算力资源进行不同的定价。不同的业务场景在计算、网络、内存、存储等方面存在相应的算力需求。根据部署方式以及部署节点的不同，部署成本存在着很大的区别。成本感知调度策略以最优部署成本为优化目标，能够完成网络资源定价，借助算网资源池成本定价模式，设计综合成本评估算网资源调度算法。

2）负载感知调度策略。负载感知调度策略以负载最优为优化目标，通过对算网资源池负载定义和计算方法进行研究，设计综合负载评估算网资源调度算法。

3）能效感知调度策略。能效感知调度策略以实现业务所需能量代价最小为优化目标，通过对各云商算网力资源池能效进行调研，对算网资源调度方案能效计算模型进行研究，设计综合能效评估算网资源调度算法。调度引擎依照能效感知调度策略将部署的容器集调度到物理机上运行，使容器可以与程序所需资源一起配置，在满足资源需求的前提下，尽量降低数据中心能耗。

4）SLA感知调度策略。SLA[7]是指提供服务的企业与客户之间就服务的品质、水准等方面所达成的双方共同认可的协议或契约，是解决Web服务中这些问题的基石。SLA有助于在网络服务之间划分责任和风险，从而使网络服务更加有序。SLA感知调度策略能够对算网资

源调度方案的网络SLA的关键特征指标进行研究，感知并学习相关SLA计算模型，进而设计出综合网络SLA评估算网资源调度算法。

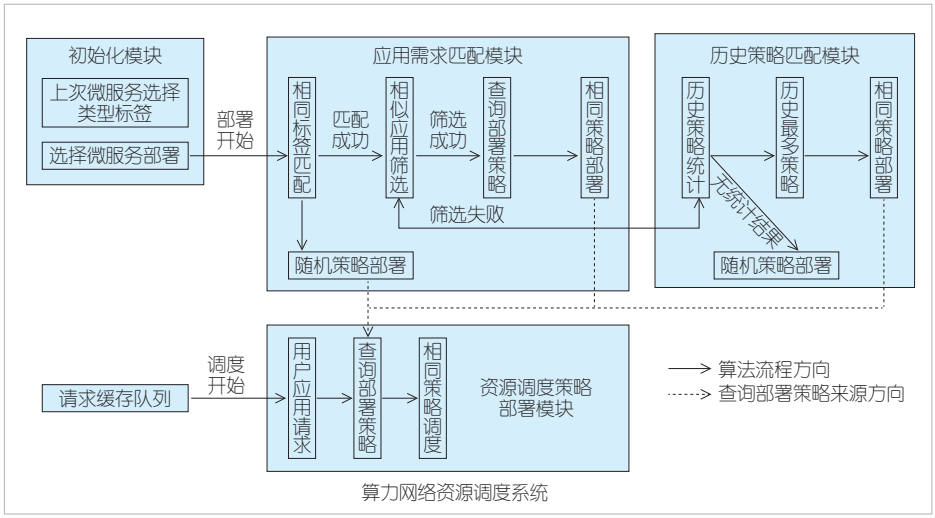
2.2 系统架构

算力网络资源调度系统根据算力和网络的抽象资源指标特征，对本文提出的负载、成本、能效、SLA 4种调度策略进行模块化组合使用，实现多目标优化。为了实现满足大量差异化需求业务的调度策略，多目标优化的算力网络资源调度系统提出了应用需求匹配算法和历史策略匹配算法。系统能够在请求到来时选择最优的资源调度策略，满足不同业务的应用需求，充分利用集群资源。图2为算力网络资源调度系统的架构图。该系统分为微服务部署和用户请求调度两个部分。微服务是小规模、松散耦合的云应用程序，是系统部署和请求调度的独立单元。云计算业务承载于微服务中[8]。

2.3 初始化模块

在用户使用算力网络资源调度系统部署微服务时，调度系统首先将用户选择的应用类型标签与一段时间内已部署过并具有相同标签的微服务进行匹配，之后将筛选与待部署微服务的算网需求相似的一个时间最近的微服务，并查询此微服务在部署时使用的部署策略。资源调度系统将为待部署微服务调用与此微服务相同的部署算法。

在进行初始化部署时，多目标优化资源调度系统主要使用业务应用标签匹配的方法。用户在上传应用时需要选择应用标签。标签包括应用的类型和使用的开发技术。在本系统中，将根据应用类型标签进行相似应用的匹配并决定部署算法调用的具体策略。系统可以按需灵活设置应用标签种类。设置的标签有电商大数据应用、交通大数据应用、AI工地上岗应用等。



▲图2 多目标优化算力资源调度系统架构图

2.4 应用需求匹配模块

初始化完成后，应用需求匹配模块将待部署的应用记为S_0，并将选取的应用类型标签记为L，以检测该应用和其他应用的相似性，实现应用的匹配。具体做法是：对于待部署的应用S_0，选取一段时间内（例如1d）已部署过且具有相同标签L的应用，并按时间由近到远的顺序将其记为集合$S=\{S_1,S_2,\cdots,S_n\}$，若S为空，则直接采取随机策略进行部署；若S非空，则进行资源需求的匹配，以进一步实现部署策略的匹配。

集合S中的应用依次与S_0进行算力资源需求和网络资源需求的相似度分析。该过程仅选取关键算力指标和网络指标进行匹配，而不必使用全部指标。

算力资源需求相似度α的计算公式为：

$$\alpha_1=\frac{S_0\text{的CPU核数需求}}{S_i\text{的CPU核数需求}}, \tag{1}$$

$$\alpha_2=\frac{S_0\text{的内存容量需求}}{S_i\text{的内存容量需求}}, \tag{2}$$

$$\alpha_3=\frac{S_0\text{的存储容量需求}}{S_i\text{的存储容量需求}}, \tag{3}$$

$$\alpha_4=\frac{S_0\text{的显存容量需求}}{S_i\text{的显存容量需求}}, \tag{4}$$

其中，第4项显存容量的比较仅针对AI类业务。

实际上，上述公式中的具体指标可以根据应用类型的实际情况进行设计。比如，如果S_0或S中参与对比的应用没有对其中某项指标提出需求，则不进行该项指标的比较，也不计入总公式。另外，算力需求相似度计算指标可以根据具体需求扩展增加。在公式（1）—（4）中，若任何有效的α_i值均在0.85～1.15内，则判断S_0与S_i算力资源需求相似。

与算力资源需求相似度α类似，网络资源需求相似度β的计算公

式为：

$$\beta_1 = \frac{S_0\text{的带宽需求}}{S_i\text{的带宽需求}}, \tag{5}$$

$$\beta_2 = \frac{S_0\text{的时延需求}}{S_i\text{的时延需求}}, \tag{6}$$

$$\beta_3 = \frac{S_0\text{的抖动需求}}{S_i\text{的抖动需求}}, \tag{7}$$

$$\beta_4 = \frac{S_0\text{的丢包率需求}}{S_i\text{的丢包率需求}}, \tag{8}$$

其中，如果S_0或S中参与对比的应用没有对其中某项指标提出需求，则不进行该项的比较，也不计入总公式。另外，网络相似度计算指标可以根据应用类型的实际情况进行不同的选取和设计。在本公式中，若任何有效的β_i值均在0.85～1.15内，则判断S_0与S_i网络资源需求相似。当S_0与S_i的算力资源需求和网络资源需求都相似时，系统则认定S_0与S_i为相似业务。此时，对S中其他应用/请求的继续比较将停止。

系统在日志/数据库中查询已部署过的S_i部署策略，并统计用户通过手动选择的4种调度策略各自的总次数，之后为待部署微服务调用选择次数最多的策略。对S_0进行部署后，系统会调用相关算法输出相应的部署参数（云、集群、副本数）。如果在标签匹配步骤中没有匹配到与待部署微服务具有相同标签的已部署微服务，则直接随机为待部署微服务调用一种部署策略。

2.5 历史策略匹配模块

若对S中的所有应用进行标签匹配后，均未发现与S_0的相似应用，则使用历史策略匹配模块进行调度。

对于集合$S=\{S_1,S_2,\cdots,S_n\}$中的所有应用，统计用户通过手动选择4种调度策略各自的次数（从日志/数据库中获知相应信息），之后使用选择次数最多的策略对S_0进行部署，并调用相关算法输出相应的部署结果（云、集群、副本数）。如果未能确定调度策略，则随机采取一种感知调度策略进行部署，并输出部署结果。

历史策略匹配模块首先需要输入集群对象列表、候选集群名称列表、微服务对象、微服务标签信息以及历史微服务数据文件地址，然后再进行初始化操作，遍历N个历史微服务，初始化微服务的对象、场景标签矩阵、开发技术标签矩阵以及资源需求参数矩阵。遍历操作会生成N个历史微服务的对象、场景标签向量、开发技术标签向量以及资源需求参数向量，使待部署微服务也有3个向量。为了计算微服务之间的相似度，流程中使用了协同过滤算法。协同过滤算法的核心思想是：通过用户的交互反馈行为，计算用户或者物品之间的相似性。公式（9）可计算对应向量的相似度，公式（10）用于计算两个微服务之间的相似度。若历史微服务综合相似度序列标准差小于1×10^{-4}且相似度均小于0.5，则表面历史微服务均不与待部署微服务相似。这时系统可随机选择成本、SLA、负载、能效感知等资源调度策略，否则选择与待部署微服务综合相似度最高的历史微服务的部署算法。

$$\sin(v1,v2)=\frac{v1\cdot v2}{||v1||\cdot||v2||}, \tag{9}$$

$$SIM=\frac{1}{3}\sum_{1}^{3}S_i。 \tag{10}$$

2.6 资源调度策略部署模块

微服务匹配筛选过程完成后，若找到了相似度较高的微服务，则按照匹配微服务使用的资源调度策略进行部署，否则随机选择一种资源调度策略部署。多目标优化的算力资源调度系统能够组合多种不同的资源调度策略。多种策略算法可以描述为公式（11）：

$$rss_i = Function(server, service, request), \tag{11}$$

其中，rss_i表示第i种资源调度策略，可以用一个函数来表示；*server*表示集群参数信息；*service*表示微服务请求信息，如SLA的请求参数、成本请求参数等；*request*为其他请求参数如输出的集群数量等。

进行第i种资源调度策略下的部署时，系统首先输入微服务信息（包括微服务请求的各项指标参数），输入集群信息（包括集群静态参数和不断变化的实时参数）；然后以第i种策略计算最优的集群如计算成本最优的集群、SLA最优的集群等策略；最后输出部署结果。

在处理请求消息缓存队列中的用户请求时，算力网络资源调度系统使用微服务部署时的算法类型为这些请求进行调度。系统调度算法输出的调度结果有：云、集群、每个集群接收的请求。

资源策略部署模块首先需要输入应用请求信息及微服务历史数据，根据微服务请求信息，构建应用请求模型；然后查找应用请求的微服务部署信息，根据微服务部署时使用的感知算法执行相同种类的应用请求调度算法。得到的输出结果包括集群名称、副本数。

使用的相同种类应用感知的应用请求调度算法过程与部署算法相似。不同的是，其输出为集群调度的概率分布。系统依据概率选择请求调度的集群。公式（12）表示：资源调度到集群1的概率为0.3，调度到集群2的概率为0.3，调度到集群3的概率为0.4。

$$res = \{集群1: 0.3, 集群2: 0.3, 集群3: 0.4\}。 \quad (12)$$

3 算力网络动态扩缩容系统

算力网络动态扩缩容系统是算力网络协同调度中的核心模块之一，对于算网应用的运行以及用户服务的处理起着至关重要的作用[9]。如图3所示，平台设计的动态扩缩容系统由监控采集模块、指标预测模块和扩缩规划模块3部分组成。动态扩缩容系统依托这些组件通过异步工作的方式实现微服务的动态扩缩容过程（增加/减少微服务实例），以实现算力资源的弹性灵活调度。

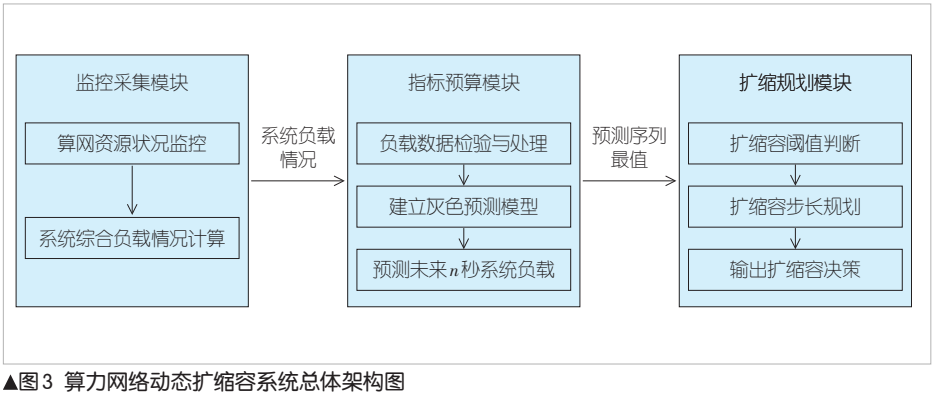
3.1 监控采集模块

算网应用进行动态扩缩容操作时，需要综合考虑内存使用、CPU使用情况、系统的磁盘利用率、系统网络带宽利用率等负载指标。这些负载指标所占的权重（权重的和是1）根据具体情况而定。定义系统的综合负载为L，从算网资源状况中获取CPU的使用情况为C，内存使用情况为M，磁盘使用情况为D，网络带宽使用情况为B，它们的权重分别定义为ω_C、ω_M、ω_D、ω_B，其中$C, M, D, B \in [0,1]$，则系统的综合负载情况为：

$$L = \omega_C C + \omega_M M + \omega_D D + \omega_B B , \quad (13)$$

$$\omega_C + \omega_M + \omega_D + \omega_B = 1 。 \quad (14)$$

监控采集模块负责对整个算力网络的状态进行实时监控。本算法主要考虑两类资源指标：一类是系统资源级别指标，另一类是自定义资源指标。其中，自定义指标是指引入三方监控提供“业务”指标类型。目前，三方监控主要有Prometheus、Microsoft Azure和Datadog Cluster等。指标适配器将指标聚合到Aggregator，由Aggregator向指标预测模块以及扩缩规划模块提供所需指标。



▲图3 算力网络动态扩缩容系统总体架构图

3.2 指标预测模块

指标预测模块一方面抓取上述监控采集模块监控到的微服务副本集的历史资源负载指标数据，并通过这些历史监控数据和预测模型来对下一时刻的微服务副本集的资源负载值进行预测；另一方面将资源负载预测值输出到扩缩规划模块，以指导后续微服务的弹性扩缩容工作。

随着系统的运行，时序性的负载指标也就产生了。这些负载指标由当前运行的CPU、内存、磁盘利用率和带宽占用情况共同组成。基于这些负载指标数据，系统可以预测出未来某段时间的系统负载情况，并根据预测的未来负载来进行资源的弹性伸缩。该问题可以看作一个时序预测问题。预测模块的输入为之前一段时间（扩缩容周期或自定义）综合负载率组成的一组时序数据；预测模块的输出为短期的未来时序数据。

当采用灰色预测模型对算网应用微服务的系统负载进行预测时，指标预测模块得到的结果为一个时序序列，它表示未来一段时间可能的资源使用量。预测序列中的最大值表示未来n秒内微服务可能达到的资源使用量的最大值，作为主要指标传递给扩缩规划模块，从而进行下一步规划。

3.3 扩缩规划模块

在算力网络动态扩缩容系统中，扩缩规划模块拟采用阶梯扩缩容的方式进行扩缩操作。该扩缩规划模块通过指标预测模块对资源负载未来的趋势进行分析，输出系统综合负载率预测序列最大值。如果该值大于扩容阈值，则扩缩规划模块计算副本数×扩容步长，并选取副本数×扩容步长与副本数最大值之间的较大值，执行扩容命令，使资源的副本数变多；如果小于缩容阈值，则计算副本数×缩容步长，选取副本数×缩容步长与副本数最小值之间的较小值，执行缩容命令，使资源的副本数变少。因此，算力网络动态扩缩容系统可以提前做出扩容或缩容操作，以实现算力资源的弹性扩缩，便于算力网

络对算力资源进行灵活调度。

4 结束语

随着算力网络相关技术的发展，全国甚至全世界范围内的云计算资源、边缘计算资源以及终端设备计算资源将组成一张庞大的算力网络。针对不同的业务需求，人们将随时随地享受到超大带宽、超低时延、海量连接、多业务承载的高品质网络服务。

参考文献

[1] 解云鹏, 马思聪, 田毅, 等. 从“东数西算”甘肃节点看中国电信的算力调度探索与实践 [J]. 通信世界, 2022(22): 34–37
[2] 罗军舟, 金嘉晖, 宋爱波, 等. 云计算: 体系架构与关键技术 [J]. 通信学报, 2011, 32(7): 3–21
[3] 施巍松, 孙辉, 曹杰, 等. 边缘计算: 万物互联时代新型计算模型 [J]. 计算机研究与发展, 2017, 54(5): 907–924
[4] 梅雅鑫. 阿里云: 打造三层边缘计算能力 构建云边端协同的开放生态 [J]. 通信世界, 2019(11): 44
[5] 雷波, 刘增义, 王旭亮, 等. 基于云、网、边融合的边缘计算新方案: 算力网络 [J]. 电信科学, 2019, 35(9)44–51. DOI: 10.11959/j.issn.1000–0801.2019209
[6] 李铭轩, 曹畅, 唐雄燕, 等. 面向算力网络的边缘资源调度解决方案研究 [J]. 数据与计算发展前沿, 2020(4): 80–91
[7] PATEL P, RANABAHU A, SHETH A. Service level agreement in cloud computing [EB/OL]. [2023–05–10]. https://corescholar. libraries. wright. edu/cgi/viewcontent.cgi?article=1077&context=knoesis
[8] 辛园园, 钮俊, 谢志军, 等. 微服务体系结构实现框架综述 [J]. 计算机工程与应用, 2018, 54(19): 10–17
[9] 赵树君, 黄倩. 基于 Kubernetes 云原生的弹性伸缩研究 [J]. 计算机与现代化, 2021(11): 28–38. DOI: 10.3969/j.issn.1006–2475.2021.11.006

作 者 简 介

彭开来，网络通信与安全紫金山实验室研究员；主要从事工业互联网、算力网络、边缘计算、时间敏感网络等方面的研究和应用；主持和参与多项工信部工业互联网创新发展工程项目；制定工业互联网标准10余项，申请国家发明专利10余项。

王旭，北京邮电大学在读硕士研究生；主要从事算力网络、边缘计算等相关研究工作。

唐琴琴，北京邮电大学在读博士后；主要从事边缘计算、算力网络、卫星互联网、网络人工智能相关研究工作；参与多个国家重点研发计划、国家自然科学基金等项目；发表论文20余篇，申请国家发明专利10余项。

新增编委介绍

王志勤

毕业于北京邮电大学，现任中国信息通信研究院副院长、中国通信标准化协会副理事长、无线通信技术委员会主席、中国通信学会无线移动委员会主任委员、IMT–2020（5G）推进组组长、IMT–2030（6G）推进组组长；在信息通信技术标准、信息化、产业与政策等方面有深入研究，致力于推动中国3G、4G及5G创新；入选国家百千万人才工程，获全国三八红旗手、全国创新争先奖、全国五一劳动奖章等国家级荣誉，获国家科学技术进步奖特等奖，并多次获得国家科学技术进步一等奖、二等奖。

面向算力网络的云边端协同调度技术

Cloud-Edge-End Collaborative Scheduling Technology for Computing Power Network

周旭/ZHOU Xu，李琢/LI Zhuo
（中国科学院计算机网络信息中心，中国 北京 100190）
(Computer Network Information Center, Chinese Academy of Sciences, Beijing 100190, China)

DOI：10.12142/ZTETJ.202304007
网络出版地址：http://kns.cnki.net/kcms/detail/34.1228.TN.20230717.1804.004.html
网络出版日期：2023-07-18
收稿日期：2023-05-23

摘要：针对网络规模庞大、资源分散和异构性等挑战，面向算力网络的云边端协同调度技术通过算力网络将云边端多级泛在算网资源整合在一起，从而形成一个庞大的、跨越多个地域的资源池。综合考虑网络实时状态、用户需求等要素，该技术能够实现对算网资源的统一管理和动态调整，提升用户体验的同时降低企业运营成本和运维复杂度。

关键词：算力网络；云边端协同；协同调度技术

Abstract: Faced with challenges such as a large scale of networks, dispersed and heterogeneous resources, the cloud-edge-end collaborative scheduling technology for computing power networks integrates multi-level ubiquitous computing network resources from the cloud, edge, and end into a large, cross-regional resource pool through the computing power network. By considering factors such as real-time network status and user demands, the technology achieves unified management and dynamic adjustment of computing network resources, improving user experience while reducing enterprise operation costs and maintenance complexity.

Keywords: computing power network; cloud-edge-end collaboration; collaborative scheduling technology

引用格式：周旭, 李琢. 面向算力网络的云边端协同调度技术 [J]. 中兴通讯技术, 2023, 29(4): 32-37. DOI: 10.12142/ZTETJ.202304007
Citation：ZHOU X, LI Z. Cloud-edge-end collaborative scheduling technology for computing power network [J]. ZTE technology journal, 2023, 29(4): 32-37. DOI: 10.12142/ZTETJ.202304007

数字化时代带来海量数据的增长。这种趋势推动了大量计算资源的需求增长，诸如数据中心、边缘计算节点和各种终端设备等构成了一个庞大且复杂的分布式计算资源网络。在这个网络中，有效地管理和调度各个节点的计算资源，以提供高效、稳定的服务，是一大挑战。传统的调度技术往往针对特定的场景和需求，无法满足多样化、动态变化的计算需求。此外，由于云端和边缘端的资源异构性、地理分布的广泛性以及网络环境复杂性等，资源的管理和调度更加困难。

为了解决这一问题，云边端协同调度技术应运而生。通过整合云、边和端的资源，实现对资源的统一管理和动态调度，以最大化资源的利用率，保障用户的服务质量。在算力网络的背景下，云边端协同调度技术的研究尤为必要[1]。算力网络将分布在各地的云端、边缘端和终端设备的计算资源通过网络连接起来，形成一个分布式的、动态的计算资源池。这个资源池可以实现对所有计算资源的统一管理和调度，从而提高资源的利用率，降低运营成本，提高服务质量[2]。在算力网络中使用云边端协同调度技术，可以充分利用网络的优势，有效满足复杂的网络环境和多样化的计算需求。因此，本文将重点阐述面向算力网络的云边端协同调度技术，探讨其在实际应用中的效果和关键技术[3-4]。

1 分布式的云边端算力

大数据和物联网等场景需要使用云计算和边缘计算技术，这样可以在全网广泛分布的设备上提供计算服务，将计算能力广泛分布到需要的地方，从而提高数据处理的效率，减少延迟，并提供更个性化、地域化的服务。

基金项目：国家重点研发计划（2018YFB1800100）；国家自然科学基金（U1909204）

1.1 云边端算力概述

为了提供更加高效、灵活和可靠的计算服务，我们提出集云计算与边缘计算于一体的计算模式——云边端算力模式。该模式综合了云计算中心的大规模存储和处理能力，以及边缘计算对于接近数据源的处理能力。在该模式中，云端负责处理大规模、复杂的计算任务，而边缘端则负责处理那些需要低延迟、快速响应的任务。通过这种方式，云边端算力能够在满足不同任务需求的同时，提高整体的计算效率。

在云边端算力模式中，数据首先会在传感器、移动设备等地方产生，再被边缘设备接收并进行初步处理。这些处理通常包括数据清理、预处理和部分分析等。随后，数据和任务会根据性质和需求，分配给云端或边端进行进一步处理。具体来讲，那些需要快速反馈的任务，通常会被留在边缘端；而那些需要大规模数据分析和深度处理的任务，则会被发送至云端。通过这种方式，云边端算力能够提供更加高效、灵活的计算服务。

1.2 分布式算力的优势和挑战

在分布式的云边端算力架构中，全网范围内的计算资源配置使得各种类型的计算任务处理更加灵活和高效。设备的资源可以根据任务需求进行动态调整，从而在维持高效运算的同时，提供稳定且高质量的服务。此外，这种配置方式还能应对各种突发的计算需求，快速进行资源调配，以满足不断变化的计算需求。

然而，分布式的云边端算力也面临着诸多挑战。在全网范围内进行高效的资源调度，需要考虑到设备状态、任务需求、网络状况等多个因素，这使得调度过程变得复杂。一方面，计算需求和设备状态可能会快速变化，因此如何在短时间内做出决策，保证服务的质量和效率，是一大挑战；另一方面，由于设备是分布式的，那么如何进行有效的分布式协调、收集设备状态信息、在设备间同步任务状态等问题也需要被解决。

1.3 分布式算力的应用和未来趋势

随着5G和6G网络技术的发展，更高的网络速度和更低的延迟推动了分布式云边端算力在更多领域中的应用，如无人驾驶、远程医疗、工业自动化等。同时，随着任务复杂性的增加，例如高精度的物理模拟、大规模的社会科学模型等，分布式计算资源的需求也将增多。此外，在机器学习和人工智能领域，分布式的云边端算力发挥着重要作用。通过利用分布式算力资源，机器学习模型能在更短的时间内进行训练和优化。

2 融合云边端的协同网络

2.1 融合云边端的网络架构

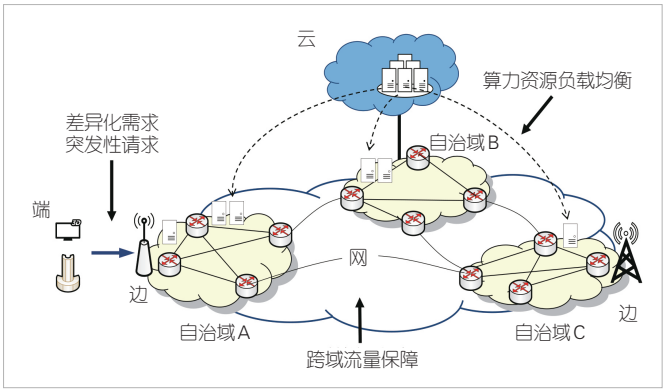
新型应用如移动互联网、物联网的普及以及新型网络技术的发展，使得传统的集中式云计算模式遇到了挑战[5]。为了应对这些挑战，我们提出融合云边端的网络架构，具体如图1所示。该架构发挥了云计算强大的计算能力和边缘计算的低延迟优势，旨在提供更高效、更灵活的计算服务，以满足不断增长的计算需求和多样化的业务需求。融合云边端的网络架构将云端的大规模数据处理能力与边缘端的实时处理能力相结合，优化了网络资源的利用，提高了网络服务的质量。

在这种融合云边端的网络结构下，资源可以被更加高效地利用，网络的可靠性和安全性得到了提高。在该网络中，云计算作为物联网的中枢，通过将大量终端或边缘无法处理的数据进行存储、整理和分析，为云边端协同网络提供强大的计算能力支持。同时，部分任务可直接在终端设备或边缘服务器中进行处理，在减少数据传输延时的同时也缓解了云数据中心的压力。

2.2 协同网络的优化与调度策略

在融合云边端的网络架构中，协同工作非常重要。这需要云计算和边缘计算紧密协作，共同完成计算任务[6]。为了实现这一目标，需要设计并实施有效的协同网络优化与调度策略。优化与调度策略的主要目标都是在满足任务需求的同时，最大程度地利用云边端的计算资源。协同网络的优化工作主要研究的是如何合理地分配任务和资源，如何实现高效的数据传输，以及如何维护网络的稳定性和可靠性等问题。而在调度策略方面，考虑的因素更为复杂，包括但不限于计算任务的性质、网络状况、设备能力、能耗和特定的应用需求等。

协同调度策略则需要将计算任务按照某种方式分配给云



▲图1 融合云边端的网络架构示意图

端或边缘设备，同时还需要考虑任务的执行顺序和资源的分配情况。在这个过程中，要尽可能地减少任务的执行时间，降低网络的传输延时，从而提升系统的整体性能。综上所述，优化与调度策略是实现云边端协同工作的关键，通过这些策略，可以更好地管理和调度网络中的资源，从而提升云边端协同网络的性能，满足多变和复杂的应用需求。

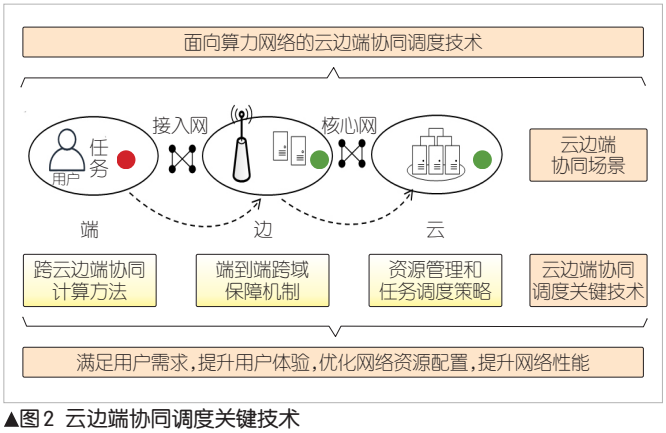
3 云边端协同调度关键技术

分布式的云边端算力提供了广阔的计算平台。融合云边端的协同网络构建了高效、灵活的计算资源调度模型[7]。然而，为了挖掘这一模型的最大潜力，还需要更多的支撑技术，如云边端协同调度关键技术等（具体如图2所示）。协同网络的优化与调度策略可以最大化地利用和分配云边端的计算资源，而这一过程需要精细化的管理和调度。精细化管理和调度的实现需要云边端协同调度关键技术做支撑，具体包括跨云边端协同计算方法、端到端跨域保障机制和资源管理和任务调度策略等。在这些技术的支持下，云边端协同网络能够在满足各种复杂、变化需求的同时，解决协同网络的优化与调度策略的问题，优化网络资源配置，进一步提升网络性能[8]。

3.1 跨云边端协同计算方法

在多终端、多任务的复杂场景下，卸载决策的制定需要综合任务计算量、数据传输量、云边端各节点计算能力和资源利用率等诸多因素。基于已有的云计算或边缘计算设计的协同计算方法不完全适用于云边端协同场景。另外，云边端设备的异构性强，这些方法也不适用于多种设备的需求。计算方法的不统一降低了处理效率，无法充分发挥协同的优势。因此，设计跨云边端的协同计算方法有很大的必要性。

为了应对云边端设备的异构性，跨云边端协同计算的方法需要考虑不同的设备需求。一种方法是根据设备的计算能力和资源利用率来分配任务，将计算密集型任务分配给计算能力较强的云端节点，而将数据密集型任务分配给边缘节点，以充分利用各节点的资源；另一种方法是利用任务切分和协同执行，将一个任务切分成多个子任务，并将这些子任务分配给不同的设备，最后将它们的结果合并得到最终结果。



▲图2 云边端协同调度关键技术

3.2 端到端跨域保障机制

在云边端协同调度中，端到端跨域保障机制主要是指延迟优化和服务质量保证。二者紧密结合，共同支撑云边端协同调度的高效运行。有效的延迟优化，可以提升系统性能；有效的服务质量保证，可以确保系统始终处于高效稳定的状态，从而进一步降低延迟。

延迟优化是指减少任务执行和数据传输的时间延迟，尤其是在需要实时反馈或高速处理的应用中。延迟优化涉及的技术包括：1）有效的任务调度，确保任务在最佳位置执行，减少数据传输和处理时间，优化网络路由和传输协议，减少网络传输延迟；2）利用边缘计算的特性，将计算任务靠近数据源，减少数据传输的时间和距离。服务质量保证则是确保系统能提供用户所需的服务，这包括满足如执行速度、响应时间、数据准确性等的各种性能要求，以及满足特定的服务等级协议。服务质量保证涉及的技术包括资源和任务调度，以及各种容错和恢复机制，以保证系统有足够的资源来满足服务需求，应对可能的错误和故障。

3.3 资源管理和任务调度策略

在云边端协同网络中，资源管理和任务调度是两项关键技术。这两项技术的主要目标是优化系统性能，提升服务质量，从而实现协同网络的优化与调度策略[9]。资源管理的核心是实现资源的高效利用，这包括了对云服务器、边缘设备和网络带宽等资源的合理分配和调度。具体来说，资源管理需要考虑系统的总体需求，以及各类资源的性能和状态，从而决定如何分配和调度这些资源。通过有效的资源管理，我们可以使系统在满足各种需求的同时，最大化地利用资源，提升系统性能。

任务调度则主要关注如何合理地分配和调度计算任务。任务调度需要考虑任务的特性，如任务的类型、大小、优先级，以及任务的执行环境等。基于这些信息，任务调度制定合适的策略，决定如何将任务分配给云服务器或边缘设备，以及如何安排任务的执行顺序[10]。

4 面向算力网络的典型应用场景

云边端协同调度技术已广泛应用于物联网与智慧城市、自动驾驶与无人机、远程医疗与虚拟现实等领域。本章节中，基于算力网络的典型应用场景，我们详细阐述了协同调度技术是如何在实际应用中发挥关键作用，优化性能并提升用户体验的。

4.1 物联网和智慧城市

在物联网环境中，大量的设备和传感器被用于收集各种类型的数据，如温度、湿度、位置等。然而，这些设备的计算能力和储存空间都非常有限。因此，分布式的云边端算力可以提供必要的计算资源，支持在边缘设备上进行数据预处理和实时分析。这大大减少了数据传输的延迟和网络带宽的需求。

智慧城市是物联网在更大规模下的应用。在智慧城市中，各种设备和系统都可以通过互联网进行连接，形成一个大型的、互相协作的网络。面向算力网络的云边端协同调度技术可以实现各种复杂的功能，如实时交通管理、能源优化、公共安全管理等。从隐私保护的角度来讲，通过在边缘设备上进行数据处理，可以确保数据的隐私和安全，同时提高服务的响应速度。

4.2 自动驾驶和无人机

对于自动驾驶来说，安全和实时性至关重要。车辆必须能够快速且准确地响应周围环境的变化，例如其他车辆的动态、行人甚至是天气状况。由于云端处理可能会引入无法接受的延迟，因此在车辆本地进行数据处理非常必要。然而，车载计算资源有限，无法处理大量的输入数据和复杂的算法。因此，云边端协同调度技术成为了一个理想的解决方案，它能够在保证实时性的同时，通过边缘计算节点的协同工作，提高数据处理能力。

无人机也有类似的需求。无人机通常需要实时的视频流处理，以进行物体检测、追踪等。这需要大量的计算资源，但无人机的载荷有限，难以满足这种需求。云边端协同调度技术可以将计算任务分配到无人机附近的边缘计算节点，从而实现实时的视频流处理。无论是自动驾驶还是无人机，云边端协同调度技术都能有效地解决有限的设备计算能力和严格的实时性要求之间的矛盾。

4.3 远程医疗和虚拟现实

远程医疗技术使医生可以在任何地方都能为患者提供服务。然而，这种类型的服务对网络的稳定性和时延有非常高的要求。面向算力网络的云边端协同调度技术可以确保数据在云端和边缘设备之间快速、准确传输，尽可能降低延迟，满足远程医疗应用的需求。虚拟现实是一个对延迟和数据处理能力要求非常高的领域，任何微小的延迟都可能导致用户体验下降，甚至引发眩晕感。同时，虚拟现实应用通常需要处理大量的图形数据和用户交互信息，这超出了大多数个人设备的计算能力。因此，利用云边端协同调度技术，可以将部分计算任务卸载到边缘设备，如附近的边缘服务器，从而降低延迟，提高数据处理能力，实现更好的体验。

5 基于云边端协同的流量调度

在云边端协同网络中，传统的网络流量调度往往采用集中式的调度算法，由中心节点对整个网络的流量进行统一管理和调度。中心节点的调度算法效率低下，对网络性能也有很大的影响。为了能够根据实时的网络状态选择流量最佳转发路径，针对云边端分布式网络中的流量调度，我们提出基于云边端协同的流量调度，由云边端各节点共同协作完成网络流量的调度，避免单一链路出现拥塞而其他链路可能有剩余带宽未被充分利用的情况。同时，通过将不同类别的流量调度到满足传输性能的链路上，降低多流并发情况下的端到端时延，提升平均吞吐率。

5.1 云边端协同流量调度模型

流量调度中最重要的指标是流完成时间和吞吐率。当网络中的流量达到一定的程度时，网络中的拥塞会增加，从而导致流的完成时间增加，吞吐率下降。因此，流完成时间和吞吐率之间存在着一种权衡关系。如果仅仅关注流的完成时间，可能会导致网络的拥塞程度加剧，从而使得网络的吞吐率下降。相反，如果仅仅关注网络的吞吐率，可能会导致一些流的完成时间变得非常长，从而影响服务质量。因此，本文定义了包含以下内容的云边端协同流量调度模型。

1）对于每个流，需要在规定的时间内完成传输，通过设置流量保障的重要程度以保证服务质量和用户体验。

2）在满足流完成时间的基础上，需要保持一定的吞吐率水平，以充分利用云边端分布式网络中的带宽资源。

将端到端流量保障模型中的优化目标设置为同时最小化流平均完成时间和最大化吞吐率，以实现流量保障和网络性能的平衡，相应的优化问题可定义如下：

$$\min \bar{\tau} + \lambda \times \frac{1}{\eta} \times \frac{1}{n} \times \sum_{i=1}^{n} \omega_i \times \tau_i$$

$$\text{s.t. } \frac{1}{n} \sum_{i=1}^{n} \tau_i = \bar{\tau}, \tag{1}$$

其中，$1/\eta$是所有流的总吞吐率的倒数，ω_i是流i的权重，表示该流对端到端流量保障的重要程度，τ_i是流i的流完成时间，λ是用来平衡完成时间和吞吐率之间的关系的系数。该优化问题的含义是：将流的平均完成时间最小化，同时通过对所有流的完成时间进行加权平均，实现网络的总体吞吐率最大化。对流进行加权的目的是为了保障重要流的服务质量，满足端到端流量保障的需求，同时适当牺牲不重要的流，以提高整体吞吐率。另外，还可以通过调整λ的值来控制吞吐率和流完成时间之间的权衡关系。

5.2 分布式强化学习算法

为求解上述协同流量调度优化问题，本小节在软策略演员–评论员（SAC）算法的基础上，设计基于SAC的分布式强化学习（DSAC）算法，具体如下：

算法1：基于SAC的分布式强化学习算法

1. **for** *iteration* = 1,2,⋯,N do
2. **for** each worker i do
3. Sample a batch of traffic flows f_j from D
4. **for** each flow f_j
5. Compute the actions a_j,k using policy $\pi(\cdot|f_j;\theta_i)$
6. Compute the scheduling policies g_k using the policy library G
7. Compute the flow rate allocation by the scheduling policies
8. Compute the delay and reward of each flow
9. **end for**
10. Add the (f_j,a_j,k,r_j,k,d_j) tuples to D
11. Compute target values
12. Update Q–function by minimizing the Bellman error
13. Update policy by maximizing the expected Q–value
14. Compute the action a using the current policy
15. Execute the action a and observe the next state s' and reward r
16. Store the transition (s,a,r,s',d) in the worker replay buffer D
17. Send the updated policy parameters θ_i and Q–function parameters ϕ_i to the parameter server
18. **end for**
19. Compute the new state representation s for each flow in the network
20. Update the network flow table with the new action a for each flow
21. **end for**

具体来讲，DSAC算法是SAC算法在分布式系统中的拓展。该算法可以将策略优化和Q值函数优化分配到多个智能体上来提高算法的效率。通过将强化学习算法中的演员–评论员架构与软策略优化相结合，实现高效的流量调度。在每个时刻，演员通过观察当前网络状态和历史流量数据来选择一个最优的流量调度决策。评论员则根据演员的决策和真实流量数据来评估演员的决策，并将其反馈给演员进行策略优化。算法可以在分布式环境中运行着多个演员–评论员框架同时协作以进行流量调度，通过软策略优化来避免对策略进行硬约束，从而使得云边端流量调度决策更加灵活。

5.3 性能仿真验证

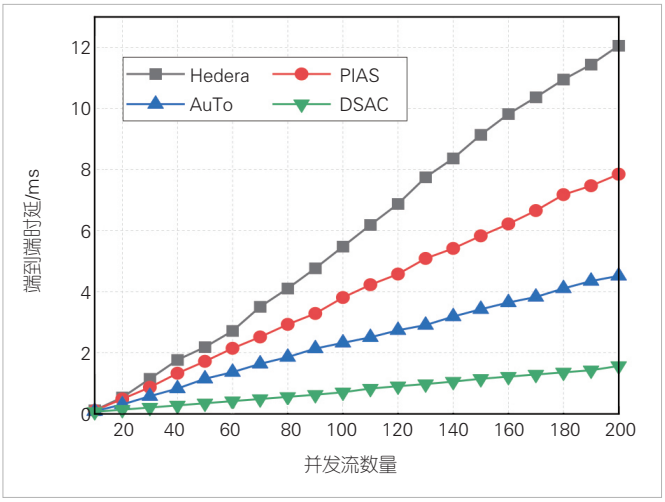
为了评估所提协同流量调度方案的有效性，我们将所提的基于分布式深度强化学习的协同流量调度方案与以下几种典型的流量调度方案进行性能比较。

1）AuTo：一种使用深度强化学习解决流量调度问题的方法。该方法根据网络中的流量负载自动地调整资源的使用，以实现更好的网络流量管理，并在保证服务质量的同时提高系统的效率[11]。

2）PIAS：一种信息不可知的流量调度算法。该算法能够动态地调度数据中心网络中的流量，以确保高效的网络运行。该算法基于实用性和信息不可知性的设计原则，通过计算流的权重和调度流来实现最佳性能和高网络利用率[12]。

3）Hedera：一种数据中心网络流调度方法。该方法使用了一种基于高负载优先的动态调度策略，即优先处理那些负载更高的网络流量，以避免网络拥塞和延迟，实现更高的网络吞吐量和更低的延迟[13]。

为了验证所提云边端协同调度方案的有效性，我们分析了不同流量调度方案在多流并发下的流量调度效果。在端到端时延方面，如图3所示，随着并发流的增加，Hedera和PIAS的端到端时延快速增加。基于深度强化学习的AuTo和



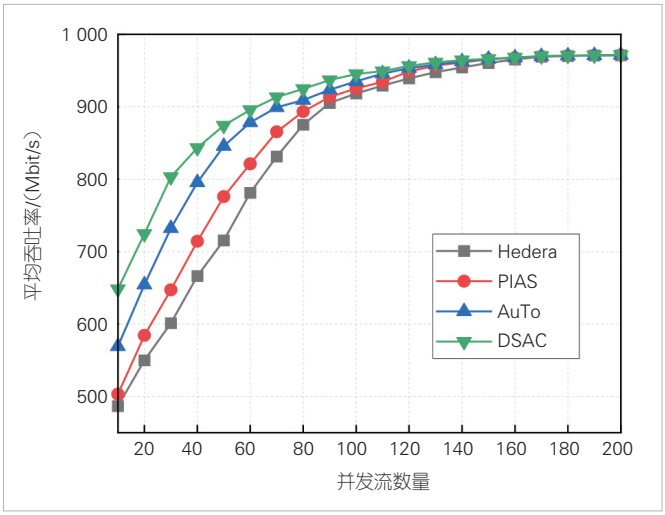
▲图3 多流并发下的端到端时延

本文所提的DSAC都能够提供较低的端到端时延。但DSAC比AuTo的时延更低。这是由于DSAC将强化学习算法中的演员–评论员架构与软策略优化算法相结合。这样一来，演员通过观察当前网络状态和历史流量数据来选择一个最优的流量调度决策，评论员则根据演员的决策和真实流量数据来评估演员的决策，再反馈给演员进行策略优化。基于DSAC算法的协同流量调度有效降低了端到端时延。

如图4所示，在平均吞吐率方面，随着并发流量数目的增多，各个算法的平均吞吐率都在增加。当并发流数量小于80时，相比于Hedera和PIAS，AuTo和DSAC获得的平均吞吐率有着明显的优势。这是由于AuTo和DRL为每个子网络分配一个深度强化学习智能体。这些智能体可以通过学习从其他子网络到自身的流量路由，实现全局流量优化，从而能够在减少端到端时延的同时提高平均吞吐率。这进一步表明了在多流并发情况下，算法的端到端时延和平均吞吐率都有着更好的表现。

6 结束语

本文首先介绍了分布式云边端算力的未来发展趋势，探讨了融合云边端的协同网络架构，分析了云边端协同调度的关键技术，明确了其在解决实时性和高性能计算问题上的重要作用；其次，结合具体的应用场景如物联网、自动驾驶、远程医疗和虚拟现实等，展示了云边端协同调度技术的实际影响和潜力；最后，以云边端协同网络中的流量调度为例，从多流并发情况下的端到端时延和平均吞吐率需求出发，提出了基于分布式强化学习的协同流量调度算法，并通过实验验证了所提协同流量调度方案的有效性。本研究推动了云边端协同调度技术在分布式的云边端算力场景中的应用。



▲图4 多流并发下的平均吞吐率

参考文献

[1] 雷波, 刘增义, 王旭亮, 等. 基于云、网、边融合的边缘计算新方案: 算力网络 [J]. 电信科学, 2019, 35(9): 44–51. DOI: 10.11959/j.issn.1000–0801.2019209

[2] 张宏科, 权伟, 刘康. 算力网络研究与探索 [J]. 中兴通讯技术, 2023, 29(1): 1–5. DOI: 10.12142/ZTETJ.202301001

[3] 段晓东, 姚惠娟, 付月霞, 等. 面向算网一体化演进的算力网络技术 [J]. 电信科学, 2021, 37(10): 76–85. DOI: 10.11959/j.issn.1000–0801.2021248

[4] ASGHAR H, JUNG E S. A survey on scheduling techniques in the edge cloud: issues, challenges and future directions [EB/OL]. [2023–05–22]. https://arxiv.org/abs/2202.07799

[5] KAI C H, ZHOU H, YI Y B, et al. Collaborative cloud–edge–end task offloading in mobile–edge computing networks with limited communication capability [J]. IEEE transactions on cognitive communications and networking, 2021, 7(2): 624–634. DOI: 10.1109/TCCN.2020.3018159

[6] YANG H, LIANG Y S, YUAN J Q, et al. Distributed blockchain–based trusted multidomain collaboration for mobile edge computing in 5G and beyond [J]. IEEE transactions on industrial informatics, 2020, 16(11): 7094–7104. DOI: 10.1109/TII.2020.2964563

[7] 金天骄, 栗蔚. 基于算力网络的大数据计算资源智能调度分配方法 [J]. 数据与计算发展前沿, 2022(6): 29–37

[8] 衷璐洁, 王目. 区块链赋能的算力网络协同资源调度方法 [J]. 计算机研究与发展, 2023, 60(4): 750–762. DOI: 10.7544/issn1000–1239.202330002

[9] 李铭轩, 曹畅, 唐雄燕, 等. 面向算力网络的边缘资源调度解决方案研究 [J]. 数据与计算发展前沿, 2020, 2(4): 80–91. DOI: 10.11871/jfdc.issn.2096–742X.2020.04.007

[10] 刘泽宁, 李凯, 吴连涛, 等. 多层次算力网络中代价感知任务调度算法 [J]. 计算机研究与发展, 2020, 57(9): 1810–1822. DOI: 10.7544/issn1000–1239.2020.20200198

[11] CHEN L, LINGYS J, CHEN K, et al. AuTO: scaling deep reinforcement learning for datacenter–scale automatic traffic optimization [C]// Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication. ACM, 2018: 191–205. DOI: 10.1145/3230543.3230551

[12] BAI W, CHEN L, CHEN K, et al. PIAS: practical information–agnostic flow scheduling for commodity data centers [C]//Proceedings of IEEE/ACM Transactions on Networking. IEEE, 2017: 1954–1967. DOI: 10.1109/TNET.2017.2669216

[13] AL–FARES M, RADHAKRISHNAN S, RAGHAVAN B, et al. Hedera: dynamic flow scheduling for data center networks [J]. NSDI, 2010, 10(8): 89–92

作　者　简　介

周旭，中国科学院计算机网络信息中心先进网络技术与应用发展部主任、研究员；主要研究领域为未来网络、5G移动网络、网络人工智能等；主持科技部、工业和信息化部、国家发展和改革委员会等重大项目10余项，获得部级科学技术奖一等奖1项、二等奖1项；发表论文70余篇，申请专利60余项（其中已授权14项），主持/参与制定国际标准/行业标准11项。

李琢，中国科学院大学在读博士研究生；主要研究方向为未来网络、分布式协同网络、多智能体系统和强化学习。

一种面向服务的算网路由架构方案

An Architecture Solution of Service-Oriented Routing for Computing and Networking

黄光平/HUANG Guangping[1,2]，谭斌/TAN Bin[1,2]，吉晓威/JI Xiaowei[1,2]
（1. 中兴通讯股份有限公司，中国 深圳 518057；
2. 移动网络和移动多媒体技术国家重点实验室，中国 深圳 518055）
(1. ZTE Corporation, Shenzhen 518057, China;
2. State Key Laboratory of Mobile Network and Mobile Multimedia, Shenzhen 518055, China)

DOI：10.12142/ZTETJ.202304008
网络出版地址：http://kns.cnki.net/kcms/detail/34.1228.tn.20230724.1536.006.html
网络出版日期：2023-07-25
收稿日期：2023-05-25

摘要：网络感知算力并据此执行算网融合路由，是算网融合在网络基础设施侧的一种重要技术方案。相对于传统基于主机地址的网络路由机制，算网路由最主要的增量是在网络侧引入分布式多算力实例的优选，因此位置和归属无关的服务标识将成为新的寻址和路由标的。阐述了一种端到端的算网路由解决方案及其对路由协议的影响，并基于典型场景和测试用例，分析了面向服务标识的算网路由架构方案在功能和性能维度的收益。

关键词：算网融合；算力路由；服务标识

Abstract: Routing based upon convergent computing and networking where the network is enabled to be aware of the computing, as far as the network infrastructure of convergence of computing and networking is concerned, is a cornerstone technique solution. Contrary to the conventional host address-based routing scheme, the above-mentioned routing brings a process of computing selection among multiple and distributed sites and nodes. Therefore, service identification which is both location and homing independent should be employed in terms of addressing and routing. The comprehensive solutions as well as their impacts on the existing routing protocols are discussed, and the gains in both function and performance of the new architecture solution are demonstrated with contexts of typical scenarios and testing analysis.

Keywords: convergence of computing and networking; routing of computing; service identification

引用格式：黄光平, 谭斌, 吉晓威. 一种面向服务的算网路由架构方案 [J]. 中兴通讯技术, 2023, 29(4): 38-42. DOI: 10.12142/ZTETJ.202304008
Citation: HUANG G P, TAN B, JI X W. An architecture solution of service-oriented routing for computing and networking [J]. ZTE technology journal, 2023, 29(4): 38-42. DOI:10.12142/ZTETJ.202304008

在网络路由和调度体系中引入算力是算网融合架构中的重要增量，网络因此扩展了对算力的感知能力。同时，路由和调度流程实现了算力和网络两个维度资源状态的融合考量，即行业通常所说的算力路由[1]。算力路由是算网融合的主要技术锚点[2-3]。算力路由从端到端协议和流程方面，打破了网络和算力这两个传统上相互隔离的技术和资源体系壁垒，实现了“网中有算，以网强算”。算力路由的本质是：面向同类算力服务的分布式等价多实例，基于算力和网络的资源状态以及业务需求，执行网络和算力联合优选寻址，即“一对多”的算网寻址路由。其中的“多”表示网络和算力均存在多路径、多实例的权衡优选。而面向用户的算力服务是位置无关的，甚至可能是归属无关的。用户对算力服务的请求仅表达意图，无须关心服务的提供方和部署位置。这是算力路由跟传统基于主机位置的IP路由最本质的区别，也是算力路由协议体系存在的主要变量之一[4]。

算力路由引入位置无关的服务标识，将其作为路由和寻址的全新对象，在使能全新的算力感知和路由功能的同时，也为现有网络路由寻址协议带来新的扩展需求和挑战。因此，新架构功能在引入的同时，需要保持与现网架构兼容。服务标识的引入在客观上打通了业务和网络之间的高效感知接口。网络通过服务标识可以精细化识别业务，并提供相应的细颗粒度网络连接服务。

面向服务的算网融合路由技术在典型的业务场景下，有独特的业务和资源应用价值。而当前的业务场景还存在一些亟待解决的问题，仍需要充分发挥面向服务的算网融合路由技术的优势。

1 算力路由在IP分组网络中面临的主要问题

算力路由是叠加在传统IP分组网络基础上的一种增强性路由。在主机IP地址路由的基础上，网络需要增强算力感知的能力，并在此基础上执行算网融合路由。这既包括对算力服务的路由寻址，也包括对算力服务节点的主机路由寻址。算力感知和算力路由引入了全新的算力因子，这给IP分组网络带来4个方面的问题。

1）IP主机地址路由体系下的算力服务路由寻址问题

基于IP分组网络的算力路由，本质上是面向服务的分布式多算力实例寻址路由，即基于算力资源状态和网络资源状态，在多实例多路径中根据服务等级协议（SLA）需求进行算力节点优选或引流。这种面向服务的分布式路由机制，跟面向IP主机地址的路由机制完全不同。后者指向全局唯一主机，且基于前缀的寻址机制是基于物理上的子网部署模式；而算力路由从语义上并不指向特定算力服务主机，而是指向特定算力服务，并且同一类算力服务可能部署在不同的物理子网内。因此，基于IP前缀的子网模式并不适用于算力服务的部署模式。同一类算力服务与多服务实例及多实例主机地址关联，算力服务仅仅充当一种抽象类型索引。网络需要在这个服务索引与它对应的算力、网络资源、服务实例主机地址之间构建动态的映射关系。

2）算力感知对IP路由协议造成的震荡问题和表项膨胀问题

网络对算力资源状态的感知，需要针对相应的接口和协议进行扩展，并且在网络路由和转发节点引入新的算力路由表项。然而，算力资源类型及其状态变更频率都非常多样化，全颗粒度算力资源状态向网络暴露，将不可避免地导致现有网络协议（如边界网关协议）收敛震荡，对现网运行造成破坏性冲击。除此之外，海量的算力资源状态必将导致网络路由和转发节点对应数量级的算力路由表项，对节点性能造成严重影响。

3）算力对网络暴露的参数类型及颗粒度问题

对IP分组网络控制面而言，算力参数可以分为算力原始状态数据和网络链路维度的算力度量折算值（即网络路由域的Metric）。

a）算力原始状态数据。算力系统通过预定接口向网络管控系统直接通告算力原始运行状态数据，如服务实例会话负荷、CPU/GPU占用率、内存占用比等。网络管控系统会对这些原始数据按照特定规则或算法进行处理，并生成对应的算网路由策略，指导网络路由和转发节点进行流量引流和路由。这种模式将显著增加网络管控系统的处理复杂度和运行负荷。

b）网络链路维度的算力Metric。算力系统将自身运行的动态数据折算成网络链路维度的Metric并向网络管控系统通告，后者据此执行传统IP路由。但是这种模式势必引入巨量的头结点路由开销，比如需要维护每实例、每出口节点、每链路的路由条目[5-6]。

4）算力与网络融合路由带来的多因子多策略问题

基于分组网络系统执行算力路由时，网络和算力融合路由将带来多因子联合优化的策略问题。算网双维度因子的全面融合将导致路由协议体系及其算路流程复杂度翻番，并严重冲击当前既有的路由和转发性能，无法实现与现网的平滑兼容。

2 基于服务标识的算力路由技术

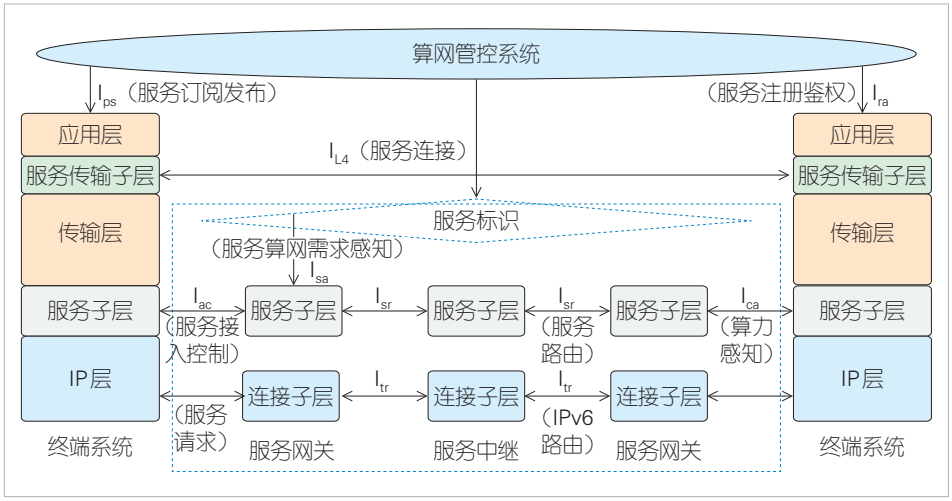
在IP路由协议体系中引入一个拥有独立语义的服务标识，将从根本上解决前文所述算力路由在IP网络中面临的主要问题，并提供统一的端到端架构解决方案。当然，服务标识也给IP网络带来一些新问题。这是在进行架构设计尤其是服务标识设计与界定的过程中需要特别考量的。

2.1 基于服务标识的算力路由架构

在算网融合调度和路由系统中引入服务标识，为IP分组网络提供了一个面向业务和算力系统的新型接口，使网络得以提供面向服务标识的路由和寻址功能。如图1所示，基于IP分组网协议的服务标识在数据面扩展定义和封装，并在控制面经由服务标识，打通算力系统动态资源和业务系统精细化SLA需求的感知接口，从逻辑上构成一个在IP分组网上的OverLay服务子层。传统分组数据网作为连接子层，为服务子层提供连接支撑能力。服务子层与连接子层之间以控制面服务标识为索引进行交互[7]。

如前文所述，服务标识在语义上与主机位置无关，因此传输层有可能通过服务标识保持业务连接，从而解决传统传输层终端或服务迁移连续性的问题，即主机地址变更导致的链路迁移仅在L3层执行，而L4层面向终端和用户的业务链接因为服务标识的位置无关属性得以维持不变，从而保障用户在这类场景下的业务体验。

数据面的服务标识是面向用户的一种轻量级算网服务能力集合表征。服务标识关联的算网质量和能力在特定算网运



▲图1 基于服务标识的算网路由架构

营管理域内可管可控，比如拥有端到端20 Mbit/s保障带宽的某种视频业务、10 ms端到端时延保障的渲染业务等。因此，服务标识内生支持精细化算网SLA需求的表征和接口。控制面基于这种服务标识的算网SLA Profile以及算网资源状态，生成以服务标识为索引的路由和转发策略，并将其下发到服务网关，指导业务流量转发。以入口服务网关对业务流量的转发和路由流程为例，用户侧报文通过服务标识表达对算力系统中特定算力服务的访问意图，以及这种服务在算网系统中的SLA需求。这里的服务标识并不指向特定的主机，而是由服务网关根据控制面的算网策略表选择特定的服务主机和网络链路，从而同时实现多服务实例间的算力优选和多网络路径中的路径优选，为对应的业务提供精细化的算网策略编排。由此可见，服务标识在东西向充当网络和算力系统之间的资源感知接口，在南北向充当网络和业务之间定制化业务SLA需求的高效感知接口。

需要指出的是，服务标识本身并不需要包含业务SLA需求的信息和参数，它仅需要在数据面和控制面之间充当映射接口即可。业务SLA需求语义由控制面来维护和表征。服务标识的可管可控、轻量级设计在解决安全性问题的同时，不会给服务网关硬件处理性能带来额外负担[8]。

1）服务标识的治理

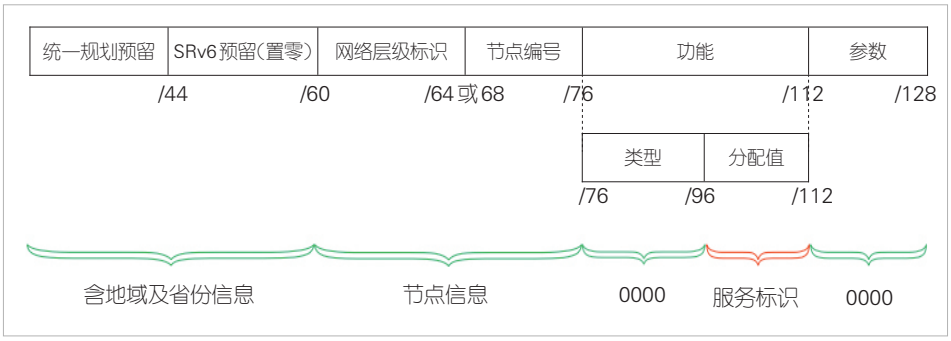
如前文所述，服务标识对用户、业务、算力和网络系统而言是一种接口。对于算网基础设施资源和业务运营方而言，服务标识是一种服务能力承诺。算网运营方应该对服务标识的全生命周期可管可控，即服务标识的注册、发布、订阅、更新和中止均应在算网运营系统的闭环治理范围内。在不同的算网运营管理域之间，服务标识的互通需要经过协商、映射甚至标准化，而这取决于特定算力服务的部署和运营模式。除了部分获得行业高度共识的基础服务涉及全网互通标准化之外，大部分服务标识的治理在单运营管理域内完成，无须标准化。

2）服务标识的封装

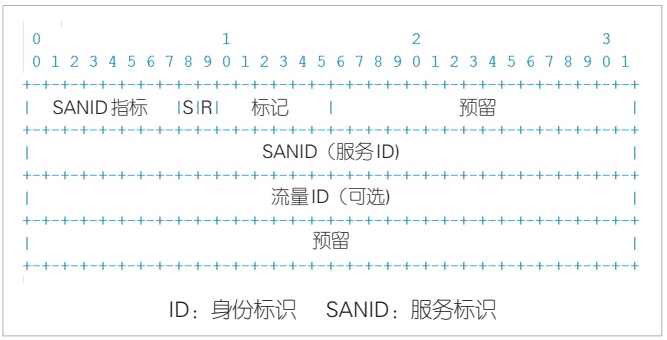
服务标识的表征对象是分布式多云部署的基础通用算力服务，因此，标识的对象空间有限。通常而言，16 ~ 32 bit足够覆盖既存的、可预见的服务类型。具体到IPv6报文头接口，服务标识的封装分为重用IPv6固定字段和扩展报文头定义封装两大类。

a）重用IPv6固定字段。源、目的地址以及流标签均可重用部分或者全部字段空间表征服务标识语义。如图2所示，基于SRv6地址结构的服务标识封装可重用功能中的低位16 bit。这种封装模式充分保留了SRv6地址原有的语义和功能。在重用IPv6固定字段的模式下，终端接口、业务请求流程以及协议栈均保持不变。此时方案落地环境兼容性较好。

b）扩展报文头定义封装。在IPv6标准扩展报文头目的选项头（DOH）、逐跳（HBH）、路由扩展头（SRH）中单独定义和封装服务标识头结构，如图3所示。服务标识头结构在服务标识之外封装了其他可选字段，用于特定场景。这种封装模式的优势是独立封装，不受服务网关节点本地处理机制的影响。服务标识可以直通算力服务系统，为算力系统网络提供增值功能，如可视化操作维护管理（OAM）、基于服



▲图2 基于SRv6地址结构的服务标识封装示例



▲图3 服务标识在IPv6扩展报文头中的独立封装示例

务标识的云内均衡和引流等。

2.2 层次化算力路由机制

将算力系统的全颗粒度算力资源状态信息通告同步到网络管控系统，将会导致现有IP分组网络协议收敛震荡和表项膨胀。为保持算力路由与现网路由协议体系的平滑兼容，需要对算力资源状态进行分类和聚合，在不同的网络节点维护不同类型的算力资源类别以及对应的算力路由表项，从而确保算力路由通告与现有IP路由之间的平滑兼容。全局算力路由表项条目数量仅与网络边缘节点有关，与云侧算力服务实例无关。这将压缩远端网络节点维护的算力路由表项空间，减轻节点的查表和处理负荷。高频变化的算力服务实例资源状态仅维护在本地网络边缘节点。这种层次化算力路由的机制，将控制面的端到端算网路由决策分层分布在网络远端和本地边缘节点，在转发流程上涉及两段路由转发：从网络远端到本地边缘节点、从本地边缘节点到算力服务实例。当然，这种层次化表项维护机制，将可能导致网络头结点算力资源信息的部分失真，可以满足绝大部分算力业务路由场景需求，但在极端异常场景下，仍需要引入丢弃或保护策略机制。

2.3 基于算力感知的算网路由解决方案

算力资源状态如何约束和影响网络边缘节点对算力和网络的选择，是算力路由的关键，也是基于IP路由的主要增量。因此，算力资源状态在网络控制面的呈现形态，是决定选择哪种端到端算力路由解决方案的关键因素。如前文第1节所述，算力参数主要有原始算力参数和网络维度算力Metric两种主要的呈现形态，与之对应的是两种不同的算力路由方案。

1） 基于算力映射的算力路由方案

算力系统向网络管控系统通告算力服务关联的原始算力状态数据。该原始算力状态数据与网络控制面路由决策系统之间的索引接口即为服务标识。网络控制面基于此类原始算力状态数据，结合网络资源状态、业务SLA需求，生成算网路由策略，完成原始算力状态数据到主机地址的映射。这个方案的优势是网络节点无须维护额外的算力路由表项。当然，在分布式路由协议方案下，算力原始状态数据的通告同样需要层次化状态维护机制，以平滑兼容现网路由协议。

2） 基于算力Metric的算力路由方案

算力系统通过一定的度量和折算机制，将算力服务关联的原始算力状态数据转换为网络维度的度量Metric，并通过特定协议接口向网络管控系统通告。具体来讲，这里的Metric可以是网络维度既有的Metric类型（如时延、带宽、等），也可以是新增的算力Metric类型。前者可以沿用既有的路由算法完成端到端算网路由编排，后者则需要扩展基于算力Metric的路由算法完成端到端路由编排。分布式路由协议方案下的层次化路由机制引入与上文所述类似，这里不再赘述。

2.4 基于算力与网络解耦的多因子多策略路由机制

在IP分组网络基础上执行算力路由，本质上是将传统IP网络的网络单维路由算法升级为算网二维路由算法。算力和网络两个维度的约束变量理论上是乘数关系，但在实际部署中，这种算网全维乘数算法将大幅增加路由算法的复杂度，甚至破坏现有IP路由协议机制的稳定性。远端网络边缘节点将“选算”和“选网”分离处理，使两类路由先决策再进行线性叠加，形成近似的算网融合优化路由策略。因此，算力和网络路由解耦，将算网二维乘数算法简化为一维线性叠加算法，并在算网融合的基础上，简化路由协议流程。需要说明的是，这种解耦机制不影响现有IP路由协议。

算网解耦以及算力、网络、业务SLA多种路由因子的引入，也为算网系统提供了多元调度机制，使能灵活的算网业务和资源运营模式。算网调度因子可以分为如下3类：

1）体验类：服务质量和体验相关的SLA指标，如时延、抖动、丢包等；

2）代价类：服务关联的算网资源成本、能耗等；

3）资源类：服务关联的算网资源的使用效率，如算网均衡度、算网利用率等。

相关算网调度策略有4种：

1）体验优先：体验类指标最优调度；

2）代价优先：体验类指标满足设定门限指标，代价类指标最优调度；

3）资源优先：体验类指标和代价类指标均设定门限指标，资源类指标最优调度；

4）资源均衡：体验类指标和代价类指标均满足设定门限指标，资源类指标均衡调度（资源使用率的方差最小）。

3 基于服务标识的算网路由评价体系及测试分析

相对于传统IP路由，算力路由带来了多方面的增量功能。这里我们从4个维度给出算力路由价值评价体系，并对方案的部分测试数据进行简要分析。

1）增强服务会话响应时延性能。算力路由通过数据面带内服务发现替代传统DNS带外服务发现。这里的服务响应时延是指：客户端首包发出到获得服务的时间间隔。DNS服务发现机制下的服务响应时延在100 ms ~ 1 s之间。本文以传输控制协议（TCP）3次握手为服务会话建立基准，使端到端响应时延低至2.76 ms，大大提高了服务会话的响应时延性能。

2）提升服务算网质量。网络感知与计算、网络质量是SLA双维度保障。在服务体验方面，网络可能会出现丢包、卡顿等现象。业务有效通量为用户的实际业务量。

3）实现资源利用率均衡。这包括网络资源利用率均衡、算力资源利用率均衡（池间），涉及网络负载偏离度和算力负载偏离度。其中，网络负载偏离度是指：调度过程中同一时刻不同网络路径的资源利用率的最大差值，算力负载偏离度是指：调度过程中同一时刻不同算力池的资源利用率最大差值。本文中，我们测试了两种机制下的资源利用率均衡度。在非均衡调度条件下，4个用户的流量均为20 Mbit/s，由资源池A提供服务，资源池B空载，此时负载偏离度比较高（>40%）；在均衡调度条件下，4个用户的流量均为20 Mbit/s，由资源池A和资源池B提供均衡服务，此时负载偏离度较低（<11%）；

4）提高资源利用效率。资源总量相同，网络可以承载更多的用户会话。

4 总结

在传统IP路由的基础上扩展算力路由功能，是实现算网融合的关键技术要素。算力路由与IP主机路由之间在机理和目标方面存在较大的差距，从而给方案部署带来诸多挑战。本文聚焦4类算力路由带来的协议和调度策略问题，并以平滑兼容现网协议和架构为目标，针对性地提出基于服务标识的算网路由架构方案。该方案的核心是引入独立于IP主机地址的服务标识，并构建用户与算网系统之间、网络与业务之间、网络与算力系统之间的简明高效互通接口。在此基础上，本文创造性地提出层次化算力路由、算力映射与算力Metric路由机制、基于算网解耦路由的多因子多策略算法等解决方案，为IP分组网络提供兼容性较好的端到端算力路由方案；同时，基于4个维度的算力路由价值评价体系，对部分典型场景测试数据进行分析。

参考文献

[1] 陈晓, 黄光平. 微服务架构下的算力路由技术 [J]. 中兴通讯技术, 2022, 27(1): 70-74. DOI:10.12142/ZTETJ.202201014
[2] 唐雄燕, 张帅, 曹畅. 夯实云网融合, 迈向算网一体 [J]. 中兴通讯技术, 2021, 27(3): 42-46. DOI:10.12142/ZTETJ.202103009
[3] 周吉喆, 杨思远, 王志勤. 面向业务感知的算网融合关键技术研究 [J]. 中兴通讯技术, 2022, 27(5): 2-6. DOI:10.12142/ZTETJ.202205002
[4] 朱海东. 云网一体使能网络即服务 [J]. 中兴通讯技术, 2019, 25(2): 9-14. DOI: 10.12142/ZTETJ.201902002
[5] 刘铎, 杨涓, 谭玉娟. 边缘存储的发展现状与挑战 [J]. 中兴通讯技术, 2019, 25(3): 15-22. DOI: 10.12142/ZTETJ.201903003
[6] 雷波, 宋军, 曹畅. 边缘计算2.0: 网络架构与技术体系 [M]. 北京: 电子工业出版社, 2021
[7] 陈晓, 郭勇, 谭斌, 等. 面向算网一体的开放服务互联架构 [J]. 信息通信技术, 2022, 16(2): 53-59
[8] HUANG D, TAN B, YANG D. Service aware network framework [EB/OL]. (2021-06-015) [2023-03-06]. https://datatracker. ietf. org/doc/html/draft-huang-service-aware-network-framework-01

作 者 简 介

黄光平，中兴通讯股份有限公司资深架构师；主要研究方向为下一代IP网络架构及关键技术，先后从事增值业务消息系统设计和开发、确定性网络以及远程宽带接入网关全球标准工作，近年聚焦算力网络架构、路由协议、算力标识等技术研究；发表论文8篇，申请专利30余项。

谭斌，中兴通讯股份有限公司未来网络技术研究项目经理；主要研究方向为IP网络、SDN系统架构与技术，先后从事有线路由器、接入产品开发、产品规划和市场等工作；申请专利2项。

吉晓威，中兴通讯股份有限公司IP产品规划工程师；长期从事IP网络、SDN/NFV等产品的规划和系统设计，目前研究方向为云网融合、算力网络；获中国算力大会创新先锋奖、SDN/NFV/网络AI最佳实践案例奖等奖项。

通用在网计算系统架构及协议设计

System Architecture and Protocol Design for Generic In-Network Computing

姚柯翰/YAO Kehan，陆璐/LU Lu，徐世萍/XU Shiping
（中国移动通信有限公司研究院，中国 北京 100053）
(China Mobile Research Institute, Beijing 100053, China)

DOI：10.12142/ZTETJ.202304009
网络出版地址：http://kns.cnki.net/kcms/detail/34.1228.TN.20230718.1826.002.html
网络出版日期：2023-07-19
收稿日期：2023-05-20

摘要：生成式人工智能大模型训练以及海量的大数据实时处理对任务处理性能提出了更高要求。在网计算在数据完成尽力而为转发的基础上，进一步将计算相关的操作卸载至转发节点，可以有效提升系统计算效率。针对当前在网计算系统设计碎片化问题开展研究，提出了满足多种在网计算场景的通用系统架构，并进行相关协议设计。通用在网计算架构兼顾了系统实现的灵活性以及应用开发友好性，为进一步推进在网计算的规模化应用提供了新思路。

关键词：在网计算；算力网络；网络架构；网络协议

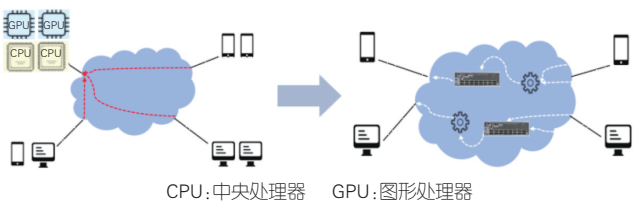
Abstract: The large model training of generative artificial intelligence and the real-time processing of big data need higher requirements for task processing performance. On the basis of completing the best-effort forwarding of data, in-network computing (INC) further offloads computing-related operations to network forwarding nodes, which can effectively improve the system computing efficiency. Aiming at solving the problem of design fragmentation of INC systems, a generic system architecture is proposed to meet different requirements of various INC scenarios, and related protocols are designed. The generic INC architecture takes into account the flexibility of system implementation and the friendliness of application development, and puts forward a new idea for further improving the scalability of INC.

Keywords: in-network computing; computing force network*; network architecture; network protocol

引用格式：姚柯翰, 陆璐, 徐世萍. 通用在网计算系统架构及协议设计 [J]. 中兴通讯技术, 2023, 29(4): 43-48. DOI: 10.12142/ZTETJ.202304009
Citation: YAO K H, LU L, XU S P. System architecture and protocol design for generic in-network computing [J]. ZTE technology journal, 2023, 29(4): 43-48. DOI: 10.12142/ZTETJ.202304009

1 在网计算的发展

在网计算（INC）指将部分计算任务卸载至网络，让数据在完成转发的同时实现数据处理，从而提升数据计算效率。如图1所示，传统的计算模式是数据由终端产生，全部送往集中的数据处理节点（如数据中心）来完成运算。



▲图1 端侧计算向在网计算演进

在网计算则可实现数据边转发边处理，大大降低数据处理节点的负载。

1.1 在网计算演进历程

在网计算的技术理念首次出现在1995年由美国国防部高级研究计划局（DARPA）提出的主动网络中[1]。在主动网络中，网络数据包不仅携带数据，还携带了数据的操作信息或程序。在网计算可以给主动网络中的转发节点使能主动计算属性，基于数据包中的程序指令对数据包进行操作，从而实现应用相关的功能，比如防火墙或网页代理等。但主动网络并未形成主流技术体系，主要原因在于其实现依赖于中央处理器（CPU）的处理能力。而网络设备的核心要务是进行

* 作者确认算力网络译为computing force network

线速数据包转发，这对转发芯片能力有严格要求。在当时，网络设备的转发芯片并不支持可编程能力，因此能够在网络设备做的计算也比较有限。

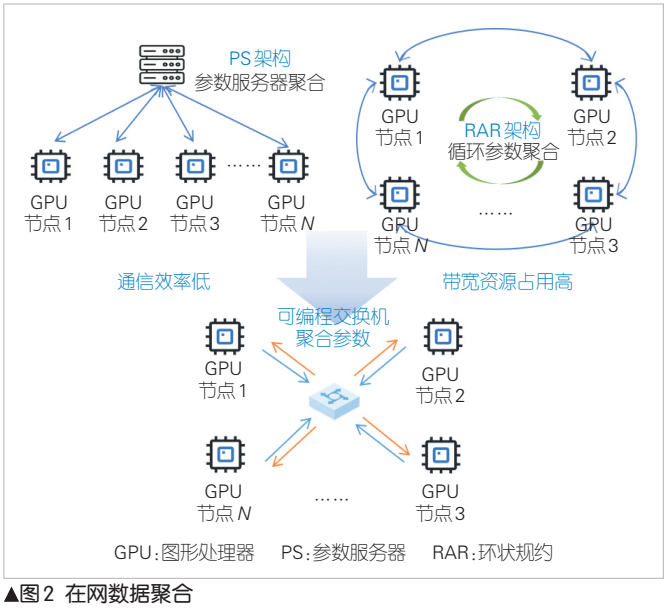
近年来，随着可编程网络硬件的发展以及软件定义网络架构的逐渐成熟，在网计算技术不断发展。斯坦福大学N. MCKEOWN教授团队在2014年发表的论文中首次提出了协议无关的包处理编程语言P4[2]，用于对网络数据平面的算法和处理逻辑进行自定义编程，从而实现更加灵活丰富的功能。目前，大量的学术研究聚焦在如何发挥可编程网络的灵活性和高性能，可编程网络成为在网计算发展的关键使能技术。

1.2 在网计算主要应用场景

在网计算[3]已广泛用于各种分布式系统。在网计算将应用相关的功能卸载至网络节点，实现分布式应用处理性能的有效提升以及网络带宽资源的合理优化。本节中，针对目前在网计算在应用加速方面的主要研究，我们进行了总结，内容主要包括在网数据聚合、在网数据推理、在网缓存以及在网共识。

1）在网数据聚合

分布式机器学习模型训练可以基于在网数据聚合进行加速。目前，主流分布式机器学习系统架构为环状规约（RAR）和参数服务器（PS），如图2所示。RAR架构对网络带宽资源占用高，完成一次完整的分布式机器学习模型训练任务需要传递约模型总参数量2倍的通信量，极易引起网络拥塞；而PS架构则由于集中式服务器节点的吞吐瓶颈问题，面临较大的聚合延迟，限制了分布式机器学习模型训练的效率，扩展性较差。



▲图2 在网数据聚合

在网计算可由网络交换节点实现参数聚合功能，既克服了聚合节点的吞吐瓶颈问题，也避免了RAR架构高额的带宽资源占用，实现了训练性能和带宽资源的有效平衡，极大地提升系统的扩展性。

2）在网数据推理

在网数据推理可实现网络流量分类和控制。业界相关的研究包括在网络转发设备实现决策树、支持向量机（SVM）、朴素贝叶斯等各种分类算法[8]，以及通过神经网络实现联邦学习，支撑网络设备在网络路径上就近返回处理结果，从而提升集群计算能力。

与基于分析服务器的推理方式相比，中间层交换机推理提前终止了终端设备发往分析服务器的原始数据流量，节省了更高层核心网络的带宽，同时利用网络设备的高速处理来减少推理时间，加速数据实时分析和控制指令响应。

3）在网数据缓存

高性能的分布式数据存储和索引需要依赖于高性能的Key-Value存储。在社交网络等高并发应用中，慢速的Key-Value存储可能导致较大的系统尾延迟，进而影响系统性能。通过设计层次化的缓存系统，在边缘网络节点部署Key-Value缓存服务，在网络设备中完成高频内容缓存以及快速查询和响应[4]。在网数据缓存机制是高性能存储系统以及高性能流式处理系统加速的关键。

4）在网数据共识

在分布式系统中，可以通过共识协议来实现对某个数据值或操作序列的一致性，比如锁定管理系统、组播通信、一致性协调。卸载共识功能的部分或全部功能卸载到网元，可以减少协调延迟，提升分布式系统的可用性。文献[5]利用可编程交换机实现了一致性算法的网内卸载，实验也证明了在网数据共识对分布式系统性能的优化。

2 通用在网计算架构

在网计算对系统的性能优化已被广泛地论证，但是在网计算在架构设计层面还面临碎片化的问题。目前，在网计算主要根据应用场景进行定制化设计，满足相关应用的个性化需求，但是这样的设计方法扩展性较差，不利于在网计算的规模化应用。

同时，对应用开发者而言，想要基于网络设备的计算能力进行系统设计，既需要了解上层系统的逻辑架构，还要了解底层物理网络的属性，包括网络设备的编程能力以及网络的规划能力。这也进一步提升了在网计算系统的设计门槛，阻碍了在网计算的应用。

为解决上述问题，我们在架构设计层面，从在网计算的通用性和应用设计的友好性出发，设计了S（任务调度层）、C（在网计算控制层）、I（基础设施层）3层通用在网计算架构，系统架构如图3所示。

2.1 基础设施层

基础设施层包含执行计算任务的端侧主机节点以及在网计算节点。由于异构网络设备在硬件架构方面存在较大差异，这些在网计算节点能够提供的计算能力也不同。这意味着同一个在网计算原语在异构网络设备内部可以有不同的实现方式。针对不同场景下在网计算原语，很多研究进行了分类和总结[7]。本文在这些研究的基础上，进一步设计了面向异构在网计算节点的统一北向接口，在网计算节点通过北向接口上报在网计算原语信息，对外提供统一的服务接口。这使得在网计算更具通用性。

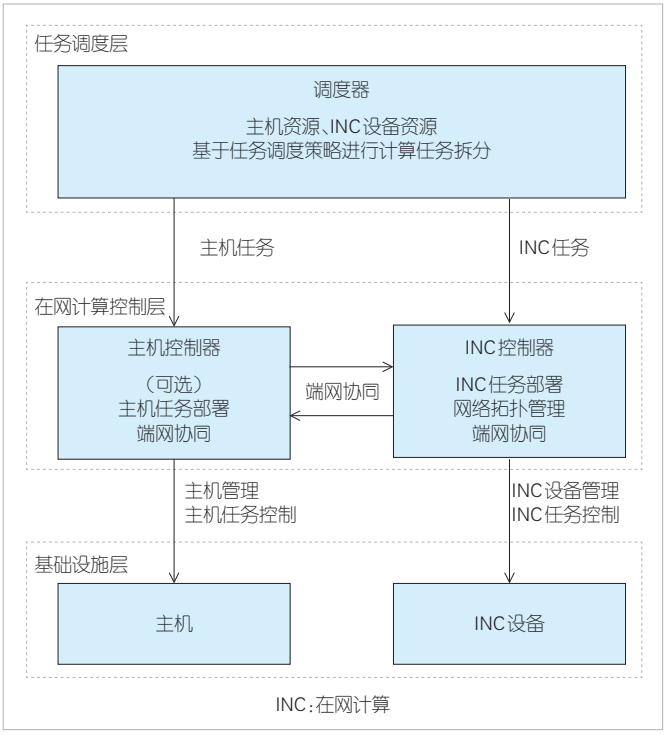
在端主机侧，应用程序也需要进行相应适配。网络无法保证大部分在网计算应用独立完成计算任务，因此需要通过端网协同机制来完成。端主机侧应用程序需要感知在网计算任务，这样可以保证计算的完整性，同时可以提高网络传输的可靠性。端主机侧通过北向接口连接端侧任务控制器，实现端侧计算任务分配。

2.2 在网计算控制层

在网计算控制层是实现端网协同在网计算的关键。控制层包含主机控制器和在网计算控制器。主机控制器根据应用场景按需部署，主要负责主机任务部署以及端到端的可靠性保障。在网计算控制器主要负责通用的网络管理以及在网计算任务部署和控制等。在网计算控制器通过南向接口实现网络管理功能，包含网络设备管理以及网络拓扑管理。网络设备管理包括网络设备状态、网络设备负载、网络设备计算能力、网络设备计算资源管理等，其中网络设备的计算能力是在网计算控制器和传统网络控制器最大的不同。网络设备的计算能力通常通过在网计算原语、在网计算数据结构来表示。网络拓扑管理包括网络拓扑更新、链路状态监控等。主机控制器和在网计算控制器共同实现端网协同控制，并根据网络资源状态综合选路，为在网计算和转发选择一条最优路径。

2.3 任务调度层

任务调度层实现在网计算系统和应用的对接。应用将任务需求提交给统一的任务调度器。任务调度器通过南向接口对接端侧控制器以及在网计算控制器，收集端侧和网侧的当前计算资源状态。任务调度器结合应用任务请求及计算资源状态，基于特定的算法进行计算图设计，生成计算节点之间的逻辑依赖关系，进而产生具体的任务分配策略。恰当的任务调度策略可以实现合理的在网计算资源分配，从而在保证任务处理性能的同时优化网络管理。



▲图3 通用在网计算架构

3 基于SRv6协议的通用在网计算的实现

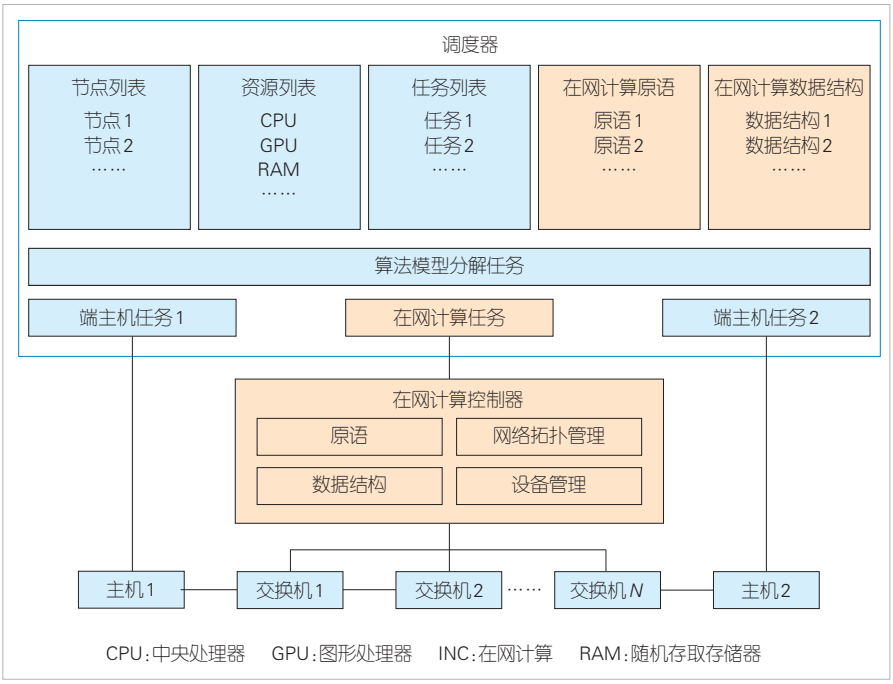
通用在网计算框架不约束数据面转发协议。本节中，我们以数据面运行基于IPv6的段路由（SRv6）协议为例，说明通用在网计算框架的工作机制。

3.1 基于SRv6协议的通用在网计算实现流程

基于SRv6的集中式通用在网计算架构如图4所示。该架构未部署主机控制器，任务调度器直接对接服务器，在网计算控制器负责承接在网计算任务，数据面运行SRv6协议，SRv6包头在接入交换机上封装。

1）管理员配置在网计算控制器，将在网计算原语和在网计算数据结构模型配置生成模板库。

2）网络设备初始化时，上报自身在网计算能力，实现标准的在网计算原语和在网计算数据结构。设备自身的实现可能有计算精度、数据范围等差异。网络设备平稳运行后，周期上报自身负载和在网计算能力变化。



▲图4 集中式通用在网计算框架

3）调度器根据任务分解策略将计算任务拆解为主机任务和在网计算任务，拆解时需要考虑主机和在网计算的能力和资源，然后告知在网计算控制器该任务具体要执行哪些在网计算原语。

4）当主机节点有数据要执行在网计算时，首先向在网计算控制器发送请求，说明在网计算任务ID、源节点、目的节点、要执行的在网计算原语，然后由UniqueID对业务分配标识。

5）在网计算控制器根据网络拓扑、在网计算能力、网络负载等情况进行综合选路，将选路结果反馈给主机侧，并在网络设备上做在网计算资源预留。

6）源服务器发送数据包，接入交换机封装SRv6头，各网络节点根据协议包头在网计算指示信息执行在网计算。

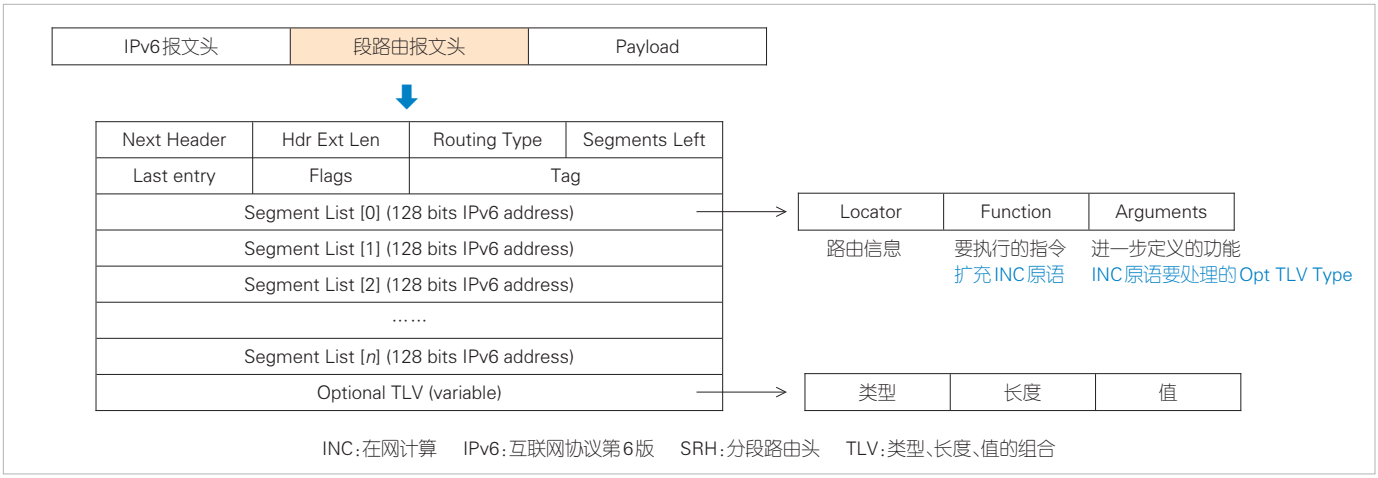
3.2 基于SRv6协议报文头扩展

国际互联网工程任务组（IETF）定义了SRv6段标识符（SID）中的Function字段的通用转发行为[9]。如图5所示，我们新增了INC Segment定义在网计算行为。其中，Locator与其他segment保持一致，表示路由位置信息；Function字段表示具体要做的在网计算原语；Arguments指示对应Optional TLV（类型、长度、值的组合）的类型，可以由应用自行定义；Optional TLV可以用来携带在网计算原语所需要的信息，例如需要处理的数据偏移、计算需要的参数等。

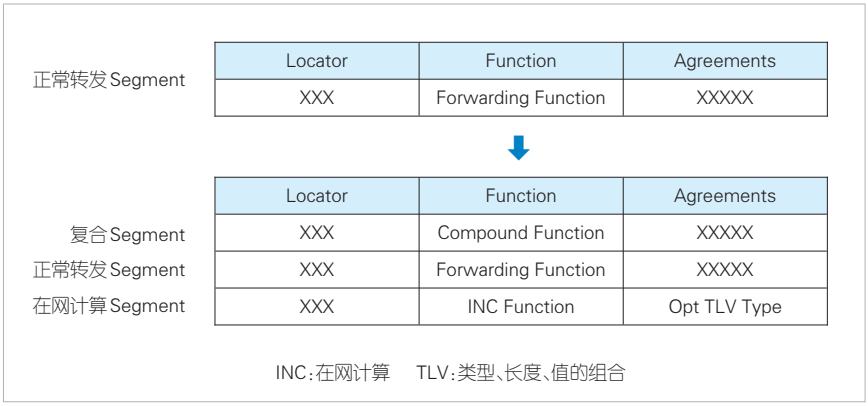
节点在执行在网计算时，不能影响正常转发，因此需要增加复合Segment，使在网计算交换机同时支持正常转发和在网计算能力，具体如图6所示。复合Segment用来指示后续两个连续的Segment都需要在本节点处理，第1个为正常转发Segment（Forwarding Segment），第2个为在网计算Segment。

3.3 交换机在网计算原语能力传播

交换机的在网计算原语可以由控制器统一管理，并基于路由协议进行信息扩展，再传递到相应的在网计算执行节点。例如，自治域内源路由使用内部网关协议（IGP）来传



▲图5 SRH协议支持在网计算协议扩充



▲图6 复合Segment使交换机同时支持转发和在网计算

播SID和对应的Function等信息。以中间系统–中间系统协议（ISIS）为例，传递在网计算可以使用两种方式。

1）扩展ISIS SRv6协议中Sub–TLV字段

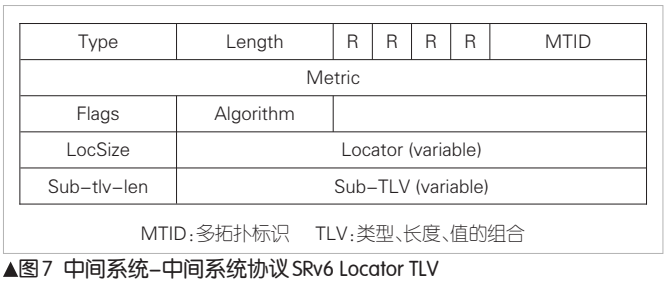
SRv6 Locator TLV用于发布SRv6 Locator以及该Locator相关的Endpoint SID。Locator具有定位功能，一般要在段路由域内唯一标识，Endpoint SID用于标识网络中的某个目的节点。

ISIS的SRv6 Locator TLV格式如图7所示。其中，Locator（variable）表示发布的SRv6 Locator，长度可变；Sub–TLVs（variable）可以根据类型不同，携带不同信息，长度可变。

因此，可以有方案1：扩展Sub–TLVs，新增一种描述在网计算原语信息的报文结构。其中，Type字段表示在网计算原语类型，Length字段表示Value长度。Value为交换机支持的在网计算原语补充信息，如果Value等于0则无补充信息。

2）扩展一种新类型的SRv6 Sub–TLV

ISIS本身有多种Sub–TLV协议报文格式，分别用来传递不同信息。其中，SRv6 Capabilities Sub–TLV用于通告SRv6



▲图7 中间系统–中间系统协议SRv6 Locator TLV

▲图8 中间系统–中间系统协议SRv6 Capabilities Sub–TLV报文格式

能力。SRv6 Capabilities Sub–TLV的格式如图8所示。

因此，可以有方案2：新定义一种用于传递在网计算原语能力的Sub–TLV类型SRv6 INC Sub–TLV。其中，Type字段为网计算原语能力，Optional Sub–sub–TLVs为交换机支持的在网计算原语。

4 在网计算发展的挑战

1）网络设备硬件资源受限

目前，可编程网络设备尚不能支持大规模或泛在的在网计算，主要原因在于可编程硬件片上资源受限。例如Tofino交换芯片，其片上的静态随机存取存储器（SRAM）、三态内容可寻址存储器（TCAM）存储空间约数十兆字节[6]，只能存储少量的带状态数据。另外，分布式机器学习及高性能计算需要在网计算具备高精度浮点数处理能力，但目前可编程交换芯片只能支持整型数据处理。

2）跨设备资源管理和任务协同

分布式系统中高并发、大数据量的处理任务对在网计算资源提出挑战，这导致在网计算的加速性能有限，因此需要设计跨交换资源的管理机制以及任务跨设备的分解调度机制，以实现在网计算的规模扩展。交换机资源如何池化以及如何利用控制器进行资源和任务协同，还有待进一步研究。

3）计算可靠性挑战

在网计算在转发的同时要实现对数据的处理，这给传统的可靠性机制带来了挑战。网络尽力而为的转发机制可能会造成在网计算结果错误。例如，在网数据在聚合过程中会丢弃已聚合的数据包，只保留最后的聚合结果，传统的可靠性机制会将这一行为判断为丢包。再如，在网计算设备可能由于资源不足或其他原因导致无法完成在网计算任务，可靠性机制需要能够灵活判断和计算。不同的场景对于可靠性的要求不同，这给在网计算的发展带来了很大的挑战。

4）安全性挑战

在网计算需要在网络转发节点终结一部分数据流并进行数据操作，这在一定程度上为网络引入了安全风险。目前，在网计算的主要应用和设计聚焦在安全可控的网络场景中。未来，面向通用泛在的在网计算应用场景，如何提升系统安全性，降低数据计算结果被篡改的风险成为挑战。

5 结束语

本文分析了在网计算在多种应用场景下的共性能力，并

针对在网计算系统碎片化问题进行架构设计，提出了S、C、I的3层通用在网计算系统架构。异构在网计算节点通过统一的北向接口向在网计算控制器上报计算能力，为不同应用场景提供共享的网络基础设施。同时架构简化了在网计算应用开发，应用只需要向任务调度器提出需求，再由任务调度器综合决策，有效避免了应用开发者对底层物理网络复杂逻辑的理解，降低了应用开发门槛。本文以SRv6数据面协议为例，设计了通用在网计算的实现机制，同时针对在网计算的通用性和扩展性的提升提出了一些需要关注的问题。

参考文献

[1] TENNENHOUSE D L, SMITH J M, SINCOSKIE W D, et al. A survey of active network research [J]. IEEE communications magazine, 1997, 35(1): 80–86. DOI: 10.1109/35.568214

[2] 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. DOI: 10.1145/2656877.2656890

[3] LAO C L, LE Y F, MAHAJAN K, et al. ATP: in–network aggregation for multi–tenant learning [C]//18th USENIX Symposium on Networked Systems Design and Implementation (NSDI 21). NSDI, 2021: 741–761

[4] SEEMAKHUPT K, LIU S H, SENEVIRATHNE Y, et al. PMNet: in–network data persistence [C]//Proceedings of 2021 ACM/IEEE 48th Annual International Symposium on Computer Architecture (ISCA). IEEE, 2021: 804–817. DOI: 10.1109/ISCA52012.2021.00068

[5] JIN X, LI X Z, ZHANG H Y, et al. Netchain: scale–free sub–RTT coordination [C]//Proceedings of the 15th USENIX Conference on Networked Systems Design and Implementation. ACM, 2018: 35–49. DOI: 10.5555/3307441.3307445

[6] CHOLE S, FINGERHUT A, MA S, et al. dRMT: disaggregated programmable switching [C]//Proceedings of the Conference of the ACM Special Interest Group on Data Communication. ACM, 2017: 1–14. DOI: 10.1145/3098822.3098823

[7] ZHAO B, WU W, XU W. NetRPC: enabling In–network computation in remote procedure calls [EB/OL]. [2023–05–20]. https://arxiv. org/abs/2212.08362

[8] ZHENG C, XIONG Z, BUI T T, et al. IIsy: practical in–network classification [EB/OL]. [2023–05–27]. https://arxiv.org/abs/2205.08243

[9] FILSFILS C, CAMARILLO P, LEDDY J, et al. Segment routing over IPv6 (SRv6) network programming: RFC 8986 [S]. 2021

作　者　简　介

姚柯翰，中国移动通信有限公司研究院研究员；研究方向包括未来网络架构、在网计算、可编程网络等。

陆璐，中国移动通信有限公司研究院基础网络技术研究所副所长、中国通信标准化协会TC5核心网组组长；长期从事移动核心网策略、演进、标准和技术研究工作，主要涉及未来网络架构、智能管道、边缘计算、算力网络等领域。

徐世萍，中国移动通信有限公司研究院研究员；研究方向包括未来IP网络、算力网络、在网计算等。

大规模语言模型的跨云联合训练关键技术

Key Technologies for Cross-Cloud Joint Training of Large-Scale Language Models

潘囿丞/PAN Youcheng，侯永帅/HOU Yongshuai，
杨卿/YANG Qing，余跃/YU Yue，相洋/XIANG Yang
（鹏城实验室，中国 深圳 518055）
(Peng Cheng Laboratory, Shenzhen 518055, China)

DOI：10.12142/ZTETJ.202304010
网络出版地址：http://kns.cnki.net/kcms/detail/34.1228.TN.20230724.1522.002.html
网络出版日期：2023-07-25
收稿日期：2023-06-08

摘要：模型参数规模的不断增加使模型训练所需的算力资源变得更加庞大，导致很多情况下单个算力集群难以满足大规模语言模型的训练需求。大规模语言模型的跨云联合训练成为解决这一问题的有效方式。以自然语言处理大模型的跨云预训练和微调为例，介绍了大规模语言模型跨云训练的主要挑战和关键技术，并探讨了这些技术在跨云训练过程中的具体应用、实际效果和未来场景。这些技术将为智能化应用和人机交互等提供有力支持。

关键词：大规模语言模型；算力资源；跨云训练；自然语言处理

Abstract: As the scale of model parameters continues to grow, the computational resources required for model training become significantly larger. This often leads to situations where a single computing cluster is insufficient to meet the training needs of large-scale language models. Cross-cloud joint training of large-scale language models has emerged as an effective solution to addressing this challenge. In this study, taking cross-cloud pre-training and fine-tuning of natural language processing models as examples, we introduce the main challenges and key technologies involved in cross-cloud training of large-scale language models. The specific applications, practical effects, and future scenarios of these technologies in the cross-cloud training process are explored. These technologies will provide strong support for intelligent applications and human-computer interaction.

Keywords: large-scale language model; computational resource; cross-cloud training; natural language processing

引用格式：潘囿丞, 侯永帅, 杨卿, 等 . 大规模语言模型的跨云联合训练关键技术 [J]. 中兴通讯技术, 2023, 29(4): 49-56. DOI: 10.12142/ZTETJ.202304010

Citation: PAN Y C, HOU Y S, YANG Q, et al. Key technologies for cross-cloud joint training of large-scale language models [J]. ZTE technology journal, 2023, 29(4): 49-56. DOI: 10.12142/ZTETJ.202304010

大规模语言模型是一种使用深度学习方法技术在大规模无标注文本语料数据上进行训练的人工智能方法。近年来，这类模型得到了快速发展，模型能力实现极大提升。然而，模型的参数规模也变得越来越大。例如，2018年谷歌的BERT-Base模型只有1.1亿个参数[1]，而到了2020年，OpenAI的GPT-3模型的参数量已经达到1 750亿个[2]。随着模型参数的增加，模型训练所需的算力资源也变得更加庞大。BERT-Base模型可以在单张图形处理器（GPU）上训练，而GPT-3模型则需要在数千张GPU上进行数月的训练。当前，单个算力集群很少具备数千张GPU算力卡的规模，即使是那些具有数千张卡的算力集群，也很难将它们在长时间内集中用于同一个任务。因此，为了满足大规模语言模型的训练需求，需要将多个算力集群的资源联合训练来提高效率。随着“东数西算”工程的逐步开展，中国各地建立了大量的算力集群。异地跨云计算将成为今后大模型训练的可行方式。

基金项目：科技创新2030—“新一代人工智能”重大项目（2022ZD0115301）

1 基于多算力集群的跨云训练方法

1.1 跨云计算的并行训练方式

在跨云集群环境中进行模型训练，需要解决不同云集群

之间参数的传递和同步问题，以及由大量数据跨云传输的时间开销导致模型训练速度慢的问题。为了提升训练速度，训练任务被拆分到多个不同的算力集群上。利用这些集群的算力，可以实现对任务的并行处理。根据不同的任务需求和场景，跨云训练可以采用不同的并行策略，包括数据并行、模型并行和流水线并行等。

数据并行是提升训练速度的一种并行策略，能够将训练任务切分到多个算力集群上。每个集群维护相同的模型参数和计算任务，只是处理不同的批数据。通过这种方式，全局的数据被分配到不同的进程，从而减轻单个集群上的计算和存储压力。

模型并行主要用于模型太大、无法在单个设备上加载的场景，对计算图按层切分以减少单个存储的容量需求，每个集群只保留模型的一部分。因此，多个算力集群可以共同训练一个更大的模型。

当模型并行在某个集群进行计算时，其余集群都会处于闲置状态，这样会极大地降低整体的使用效率。于是，在模型并行的基础上，如图1所示，把原先的批数据再划分成若干个微批次，按流水线方式送入各个算力集群进行训练，也就是流水线并行[3]。

当在跨云场景下进行大规模语言模型训练时，由于巨大的数据量和参数规模，不论是对训练数据还是模型张量进行切分，在进行跨云同步传输时都会产生较大的耗时，会影响整体的训练速度。由此可见，数据并行和模型并行这两种方式能够支持的模型参数规模有限。而流水线并行训练则将模型参数按照层次进行拆分，把不同层的模型参数放到不同集群中进行训练。训练过程中不需要同步全部模型参数，集群之间只需要串行传递训练过程的中间计算变量。该方法受模型参数规模影响较小，更适合大规模语言模型的跨云训练。

1.2 跨云流水线并行的主要挑战及关键技术

跨云流水线并行和普通流水线并行的最大区别在于处理通信数据的方式。目前，普通流水线并行策略通常仅在单个计算资源中心内部使用，这意味着计算设备之间存在专用的高带宽网络连接。此时，通信代价极低，通常可以忽略不计。然而，当普通流水线并行策略应用于跨云场景时，计算设备之间的连接带宽远低于上述连接，通信代价将显著增加，这将极大地影响训练效率。图1的左图和右图分别展示了普通流水线并行和跨云流水线并行的处理流程。

普通流水线并行的效率评价指标为并行空泡占用率比例（parallelism bubble ration），该比例越小代表效率越高。假设并行的阶段（stage）数为 p，微批次的数量（micro-batch）为 m，每个micro-batch的前向和后向执行时间为 t_f 和 t_b，则空泡率为：

$$bubbleration = \frac{p-1}{m+p-1}。\tag{1}$$

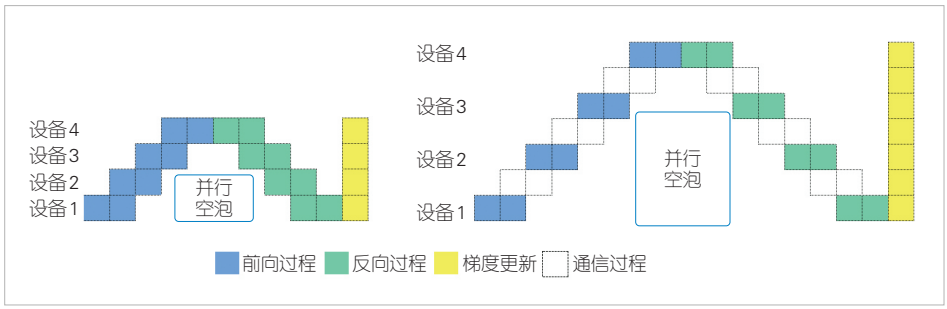
而在跨云流水线并行中，会出现因为通信而导致的额外空泡。假设通信时间为 t_t，在不做任何处理的情况下，前向和后向的通信时间相等，此时空泡率为：

$$bubbleration = \frac{(p-1)(t_f+t_b+2mt_t)}{m(t_f+t_b)+(p-1)(t_f+t_b+2mt_t)}。\tag{2}$$

因此，跨云流水线并行所面临的主要挑战是如何提高训练效率，即如何降低并行空泡的占用率。从上述公式（2）中可以看出，在跨云场景中，与普通流水线并行不同，增加微批次的数量并不一定会提高效率，需要根据实际情况进行分析，并计算出最优的微批次数量。此外，公式（2）还表明，缩短通信时间、减少阶段数量均有助于降低空泡率。特别是由于通信时间的存在，阶段数量对空泡率的影响更为显著。因此，减少阶段数量可以带来更大的收益。下面我们将从这两个方面介绍相关的技术。

缩短通信时间的核心在于减少通信的数据量。为此，可以采用稀疏化、量化和低秩训练等技术。另外，阶段数量主要受到节点总内存的限制。如果能够降低训练占用的内存，就可以使每个节点容纳更多的参数，从而有可能降低阶段数。需要注意的是，在此处，以增加通信量为代价来降低内存的方案并不适用。

稀疏化的主要思想是，神经网络层的输出中绝对值较大的数值通常承载了更多的信息量。因此，将中间层数据中的大多数数值变为0就不会损失主要信息。对此可以利用稀疏化数据的表示方式来压缩数据，从而减少通信量和存储空间



▲图1 普通流水线并行和跨云流水线并行

的占用。

量化则是将传输的中间结果从原本32位比特的浮点数映射到8位或者更少比特表示的整型数据上。这种方式可以有效压缩通信数据，但是会带来额外的误差，进而会影响到训练的精度。因此，需要根据实际的数据分布情况来设计量化的位数和方式。

大型模型通常存在“过参数化”的问题，即虽然模型的参数众多，但实际上模型主要依赖于低秩维度的内容。为此，可以采用一些基于低秩分解的训练方法，例如低秩适应（LoRA）[4]算法。该方法新增了一个先降维再升维的旁路。这样的设计可以天然地降低中间数据的维度。将降维矩阵的输出位置作为切分点也可以达到减少通信时间的目的。

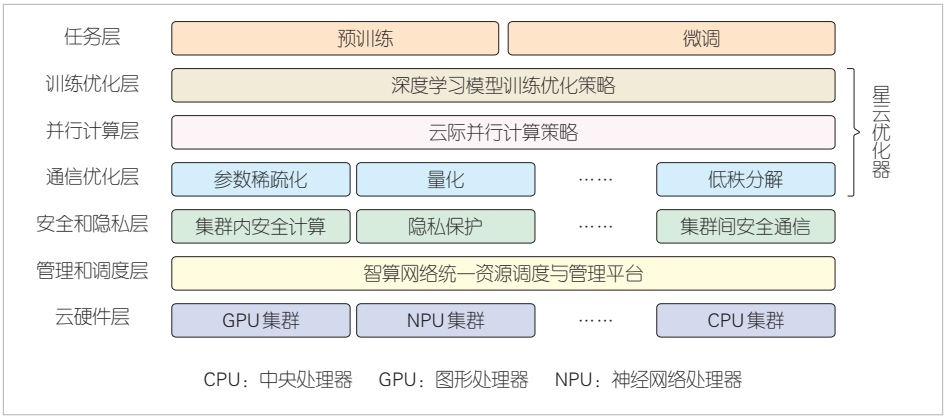
2 一种面向大规模语言模型的跨云训练方法

大规模语言模型的训练任务包括语言模型预训练和下游任务微调两个阶段。为了应对跨云模型训练的挑战，本文中我们将介绍一种基于跨云大模型训练框架“星云”[5]的预训练和微调方法。如图2所示，“星云”是一个专门面向云际环境的深度学习模型统一训练框架，该框架包含了任务层、训练优化层、并行计算层、通信优化层、安全和隐私层、管理和调度层以及云硬件层等7个功能层，支持在低带宽网络环境下，利用不同算力集群的异构算力进行大模型的跨云训练，在通信优化方面采用了参数稀疏化、量化以及低秩分解等有效技术来确保集群间信息传输的轻量化和最小化模型精度损失，并主要采取流水线并行的方式来实现在多个算力集群间的并行计算。

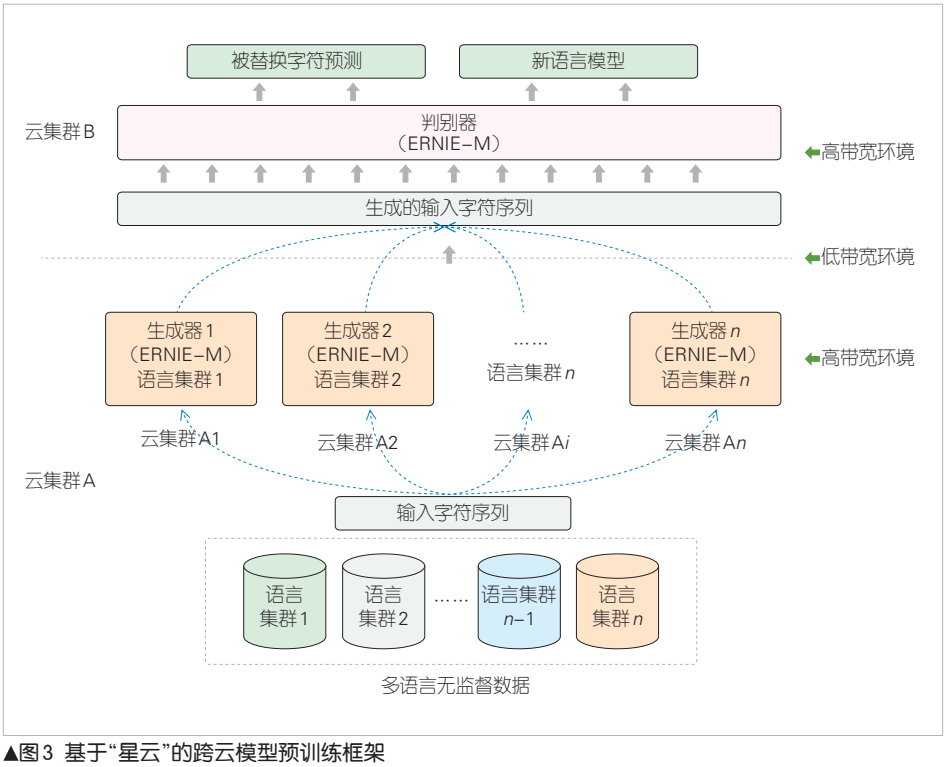
2.1 多语言大模型的跨云预训练方法

针对多语言模型预训练任务，我们基于“星云”实现了一套支持跨云多源数据训练的多语言模型预训练方案，如图3所示。为了优化训练过程，该方案参考ELECTRA[6]架构设计了一种适合跨云使用的模型架构，由生成器（Generator）和判别器（Discriminator）两部分组成。其中，生成器根据输入内容生成对应的字符序列，判别器则对生成的字符序列进行判断，以达到优化训练的目的。

在模型训练过程中，生成器只需要将输出的字符序列单向传递给判别器。当进行跨云训练时，生成器和判别器会被部署在不同的云集群上，此时生成器只需向判别器传输字符串序列即可。在这个过程中，所需的数据传输量较少，带宽需求也较低，这有利于跨云大模型的训练。此外，通过共享生成器和判别器间的词表、跨云只传输字符ID序列的方式



▲图2 “星云”的框架结构示意图



▲图3 基于“星云”的跨云模型预训练框架

不仅可以进一步减少数据传输量，还可以避免数据泄露。

为了支持多源数据多方协同训练，该架构需要使用多个生成器来共同训练判别器。不同的生成器对应不同的训练数据和不同的预训练模型，例如：可以让每个生成器负责一个语种的生成，多个生成器共同支持多语言判别器的训练，这样可以提高训练效率，增强判别器的泛化能力。

在模型训练过程中，生成器和判别器之间只有单向的字符标识序列传输，数据量小，受网络带宽瓶颈影响较小。为了提高集群资源的利用率和训练速度，本文中我们采用了数据并行的方式在生成器集群和判别器集群内部分别进行训练。为了验证该框架在异构算力环境下的模型训练能力，我们将生成器部署在GPU算力集群，将判别器部署在NPU算力集群。该框架的跨云集群部署及并行计算方式如图4所示。这种部署和计算方式可以提高训练效率，优化资源利用率。

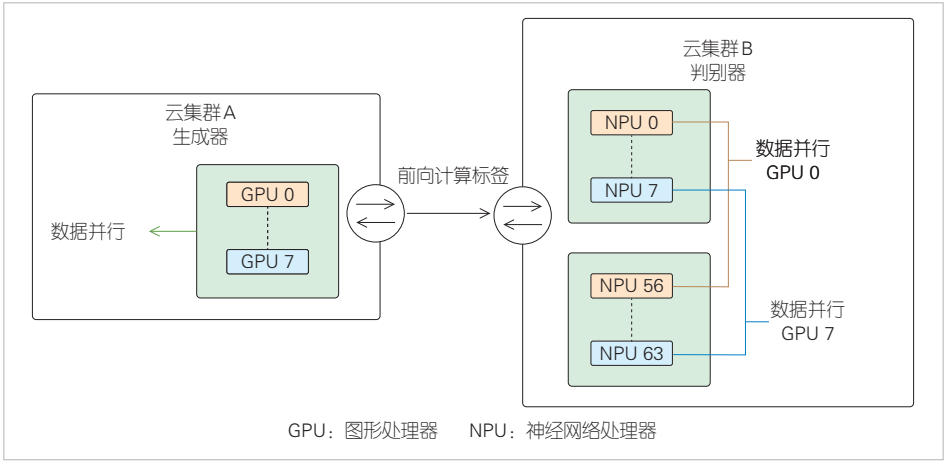
为了测试跨云模型预训练的效果，实验中我们利用包含116种语言的单语数据和15种语言的平行语料数据，进行基于生成器–判别器架构的跨云大模型训练。使用多语言预训练语言模型ERNIE–M–Base来初始化生成器，使用ERNIE–M–Large来初始化判别器，训练得到的判别器ERNIE–M–Extra则作为最终的多语言大模型。为了测试ERNIE–M–Extra模型的多语言能力，本文中我们首先使用英语数据进行微调，然后在15种语言的跨语言推理任务上进行了测试。测试结果如表1所示。

由表1可知，ERNIE–M–Extra模型在15种语言的跨语言推理任务中表现出最优的平均成绩，相比于基础模型ERNIE–M–Large，其精度提高了0.2。

为了测试模型训练过程的吞吐率，我们进行了在云集群内和跨云集群环境下的测试。实验结果显示，跨云训练的吞吐率达到了单云集群训练的85%。在GPU算力集群和NPU算力集群环境下，针对异构环境下硬件加速效果进行了实验，并对比了由8卡NPU算力增加到64卡的模型训练速度。实验结果表明，增加算力卡后训练速度提高了4.34倍。

为了验证模型在跨云集群训练中的有效性，本文对比了单云环境和跨云环境下模型训练的损失曲线，如图5所示。可以看出，跨云集群训练可以保持训练过程的持续收敛。

综上所述，采用生成器–判别器架构进行多语言大模型训练，可以在跨云环境下保持较高的吞吐率，确保训练过程持续收敛。此外，增加算力资源可以有效提高训练速度。



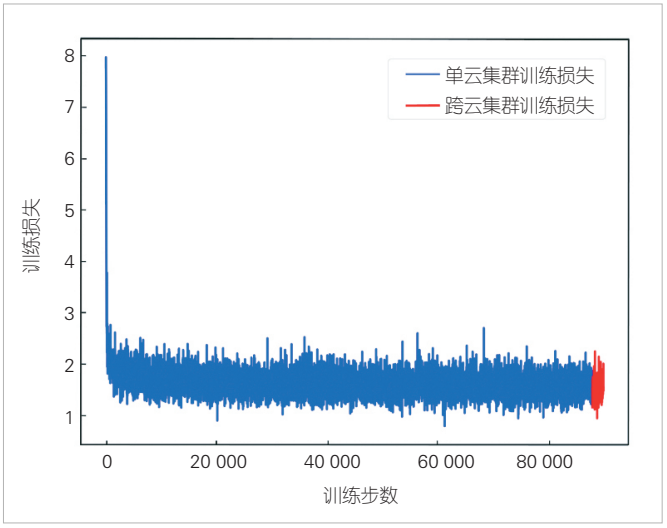
▲图4 跨云预训练集群算力互联及并行计算方式

2.2 大规模语言模型的跨云微调方法

微调是指在预训练大模型的基础上，为了特定的任务进行有针对性的模型训练。本文中我们将分别介绍基于编码器–解码器架构的自然

▼表1 跨云模型预训练最终模型精度对比

模型	En	Fr	Es	De	El	Bg	Ru	Tr	Ar	Vi	Th	Zh	Hi	Sw	Ur	平均
XLM[7]	85.0	78.7	78.9	77.8	76.6	77.4	75.3	72.5	73.1	76.1	73.2	76.5	69.6	68.4	67.3	75.1
Unicoder[8]	85.1	79.0	79.4	77.8	77.2	77.2	76.3	72.8	73.5	76.4	73.6	76.2	69.4	69.7	66.7	75.4
XLM–R[9]	85.8	79.7	80.7	78.7	77.5	79.6	78.1	74.2	73.8	76.5	74.6	76.7	72.4	66.5	68.3	76.2
INFOXLM[10]	86.4	80.6	80.8	78.9	77.8	78.9	77.6	75.6	74.0	77.0	73.7	76.7	72.0	66.4	67.1	76.2
ERNIE–M[11]	85.5	80.1	81.2	79.2	79.1	80.4	78.1	76.8	76.3	78.3	75.8	77.4	72.9	69.5	68.8	77.3
XLM–RLARGE[9]	89.1	84.1	85.1	83.9	82.9	84.0	81.2	79.6	79.8	80.8	78.1	80.2	76.9	73.9	73.8	80.9
INFOXLMLARGE[10]	89.7	84.5	85.5	84.1	83.4	84.2	81.3	80.9	80.4	80.8	78.9	80.9	77.9	74.8	73.7	81.4
VECOLARGE[12]	88.2	79.2	83.1	82.9	81.2	84.2	82.8	76.2	80.3	74.3	77.0	78.4	71.3	80.4	79.1	79.9
ERNIE–MLARGE[11]	89.3	85.1	85.7	84.4	83.7	84.5	82.0	81.2	81.2	81.9	79.2	81.0	78.6	76.2	75.4	82.0
ERNIE–M–Extra	89.4	85.1	86.0	84.5	84.4	84.6	81.8	81.7	81.8	81.9	79.3	81.2	79.1	76.3	75.7	82.2



▲图5 单云训练和跨云训练损失对比

语言生成微调训练和基于编码器架构的自然语言理解微调训练。

2.2.1 针对自然语言生成任务的微调

针对基于编码器–解码器架构的自然语言生成模型，本文以机器翻译任务为例，参照ABNet[13]模型架构设计，实现基于“星云”的跨云机器翻译模型微调训练。ABNet是一种用于微调训练的模型架构，在编码器和解码器的各个子层之间插入需要训练的适配器模块。在训练过程中，预训练模型的参数被冻结。该微调方法利用预训练语言模型的知识，但不调整预训练模型的参数。如图6所示，针对源语言和目标语言的预训练模型分别被部署在两个云集群中。

在模型训练时，每进行一步前向计算和反向传播，编码端和解码端都需要进行一次跨云中间数据传输。数据传输量与数据批处理大小（B）、序列长度（S）、隐藏层维度（H）等因素相关。需要传递的数据规模如公式（3）所示：

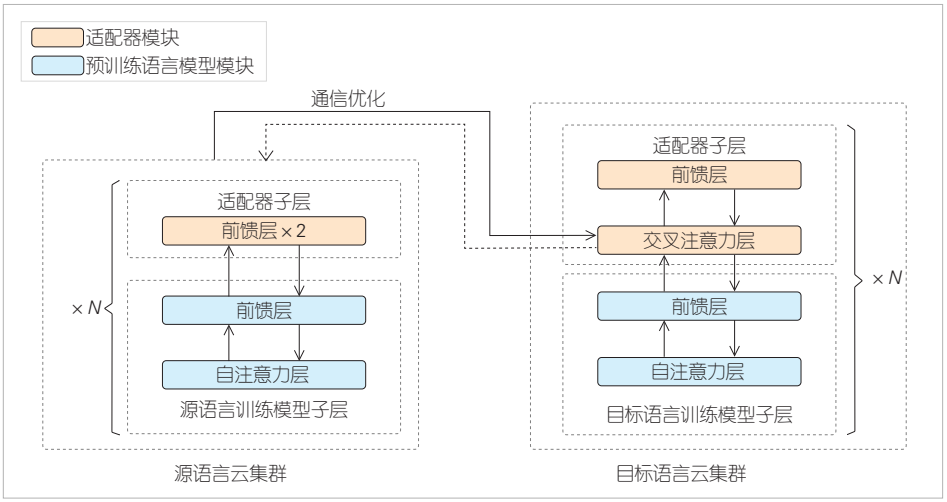
$$Data\ size = B \times S \times H。 \quad (3)$$

在微调训练过程中，数据传输占用了大量的网络带宽资源。传输时间的长短对训练速度的影响很大。当网络带宽过低时，跨云训练就无法达到加速训练的目的。因此，为了提高模型的训练速度，“星云”框架从云间通信和并行训练两个方面进行综合优化。

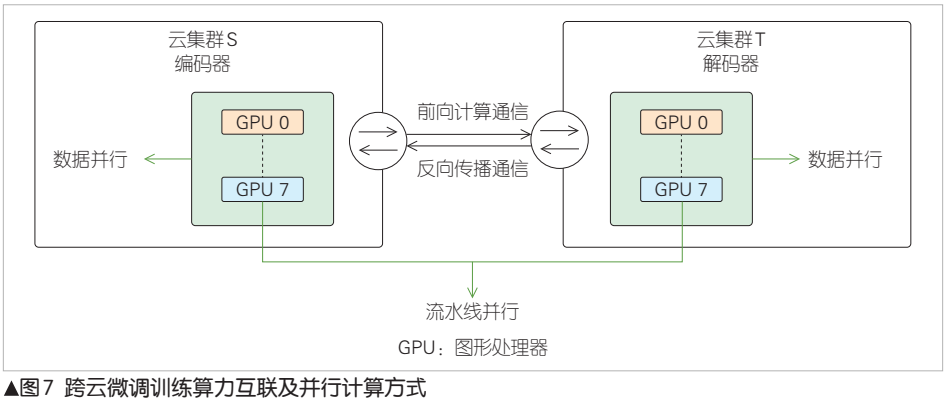
为了解决在训练过程中数据传输量大、传输时间长的问题，针对需要跨云传输的中间数据，可以采用压缩通信的策略进行优化，以减少单次传输的数据量。可采用的压缩通信方法主要包括量化、稀疏化、低秩分解等。为了减小压缩通信对模型精度的影响，可以组合使用不同的压缩策略，并在训练的不同阶段采用不同的压缩传输策略。

为了解决在模型训练过程中由串行计算导致的资源利用率不高的问题，“星云”采用并行优化策略来优化训练过程。在云集群间采用流水线并行，云集群内采用数据并行的方式，采用多微批次以流水线并行的方式在云集群间执行计算和数据传输任务，可以减少同一时刻资源的停等，提高参与训练各资源的利用率。ABNet架构在跨云环境的部署及并行计算方式如图7所示。

为了进行跨云集群模型微调的实验，我们选择IWSLT’



▲图6 基于ABNet的跨云微调训练方法



▲图7 跨云微调训练算力互联及并行计算方式

14的西班牙语（Es）到英语（En）的机器翻译任务，并采用ABNet跨云架构基于预训练语言模型进行微调训练。在该实验中，我们使用多语言预训练模型ERNIE-M-base-cased作为编码端，使用英文预训练模型BERT-Base作为解码端，并将它们分别部署在两个配备了8张NVIDIA V100 GPU显卡的云集群上。

实验结果显示，完全重新训练的Transformer-Base模型[14]的双语评估替换（BLEU）值[15]为39.60，在本地微调训练的ABNet-Local模型为43.19，采用跨云微调训练的ABNet-Cloud模型为41.92。实验结果表明，采用基于预训练模型微调的翻译模型性能优于仅使用训练数据重新训练的Transformer-Base模型。相对于仅在本地集群训练的ABNet-Local模型，跨云微调的ABNet-Cloud模型的BLEU值降低了1.27个，这是由于压缩通信导致了模型精度损失。然而，相对于Transformer-Base模型，ABNet-Cloud仍然提高了2.32个BLEU值。这表明在跨云环境中，基于预训练语言模型进行微调训练可以复用预训练模型的知识，从而提高最终翻译模型的精度。

为了研究压缩通信策略对模型训练的影响，我们对不同压缩通信策略下的模型训练速度和最终模型精度进行了对比。其中，前向计算数据传输采用FP16半精度及其与不同压缩率的SVD分解的组合，反向传播采用固定的INT8量化压缩。实验结果如表2所示，压缩率越高，模型训练速度越快。在FP16(SVD(0.2))+INT8的压缩策略下，模型训练单步消耗时间仅为不压缩训练的19%。然而，该策略下模型精度损失了4.19个BLEU值。在所验证的压缩策略中，FP16(SVD(0.6))+INT8策略下得到的模型精度最佳（达到41.92），单步训练时间仅为不压缩的32%，训练速度提升了3倍以上。

▼表2 不同压缩通信方法性能对比

压缩方法	BLEU	训练速度(s/步)
ABNet-Local	43.19	4.42
FP16+INT8	38.82	1.60
FP16(SVD(0.8))+INT8	41.15	1.50
FP16(SVD(0.6))+INT8	41.92	1.42
FP16(SVD(0.4))+INT8	39.56	0.94
FP16(SVD(0.2))+INT8	39.00	0.86

BLEU：双语评估替换　SVD：奇异值分解
FP16：半精度　INT8：8比特量化

2.2.2 针对自然语言理解任务的微调

自然语言理解包括文本分类、文本蕴含、阅读理解等任务。通常人们采用基于编码器类型的预训练模型进行微调训练。为了在跨云环境下微调这类模型，可以采用低秩结构的思想对通信数据进行压缩[16]。具体的做法如下：

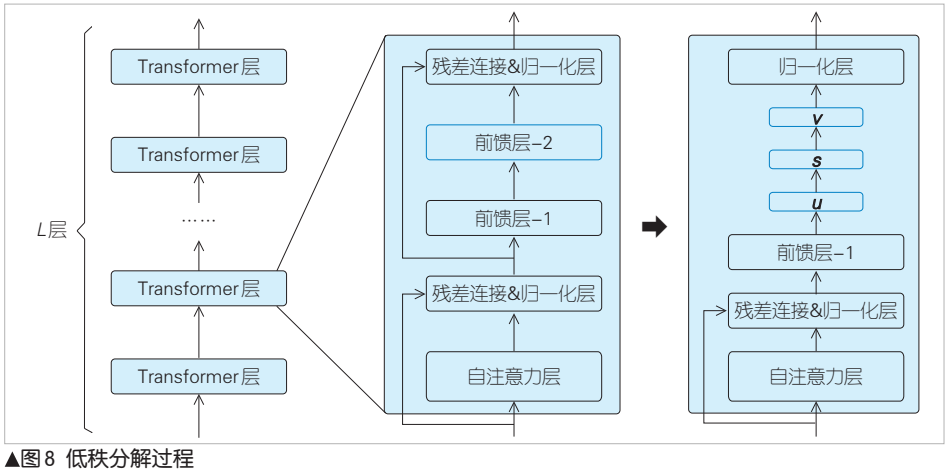
1）对于模型中的每一个Transformer块，假设其输入和输出矩阵的维度为$\boldsymbol{R}^{b\times d}$，即在跨云训练时，通信数据的维度也为$\boldsymbol{R}^{b\times d}$。其中，$b$表示batch_size，$d$表示模型的维度参数。

2）对于其中一个Transformer块的线性层，可以进行奇异值分解来降低通信数据的维度。具体做法是：将该线性层的权重矩阵$\boldsymbol{W}\in\boldsymbol{R}^{m\times d}$进行奇异值分解，选取前$r$个奇异值，得到3个矩阵$\boldsymbol{u}$、$\boldsymbol{s}$和$\boldsymbol{v}$，维度分别为$\boldsymbol{R}^{m\times r}$、$\boldsymbol{R}^{r\times r}$和$\boldsymbol{R}^{r\times d}$；然后，使用3个连续的线性层来替代原始的线性层，这3个线性层的权重分别为U、S和V，如图8所示。

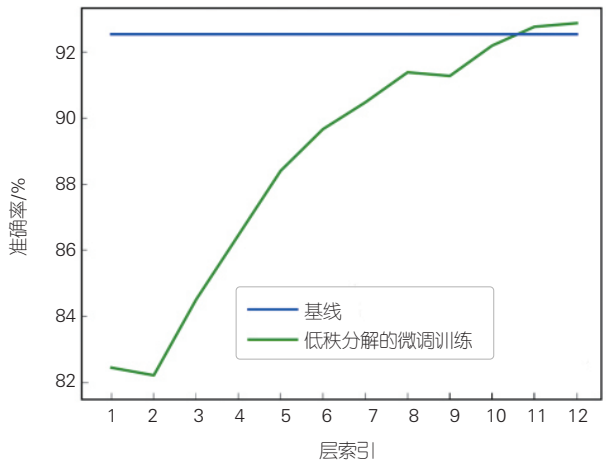
3）将用于跨云通信的模型拆分点设置在S和V层之间，并移除该Transformer块的直接连接分支。这样，通信数据的维度会变成$\boldsymbol{R}^{b\times r}$，即原有数据的$r/d$倍。

根据上述的压缩方案，以BERT-Base为基础模型，在GLUE数据集[17]和SQuAD数据集[18]上进行跨云微调训练，并分析该算法在不同层索引上对训练精度的影响。将上述算法中的r设置为8，实验结果如图9所示。其中，横轴表示拆分的层级索引，纵轴表示准确率。需要说明的是，由于各个数据集表现出的规律一致，这里仅以SST-2和QNLI数据集为代表。

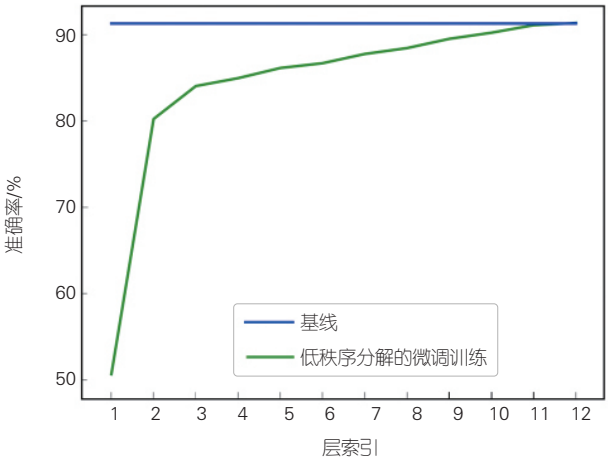
由图9可知，r值较小且拆分位置处于模型的底层会导致训练精度显著下降。但是，当拆分位置位于模型的高层时，r值的大小对训练精度没有影响。在实验中，我们选择



▲图8 低秩分解过程



（a）SST-2



（b）QNLI

注：QNLI、SST-2是两种数据集

▲图9 基于低秩分解的跨云微调

将模型拆分在第11层，然后针对不同的r值（分别为8、16和32）进行测试，结果如表3所示。特别地，在r等于8的情况下，传输数据量降为原有的1/96，同时精度维持在原有模型的相当水平。

通过跨云场景的模型微调训练实验验证，我们证实了跨云微调的可行性。用户可以利用分布在不同云集群上的预训练模型来微调目标任务模型，并通过复用已有模型的知识来提升模型性能。这比仅使用自身数据训练模型更为优越。由于模型被拆分成多个部分，用户可以将模型的底层部分置于可信集群上，从而确保其他集群无法获得标注数据，保障用户标注数据的安全性。

3 跨云训练算力互联及未来场景

生成算法、预训练模型、多模态等技术的融合催生了以ChatGPT为代表的人工智能生成内容（AIGC）的爆发，进而带来了高算力需求。以ChatGPT为例，它使用了10 000块A100 GPU进行训练。此外，它的部署成本也很高，根据国盛证券报告估算，它的每日咨询量对应的算力需求达到了上万块A100。所以，利用跨云训练可以将广泛分布的算力结合起来，这是应对大模型对算力高需求的一种解决方案，从而有效应对算力对大模型训练的制约。同时，跨云训练可以利用闲散算力，有效解决碎片化问题，提高云集群资源的利用率。

除了算力限制，与个人信息强相关的应用，例如语音助手、心理咨询等，也关注隐私保护问题。跨云训练机制具备较好的隐私保护能力。用户可以通过构建本地设备与云的协同训练来实现个人信息在本地处理、云端提供算力的方式，从而保证个人信息不被泄露。

4 结束语

本文的研究表明，在跨云环境下进行大规模语言模型训练是可行的，是一种提高算力利用率的方案。通过采用模型分割、拆分学习、跨云协同、压缩通信和模型复用等关键技术，该方案能够有效解决跨云训练过程中可能出现的算力和数据不足的问题，并提高训练速度和效率。这些技术在自然语言处理领域的应用将有望带来更为精准和高效的文本处理和语义分析结果，并具备较好的隐私保护能力，为智能化应用和人机交互等领域的发展提供有力的支持。

致谢

感谢百度飞桨团队吴志华和巩伟宝，以及哈尔滨工业大

▼表3 11层拆分微调结果（k表示1 000）

	SST-2（67k）	QNLI（105k）	MNLI（364k）	QQP（91.2k）	CoLA（8.5k）	RTE（2.5k）	STS-B（7k）	MRPC（3.7k）	SQuAD（88k）
基线模型	92.54	91.24	84.56	90.73	55.3	66.06	88.38	85.33	88.25
r=8	92.43	90.98	83.98	90.93	57.13	64.25	86.46	84.81	88.33
r=16	92.31	91.22	84.33	90.75	57.35	62.09	86.78	83.47	88.75
r=32	92.77	91.04	84.27	90.99	57.87	62.81	87.46	84.23	88.56

学(深圳)施少怀教授对本文写作提供的帮助!

参考文献

[1] DEVLIN J, CHANG M W, LEE K, et al. BERT: pre-training of deep bidirectional transformers for language understanding [EB/OL]. [2023-06-08]. https://arxiv.org/abs/1810.04805
[2] BROWN T B, MANN B, RYDER N, et al. Language models are few-shot learners [EB/OL]. [2023-06-08]. https://arxiv.org/abs/2005.14165
[3] HUANG Y P, CHENG Y L, CHEN D H, et al. GPipe: efficient training of giant neural networks using pipeline parallelism [EB/OL]. [2023-06-08]. https://arxiv.org/abs/1811.06965
[4] HU E J, SHEN Y L, WALLIS P, et al. LoRA: low-rank adaptation of large language models [EB/OL]. [2023-06-08]. https://arxiv.org/abs/2106.09685
[5] XIANG Y, WU Z H, GONG W B, et al. Nebula-I: a general framework for collaboratively training deep learning models on low-bandwidth cloud clusters [EB/OL]. [2023-06-08]. https://arxiv.org/abs/2205.09470
[6] CLARK K, LUONG M T, LE Q V, et al. ELECTRA: pre-training text encoders as discriminators rather than generators [EB/OL]. [2023-06-08]. https://arxiv.org/abs/2003.10555
[7] LAMPLE G, CONNEAU A. Cross-lingual language model pretraining [EB/OL]. [2023-06-08]. https://arxiv.org/abs/1901.07291
[8] HUANG H Y, LIANG Y B, DUAN N, et al. Unicoder: a universal language encoder by pre-training with multiple cross-lingual tasks [EB/OL]. [2023-06-08]. https://arxiv.org/abs/1909.00964
[9] CONNEAU A, KHANDELWAL K, GOYAL N, et al. Unsupervised cross-lingual representation learning at scale [EB/OL]. [2023-06-08]. https://arxiv.org/abs/1911.02116
[10] CHI Z W, DONG L, WEI F R, et al. InfoXLM: an information-theoretic framework for cross-lingual language model pre-training [EB/OL]. [2023-06-08]. https://arxiv.org/abs/2007.07834
[11] OUYANG X, WANG S H, PANG C, et al. ERNIE-M: enhanced multilingual representation by aligning cross-lingual semantics with monolingual corpora [EB/OL]. [2023-06-08]. https://arxiv.org/abs/2012.15674
[12] LUO F L, WANG W, LIU J H, et al. VECO: variable and flexible cross-lingual pre-training for language understanding and generation [EB/OL]. [2023-06-08]. https://arxiv.org/abs/2010.16046
[13] GUO J L, ZHANG Z R, XU L L, et al. Incorporating BERT into parallel sequence decoding with adapters [C]//Proceedings of the 34th International Conference on Neural Information Processing Systems. ACM, 2020: 10843 - 10854. DOI: 10.5555/3495724.3496634
[14] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all You need [C]// Proceedings of the 31st International Conference on Neural Information Processing Systems. ACM, 2017: 6000 - 6010. DOI: 10.5555/3295222.3295349
[15] PAPINENI K, ROUKOS S, WARD T, et al. BLEU: a method for automatic evaluation of machine translation [C]//Proceedings of the 40th Annual Meeting on Association for Computational Linguistics - ACL '02. Association for Computational Linguistics, 2001: 311-318. DOI: 10.3115/1073083.1073135
[16] SHI S H, YANG Q, XIANG Y, et al. An efficient split fine-tuning framework for edge and cloud collaborative learning [EB/OL]. [2023-06-08]. https://arxiv.org/abs/2211.16703
[17] WANG A, SINGH A, MICHAEL J, et al. GLUE: a multi-task benchmark and analysis platform for natural language understanding [C]//Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP. Association for Computational Linguistics, 2018: 353 - 355. DOI: 10.18653/v1/w18-5446
[18] RAJPURKAR P, ZHANG J, LOPYREV K, et al. SQuAD: 100, 000+ questions for machine comprehension of text [C]//Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2016: 2383-2392. DOI: 10.18653/v1/d16-1264

作 者 简 介

潘囿丞，鹏城实验室网络智能部云计算所在站博士后；主要研究方向为自然语言处理、文本生成、机器翻译等；发表论文8篇，获授权国家发明专利1项。

侯永帅，鹏城实验室网络智能部云计算所工程师；主要从事智能问答、机器翻译、云际协同学习等方面的研究；发表论文15篇，获授权国家发明专利2项。

杨卿，鹏城实验室网络智能部云计算所工程师；主要从事自然语言处理、协同计算等方面的研究。

余跃，鹏城实验室人工智能开源技术总师、AITISA联盟智算中心和智算网络标准工作组联合组长、算力网络推进组组长；主要从事智能计算、云计算、开源软件等相关领域的研究工作；作为技术负责人负责AITISA联盟智能计算中心与算力网相关标准体系的制定与开源平台研发；发表论文50余篇。

相洋，鹏城实验室网络智能部云计算所副所长、深圳“孔雀计划”海外高层次人才；主要研究方向为自然语言处理、人工智能、大模型、云计算等；主持两项国家级科研项目，参与多项重大科研攻关项目；获深圳市科技进步二等奖1项；发表论文80余篇。

面向新型智能计算中心的全调度以太网技术

Global Scheduling Ethernet for New Intelligent Computing Center

段晓东/DUAN Xiaodong，程伟强/CHENG Weiqiang，
王瑞雪/WANG Ruixue，王雯萱/WANG Wenxuan

（中国移动通信有限公司研究院，中国 北京 100053）
(The Research Institution of China Mobile, Beijing 100053, China)

DOI：10.12142/ZTETJ.202304011
网络出版地址：http://kns.cnki.net/kcms/detail/34.1228.TN.20230724.1702.010.html
网络出版日期：2023-07-25
收稿日期：2023-06-08

摘要：智能计算中心网络作为智能计算中心的连接底座，需要具备高性能、低时延的通信能力。一旦网络性能不佳，就会严重影响分布式训练的效果。智能计算中心网络体系是一个多要素融合的复杂系统，依赖于智能计算业务、网络设备、交换芯片、网卡、仪表等上下游产业协同创新。提出一种新型全调度以太网（GSE）技术架构，在最大限度地兼容以太网生态链的前提下，基于报文容器（PKTC）转发、负载均衡机制以及基于报文容器的动态全局调度队列（DGSQ）全局调度技术，构建超大规模、超高带宽、超低时延、超高可靠的智能计算中心网络，助力AI产业发展。

关键词：人工智能生成内容；智算中心网络；全调度以太网；报文容器；动态全局调度队列

Abstract: The intelligent computing center network, as the connection base of the intelligent computing center, needs to have high performance and low latency communication capability. Once the network performance is poor, it can seriously affect the effect of distributed training. An intelligent computing center network system is a multi-element integration of complex systems, relying on intelligent computing services, network equipment, switching chips, network cards, instruments, and other upstream and downstream industry collaborative innovation. A new global scheduling Ethernet (GSE) technology architecture is proposed to build an ultra-large scale, ultra-high bandwidth, ultra-low latency, and ultra-reliable intelligent computing center network with maximum compatibility with the Ethernet ecosystem, innovative forwarding and load balancing mechanism based on packet containers (PKTC) and dynamic global scheduling queue (DGSQ) global scheduling technology based on packet containers to help the development of AI industry.

Keywords: artificial intelligence-generated content; intelligent computing center network; GSE; packet container; dynamic global scheduling queue

引用格式：段晓东, 程伟强, 王瑞雪, 等. 面向新型智能计算中心的全调度以太网技术 [J]. 中兴通讯技术, 2023, 29(4): 57-63. DOI: 10.12142/ZTETJ.202304011

Citation: DUAN X D, CHENG W Q, WANG R X, et al. Global scheduling Ethernet for new intelligent computing center [J]. ZTE technology journal, 2023, 29(4): 57-63. DOI: 10.12142/ZTETJ.202304011

1 AI业务与智能计算中心产业的发展

1.1 AI业务发展趋势

人工智能（AI）业务发展经历了漫长的历程。20世纪50年代，人们开始尝试模拟人脑的神经网络来解决计算机视觉和语音识别的问题。但由于当时无法解决神经网络计算复杂度高和可解释性差的问题，AI技术进入了“寒冬”。2012—2017年，Hinton等提出卷积神经网络，大大推动计算机视觉和深度学习的发展。同时，基于深度学习的AlphaGo战胜围棋世界冠军，进一步点燃人们在深度学习领域探索的热情与信心。2017—2022年，基于大型神经网络的Transformer架构出现，该模型可以更好地捕捉序列之间的依赖关系，开启了基于深度学习的AI新时代。2022年11月，OpenAI公司开发的大规模智能语言模型ChatGPT横空出世。ChatGPT结合了GPT-3.5和GPT-4系列的大型语言模型，展现出惊人的语言能力[1]。该模型深入各个领域，在引爆全球科技领域的同时，推动AI产业全面进入大模型时代。因此，ChatGPT的出现具有跨时代的意义。

近年来，随着算力经济的高速发展[2]，AI业务在自动驾驶、语音识别和自然语言处理等领域取得了许多重大成就，并涌现出人工智能即服务（AIaaS）和模型即服务（MaaS）两种新型服务模式。当前，教育、医疗、智慧城市和智能制造等行业迫切需要AI赋能，例如：华为云、百度云、阿里云和腾讯云等提供AIaaS的企业均为用户提供高品质的人工

智能服务。MaaS拥有经过大量数据集训练和优化的模型，可为用户提供图像识别、自然语言处理、预测分析和欺诈检测等服务。

为推动AI业务的发展，中国陆续给予政策方面的扶持和激励，特别是东数西算工程的全面启动，给AI大模型在智能计算（后文简称为“智算”）中心的快速发展注入强大的助推剂[3]。AI大模型的参数量呈指数级增长，有力地驱动了“大模型”向“超大模型”演进。与此同时，智算规模和智算需求也呈指数级增长。预计截至2030年，智算占比将达到70%，AI技术将广泛落地，中国将迎来智算中心建设的热潮。

1.2 智算中心产业发展趋势

为加速智能经济发展和产业数字化转型，智算中心作为一种新的关键性信息基础设施进入公众视野。智算中心既不同于超算中心，也不同于互联网企业和运营商的云计算中心。智算中心既要借鉴超算中心分布式集群计算架构，以支持超大规模、复杂度高及多样性的数据处理，又要参照云计算服务模式，采用统一的架构和统一的应用程序编程接口（API），以屏蔽底层技术细节，降低使用门槛，向不同行业提供普适且灵活多样的智算服务。

随着业内领军企业竞相推出千亿、万亿级参数量的大模型，以图形处理器（GPU）、神经网络处理器（NPU）为代表的AI算力设施迅猛发展，使得智算中心底层GPU算力部署规模达到万卡级别。基于数据并行、模型并行的分布式训练成为处理超大模型和超大数据集的关键手段。智算中心集群算力与GPU算力、节点数量、线性加速比、有效运行时间等呈正相关，需要计算、存储和网络资源的协同设计，具体表现在以下几个方面：在计算方面，单机算力无法支撑海量训练数据，需要将计算任务切分到单机级别，以并行计算的集群架构方式提供算力服务；在存储方面，为突破计算节点中大量密集数据存取带来的算力瓶颈，搭建机械硬盘（HDD）、固态硬盘（SSD）、存储类内存（SCM）等异构存储集群，以降低数据访问时延；在网络方面，构建连接中央处理器（CPU）、GPU、存储等异构算力资源的总线级、高性能无阻塞交换网络，以提升网络通信性能和稳定性；在机房建设方面，提前规划“风火水电”等基建设施，引入液冷系统，实现低电源使用效率（PUE）数据中心的高能效利用。由此可见，传统智算中心正在向新型智算中心演进。

面向智能计算业务的发展，新型智算中心围绕“算、存、网、管、效”五大核心技术全面升级，以提升GPU集群算力，打造多元融合存储，构建高速无损网络，管控异构算力池化，以高效节能控制为目标，构建标准统一、技术领先、兼容开放的智算底座。

2 智算中心网络演进趋势与挑战

2.1 智算中心网络关键特征

随着GPU高速发展和算力需求的激增，算力中心正向集约化方向发展，数据中心从“云化时代”转向“算力时代”。在传统云数据中心中，传统的计算处理任务或离线大数据计算任务以服务器或虚拟机（VM）为池化对象，网络负责提供服务器或VM之间的连接，并聚焦业务部署效率及网络自动化能力；而智算中心是服务于人工智能的数据计算中心，以GPU等AI训练芯片为主，并以提升单位时间、单位能耗下的运算能力和质量为核心诉求，为AI计算提供更大的计算规模和更快的计算速度。传统数据中心通过CPU来执行计算任务，且网络带宽需求为10~100 Gbit/s，并通过使用传输控制协议（TCP）来完成内存数据的读取；而智算中心网络主要用于承载AI训练业务，其GPU算力与CPU相比拥有更高的计算性能，且网络带宽需求为100~400 Gbit/s（甚至达到800 Gbit/s），并可以通过远程直接内存访问（RDMA）来减少传输时延。由于RDMA网络对于丢包异常敏感，0.01%的丢包率就会使RDMA吞吐率变为0，因此大模型训练的智算中心网络需要缩短迭代过程中通信传输数据的时间，降低通信开销，从而减少GPU的计算等待，提升计算效率。综上所述，零丢包、大带宽、低时延、高可靠是智算中心网络最为关键的特征。

2.2 智算中心网络面临的挑战

与传统数据中心不同，智算中心主要用于承载AI模型训练业务，其通信流量主要具备周期性、流量大、同步突发等特点。在大模型训练过程中，通信具有非常强的周期性，且每轮迭代的通信模式保持一致。在每一轮的迭代过程中，不同节点间的流量保持同步，同时流量以on-off的模式突发式传输。以上通信流量的特点对智算中心网络提出了3个需求：

1）高接入带宽是基础。大模型训练对带宽比较敏感。网络对通信影响最大的是序列化时延，网络通信质量主要取决于有效带宽。但由于网络交换的时间占比不高，静态时延对模型训练效率影响不大。

2）网络级负载均衡是关键。保证通信的有效带宽是模型训练的关键因素之一。负载均衡技术是保证有效带宽的关键。集合操作通信的完成时间由最慢节点的完成时间决定。

在无阻塞网络中，若链路负载不均衡，则会导致冲突流有效带宽下降，冲突流的序列化时间增加。

3）高健壮网络是保障。网络持续高可用、故障业务无中断是分布式系统运行的基础。在高健壮网络中，链路故障时网络会达到亚毫秒级的自动收敛，降低了网络故障对网络拥塞的影响。

如今，基于融合以太网承载远程直接内存访问（RoCE）协议的智算中心网络，通常采用五元组哈希实现链路负载分担技术，以及基于优先级的流量控制（PFC）、显式拥塞通告（ECN）协议实现网络无损，该方案对智算中心网络提出4个挑战：

挑战1：传统基于逐流的等价多路径路由（ECMP）负载均衡技术在流量数小的情况下会失效，导致流量在交换网络发生极化，链路负载不均。当智算中心网络中存在大象流时，很容易发生多个流被散列到相同的路径上的情况，从而导致链路过载，造成某个物理链路负载过大，甚至会出现拥塞而导致报文丢弃。

挑战2：随着网络规模的不断提升，报文交换方式由单网络节点内实现到单网络节点间多跳实现转变，各节点间也从松耦合关系变化为联合转发。业界通过Clos架构搭建大规模分布式转发结构来满足日益增长的转发规模需求。在该架构下，各节点分布式运行和自我决策转发路径导致无法完全感知全局信息和实现最优的整网性能。

挑战3：当前流量进入网络时，在不考虑出端口转发能力的情况下，流量会以“推”的方式进入网络。分布式训练的多对一通信模型产生大量In-cast流量，造成设备内部队列缓存的瞬时突发而导致拥塞甚至丢包，造成应用时延的增加和吞吐的下降。PFC和ECN都是拥塞产生后的事后干预的被动拥塞控制机制，它们无法从根本上避免拥塞。

挑战4：AI训练网络是一个封闭的专用网络，针对训练效率，通过Underlay直接承载AI训练任务，不再划分Overlay平面，使传统SDN能力失效。同时，传统的智能流分析技术已无法满足高性能无损网络隐患识别、故障预测和闭环等运维可视化需求。

2.3 智算中心网络的演进趋势

综合当前所面临的挑战，未来智算中心网络将向3个方向进行演进：一是从“流”分发到“包”分发演进，即通过提供逐报文容器动态负载均衡机制，消除哈希极化问题，实现单流多路径负载分担，提升有效带宽，降低长尾时延；二是从“局部”决策到“全局”调度演进，即实现全局视野的转发调度机制，并实现集中式管理运维、分布式控制转发，提高网络可用性；三是从“推”流到“拉”流演进，即从被动拥塞控制向依赖“授权请求”和“响应机制”的主动流控转变，最大限度地避免网络拥塞产生，同时需要引入全局集中式管理系统，提升网络自动化及可视化能力。

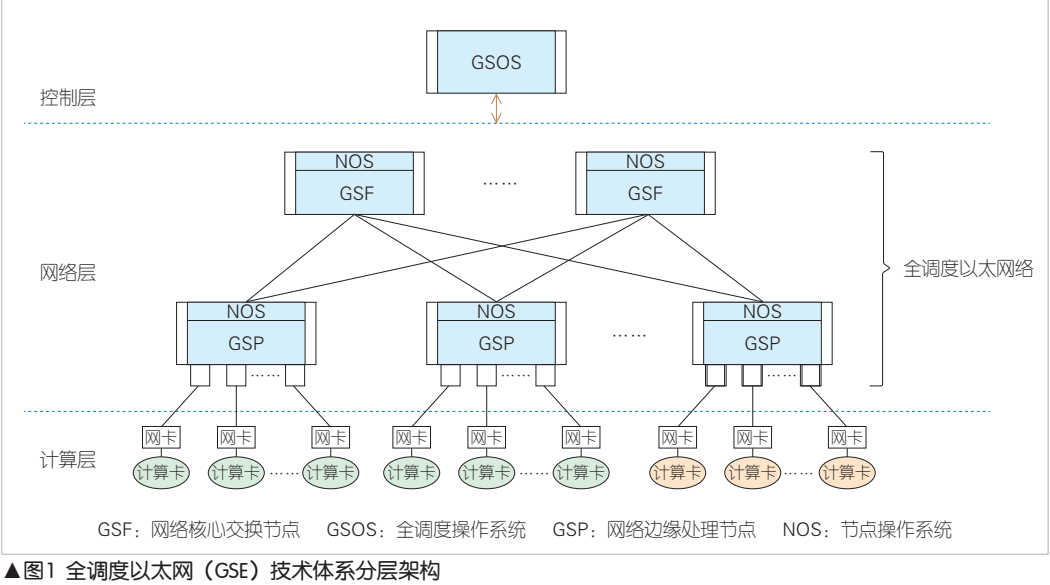
基于以上面向未来智算中心的三大演进方向，我们创新性地提出一种全调度以太网（GSE）技术方案，打造无阻塞、高带宽、低时延、自动化的新型智算中心网络，助力AIGC等高性能业务快速发展[4]。

3 新型GSE架构体系

3.1 GSE架构介绍

为打造无阻塞、高带宽、低时延的高性能网络，GSE架构应运而生，如图1所示。该架构主要包括计算层、网络层和控制层3个层级，包含计算节点、网络边缘处理节点（GSP）、网络核心交换节点（GSF）及全调度操作系统（GSOS）4类设备[4]。

1）控制层：包含全局集中式GSOS，以及GSP和GSF设



▲图1 全调度以太网（GSE）技术体系分层架构

备端分布式节点操作系统（NOS）。其中，集中式GSOS用于提供网络全局信息，实现基于全局信息编址、日常运维管理等功能；设备端NOS可实现动态负载均衡、动态全局调度队列（DGSQ）调度等分布式网络管控功能。

2）网络层：GSE网络主要实现GSP和GSF协同，构建出具备全局流量调度、链路负载均衡、流量精细反压等技术融合的交换网络。其中，Fabric部分可支持二层GSF扩展，以满足更大规模组网需求。

3）计算层：即GSE网络服务层，包含高性能计算卡（GPU或CPU）及网卡，初期将计算节点作为全调度以太网边界，仅通过优化交换网络能力提升计算集群训练性能。未来计算将与网络深度融合，以进一步提升高性能计算能力。

GSE 3层架构涉及计算节点、GSP、GSF及GSOS 4类设备，各设备分工如下：

1）计算节点：即服务器侧的计算卡、网卡，提供高性能计算能力。

2）GSP：即网络边缘处理节点，用以接入计算流量，并对流量做全局调度；流量上行时具备动态负载均衡能力，流量下行时具备流量排序能力。

3）GSF：即网络核心交换节点，作为GSP的上一层级设备，用于灵活扩展网络规模，具备动态负载均衡能力，以及反压信息发布能力。

4）GSOS：即全调度操作系统，提供整网管控的集中式网络操作系统能力。

3.2 GSE技术三大核心理念

3.2.1 基于报文容器的转发及负载分担机制

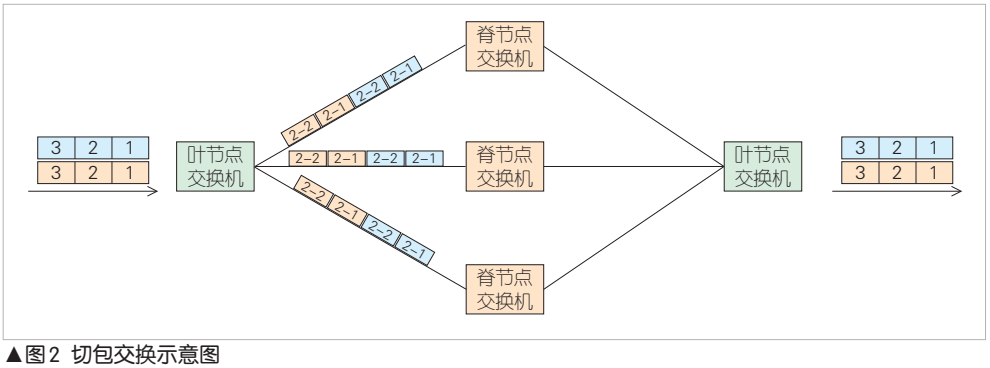
智算中心网络通常采用胖树（Fat-Tree）架构，任意出入端口之间存在多条等价转发路径。与云数据中心业务流量不同，智算业务流量具有“数量少，单流大”的特点。传统以太网逐流负载分担方式导致链路利用率不均，从而引起网络拥塞。单流多路径是提升智算中心网络有效带宽、避免网络拥塞的关键技术手段。业界传统网络中实现单流多路径的技术方案包括切包交换、逐包交换和组包交换。

1）切包交换：核心思想是在网络入口将数据包逻辑切分成若干个信元，将属于同一个数据包的信元调度到不同路径进行传输，在网络出口侧对信元进行排序及重组，如图2所示。该方式可充分利用多路径交换能力，最大程度实现链路负载均衡。但在高带宽演进趋势下，由于被切分后的信元长度短，信元头部开销带来较多的带宽损耗，且极高的均衡调度频率对硬件有较高的要求。

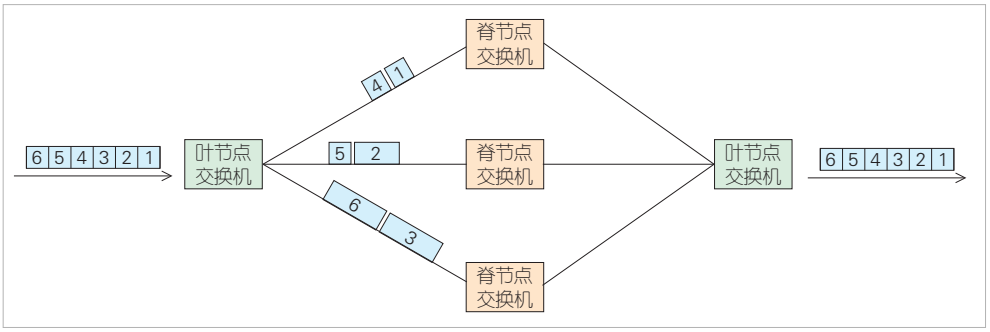
2）逐包交换：核心思想是不对数据包进行处理，直接通过轮询或权重等机制将数据包发往不同路径进行传输，在网络出口侧对报文进行排序，如图3所示。该方式不存在额外的报文开销，也无需高频的均衡调度周期。但由于数据包长度分布连续，难以准确根据已发往每条路径的数据包总数据量来实现均衡负载，链路负载均衡性差，易受微突发影响导致网络拥塞甚至丢包。

3）组包交换：核心思想是将数据包组装成定长且长度较长的数据帧，并为数据帧添加帧头（用于组装和还原）。当数据包不足以填充一个大帧时，就需要填充冗余数据成帧，并利用网络各节点对大数据帧进行存储转发，如图4所示。该方式下大帧均衡调度的周期短，可适应高带宽的转发需求。但帧头及冗余数据填充及存储转发机制会带来一定程度的带宽和时延损耗。

基于上述分析，面向后续智算中心高带宽、低时延的网



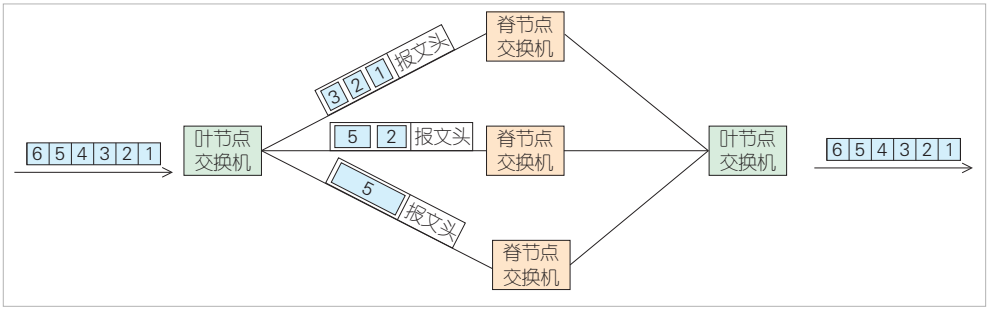
▲图2 切包交换示意图



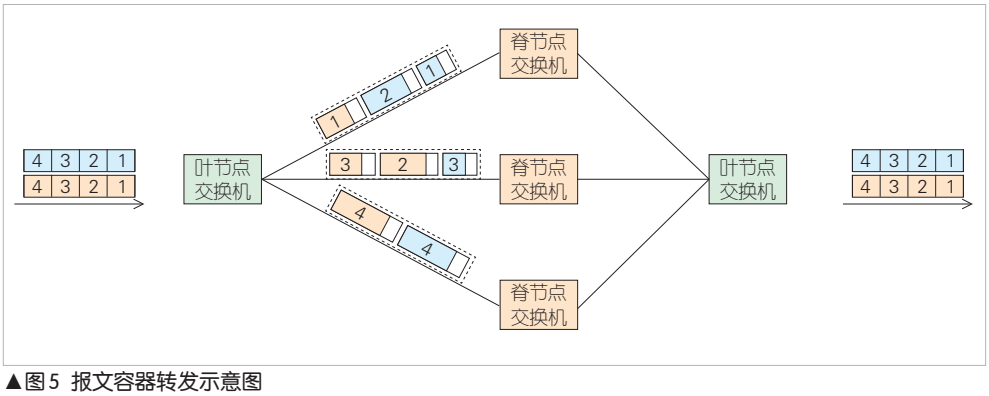
▲图3 逐包交换示意图

络需求，并结合逐包交换方式下即来即转的低时延特性、组包交换方式下的高带宽特性，本文在GSE技术架构中提出一种基于报文容器（PKTC）的转发及负载分担机制。该机制根据最终设备或设备出端口，将数据包逻辑分组，并组装成长度较长的“定长”容器进行转发。属于同一个报文容器的数据包被标记为相同的容器标识，沿着相同路径进行转发，以保证同属于一个报文容器的数据包保序传输，如图5所示。

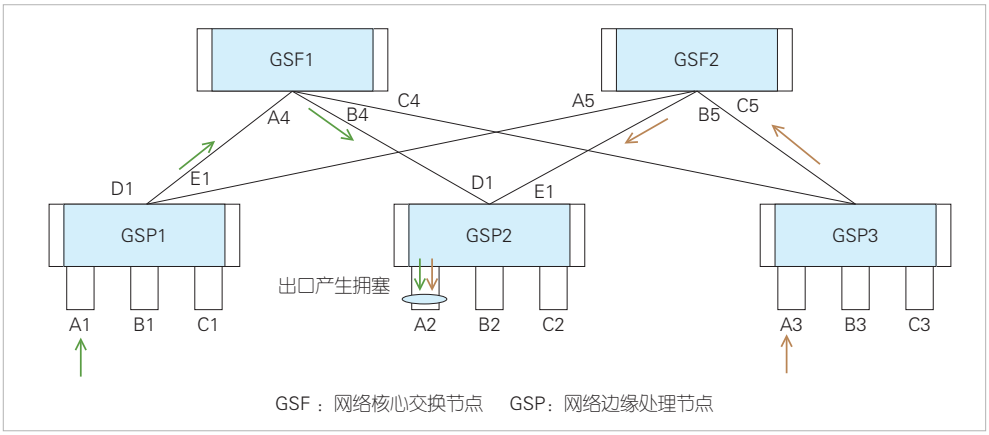
在负载均衡调度时，报文容器被作为转发单位。但由于报文是逻辑组装，无需额外的硬件开销来对数据包进行组装和还原。在网络中转发时添加的报文容器标识，仍以数据包的形式传输，且无冗余数据填充的问题，带宽损耗小。



▲图4 组包交换示意图



▲图5 报文容器转发示意图



▲图6 网络In-cast流量发生场景

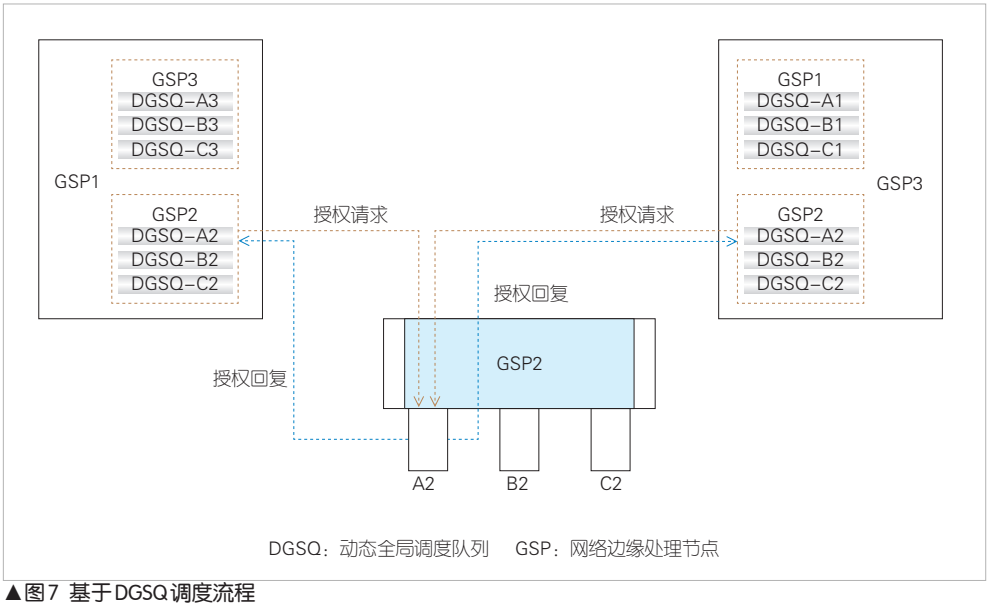
3.2.2 基于报文容器的DGSQ全局调度技术

分布式高性能应用的特征是多对一通信的In-cast流量模型。如果这种流量是短暂的，在出口处可以通过一定的Buffer进行吸收；如果时间持续过长且多个入口的流量相加远大于出口的线速带宽，为了避免丢包，出口设备需启用反压机制保护流量。而反压一旦出现，网络的转发性能就会大幅度下降，从而损害分布式应用的性能。

DCQCN目前是RDMA网络应用最广泛的拥塞控制算法，也是典型的被动拥塞控制算法。发送端根据接收到的拥塞通知报文（CNP）动态调整发送速率。由于1个比特的ECN信号仅能定性表示网络产生拥塞，但无法定量地表示拥塞程度，所以端侧需要探测式调整发送速率。此外，收敛速度慢会导致网络吞吐性能下降。解决网络拥塞丢包最直接的手段是防止过多的数据注入到网络中造成拥塞，保证网络中任意设备端口缓存或链路容量不会过载。

如图6所示，GSP1的A1口和GSP3的A3口同时向GSP2的A2口发送流量，且流量相加大于A2的出口带宽。这造成A2口出口队列拥塞。这种情况仅通过负载均衡是无法规避的，需要全局控制保证送到A2的流量不超过其出口带宽。因此，引入基于全局的转发技术和基于DGSQ的调度技术，才可实现全局流量的调度控制。

基于DGSQ的全局调度技术如图7所示，在GSP上建立网络中所有设备出口的虚拟队列，用以实现本GSP节点到对应所有出端口的流量调度。本GSP节点的DGSQ调度带宽依赖授权请求和响应机制，由最终的设备出口、途经的设备统一进行全网端到端授权。由于中间节点的流量压力差异，GSP去往最终目的端口不再通过等价多路径路由（ECMP）（路径授权权重选择路径，而是需要基于授予的权重在不同



▲图7 基于DGSQ调度流程

的路径上进行流量调度。这种方式可保证全网中前往任何一个端口的流量既不会超过该端口的负载能力，也不会超出中间任一网络节点的转发能力，可降低网络中In-cast流量产生的概率，减少全网内部反压机制的产生。

基于PKTC的负载均衡技术和DGSQ全局调度技术在平稳状态下可很好地进行流量调控与分配。但在微突发、链路故障等异常场景下，短时间内网络还是会产生拥塞，这时仍需要依赖反压机制来抑制源端的流量发送。传统PFC或FC都是点到点的局部反压技术，一旦触发扩散到整个网络中，会引起头阻HoL、网络风暴等问题。全调度以太网技术需要精细的反压机制来守护网络的防线，通过最小的反压代价来实现网络的稳定负载。

3.2.3 全调度以太网的GSOS

综合考虑分布式NOS、集中式SDN控制器的优势，全调度以太网的GSOS分为全调度控制器、设备侧NOS两大部分，可全面提升GSE网络自动化及可视化能力。

GSP和GSF的盒式设备支持独立部署NOS，有助于构建出分布式网络操作系统。每台GSP和GSF具备独立的控制面和管理面，可以运行属于设备自身的网络功能，提升系统可靠性，降低部署难度。分布式NOS可以将单点设备故障限制在局部范围，避免对整网造成影响。

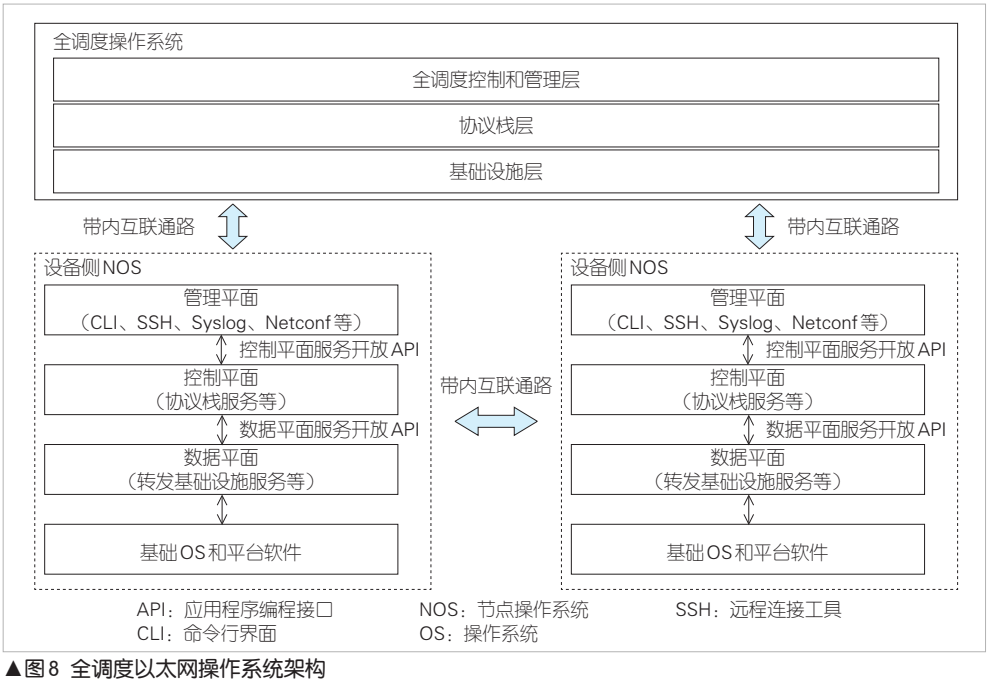
集中式GSOS提供更好的网络全局信息，简化基于全局端口信息的DGSQ系统的建立和维护。同时GSOS也是整网运维监控的大脑，可协同设备实现对实时路径、历史的记录及呈现，以支撑网络运维。

3.3 GSE网络工作机制

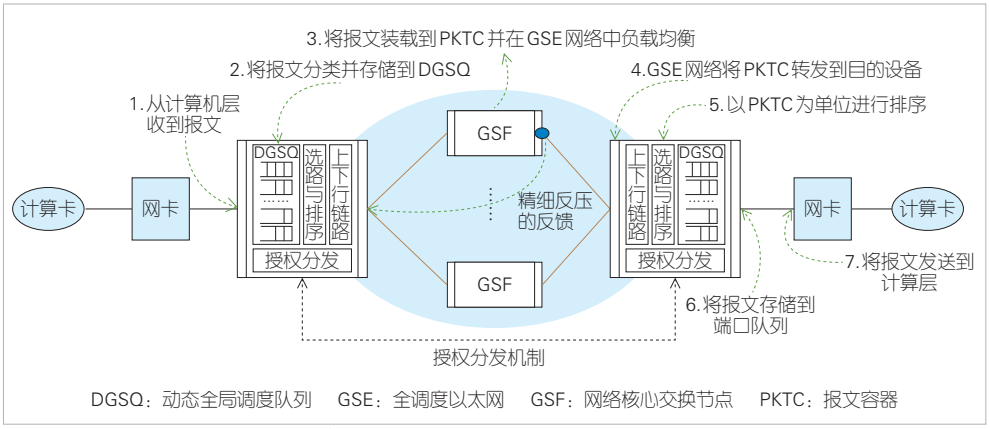
GSE交换网络采用定长的PKTC进行报文转发及动态负载均衡，通过构建基于PKTC的DGSQ全调度机制、精细的反压机制和无感知自愈机制，实现微突发及故障场景下的精准控制，全面提升网络有效带宽和转发延迟稳定性。相关的具体端到端转发流程图9所示。

1）源端GSP设备从计算侧收到报文后，通过转发表找到最终出口，并基于最终出口按需将报文分配到对应的DGSQ中进行授权调度。

2）源端GSP设备获得授权后，遵循PKTC的负载均衡



▲图8 全调度以太网操作系统架构



▲图9 GSE网络端到端流量转发示意图

要求，将报文发送到GSE网络中。

3）当到达目的端GSP设备后，报文先进行PKTC级别的排序，再通过转发表存储到物理端口对应队列，最终通过端口调度发送到计算节点。

作为一种标准开放的新型以太网技术，GSE可采用网卡侧无感知的组网方案，即网络侧采用GSE技术方案，网卡侧仍采用传统RoCE网卡。此外，也可以结合网卡能力演进，将GSE方案各组件的功能在网络组件中重新分工，将部分或全部网络功能下沉到网卡侧来实现。也就是说，在未来的实际应用中，可以将GSP的功能全部下沉到网卡以提供端到端的方案，也可以将网络的起终点分别落在网络设备和网卡上，为后续网络建设和设备选型提供灵活的可选方案。

4 结束语

新型智算中心网络技术已逐渐成为全球创新焦点。智算中心网络是一个多要素融合的复杂系统，是算网的深度融合，它依赖于AI业务、网络设备、交换芯片、网卡、仪表等上下游产业的协同创新。如何提升网络规模和性能，构建超大规模、超高带宽、超低时延的高性能智算中心网络，是提升算力水平的关键。

GSE面向无损、高带宽、超低时延等高性能网络需求业务场景，兼容以太网生态链，通过采用全调度转发机制、基于PKTC的负载均衡技术、基于DGSQ的全调度技术、精细的反压机制、无感知自愈机制、集中管理及分布式控制等技术，实现低时延、无阻塞、高带宽的新型智算中心网络[4]。该技术架构旨在构建一个标准开放的高性能网络技术体系，助力AIGC等高性能产业快速发展。由于该架构创新难度大、开发周期长，我们希望各个行业能够携手合作，持续推动相关技术标准发展。

参考文献

[1] 姚惠娟, 陆璐, 段晓东. 算力感知网络架构与关键技术 [J]. 中兴通讯技术, 2021, 27(3): 7–11. DOI:10.12142/ZTETJ.202103003
[2] 中国移动通信研究院. 面向AI大模型的智算中心网络演进白皮书 [R]. 2023
[3] 中国移动通信有研究院. 新一代智算中心网络技术白皮书 [R]. 2022
[4] 中国移动通信有研究院. 全调度以太网技术架构白皮书 [R]. 2023

作　者　简　介

段晓东，中国移动通信有限公司研究院副院长、“新世纪百千万人才工程”国家级人选、教授级高级工程师；长期从事下一代互联网、算力网络、5G网络架构、6G网络架构、SDN/NFV等技术研究工作。

程伟强，中国移动通信有限公司研究院基础网络技术研究所副所长、教授级高工；长期从事下一代互联网、数据中心网络、传输网等方面的技术研究和标准推动工作。

王瑞雪，中国移动通信有限公司研究院基础网络技术研究所技术经理、SDN/NFV/AI标准与产业推进委员会（TC610）SDN/NFV技术工作组组长、算网融合产业及标准推进委员会（TC621）国际/开源合作工作组组长；主要研究领域为数据中心网络、SDN/NFV、算力网络等。

王雯萱，中国移动通信有限公司研究院基础网络研究所项目经理；主要从事数据中心网络技术与方案研究工作。

数据管理系统发展趋势与挑战

Development Trends and Challenges of Data Management Systems

韩银俊/HAN Yinjun[1,2]，牛家浩/NIU Jiahao[1,2]，
屠要峰/TU Yaofeng[1,2]
（1. 中兴通讯股份有限公司，中国 深圳 518057；
2. 移动网络和移动多媒体技术国家重点实验室，中国 深圳 518055）
(1. ZTE Corporation, Shenzhen 518057, China;
2. State Key Laboratory of Mobile Network and Mobile Multimedia, Shenzhen 518055, China)

DOI：10.12142/ZTETJ.202304012
网络出版地址：http://kns.cnki.net/kcms/detail/34.1228.TN.20230725.1301.002.html
网络出版日期：2023-07-26
收稿日期：2023-06-10

摘要：数据是数字经济时代重要的生产要素。数据管理成为释放数据价值的重要引擎。回顾了数据管理技术的迭代变迁历程，分析了构建新一代数据管理基础设施的关键技术及挑战。结合中兴通讯在数据管理领域进行的创新和研发实践，展示了应对这些挑战的思路、方案及取得的成效。最后，对数据管理技术发展进行了总结和展望。指出数据域技术栈应当走低碳高效、可持续发展路线，而高能效数据管理技术是可持续发展的关键。

关键词：数据管理；数据分析；数据库；大数据；人工智能

Abstract: Data is an important factor of production in the era of digital economy. Data management is an important engine for releasing the value of data. The iterative evolution of data management technology is reviewed, and the key technologies and challenges of building a new generation of data management infrastructure are analyzed. Combined with ZTE's innovation and R&D practices in the field of data management, the ideas to solve these challenges, solutions, and results achieved are demonstrated. Finally, the development of data management technology is summarized and prospected. It is pointed out that the data domain technology stack should follow a low-carbon, efficient, and sustainable development path, and high energy efficiency data management technology is the key to sustainable development.

Keywords: data management; data analysis; database; big data; artificial intelligence

引用格式：韩银俊, 牛家浩, 屠要峰. 数据管理系统发展趋势与挑战 [J]. 中兴通讯技术, 2023, 29(4): 64-71. DOI: 10.12142/ZTETJ.202304012
Citation：HAN Y J, NIU J H, TU Y F. Development trends and challenges of data management systems [J]. ZTE technology journal, 2023, 29(4): 64-71. DOI: 10.12142/ZTETJ.202304012

数据作为新型生产要素，对传统生产方式变革具有重大影响，要构建以数据为关键要素的数字经济。数据、算法、算力是数字经济时代核心的3个要素。其中，数据具有可共享、可复制、可无限供给等特征，是推动数字经济发展的关键生产要素，已上升到国家战略高度。

随着应用需求的发展，数据管理系统也在不断完善，每10年会出现一次比较大的技术变革，产品形态不断繁荣发展——从20世纪60年代的文件系统、数据库、数据仓库、数据湖发展到现在的湖仓一体，产业规模也在持续扩大。数据管理系统如今已在各个行业得到广泛应用，成为数字经济不可或缺的通用基础设施。

随着信息技术的高速发展和数据量的迅速膨胀，大规模、高性能的新型数据管理系统不断涌现。云基础设施的逐渐成熟以及企业用户需求的推动，使得云原生数据管理系统近年来蓬勃发展，催生出各类基于云架构的数据管理服务。人工智能（AI）技术和数据管理技术相辅相成：AI技术越来越多地应用在数据管理系统的计算、存储和运维等方面，数据管理系统为AI训练和推理提供高效的数据服务。异构处理器、新型存储和网络技术的快速发展，正在改变数据管理系统依赖的底层环境，给数据管理与分析技术的发展带来新的机遇与挑战。湖仓一体为用户提供的数据管理平台不仅具有数据仓库的结构化和治理优点，还拥有数据湖的扩展性和机器学习的便利性。数据要素的可信流通使得数据安全成为热点。如何保证数据的安全和隐私成为数据管理系统的关键诉求。

1 数据管理技术的迭代变迁

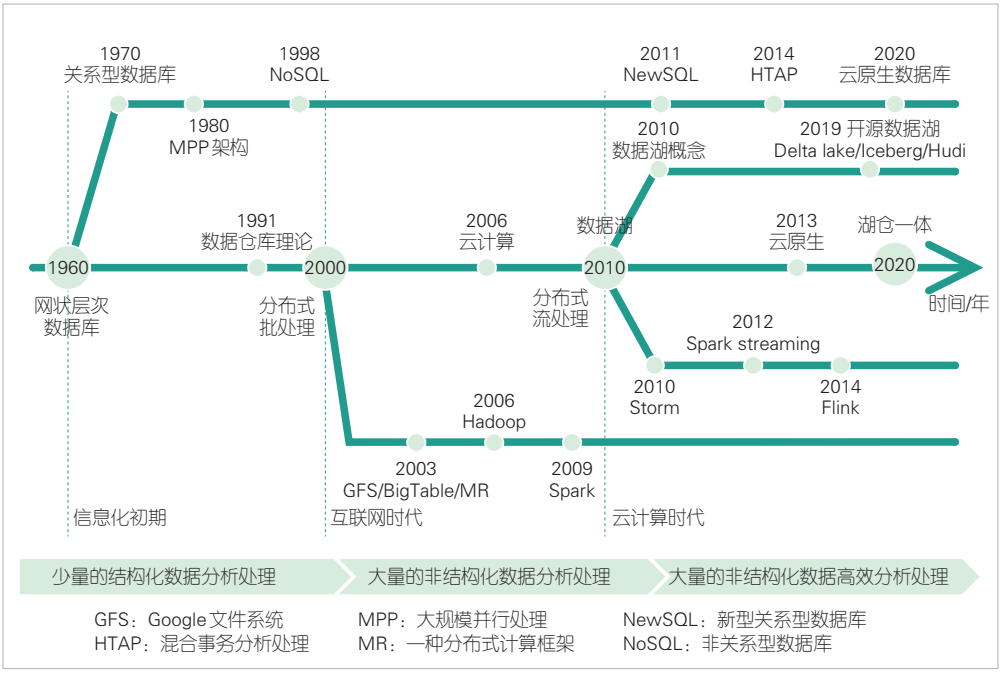
数据管理是计算机科学中一个非常重要的领域，涉及大量的技术创新和研究成果。该领域共获得5次计算机图灵奖，并衍生出网状层次数据库、关系数据库、数据仓库、

NoSQL（指非关系型数据库）、NewSQL（指新型关系型数据库）、数据湖、湖仓一体等面向不同场景、具有多种形态的数据管理系统。如图1所示，以应用需求变更为主线，数据管理系统的发展历程分为信息化初期、互联网时代、云计算时代3个阶段。

在信息化初期阶段，数据管理系统主要针对少量、结构化数据进行管理。第1代数据管理系统是网状层次数据库。网状层次数据库很好地解决了数据集中和共享问题，但是在易用性、数据独立性和抽象方面仍有很大欠缺[1]。1970年，IBM提出数据关系模型的概念[2]。关系模型把现实世界抽象为二维表，借助关系代数的集合运算和关系运算，具有强大的查询表达能力，有力地支撑了信息化初期的数据管理需求。因此，关系模型迅速取代了层次模型和网状模型，成为数据库事实标准。早期的关系数据库主要面向实时交易，支持多并发、快速增删改查。这类应用被称为联机事务处理（OLTP）[3]。20世纪90年代中期，以MySQL、PostgreSQL为代表的OLTP开源数据库迅猛发展。早期MySQL抓住了开源LAMP（指Linux-Apache-MySQL-PHP）架构的先机，在互联网的快速发展中获得广泛应用。近年来，由于拥有更强大的技术先进性和更友好的开源协议，PostgreSQL发展势头强劲。随着累积的历史数据越来越多，如何让这些数据发挥更大的作用是一个亟待解决的问题。1991年B. INMON提出了数据仓库建设方法。1993年E. F. CODD提出联机分析处理（OLAP）[4]的概念，以便满足决策支持、报表展示以及多维数据查询的需求。

20世纪90年代，随着互联网的快速发展，数据量急剧增加。严格的事务一致性要求制约了关系数据库的系统扩展能力，使得低成本的弹性扩展成为数据库的首要需求。在此背景下，以Google为代表的互联网公司开发了NoSQL数据库，在牺牲数据库的事务特性和某些结构化查询语言（SQL）功能的前提下获得了较强的可扩展性。NoSQL泛指非关系型数据库，不同的NoSQL数据库有不同查询语言，难以统一应用程序接口，不具备结构化查询功能。为了解决这些问题，NewSQL[5]数据库被提出。NewSQL是各种新的可扩展和高性能数据库的简称，这类数据库在具有高可扩展性的同时，又保留了传统关系数据库的原子性、一致性、隔离性、持久性（ACID）等特性。由于互联网的高速发展，数据越来越多，数据类型也越来越丰富，传统数据库存不下、无法建模、无法及时入库等问题逐渐凸显。在此背景下，Google相继提出GFS[6]、MapReduce[7]和Bigtable[8]，开启了大数据时代。2006年开源生态Hadoop[9]的诞生，改变了企业对数据的存储、处理和分析的过程，加速了大数据的发展，带来了行业变革。

随着云计算的发展，数据形式及应用场景变得更加多样化。数据管理系统需要基于云计算基础设施提供更加灵活、高效、可靠、安全的解决方案。云原生数据管理是基于云计算架构而设计和构建的，充分利用云基础设施的能力，具备弹性伸缩、多租户、分布式部署等特性，满足多源异构的大规模数据处理需求。实时推荐、即时决策等场景提出了海量数据联机处理与实时分析的需求。实时数据仓库和流式计算引擎（Storm[10]、Spark Streaming[11]、Flink[12]）等应运而生，可满足一些实时性要求高的场景。Hadoop因生态复杂、事务支持能力弱、交付及运维成本高，无法替代核心数仓，逐渐形成了自身特殊的定位——数据湖（Data Lake）。数据湖[13]是一种数据存储方法，即在系统或存储库中以自然格式存储数据的方法，通常是企业中全量数据的单一存储，可提供各类报表、数据可视化、高级分析和机器学习等



▲图1 数据管理技术的迭代变迁

服务。数据湖提供了更为完善的数据管理能力，但仍无法满足用户在性能、事务等方面的要求。2020年Databricks提出了Lakehouse和面向湖仓一体的体系架构[14]。Lakehouse是由Data Lakes与Data Warehouses组合而成的一种新的数据架构，目的是打破数据湖与数据仓库割裂的关系，结合数据仓库企业级能力与数据湖的灵活性，同时满足商业智能（BI）与AI两类场景需求。湖仓一体要在数据处理方面实现数据湖和数据仓库的互通，是数据一体化思想的体现。随着数据要素的流通和发展，湖仓一体将被赋予更多的含义和价值。

由数据管理技术60多年的迭代变迁历程可以看出，计算模式的改变和应用需求的变化，对数据管理系统形态的发展起到了至关重要的作用，数据管理技术和架构也随之不断迭代更新。在负载特征方面，针对不同业务场景的数据管理系统不断涌现，包括联机事务处理（OLTP）、联机分析处理（OLAP）、混合事务分析处理（HTAP），以及面向流批计算和湖仓融合的数据处理；在数据模式方面，数据模型从关系型向非关系型拓展，包括键值、文档、图、列族和时序等；在系统架构方面，传统单机数据库通过主从复制的方式满足数据库的可用性，而分布式和多主架构则进一步满足数据管理容量和性能的需求。此外，云计算和AI的普及，使得数据管理更具弹性和智能。

2 数据管理的关键技术及挑战

近年来，云原生、AI、新型硬件、安全隐私以及大模型等技术迅速发展，为数据管理系统的创新带来机遇和挑战[15]。利用新型交叉学科技术构建的新一代数据管理基础设施正在兴起。

2.1 云原生数据管理

随着云基础设施的逐渐成熟以及企业用户需求的推动，云原生数据管理近年来得到了蓬勃发展，催生出各类基于云架构的数据管理服务。目前云数据库包含数据库云服务和云原生数据库两大类。数据库云服务主要采用云托管的形式，即云服务商将数据库看作一种部署到云平台的普通软件，在架构层面没有质变，无法充分复用云平台的强大能力，存在计算存储紧耦合、数据存储冗余、同步延时严重等问题。云原生数据库则是为云架构而原生设计的数据库。Amazon Aurora[16]和Snowflake[17]分别是云原生OLTP数据库和云原生OLAP数据库的全球引领者。云原生数据库采用计算存储分离的架构，遵循“日志即数据”的原则，计算层能够自动实现读写分离，扩缩容过程对上层透明，存储层采用分布式高可用存储系统，该架构实现了独立的计算节点弹性伸缩和存储节点弹性扩缩容，进而提升了数据库性价比。

通过存储与计算分离，云原生数据库很好地解决了数据库云服务的高可靠、高可用和高可扩展性问题，但还存在诸多挑战：首先，存储和计算分离带来存储和计算之间访问时延的开销；其次，当前云原生数据库基本都只支持一写多读，不能实现多节点写，造成了写扩展性受限，特别是不能支持对写需求大的应用；此外，当前云原生数据库往往是针对一种负载类型设计的，对于HTAP的混合负载数据库缺乏有效的支持。

为了应对上述挑战，中兴通讯基于电信云基础设施（TCF）研发了云原生数据库EBASE-C和云原生数据仓库EBASE-A。EBASE-C采用存储与计算分离架构，利用全局事务处理模块，将多个节点读写的事务ID的分配和事务并发控制进行统一协调处理，支持基于多节点的读写功能，提升了数据库的读写扩展性；引入全局缓存，通过高性能的网络把各个节点的共享缓冲池连成一个整体，并对外提供高效、一致的缓存服务，减少了网络数据传输；在计算节点之间仅同步Redo Log相关的元数据信息，降低了节点间的复制延迟。EBASE-A在计算层引入向量化加速引擎，利用指令集的原生加速实现高效OLAP查询，借助算子下推能力将SQL操作下推到存储层中，在存储层过滤掉不必要的数据，减少了计算节点和存储节点之间数据传输的开销；在存储层采用行列混合的存储方式，支持数据压缩，有效支持了HTAP混合负载的访问；利用统一元数据架构，提供统一数据资产视图，管理全局事务和全局对象，打破了数据湖与数据仓库之间的界限，实现了湖仓一体化实时分析。

Serverless是云原生数据管理的下一个阶段，通过隐藏服务器，提供突出的弹性伸缩和按需服务能力，兼容处理各种类型的负载，实现更细粒度、更精准的资源调度。

2.2 智能化数据管理

传统的数据管理系统在大规模服务、性能调优和运维管理等方面面临很多挑战。AI技术因其强大的学习、推理、规划能力，为数据管理提供了新的发展机遇。AI赋能的数据管理技术得到了广泛关注。

以AI4DB为代表的智能化数据管理将AI技术应用到数据管理领域，提供自检测、自配置、自调优、自诊断、自愈、自安全和自组装等功能。从AI与数据管理系统的作用关系看，AI4DB分为外置AI优化和内置AI优化。其中，外置AI优化主要充当数据库管理员（DBA）的角色，对数据库进行调优和诊断，包括参数配置、参数调优、SQL改写、索引推荐、根因分析等；内置AI优化则包括基数估计、查

询优化和学习型索引等。基数估计是数据管理系统查询优化的一大核心问题，更精确的基数估计能够帮助优化器选择更优的查询计划。AI驱动的学习型基数估计方法将基数估计作为回归问题，该类方法收集具有真实基数（作为标签的查询池），提取查询特征并将它们编码为向量，随后训练模型并将查询映射到基数。在推理时，查询被编码为特征向量，通过输入回归模型得出基数估计结果。由AI驱动的学习型查询优化器受到研究者的广泛关注。Neo[18]是第一个学习型查询优化器，通过强化学习方法生成延迟最低的执行计划。这类优化器能够以更少的代价取得更好的性能。麻省理工学院首次提出学习型索引[19]概念，使用机器学习模型替代传统的索引结构。学习型索引可以大幅降低传统索引的空间代价，提高查询性能。

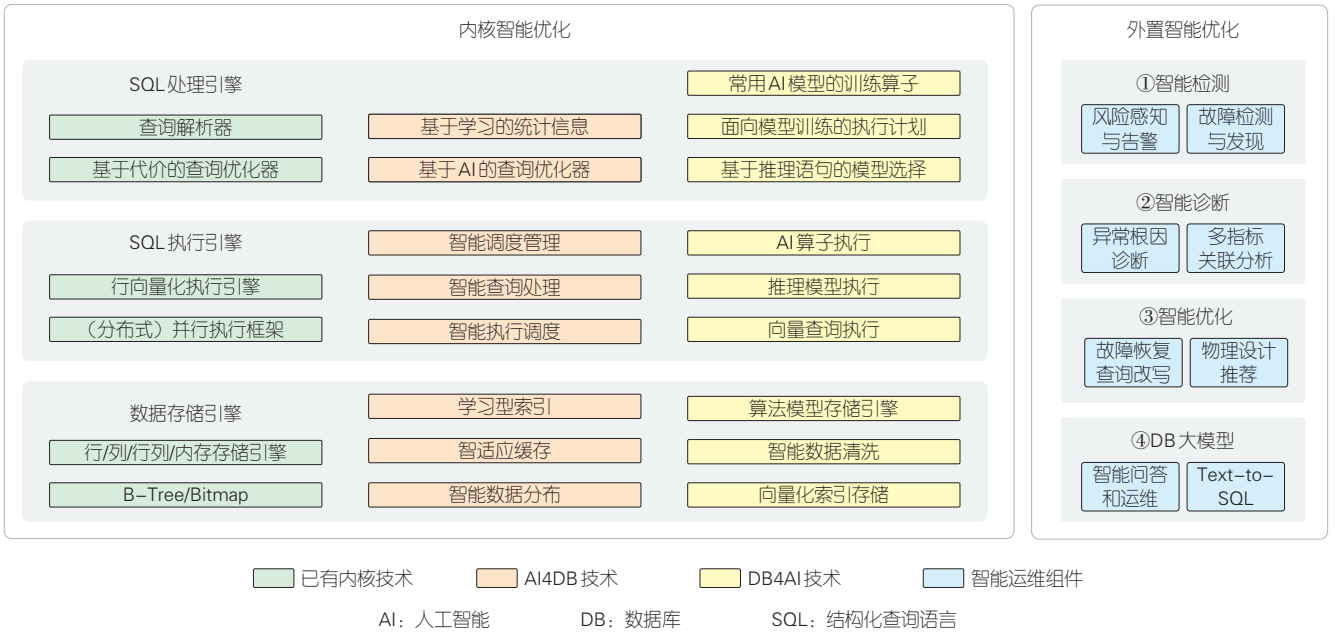
中兴通讯-北京大学联合实验室围绕智能化数据管理进行创新和实践，研发了智能化数据管理模块DBRobot，如图2所示。

DBRobot包括外置智能优化和内核智能优化两大功能。外置智能优化实现了业务无感一键式诊断优化，包括智能监控、智能诊断、智能优化和数据库（DB）大模型4个部分。其中，智能监控模块采集日志和参数等多维指标，进行趋势预测和异常检测，对发现的异常及时告警；智能诊断模块通过细粒度性能诊断、异常分析和多指标关联分析等手段实现慢SQL诊断、系统亚健康诊断和系统故障诊断，识别问题根因；智能优化模块针对问题根因通过智能参数调优、索引智能推荐、SQL智能重写等技术，排除诊断出的故障；DB大模型模块利用大语言模型的上下文学习和思维链能力实现数据库的智能问答、智能运维和Text-To-SQL等功能。

内核智能优化聚焦AI4DB和DB4AI两个方向。在AI4DB方向，EBASE实现了基于AI的查询优化器LOGER[20]。LOGER使用深度强化学习方法，在搜索过程中对部分查询计划进行评价，并生成完整的查询计划。在DB4AI方向，引入支持向量计算的训练算子，可实现库内数据训练和训练模型的存储；引入模型调用接口，使库内数据能够在查询后进行推理分析。

ChatGPT引发的大模型浪潮，催生了向量数据的存储、检索需求。传统的数据库索引结构难以有效地处理向量之间的相似度搜索和邻近性查询。向量数据库应运而生。向量数据库的核心思想是：将向量和对应的标识符存储在数据库中，并构建索引以加速相似度搜索，满足如图像检索、推荐系统、人脸识别和语义搜索等应用的需求。中兴通讯向量数据库EBASE-Vector能够高效地解决向量相似度检索和高密度向量聚类等问题，支持拍字节（PB）级向量数据的管理，通过与大模型技术和LangChain生态的融合，在高效存储和检索向量数据的同时，使得AI应用开发更加高效便捷。中兴通讯EBASE在大模型与数据管理融合领域持续创新，发布了业界领先的数据库大模型Nebula-EBASE。该模型具备Text-To-SQL、智能问答和智能运维等能力。

近几年，AI技术被广泛应用在数据管理领域中。总体



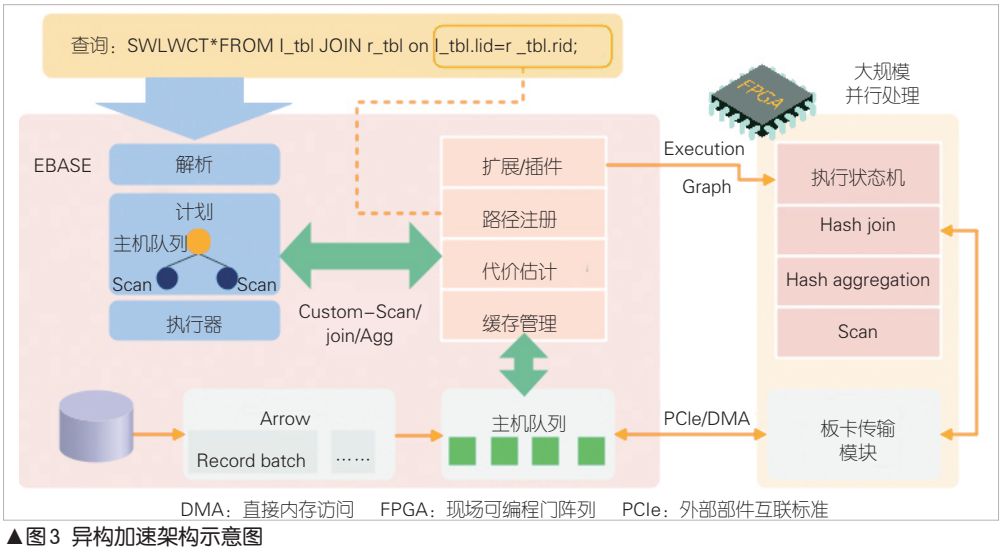
▲图2 中兴智能化数据管理模块DBRobot

上讲，AI在智能运维和系统管理方面的应用较为成熟，但在系统内核的智能化和DB4AI方面还需要不断探索。

2.3 新型硬件适配

数据管理系统在基础硬件和上层软件之间起到“承上启下”的作用，向上支撑上层应用，向下发挥硬件算力作用。以高性能处理器和硬件加速器、非易失内存（NVM）和远程直接内存访问（RDMA）高性能网络为代表的新硬件技术，正在改变传统的数据管理系统的底层载体支撑。数据管理系统将向异构计算架构、混合存储环境和高性能互联网络逐步演进[21]。在存储层面，按字节存取的持久内存（PMEM）在提供更高的事务吞吐量的同时，也引入了一致性挑战。如何针对PMEM的特性管理设计高效的索引结构是一个关键问题。在网络层面，RDMA极大降低了主机间数据传输的时延，有效改善了分布式系统的运行环境。但由于RDMA在内存之间直接访问，系统设计需要重新考虑如何有效管理和分配内存资源，对事务一致性也提出了更高的要求。在计算层面，众核高性能处理器和各类硬件加速器，例如图形处理器（GPU）、现场可编程门阵列（FPGA）等，已被广泛用于加速处理数据。在系统设计时需要重新优化计算模型，以充分利用异构处理器的并行计算能力，面临着任务划分、资源调度、数据分发和算法优化等方面的挑战。

中兴通讯EBASE围绕新型硬件技术和软硬协同进行创新研发。在PMEM适配方面，EBASE绕过原有的文件系统内核输入输出（IO）层，直接对PMEM进行操作，实现了PMEM原生的日志机制和存储引擎。针对PMEM跨非统一内存访问（NUMA）带来的数据访问性能下降问题，EBASE实现了NUMA感知的数据访问机制，能够将同一个NUMA节点内的中央处理器（CPU）和PMEM设备进行绑定，确保了数据访问的局部性。EBASE利用PMEM大容量来扩大内存空间，基于DRAM/PMEM两级内存的缓冲区，实现热度感知的高速缓冲，提高系统查询处理的性能。在异构处理器加速方面，EBASE采用与CPU协作的加速器方式，将Join、Agg、Scan等算子或算子组合卸载到异构处理器FPGA设备上，与CPU协同完成查询语句的执行。如图3所示，异构加速架构通过加速扩展层实现异构算子路径注册、异构算子调用、数据传输等功能，由查询优化器自动选择最优的物理计划，无须上层业务干预，实现了异构加速与数据管理系统的无缝集成。



▲图3 异构加速架构示意图

新的硬件还在不断发展演进，以CXL[22]为代表的高速总线协议将有效提升处理器和设备之间的内存互联互通的效率，将带来更大的内存扩展空间。如何对此进行软件层面的优化和适配，是数据管理系统后续的改进方向。

2.4 湖仓融合的数据一体化

湖仓一体是新的概念，目前并没有统一且成熟的定义。各大厂商均对湖仓一体进行探索和实践。有些厂商基于数据湖架构对数据仓库进行能力扩展，通过在开放文件存储格式之上构建一套表格式Table Format和元数据管理系统，使数据湖具备ACID事务能力，并提高了数据管理水平，如开源系统Apache Iceberg[23]、Apache Hudi[24]、DeltaLake[25]。有些厂商推出的方案基于数据仓库向数据湖能力扩展，通过各种连接器以外部表的方式访问数据湖底层存储系统中的数据，多采用存算分离的架构来完善自身的调度、计算、存储功能，扩展自身的能力，使自身形成一个数据处理平台。相关的技术方向往往更注重实时高并发场景应用和非结构化数据治理。

湖仓一体在成本和性能上还不足以与传统成熟的大数据存储解决方案竞争，成熟的产品和系统较少。在海量存储上搭建能够保证ACID的高性能湖仓一体架构仍然是主要挑战。在湖仓存储层，随着文件数量大幅增长，数据湖存储Hadoop分布式文件系统（HDFS）的NameNode节点遇到了元数据容量瓶颈，这限制了湖仓存储能力。同时，大集群的

NameNode启动速度非常缓慢，其全局锁处理机制大大限制了并发访问能力。湖仓元数据和计算层面临着ACID事务性能提升、高效并发更新及写入、海量元数据管理、查询优化等方面的挑战[26-28]。

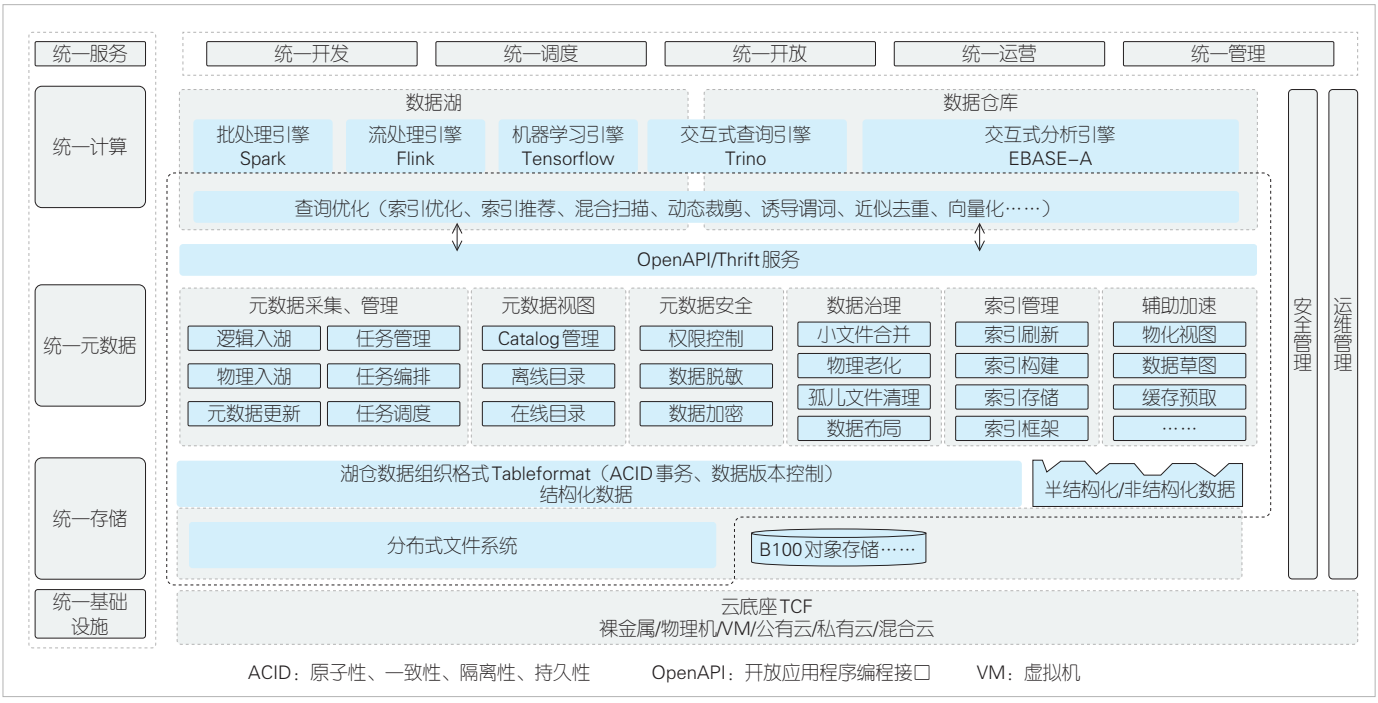
湖仓一体不仅仅在数据处理上将数据湖和数据仓库互相打通，还实现了数据一体化。通过整合大数据、数据仓库、AI等技术，中兴通讯研发了新一代面向湖仓融合的数据管理系统DAIP。DAIP兼顾性能和成本效率，通过以表格式Table Format为代表的新技术，将数据湖和数据仓库功能融合，实现一体化存储，形成一套基于统一元数据的数据服务系统；结合云原生技术，采用存算分离架构，提供统一开放的存储接口；对接多样的计算引擎，实现存储和计算灵活部署，实现资源按需使用。DAIP能够有效简化企业的数据基础设施架构，让数据管理的灵活性与成长性得到了统一。DAIP架构如图4所示，其中虚线内功能模块表示中兴通讯自主研发的功能或者在开源基础上实现的增强功能。

DAIP基于自研大容量分布式元数据持久化技术，突破了原生NameNode元数据全内存架构的限制，纵向扩展了单NameNode节点的元数据容量。目前，单个NameNode可支持的文件数量达到百亿以上，与原生联邦横向扩展架构兼容。二者叠加可满足千亿级文件存储的需求，有效应对大容量湖仓存储挑战。在流处理场景下应用不断向数据湖表中数据插入数据或者进行merge、update等操作时，会产生大量的小文件。过多的小文件会导致计算引擎的查询过程变慢，并且会引起系统扩展性和稳定性问题。DAIP研发了数据湖治理功能，支持压缩合并表文件、物理老化和孤儿文件清理等，实现了自动数据布局和优化，以保持文件访问最佳性能，将查询运行时间和占用存储容量减少了10%以上。围绕大规模元数据管理及查询优化技术进行创新和实践，DAIP构建了一个面向湖仓融合的低成本索引系统，为湖仓不同计算引擎提供统一计算加速能力。通过将元数据管理与数据管理同等看待，以分布式方式管理和处理元数据，该系统可以存储非常丰富的元数据并扩展到非常大的表，可同时嵌入到多个计算引擎中，结合查询优化技术允许各个计算引擎直接跳过无关文件，以提升实时数据分析及查询效率。

湖仓一体技术仍在不断迭代发展。中兴通讯新一代面向湖仓融合的数据管理系统将以提升用户体验为目标，为湖仓提供更大的容量、更快的速度、更好的稳定性，并构建智能数仓、流式数仓等外围生态，在数字经济建设中发挥更重要的作用。

2.5 数据要素可信流通

数据作为新型生产要素，是数字化、智能化的基础，已快速融入生产、分配、流通、消费和服务等各环节。如何保证不同场景下数据要素安全可信流通，构建数据治理新体系，是工业界与学术界研究的热点问题。



▲图4 新一代面向湖仓融合的数据管理系统架构图

隐私计算[29]是涵盖众多学科的交叉融合技术，目前主流的隐私计算技术主要分为三大方向：1）以多方安全计算为代表并基于密码学的隐私计算技术；2）AI与隐私保护融合而衍生的技术；3）以可信执行环境为代表并基于可信硬件的隐私计算技术。借助隐私计算机制，在技术层面可通过隐私计算技术，从数据采集、存储、协作等方面提升数据安全和隐私保护水平，保护数据全生命周期的安全，将数据所有权与使用权分离，使计算过程中不发生数据所有权的转移，从而实现“数据可用不可见”，为数据要素安全可信流通提供有力支撑。如何保证数据不受恶意篡改是数据维护中关乎数据安全的基础性问题。随着数据规模的不断增长和云服务的逐渐普及，传统防篡改机制难以适应在复杂环境下对大规模数据的保护要求。如何构建高效的数据防篡改机制，如何在不可信环境下保护数据安全，都是亟待解决的问题。不可篡改性、去中心化、可追溯性等特性保证了区块链能在不可信环境中构建可信的计算环境。

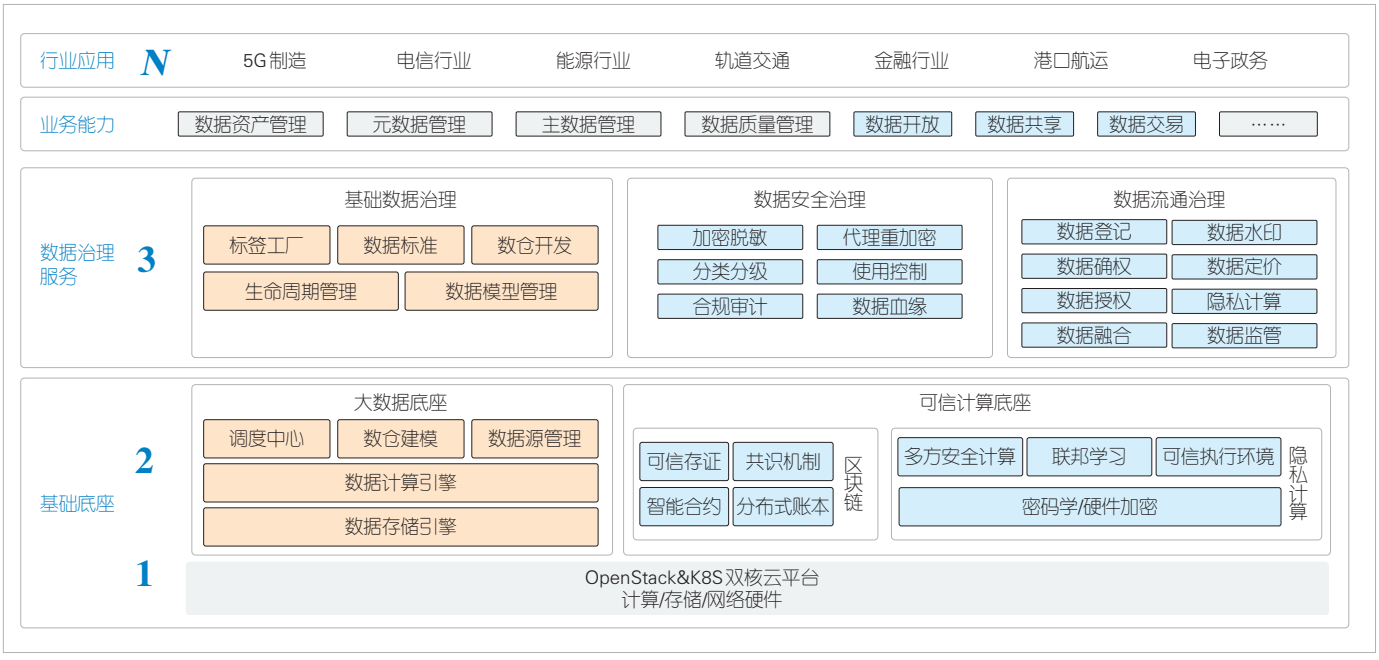
中兴通讯结合自身在区块链和隐私计算领域的多年深耕与积淀，提出数据要素可信流通1+2+3+*N*架构[30]，成功实现了两者的融合部署应用，如图5所示。

隐私计算技术能解决区块链的扩展和隐私保护问题，区块链技术也能使得隐私计算技术更加安全、更加可信赖。通过两者结合，中兴通讯进一步拓展了各个垂直行业应用的场景，实现了数据在存储、流通和计算过程中端到端的安全和可审计，为数据要素可靠、安全、合规和高效流通奠定了扎实技术基础。

3 数据管理技术发展展望

回顾数据管理60年的发展历程，计算模式和应用需求的变化对数据管理系统形态产生了重要影响，推动了数据管理架构和技术迭代更新。近年来，全球数据管理新技术蓬勃发展，但仍然存在两个亟待解决的问题：1）数据规模高速增长，计算处理能力依然是主要瓶颈；2）数据虽上升为生产要素，但数据价值释放不充分。

全球数据量的持续高速增长，“碳达峰、碳中和”目标的提出，都要求数据域技术栈必须走低碳高效、可持续发展的路线。因此，高能效数据管理技术是可持续发展的关键。云数据管理系统具有资源共享、节能高效的特点，将是未来数据管理的主要基础形态。数据管理与处理的成本成为重要考量因素。数据管理系统的设计理念从传统的“扩展性优先”向“以性能优先”转变。智能化数据管理、近数处理、新型硬件驱动等新兴管理和处理方法，成为性能优先设计的重要技术手段。GPU、FPGA、深度学习处理器（DPU）等专用加速器从专用领域走向通用计算，对数据管理技术产生重要影响，特别是在高维数据分析和大规模非结构化数据处理方面。近年来，不少国家在云数据管理的基础上开始探索国家范围内的一体化高能效数据管理。中国提出了算力网络的概念并制定相关国际标准，正式启动“东数西算”工程。由于算力和数据要素的大规模调度与流通，如何在云数据管



▲图5 中兴通讯数据要素可信流通平台“1+2+3+*N*”架构

理基础上进行云、边、端以及多云之间数据和计算的协同，实现全国一体化的高能效数据管理，形成低碳发展新格局，成为未来数据管理的主要方向。

在海量数据和丰富应用场景的驱动下，更多的数据技术和应用创新将全面落地。数据采集、数据治理、数据流通、数据开发利用、数据安全保护等各方面将协同推进。数据要素规模化产业集群和规范化产业生态将逐步形成。数据要素的价值将得到充分挖掘和释放，从而进一步促进数字经济和实体经济深度融合，助力数字经济高质量可持续发展。中兴通讯将持续致力于新型数据管理系统的研发，协同推动数据一体化、新硬件加速、智能化数据管理等新技术的快速商用落地，实现横向跨域拉通和智能敏捷赋能，繁荣生态合作，助力客户在数字经济时代建立可持续的竞争优势。

参考文献

[1] 杜小勇, 卢卫, 张峰 . 大数据管理系统的历史、现状与未来 [J]. 软件学报, 2019, 30(1): 127−141. DOI: 10.13328/j.cnki.jos.005644
[2] CODD E F. A relational model of data for large shared data banks [J]. Communications of the acm, 1983, 26(1): 64−69
[3] GRAY J, REUTER A. Transaction processing: concepts and techniques [M]. San Francisco: Morgan Kaufmann Publishers Inc., 1992
[4] HAN J W, KAMBER M. Data mining: concepts and techniques, second edition [M]. San Francisco: Morgan Kaufmann Publishers Inc., 2002
[5] PAVLO A, ASLETT M. What's really new with NewSQL [J]. ACM sigmod record, 2016, 45(2): 45−55
[6] GHEMAWAT S, GOBIOFF H, LEUNG S T. The Google file system [C]// Proceedings of the nineteenth ACM symposium on Operating systems principles. ACM, 2003: 29−43. DOI: 10.1145/945445.945450
[7] DEAN J, GHEMAWAT S. MapReduce [J]. Communications of the acm, 2008, 51(1): 107−113. DOI: 10.1145/1327452.1327492
[8] CHANG F, DEAN J, GHEMAWAT S, et al. Bigtable: a distributed storage system for structured data [J]. ACM transactions on computer systems, 2008, 26(2): 1−26. DOI: 10.1145/1365815.1365816
[9] Apache Software Foundation. Apache Hadoop [EB/OL]. [2023−05−25]. https://hadoop.apache.org.
[10] HUSSAIN IQBAL M, SZABIST, RAHIM SOOMRO T. Big data analysis: apache storm perspective [J]. International journal of computer trends and technology, 2015, 19(1): 9−14. DOI: 10.14445/22312803/ijctt−v19p103
[11] ZAHARIA M, XIN R S, WENDELL P, et al. Apache spark [J]. Communications of the ACM, 2016, 59(11): 56−65. DOI: 10.1145/2934664
[12] CARBONE P, KATSIFODIMOS A, EWEN S, et al. Apache flink: stream and batch processing in a single engine [EB/OL]. [2023−05−25]. http://sites.computer.org/debull/A15dec/p28.pdf
[13] 维基百科 . 数据湖 [EB/OL]. [2023−05−25]. https://zh.wikipedia.org/wiki/%E6%95%B0%E6%8D%AE%E6%B9%96
[14] ZAHARIA M, GHODSI A, XIN R, et al. Lakehouse: a new generation of open platforms that unify data warehousing and advanced analytics [EB/OL]. [2023−05−25]. https://cs. stanford. edu/~matei/papers/2021/cidr_lakehouse.pdf
[15] 李战怀, 李国良, 陈跃国 . "十四五"数据库发展趋势与挑战 [J]. 中国计算机学会通讯, 2022, 6: 8−11
[16] VAN AKEN D, PAVLO A, GORDON G J, et al. Automatic database management system tuning through large−scale machine learning [C]// Proceedings of the 2017 ACM International Conference on Management of Data. ACM, 2017: 1009−1024. DOI: 10.1145/3035918.3064029
[17] VUPPALAPATI M, MIRON J, AGARWAL R, et al. Building an elastic query engine on disaggregated storage [C]//NSDI. USENIX, 2020: 449−462
[18] MARCUS R, NEGI P, MAO H Z, et al. Neo [J]. Proceedings of the VLDB endowment, 2019, 12(11): 1705−1718. DOI: 10.14778/3342263.3342644
[19] KRASKA T, BEUTEL A, CHI E H, et al. The case for learned index structures [C]//Proceedings of the 2018 International Conference on Management of Data. ACM, 2018：489−504
[20] CHEN T Y, GAO J, CHEN H D, et al. LOGER: a learned optimizer towards generating efficient and robust query execution plans [J]. Proceedings of the VLDB endowment, 2023, 16(7): 1777−1789. DOI: 10.14778/3587136.3587150
[21] 梅宏, 杜小勇, 金海, 等 . 大数据技术前瞻 [J]. 大数据, 2023, 9(1): 1−20
[22] DAS SHARMA D. Compute Express Link: an open industry−standard interconnect enabling heterogeneous data−centric computing [C]// Proceedings of 2022 IEEE Symposium on High−Performance Interconnects (HOTI). IEEE, 2022: 5−12. DOI: 10.1109/HOTI55740.2022.00017
[23] Apache Iceberg. Apache Iceberg homepage [EB/OL]. [2023−05−25]. https://iceberg.apache.org
[24] Apache Hudi. Apache Hudi hompage [EB/OL]. [2023−05−25]. https://hudi.apache.org
[25] Delta Lake. Delta Lake hompage [EB/OL]. [2023−05−25]. https://delta.io
[26] EDARA P, PASUMANSKY M. Big metadata [J]. Proceedings of the VLDB endowment, 2021, 14(12): 3083−3095. DOI: 10.14778/3476311.3476385
[27] POTHARAJU R, KIM T, SONG E, et al. Hyperspace [J]. Proceedings of the VLDB endowment, 2021, 14(12): 3043−3055. DOI: 10.14778/3476311.3476382
[28] TA−SHMA P, KHAZMA G, LUSHI G, et al. Extensible data skipping [C]// Proceedings of 2020 IEEE International Conference on Big Data (Big Data). IEEE, 2021: 372−382. DOI: 10.1109/BigData50022.2020.9377740
[29] 全国信标委大数据标准工作组 . 2022. 数据要素流通标准化白皮书（2022版）[R]. 2022
[30] 中兴通讯 . 2023 Netnumen Ztppc数据要素可信流通平台 [R]. 2023

作　者　简　介

韩银俊，中兴通讯股份有限公司数据智能研发总工、中兴通讯青年领军人才；负责新型数据库技术、大数据湖仓融合技术和高性能存储相关技术的研发工作，主要研究方向为数据库、人工智能、大数据及存储；获省级科技进步奖一等奖3项。

牛家浩，中兴通讯股份有限公司数据智能系统架构师、中兴通讯青年领军人才；负责大数据平台规划和技术研究工作，主要研究方向为大数据、湖仓融合、数据安全及隐私保护等。

屠要峰，中兴通讯股份有限公司中心研究院副院长、数据库技术专家委员会主任，中国计算机学会杰出会员、南京分部副主席、数据库专委、大数据专委、信息存储常委，中国人工智能学会常务理事，中国开源软件联盟理事；负责中兴通讯数据智能平台研发，主要研究方向为大数据、人工智能、数据库及存储。

基于5G连接的集中式PLC新型工业组网架构

5G-Based Centralized PLC New Industrial Networking Architecture

邓伟/DENG Wei，于天意/YU Tianyi，侯庆东/HOU Qingdong
（中国移动通信有限公司研究院，中国 北京 100053）
(Research Institute of China Mobile, Beijing 100053, China)

DOI：10.12142/ZTETJ.202304013
网络出版地址：http://kns.cnki.net/kcms/detail/34.1228.TN.20230720.1158.002.html
网络出版日期：2023-07-20
收稿日期：2023-03-25

摘要：提出基于5G连接的集中式可编程逻辑控制器（PLC）组网架构。通过将生产环节中的物理实体PLC虚拟化，并将虚拟化PLC集中部署在靠近工业现场的边缘算力载体上，结合5G系统确定性增强，实现PLC全向无线化。围绕“基于软性确定性连接的全局集中式协同控制”思路，分析集中式PLC组网架构中涉及的确定性、差异化、降成本和安全性等挑战，并重点给出关键使能技术体系。

关键词：集中式；可编程逻辑控制器；5G连接；降本增效

Abstract: A centralized programmable logic controller (PLC) networking architecture based on the 5G connection is proposed. By virtualizing the physical entity PLC in the production link, and centrally deploying the virtualization PLC on the edge computing power carrier near the industrial field, the PLC omnidirectional wireless is realized. Focusing on the idea of "global centralized collaborative control based on soft deterministic connection", the challenges of determinism, differentiation, cost reduction, security involved in the centralized PLC networking architecture are analyzed, and the key enabling technology system is focused on.

Keywords: centralized; PLC; 5G connection; cost reduction and benefit increasing

引用格式：邓伟，于天意，侯庆东．基于5G连接的集中式PLC新型工业组网架构 [J]. 中兴通讯技术，2023，29(4)：72-77. DOI：10.12142/ZTETJ.202304013
Citation：DENG W, YU T Y, HOU Q D. 5G-based centralized PLC new industrial networking architecture [J]. ZTE technology journal, 2023, 29(4): 72-77. DOI: 10.12142/ZTETJ.202304013

传统的工业网络一般分为企业层、车间层、控制层和设备层[1]。其中，企业层与车间层属于信息技术（IT）网络，对网络性能要求不高，一般采用标准以太网协议。控制层与设备层位于生产现场，属于运营技术（OT）网络，对网络性能要求苛刻，一般采用定制工业以太网/现场总线协议。

为提高生产效率，工业生产大部分采用流水线作业方式，工业控制（后文简称为“工控”）设备位于工业现场，与机器人、生产装备等组成具备某一生产能力的工位。各工位之间通过传送带、物料架等连接，并通过统一精准的协同控制完成生产流程的闭环。高精度、高可靠的自动化生产对控制环节的确定性要求很高，工业现场普遍采用基于刚性确定性规划协同的离散分布式控制系统，来保障生产的连续性和安全性。工控通信网络整体产业链条长、产业难度大，目前高端工控设备及工业协议严重依赖进口，制约了中国智能制造产业的高质量发展。

近年来，随着5G、云计算等新一代信息技术的发展，无线通信、大规模计算能力不断提升，使用成本不断降低。随着5G与工业融合的逐步深入，业界正在探索5G与OT的深度融合，使能工业现场无线化，真正实现柔性生产。基于此，本文提出了“基于软性确定性连接的全局集中式协同控制”思路，以替代传统“基于刚性确定性规划协同的离散分布式控制”思路，实现降本增效。一方面，具备确定性增强能力的5G系统内生云化/集中化可编程逻辑控制器（PLC），可承载工业网络IT域、OT域的所有需求，实现“一网到底”；另一方面，PLC在云化/集中化后，通过大范围协同、组态逻辑的优化、网络与业务的协同，降低了控制器与输入/输出模块（C2IO）间通信对连接确定性的要求。同时，控制功能实现云化/集中化后，可复用当前通用的算力资源，提升工控的灵活性，降低部署成本。

1 业界对控制集中化的组网架构的探索

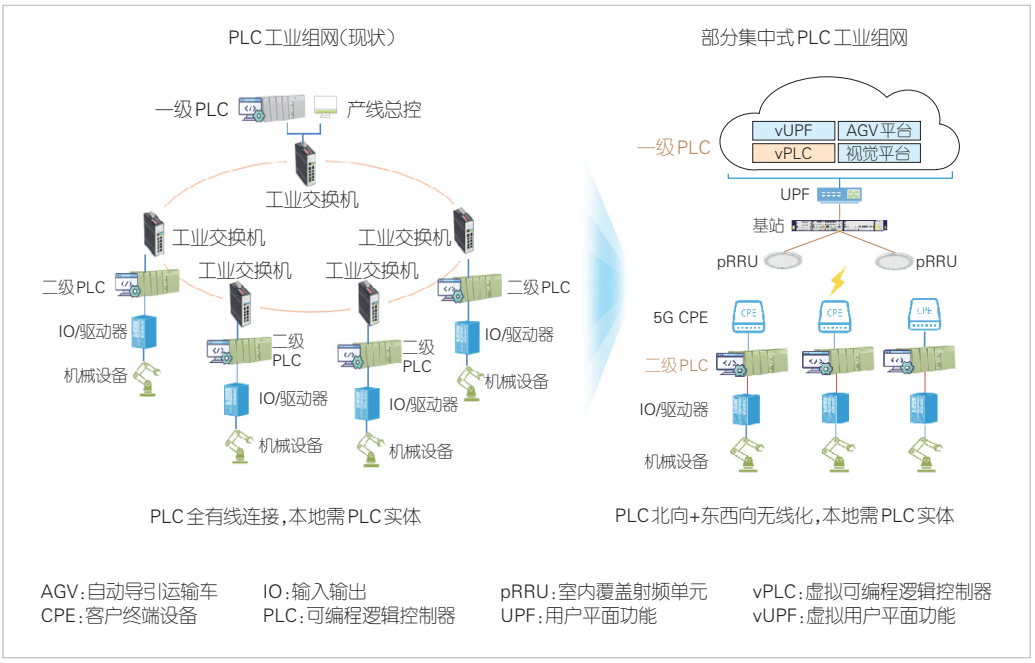
自工业自动化技术发展以来，工业界从未停止对控制器的创新。随着服务器能力的不断提升，用服务器承载所有控制实体的集中化组网架构逐渐进入工业现场。工业集中化组网架构以统一的服务器为载体，将网络进程间通信（IPC）、设备控制器、PLC等进行云化/集中化部署。在该架构下，控制器实体部署在工位内部，不同级的设备间为有线连接，未能做到产线级乃至工厂级的完全云化/集中化部署，因此也就无法解决不同网络间、不同工控设备间的数据互通问题。

5G因其统一的标准、极致的性能、成熟的上下游产业链及生态，成为业界探索工业控制网络无线化的主赛道。近几年，“5G+云化PLC”成为通信界的研究热点。如图1所示，一级PLC被云化/集中化部署到用户端口功能（UPF）/多接入边缘计算（MEC）等边缘算力内；二级PLC依然采用有线分布式的方式来部署，实现了部分云化/集中化。该架构在一定程度上简化了工业网络架构，实现一级PLC间的协同增效。但工业现场存在多张网络，仍无法解决不同网络间、不同工控设备间数据互通的一系列问题。同时，二级PLC与输入输出（IO）/驱动器之间仍然采用有线连接，仅实现了部分柔性。

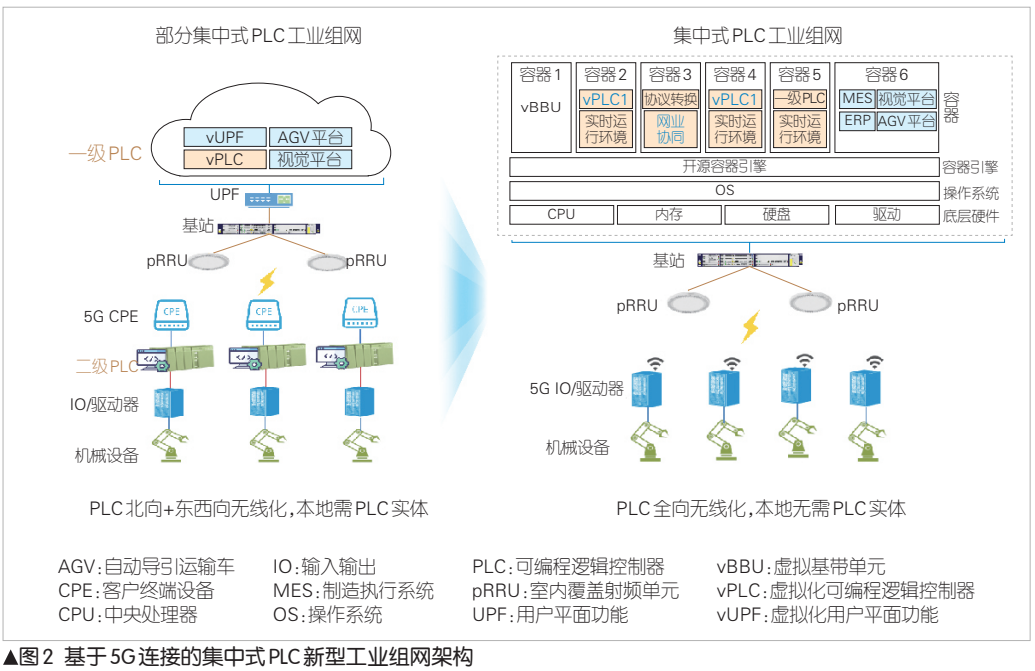
2 新型工业集中控制组网架构

为进一步简化工业现场组网，实现全向柔性，本文提出了基于5G连接的集中式PLC新型工业组网架构，如图2所示。该架构基于工业现场的边缘算力载体来部署云化PLC，通过5G网元（如工业现场基站、边缘计算平台、园区用户端口功能等）与PLC的集成部署与协同优化，实现基于软性确定性连接的全局集中式协同控制。

集中式PLC组网架构采用统一云底座，一方面通过软件化、虚拟化实现虚拟化PLC（vPLC），取代现场所有的硬件PLC，实现PLC的全向无线化；另一方面，vPLC的集中化带来组态逻辑设计的转变，使高效的协同互动替代了静态的流



▲图1 通信界对控制集中化组网架构的探索



▲图2 基于5G连接的集中式PLC新型工业组网架构

程规划，从而提升生产效率。该架构通过统一开放的编程接口实现集中化的调试升级，提升运维效率。同时，该架构依托边缘算力集中化部署协议转换，可以降低端侧复杂度，以及行业终端成本。

▲图3 集中式PLC新型工业组网架构下的关键使能技术体系

3 面临的挑战及关键技术体系

3.1 面临的挑战

目前，基于5G连接的集中式PLC新型工业组网架构面临如下4个方面的挑战：

1）如何保障端到端传输的确定性？PLC部署位置的变化对5G系统端到端的确定性传输提出了更高的要求：一方面PLC与IO设备间的接口从有线连接变成了无线连接，控制指令在无线空口环境中传输的确定性面临极大挑战；另一方面，PLC虚拟化后所在宿主操作系统的实时性能否满足PLC业务的需求，也是需要考虑的内容。

2）如何进一步降低部署及使用成本？PLC形态的变化节省了PLC实体设备的成本，但5G复杂的网络架构及高昂的终端价格也在很大程度上影响了5G深入工业现场的节奏。针对工厂局域场景，亟需一种低成本、极简的5G端到端组网方案。

3）如何高效使能柔性生产？现阶段，工业生产模式逐渐由单一产品的大规模量产逐渐转向消费者定制的个性化生产，由此衍生了柔性制造的概念。柔性制造对生产线生产设备快速灵活的调整、产线组态逻辑按需快速的变化与组合提出了较高的要求。

4）如何保障集中化架构下CT与OT的安全性？集中化架构带来了两种状态的改变：一是PLC设备间原生物理隔离的状态因集中化变为虚拟隔离；二是CT与OT各自独立的状态因集中化部署在同一设备上出现了交集。因此，如何确保容器间的异常不会互相影响、CT与OT之间不能随意互访、边缘算力载体内外数据安全可靠都是需要考虑的问题，因此亟需一种本地化高隔离、高安全的解决方案。

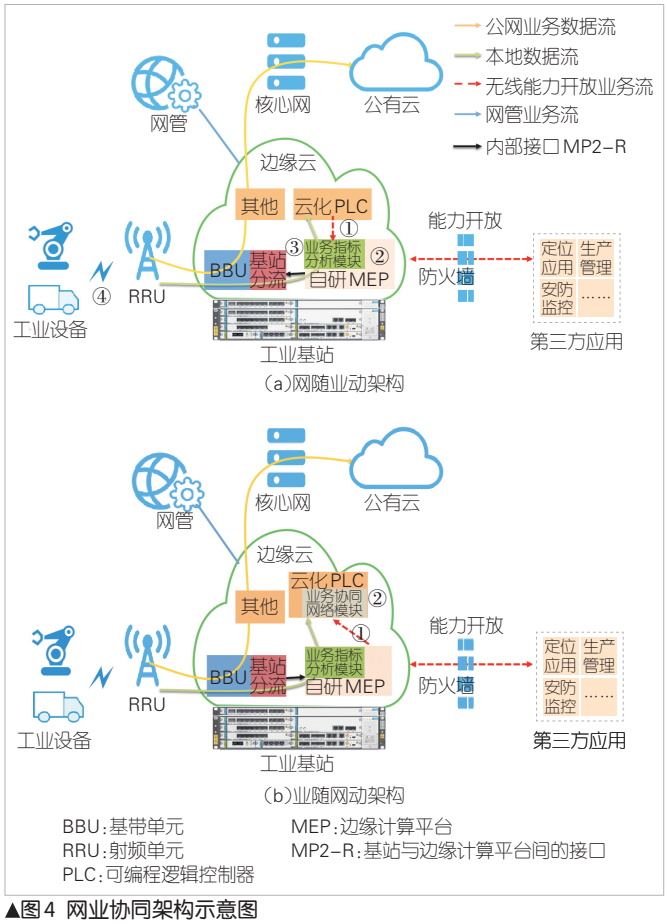
3.2 关键技术体系

针对上述挑战，如图3所示，本文提出集中式PLC新型工业组网架构下的关键使能技术体系。通过在网络侧引入资源自适应的高确定性传输技术、灵活帧结构低时延技术，在端侧引入RedCap技术，构建高确定、低时延、低成本的连接层；通过引入多层级安全隔离技术、基于实时能力增强的云平台技术，构建安全、实时的平台层。基于增强的5G连接层与平台层，网络与业务可以进行协同优化，业务逻辑可以按需灵活编排，工业应用可以按需一体化部署，从而加速工厂各生产要素的“全连接”。

1）连接层增强技术。连接层通过引入网业协同、低时延、RedCap等端到端增强技术，具备高确定、低时延、低成本的特点，高效使能柔性制造。

a）基于资源自适应的高确定性传输技术——网业协同。一方面，网随业动，在集中式PLC组网架构下，5G网络通过网业协同，对业务流进行智能识别及精准调度，同时通过全局的流量编排和门控来减少端到端的抖动，使5G具备大部分场景下对确定性的要求；另一方面，业随网动，通过本地能力开放将网络状态、资源使用情况、排队等待时间等告知业务，业务结合网络情况调整发包特征。网业协同架构图如图4所示。

b）灵活帧结构等低时延技术。一方面，5G系统持续增强空口原子能力，降低端到端的平均时延，提升可靠性。其



▲图4 网业协同架构示意图

中，降低时延的关键原子能力包括超短帧（DS帧结构）、非时隙调度（Mini-Slot）、半持续调度/免授权调度（SPS/CG）以及双激活协议栈（DAPS）等，提升可靠性的关键原子能力包括低码率调制解调（MCS）、时隙（Slot）重复以及分组数据汇聚协议（PDCP）复制[2]等。另一方面，5G系统采用分级分档技术体系[3]将众多原子能力根据不同的业务需求进行灵活组合，以提供差异化的业务保障。

c）RedCap。通过引入RedCap IO等低成本5G终端，大幅降低端侧成本。

2）平台层增强技术。平台层通过引入多种实时性技术、多层级安全隔离技术，保障集中化架构下OT/CT业务的实时性、安全性。

a）实时性关键技术。可以采用实时操作系统（RTOS）、打实时（RT）补丁、绑核等技术来减少宿主操作系统在任务处理时延及任务调度等方面对云化PLC运行的影响。

b）多层级安全隔离技术。在边缘算力载体内，针对集中式PLC组网架构下的边缘算力载体，需实现APP的安全隔离、安全管理，以及生命周期安全。同时，可以采用传统容器隔离、镜像安全等技术来提升容器安全可靠性。在边缘算力载体与其他电信设备之间，边缘算力载体在对接其他电信设备时可采用不同网络平面，或使用不同业务IP段来实现隔离。边缘算力载体对外可通过防火墙实现外网隔离。解决方案如图5所示。

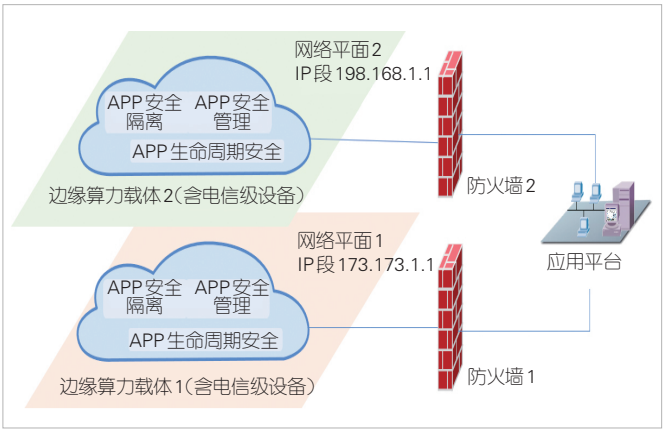
3）应用层增强技术。基于统一云底座，应用层可实现业务逻辑灵活编排、行业应用按需部署，降低部署及使用成本。

a）业务逻辑灵活编排。集中式PLC组网架构采用云化技术部署后，不再受物理分布影响，可对PLC的业务逻辑进行整合与拆分，按需编排，灵活设计。

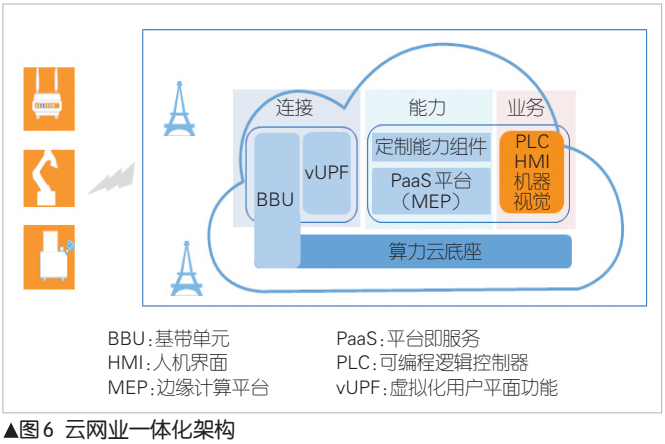
b）行业应用按需部署。承载集中式PLC组网架构的边缘算力平台采用统一云底座[4]，如图6所示。虚拟化技术可以对基站硬件资源进行统一的池化管理，根据业务需求实现按需弹性扩展、快速部署以及业务间的隔离性要求。依托网云业一体化的架构，基站可以灵活集成各种多样化的工业应用，更好地实现设备快速部署、业务快速开通，降低使用成本。

4 新型工业集中控制架构在家电行业中的测试与验证

为了验证集中式PLC组网架构在实际场景中的作用，我

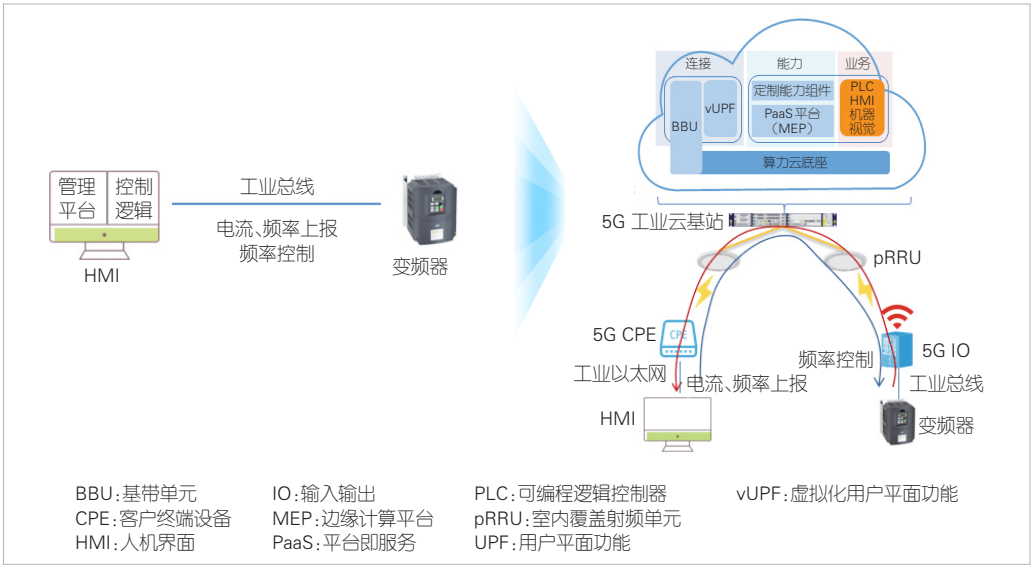


▲图5 容器形态下的集中式PLC架构安全隔离解决方案



▲图6 云网业一体化架构

们在某家电制造工厂进行新型组网架构测试验证。本次试点为生产线皮带线速调节场景，在产线工人数量、产品规格、产品生产速度、产品合格率变化时，产线生产需要根据生产资源和生产要求的变化进行调节。每段产线的生产线速由一个变频器控制，并通过PLC控制多个变频器的频率，进而调整皮带线速。由于工厂环境限制，需要产线变频器频率实现远程调节，保证在高温等恶劣生产条件下，正常进行生产节奏调节。



▲图7 变频器调节场景改造前、改造后架构示意图

在工厂的原有架构中，变频器的频率可以通过产线人机界面（HMI）进行调节。HMI以有线方式连接至产线各变频器，集成了管理平台和变频器控制单元。变频器控制单元与HMI紧耦合，无法实现HMI管理平台远程控制变频器频率。

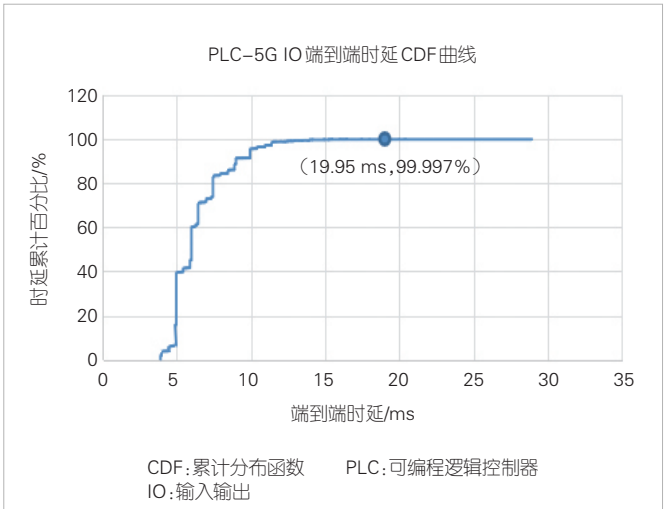
在本案例中，为实现变频器远程调节，应用5G云化工业基站对产线进行改造。5G工业云基站通过容器化方式部署了vPLC，使产线的控制能力集中在工业基站中。为满足PLC对实时性的需求，基站的操作系统需要进行实时性增强（打RT kernel补丁），同时为PLC所在容器分配了固定的中央处理器（CPU）核及存储空间，以保证PLC程序在独占的资源中以高隔离性运行。

在产线改造的过程中，通过工业基站将HMI的控制单元与管理平台分离，将管理平台无线化迁移到PC端，将PLC迁移至工业基站中，实现集中化部署。通过5G网络控制变频器频率调节和状态上报，实现变频器频率调节无线化，因此工厂可基于远端操作界面进行频率调节。变频器调节场景改造后的架构如图7所示。经压力测试，PLC南向无线网络可靠性可达到99.99%@20 ms，如图8所示，满足产线业务需求。

本次试点通过5G工业云基站的应用，可以在以下几方面帮助工厂实现降本增效：

1）PLC的软件化和无线化可以减少生产现场中PLC、线缆等硬件部署，降低产线组网成本；

2）PLC的集中化和无线化、HMI和PLC解耦分离简化了组网架构，使组网更加扁平，从而减少不必要的层级交互，提升整体效率；



▲图8 压力测试结果

3）PLC南向和HMI无线化可以实现HMI远程操作，使产线部署和控制更加灵活，赋能柔性制造；

4）PLC的云化部署可以采取线上、远程等方式进行问题检测和软件升级，更加高效地推动工业控制运维向智能化的方向发展。

5 结束语

本文围绕“基于软性确定性连接的全局集中式协同控制”思路，提出基于5G连接的集中式PLC工业组网新型架构。在新架构下，首先，通过空口增强、网业协同等保障控制信令的端到端确定性传输；其次，基于统一的云底座，将

核心控制器件软件化集中部署，可以实现PLC逻辑按需编排、灵活设计，真正使能柔性制造；最后，网云业一体的设计，支持行业应用按需部署、数据本地流转，从而实现多层级安全保障，为行业提供了一种高安全性的本地低成本解决方案。

未来，我们期待与产业链各领域合作伙伴一起，推动5G与OT从“互通”走向“融合”，探索工业控制协议国产化。在该阶段，以5G为基础面向工控进行定制增强，联合产业定义工业应用协议标准，与底层无线协议内生融合，形成性能极致、更具鲁棒性的自主可控的工业无线控制系统。同时，为更好地发挥5G作为新型基础设施的作用，需要推动工控设备内生5G，提升现场设备的智能化，助力制造强国的实现。

参考文献

[1] 汪晋宽, 马淑华, 吴雨川. 工业网络技术 [M]. 北京: 北京邮电大学出版社, 2007
[2] 中国移动通信有限公司研究院. 面向URLLC场景的无线网络能力白皮书 [R]. 2020
[3] 孙朝, 徐芙蓉. 5G网络业务分级分档保障方案研究, 电信科学, 2022, 38 (S1): 134–142
[4] 中国移动通信有限公司研究院. 5G智简行业网白皮书 [R]. 2021

作 者 简 介

邓伟，中国移动研究院无线与终端技术研究所所长；长期从事3G、4G、5G相关策略和技术研究，先后作为中国移动集团4G/5G项目的项目经理和项目总监，组织开展技术和策略研究、规模试验和商用攻关等工作，推动TD-LTE全球成功商用，以及5G在技术、产业、商用上的全球引领；获得多次省部级科技奖项。

于天意，中国移动研究院无线与终端技术研究所研究员；主要从事5G工业互联网关键技术研究，包括集中式PLC组网架构、5G与工业融合演进协议等。

侯庆东，中国移动研究院无线与终端技术研究所研究员；现从事5G面向行业落地应用的研究工作，主要研究方向为5G无线网络架构、5G+工业互联网等。

基于数字子载波和概率整形的相干光通信系统设计及应用

Coherent Optical Communication System Based on Digital Subcarrier and Probabilistic Shaping: Design and Application

陆源/LU Yuan[1]，牛文林/NIU Wenlin[2]，
王永奔/WANG Yongben[3]，胡子荷/HU Zihe[3]
（1. 山东省邮电规划设计院有限公司，中国 济南 250101；
2. 中国联通山东省分公司，中国 济南 250000；
3. 中兴通讯股份有限公司，中国 深圳 518057）
(1. Shandong Posts and Telecommunications Planning and Design Institute Co. , Ltd, Jinan 250101, China;
2. China Unicom Shandong Branch, Jinan 250000, China;
3. ZTE Corporation, Shenzhen 518057, China)

DOI：10.12142/ZTETJ.202304014
网络出版地址：http://kns.cnki.net/kcms/detail/34.1228.TN.20230724.1927.013.html
网络出版日期：2023-07-25
收稿日期：2023-06-15

摘要：相比于传统的单载波架构，数字子载波架构更能抵抗光纤传输中的信道损伤。概率整形技术相比于传统的常规调制格式可以更有效地抵抗光放大器引入的噪声。阐述了数字子载波和概率整形技术的原理以及它们在800G系统中的应用。数字子载波复用技术和概率整形技术的联合使用，可以进一步提升系统的可重构光分插复用器穿通能力和传输容量。现网测试验证了两种技术在800G相干系统中的优越性。

关键词：数字子载波复用；概率整形；相干光通信；现网测试

Abstract: Compared with the traditional single-carrier architecture, the subcarrier architecture is more resistant to damage in optical fiber transmission. Probabilistic shaping technology can resist noise more effectively than traditional conventional modulation formats. The principles of these two new technologies and their applications in 800G systems are discussed. The combination of digital subcarrier multiplexing technology and probabilistic shaping technology can further enhance the system´s reconfigurable optical add-drop multiplexer passthrough capability and transmission capacity. The network testing has verified the superiority of these two technologies in 800G coherent systems.

Keywords: digital subcarrier multiplexing; probabilistic shaping; coherent optical communication; network test

引用格式：陆源，牛文林，王永奔，等. 基于数字子载波和概率整形的相干光通信系统设计及应用 [J]. 中兴通讯技术, 2023, 29(4): 78-82. DOI: 10.12142/ZTETJ.202304014
Citation：LU Y, NIU W L, WANG Y B, et al. Coherent optical communication system based on digital subcarrier and probabilistic shaping: design and application [J]. ZTE technology journal, 2023, 29(4): 78-82. DOI: 10.12142/ZTETJ.202304014

光通信在过去几十年里得到了巨大发展。就在10年前，主流的商用系统还是直接探测的10G系统。如今，互联网时代的流量大爆发推动了100G系统的迅速发展。100G系统的实现得益于相干探测和数字信号处理（DSP）以及高速模数转换（ADC）/数模转换（DAC）的联合使用[1]。

随着5G时代的到来，运营商对系统的容量需求急剧增加，互联网流量增速已大大超过光网络传输容量的增速。为了应对容量需求的进一步增长，未来的光通信系统有以下几个研究方向：1）基于灵活和感知光网络的高效资源管理；2）采用更高阶的调制格式和更高的波特率；3）采用更先进的DSP和前向纠错码（FEC）算法；4）支持光纤多波段的传输器件；5）少模多芯等新型光纤的应用。高效的光网络资源管理是基础，因为它可以通过高效地利用现有的光纤基础设施，来推迟昂贵的新光纤和新器件的部署。采用更高阶的调制格式（例如32符号正交幅度调制）和更高的波特率（>90 GBd）可以实现单波800 Gbit/s的传输速率。而这会导致更严重的信道损伤，例如窄带滤波效应、器件非线性效应以及I/Q之间的时延效应。这些都需要更复杂的DSP算法以及更先进的FEC方案进行补偿。此外，采用先进的数字通信技术（例如概率整形和数字子载波复用）既可以实现灵活的频谱效率，也可以提升传输性能。数字子载波复用技术更适用于灵活光网络，可以更方便地处理信道的动态业务。此

外，使用C+L波段还可以进一步提升整根光纤的传输容量。

本文重点介绍数字子载波技术和概率整形技术的理论背景以及它们在单波800G系统中的优势，并给出中国联通研究院携手中兴通讯完成的单波800G现网测试结果。

1 数字子载波复用技术

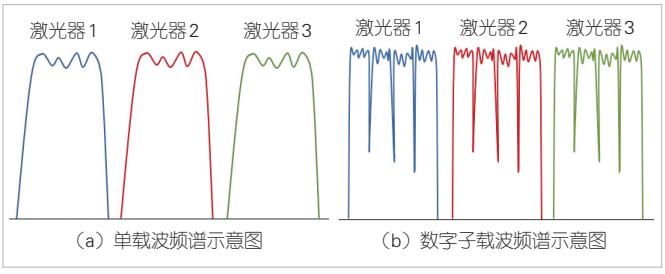
在传统的相干光通信系统中，单个激光器产生的光信号占据一段连续的频谱，如图1（a）所示。而对于数字子载波复用技术，在发端通过数字域上的特殊处理，可以将频谱上连续的光载波分割成若干个独立的奈奎斯特子载波，如图1（b）所示。子载波的个数可以根据总的波特率和应用场景来设定，一般选择4、8或者16。

1.1 数字子载波的优势

（1）降低色散补偿复杂度

目前大部分相干光通信系统会在收端数字域上补偿光纤链路累积的色散。主流的色散补偿算法对信号做快速傅里叶变换（FFT），并在频域上进行重叠存储补偿。补偿色散需要的滤波器阶数与色散和波特率大小密切相关[2]。假设采用4个子载波，在总波特率相同的情况下，每个子载波的波特率是单载波的1/4，那么在色散补偿时需要的滤波器阶数就只有单载波的1/16。这可以大大节省色散补偿所需要的硬件资源。

除了降低色散补偿的复杂度，数字子载波技术也可以降低色散补偿导致的传输代价。对于目前收端通用的相干DSP处理结构，均衡增强相位噪声（EEPN）效应并没有得到很好的解决[3]。EEPN效应主要是由收端激光器线宽导致的相位变化和色散补偿滤波器的共同作用引起的。当采用低阶调制格式且波特率较小时，EEPN的影响很小，可以忽略。然而，随着高阶调制格式的引入和波特率的上升（如上升到64 GBd和128 GBd），EEPN导致的色散补偿代价就不能再忽略。例如，对于64 GBd的400G 16QAM传输系统，当收端激光器线宽为300 kHz，累积色散为20 000 ps/nm时，EEPN带来的光信噪比（OSNR）代价有0.2 dB。而如果采用数字子载波技术，当每个子载波的波特率为16 GBd时，EEPN的代价就不到0.05 dB。



▲图1 单载波频谱与数字子载波频谱

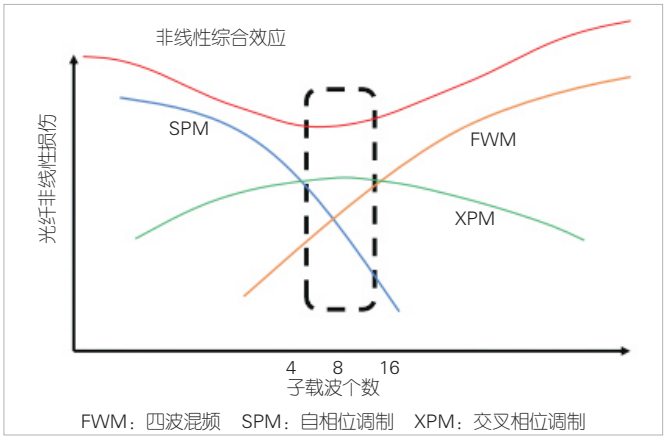
（2）减少非线性传输代价

光纤中的非线性效应，包括自相位调制（SPM）、交叉相位调制（XPM）、四波混频（FWM），已经成为限制光通信系统传输距离的主要因素之一。而光纤的非线性损伤补偿也是业界面临的挑战之一。一些有效的非线性补偿算法如数字反向传输补偿算法（DBP），因为复杂度太高而无法在硬件上实现，不具有实用性。因此，有研究人员提出，不采用补偿方法，直接从源头降低非线性效应。这也是数字子载波技术诞生的原因之一。

通过优化每个子载波的波特率、子载波的个数，数字子载波复用技术使SPM、XPM和FWM这3种非线性效应的综合效应最小，从而减少非线性的损伤。如图2所示，FWM效应随着子载波个数的增加而增加，SPM效应随着子载波个数的增加而减少，XPM效应介于两者之间。可以看出，子载波数目存在一个最优区间，它能使得三者的综合效应达到最小。文献[4]深入分析和比较了多子载波和单载波的非线性传输性能。实验结果表明，与单载波相比，通过优化子载波数目，100G正交相移键控（QPSK）采用数字子载波技术可以额外传输25%的距离。

（3）提高频谱利用率

在发射端对数字信号做奈奎斯特整形，可以有效减少信号占用的频谱带宽。这项技术已经被普遍应用在单载波系统中。由于在收端需要恢复信号时钟，因此发端奈奎斯特整形的滚降系数不能为0（一般设置在0.1~0.2之间）。在数字子载波系统中，当其中一个子载波的滚降系数设置高一些（用来做时钟恢复）时，其余子载波的滚降系数就可以设置低一些。假设一个32 GBd的单载波系统的滚降系数为0.2，那么其占用的总频谱带宽为32 × 1.2=38.4 GHz。对于数字子载波系统，假设子载波数目为4，每个子载波的波特率为8 GBd，



▲图2 光纤不同非线性效应与子载波个数关系

用来做时钟恢复的子载波的滚降系数为0.2，其余子载波的滚降系数为0.025，那么其占用的总频谱带宽为8×1.2+24×1.025=34.2 GHz。这样总的频谱带宽占用就比单载波系统少4.2 GHz，有效提高了频谱效率。

1.2 数字子载波系统和单载波系统仿真性能对比

为了进一步验证数字子载波架构相比于单载波架构的性能优势，我们在800G系统下对两种架构做了仿真研究，选择调制格式为PM-16QAM，波特率为128 GBd，子载波个数为16。背靠背条件下子载波架构和单载波架构的OSNR曲线如图3所示。其中，蓝色曲线表示在128 GBd PM-16QAM理论情况下误码率（BER）随OSNR的变化曲线。当软判决前向纠错（SD-FEC）的纠错门限为0.02时，其对应的OSNR容限为22.8 dB。黑色和红色曲线分别代表单载波128 GBd PM-16QAM和16×8 GBd PM-16QAM子载波系统下的BER随OSNR的变化曲线。两种方式对应纠错门限的OSNR都在23.3 dB左右，与理论值相比代价约为0.5 dB。因此，在背靠背传输条件下，子载波系统的OSNR代价与单载波基本相当。

在长距离传输情况下我们对比了两种不同架构的性能。图4中黑色曲线表示单载波128 GBd PM-16QAM传输下BER随传输距离变化的曲线，其BER达到0.02时对应第22跨段（1 760 km）。红色曲线代表子载波系统中BER（16个子载波的平均值）随传输距离变化的曲线，其BER达到0.02时对应第27跨段（2 160 km）。仿真结果表明，16×8 GBd子载波系统明显要优于单载波系统。在达到相同BER要求的情况下，子载波系统要比单载波系统多传输4~5个跨段（320~400 km）。这是因为子载波架构的传输方式对光纤传输中产生的非线性效应有较好的容忍性。

2 概率整形技术

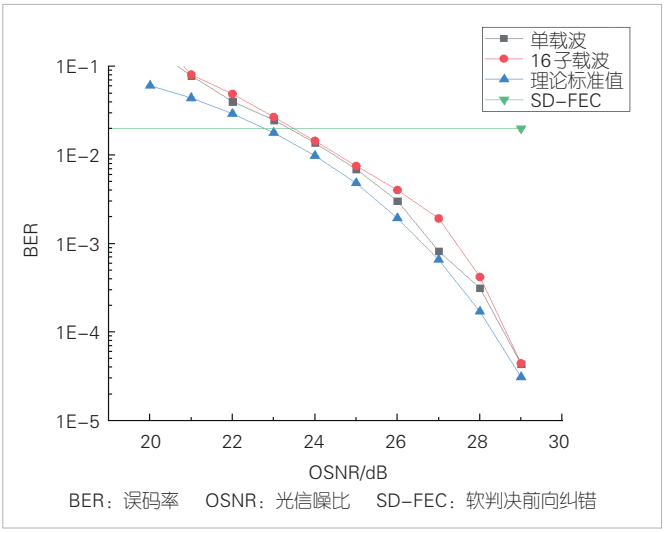
2.1 概率整形原理

1948年，香农在信息论中提出：当信源的分布符合信道分布时，信道的传输容量达到最大。类似地，在光纤通信系统模型中，当发端的离散星座点近似高斯分布时，传输性能达到最优[5]。传统的QAM调制格式的星座点是均匀分布的，每个点出现的概率相同。概率整形技术通过改变星座点的概率让其近似高斯分布，进而提升传输性能。在概率整形技术中，星座图中各个星座点的间隔是等距的，但是每个星座点具有不同的概率。相比于传统星座点的等概率分布，概率整形技术可以让能量低的符号比能量高的符号出现的次数更多，即能量低的星座点出现概率大。这样可以降低平均发射功率，有效应对噪声等因素带来的损伤。概率整形16QAM的互信息高于均匀16QAM，更加接近香农容限。

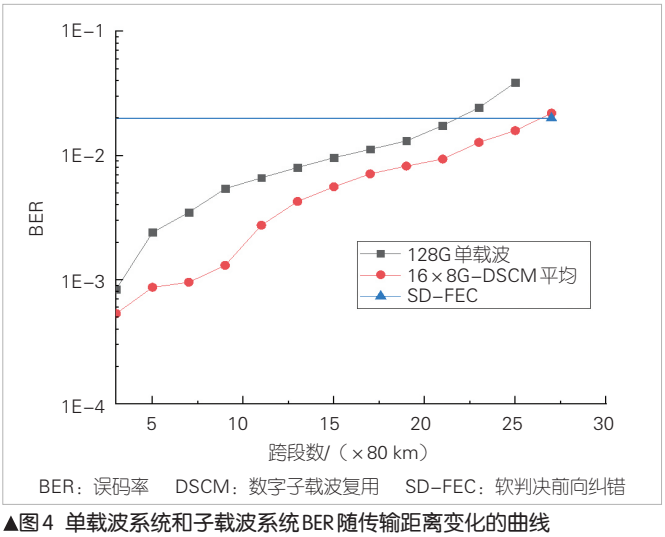
概率整形的另一个优点是可以实现灵活的频谱效率调整，以适应不同的信道环境。改变概率编码模块的输入输出比例有助于实现不同方差的高斯分布，调整有效的传输比特个数。不同的频谱效率可以应用于200~800 Gbit/s的传输速率，满足长距离骨干、中距离城际以及短距离数据中心等不同应用场景的需求。

2.2 概率分布匹配器的实现

分布适配器（DM）是概率整形能够实现的关键技术。它的作用是让均匀分布的01输入比特变成非均匀的映射符号。目前业界的主流DM算法为恒定成分分布适配器（CCDM）算法。该算法的性能好，编码损失小，但是计算



▲图3 背靠背单载波系统和子载波系统BER随OSNR变化曲线



▲图4 单载波系统和子载波系统BER随传输距离变化的曲线

量非常大，主要用于离线实验研究。文献[6]提出了一种基于级联查找表形式的DM方案（HiDM）。该方案不需要高精度的乘法运算，使用二叉树形式的并行结构，因此非常适用于硬件实现。以82输入128输出概率整形比例为例，我们设计了一个5层级联查找表的DM实现方式，具体结构如图5所示。该结构主要具有以下几点特征：

1）第n+1层的查找表个数是第n层的2倍，可形成二叉树形式级联，即$T_{n+1}=2T_n$；

2）每层所有的查找表都输出8个比特，即$U_n=8$；

3）第n+1层的查找表输入由第n层的一半输出（图中红色比特值）和该层的净荷（图中蓝色比特值）共同构成，其中净荷处于低位。此时，$V_{n+1}=S_{n+1}+U_n/2$；

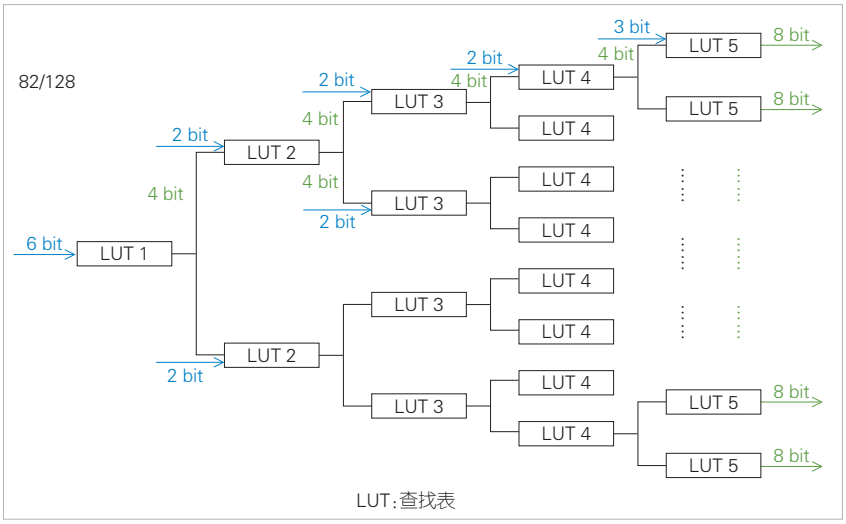
4）总净荷比特数为$\sum T_n \times S_n$。

2.3 基于HiDM概率整形的FPGA验证

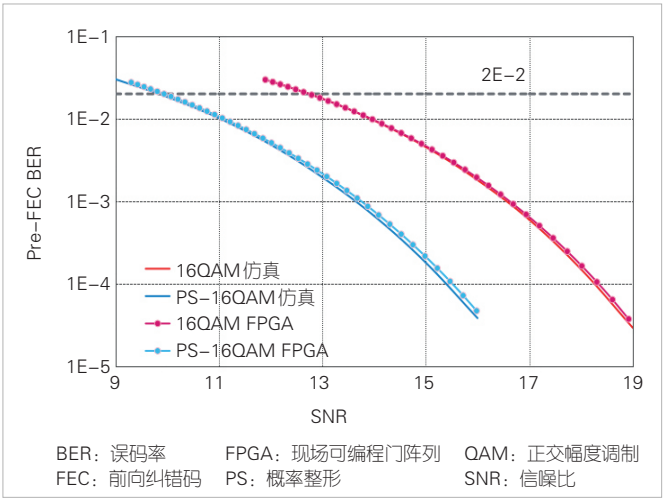
我们做了仿真和现场可编程门阵列（FPGA）实验来比较概率整形16QAM的性能优势。仿真和FPGA实验结果如图6所示。可以看出，FPGA实验结果与仿真结果高度吻合。在FEC纠错门限2E–2处，相比于均匀16QAM，PS–16QAM大概有2.9 dB的信噪比（SNR）优势。在考虑概率整形编码冗余之后，PS–16QAM仍然有1.84 dB的净速率优势。

3 数字子载波技术与概率整形的结合

在长距离传输中，有些场景会级联很多个可重构光分插复用器（ROADM），这会增加滤波效应，进而影响传输性能。在正交频分复用系统（OFDM）中，这种频率相关的信噪比衰减损伤通常采用注水算法进行补偿，即不同的频段加载不同的调制格式[7]。数字子载波和概率整形的结合就可以达到类似的效果。对于靠近中心的子载波，我们可以调制频谱效率高的弱整形64QAM；对于频率衰减比较严重且远离中心的子载波，我们可以调制频谱效率低的强整形64QAM，以应对更高的信道损伤和畸变。



▲图5 82输入128输出的HiDM结构图

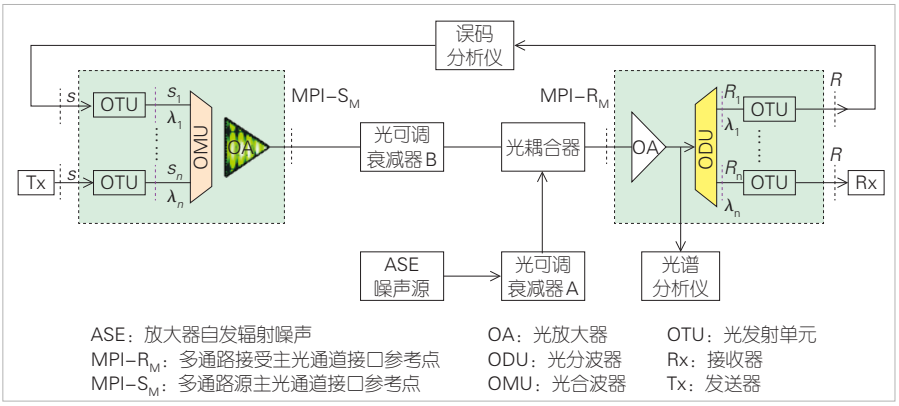


▲图6 PS–16QAM和16QAM仿真和FPGA实验结果对比

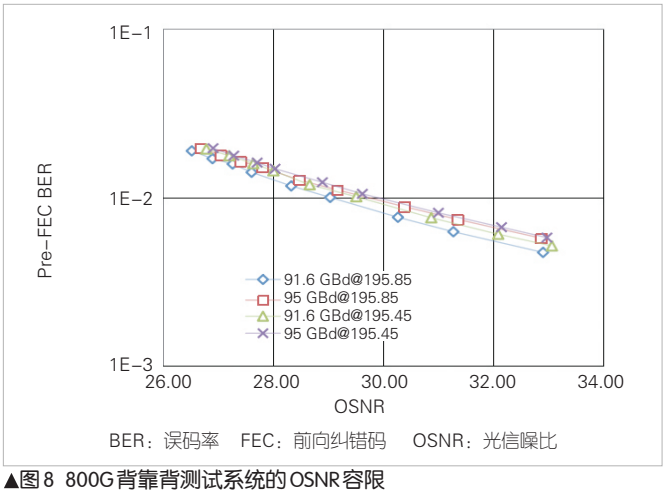
4 基于数字子载波和概率整形技术的800G现网测试

山东联通携手中兴通讯进行了单波800G现网传输测试。该测试采用新一代基于数字子载波和概率整形技术的800G光传输网（OTN）板卡。背靠背测试系统配置如图7所示。该系统最高波特率为95 GBd，调制格式为PS–64QAM。

图8给出了背靠背OSNR–PreBER测试结果。单波800G的OSNR容限约为26.2 dB，对应纠错门限为1.9E–2。通过使用色散预补偿和数字子载波技术，该系统支持404 000 ps/nm的色散补偿容限。得益于数字子载波的带宽优势，当采用95 GBd的波特率时，800G速率只需要100 GHz的通道间隔，进一步提升了频谱利用率。引入C++波段可提升传输带宽，使整根光纤的传输容量达到48 Tbit/s。通过采用800G PS–64QAM调制方式和C++传输系统，现网800G传输完成了1×40、2×40、3×40、1×90 km等应用场景测试。能达到这样的测试效果与概率整形、数字子载波这两项关键技术密切相关。如果采用常规的均匀调制64QAM，800G速率的OSNR容限在仿真系统中会高达27.5 dB。概率整形将OSNR需求至少降低1.3 dB。数字子载波的引入同样显著提升了色散补偿容限。常规单载波架构系统的



▲图7 800G背靠背测试配置



▲图8 800G背靠背测试系统的OSNR容限

色散补偿能力一般只有20 000～30 000 ps/nm。测试充分验证了传输后OSNR、长期稳定性、保护倒换能力、频谱分配等系统传输指标。结果表明，各项性能指标表现良好。

5 结束语

本文讨论了数字子载波复用技术和概率整形技术在高速相干光通信系统中的优势。数字子载波技术可以大幅降低色散补偿复杂度，使非线性损伤容忍度高于单载波，还可以进一步节省带宽，提高频谱效率。概率整形技术可以进一步接近香农容限，实现频谱效率的灵活可调。数字子载波复用技术和概率整形技术的联合使用，能够提升系统的ROADM穿通能力和传输容量。现网测试结果表明，这两项技术在800G相干系统中具有很高的优越性。

参考文献

[1] ROBERTS K, O'SULLIVAN M, WU K T, et al. Performance of dual–polarization QPSK for optical transport systems [J]. Journal of lightwave technology, 2009, 27(16): 3546–3559. DOI: 10.1109/JLT.2009.2022484
[2] GEYER J C, FLUDGER C R S, DUTHEL T, et al. Efficient frequency domain chromatic dispersion compensation in a coherent Polmux QPSK–receiver [C]// Proceedings of 2010 Conference on Optical Fiber Communication (OFC/NFOEC), collocated National Fiber Optic Engineers Conference. IEEE, 2010: 1–3
[3] SHIEH W, HO K P. Equalization–enhanced phase noise for coherent–detection systems using electronic digital signal processing [J]. Optics express, 2008, 16(20): 15718–15727. DOI: 10.1364/oe.16.015718
[4] POGGIOLINI P, NESPOLA A, JIANG Y C, et al. Analytical and experimental results on system maximum reach increase through symbol rate optimization [J]. Journal of lightwave technology, 2016, 34(8): 1872–1885. DOI: 10.1109/JLT.2016.2516398
[5] CHO J, WINZER P J. Probabilistic constellation shaping for optical fiber communications [J]. Journal of lightwave technology, 2019, 37(6): 1590–1607. DOI: 10.1109/JLT.2019.2898855
[6] YOSHIDA T, KARLSSON M, AGRELL E. Hierarchical distribution matching for probabilistically shaped coded modulation [J]. Journal of lightwave technology, 2019, 37(6): 1579–1589
[7] JANG J, LEE K B. Transmit power adaptation for multiuser OFDM systems [J]. IEEE journal on selected areas in communications, 2003, 21(2): 171–178. DOI: 10.1109/JSAC.2002.807348

作 者 简 介

陆源，山东省邮电规划设计院有限公司传输网支撑中心主任、中国联通高级工程师、全国优秀通信设计工作者；主要从事干线传输和本地传送网工程项目的咨询设计与管理工作；主持设计的项目获省部级奖项20余项，参与制定通信行业标准1项，发表论文20余篇，拥有国家专利2项。

牛文林，中国联通山东省分公司云网运营中心云网承载组组长、技术总监，中国联通B级传送网专家人才，中国联通高级工程师；具有丰富的网络建设和管理经验，担任多年省公司专家组成员，在推动数字化转型、网络能力产品化、智能化运营等方面做出突出贡献；带领团队获得2020年信息通信行业“质量信得过班组”。

王永奔，中兴通讯股份有限公司算法系统工程师；主要从事相干光通信DSP算法的研究与应用。

胡子荷，中兴通讯股份有限公司算法系统工程师；主要从事相干光通信DSP算法的研究与应用。

中兴通讯技术杂志社
促进产学研合作青年专家委员会

刊物相关信息

投稿须知　投稿平台　过刊下载　论文索引与引用指南

中兴通讯技术

(ZHONGXING TONGXUN JISHU)

办刊宗旨：

以人为本，荟萃通信技术领域精英
迎接挑战，把握世界通信技术动态
立即行动，求解通信发展疑难课题
励精图治，促进民族信息产业崛起

双月刊　1995 年创刊　总第 171 期
2023 年 8 月　第 29 卷 第 4 期

主管：安徽出版集团有限责任公司
主办：时代出版传媒股份有限公司
　　深圳航天广宇工业有限公司
出版：安徽科学技术出版社
编辑、发行：中兴通讯技术杂志社

总编辑：王喜瑜
主编：蒋贤骏
执行主编：黄新明
编辑部主任：卢丹
责任编辑：徐烨
编辑：杨广西、朱莉、任溪溪
设计排版：徐莹
发行：王萍萍
编务：王坤

《中兴通讯技术》编辑部
地址：合肥市金寨路 329 号凯旋大厦 1201 室
邮编：230061
网址：tech.zte.com.cn
投稿平台：tech.zte.com.cn/submission
电子信箱：magazine@zte.com.cn
电话：（0551）65533356

发行方式：自办发行
印刷：合肥添彩包装有限公司
出版日期：2023 年 8 月 1 日
中国标准连续出版物号：ISSN 1009-6868
CN 34-1228/TN
定价：每册 20.00 元