

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 Kailai¹, 王旭/WANG Xu²,
唐琴琴/TANG Qinqin²

(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

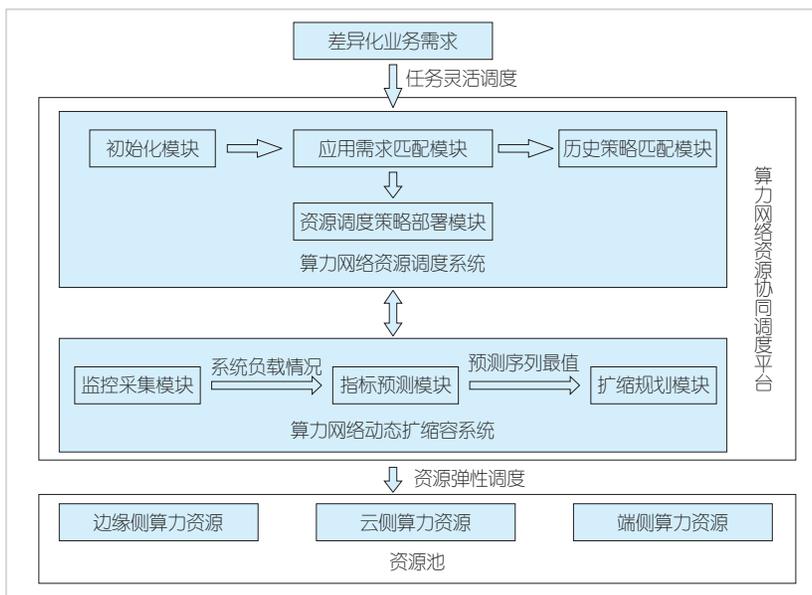
前算力资源不再集中分布在云数据中心，而是广泛分布在大量的边缘节点及海量的终端设备上。算力资源正由集中式云计算转向云边端泛在分布，算力架构逐渐演变为云、边、端三级架构形式^[4]。

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 and 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。



▲图1 算网一体化调度平台技术架构

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 and 用户无感业务扩缩容。

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评估算力资源调度算法。

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。

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 $S = \{S_1, S_2, \dots, S_n\}$,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:

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

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

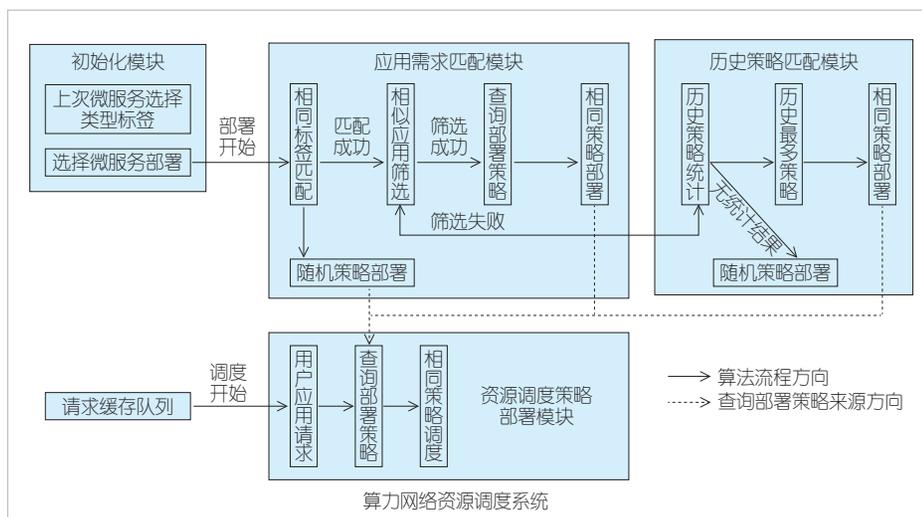
$$\alpha_3 = \frac{S_0 \text{的存储容量需求}}{S_i \text{的存储容量需求}}, \tag{3}$$

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

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



▲图2 多目标优化算力资源调度系统架构图

式为:

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

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

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

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

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 $S = \{S_1, S_2, \dots, S_n\}$ 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, 使得部署微服务也有 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。

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

$$SIM = \frac{1}{3} \sum_{i=1}^3 S_i. \quad (10)$$

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):

$$rss_i = Function(server, service, request), \quad (11)$$

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。

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

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。

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。

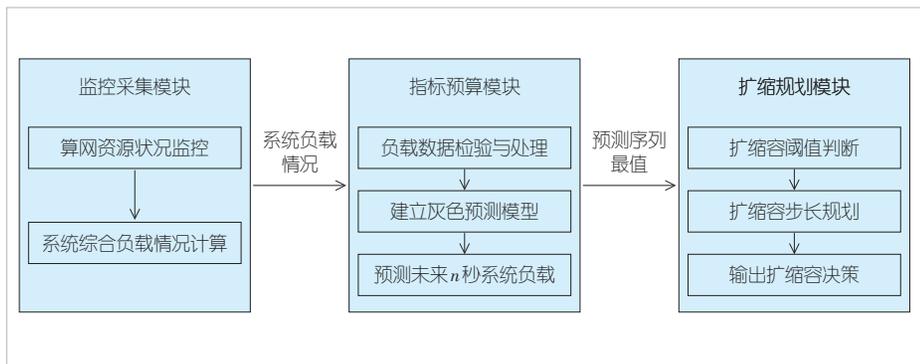
随着系统的运行，时序性的负载指标也就产生了。这些负载指标由当前运行的CPU、内存、磁盘利用率和带宽占用情况共同组成。基于这些负载指标数据，系统可以预测出未来某段时间的系统负载情况，并根据预测的未来负载来进行资源的弹性伸缩。该问题可以看作一个时序预测问题。预测模块的输入为之前一段时间（扩缩容周期或自定义）综合负载率组成的一组时序数据；预测模块的输出为短期的未来时序数据。

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 \times 扩容步长，并选取副本数 \times 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 \times 缩容步长，选取副本数 \times 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



▲图3 算力网络动态扩缩容系统总体架构图

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 构建云边缘协同的开放生态 [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。