计算
Computing Magazine of the CCF

面向云边端分布式协同的智能原生计算

李博睿 王帅 王维龙 刘佳伟 刘天恩

AI-Native Computing for Cloud-Edge-Device Distributed Collaboration

LI Borui, WANG Shuai, WANG Weilong, LIU Jiawei, LIU Tianen

在线阅读 View online: http://js.xml-journal.net/article/doi/10.11991/cccf.202511107

面向云边端分布式协同的智能原生计算

李博睿　王　帅　王维龙　刘佳伟　刘天恩
东南大学

引言

随着信息物理系统（cyber-physical systems, CPS）的快速发展，其复杂度、智能化水平不断提升，对于算力与时延的需求也随之增长。在此情况下，传统以端侧设备或云服务器为核心的集中式计算架构已无法满足物理世界与数字空间融合产生的复杂计算需求，信息物理系统逐渐向云边协同（cloud-edge collaboration）的分布式计算架构靠拢：云计算凭借弹性扩展能力支撑大规模数据处理与模型训练，边缘计算则通过算力本地化部署实现服务延迟的数量级降低，二者协同支撑了从智能交互终端到工业级实时控制的各种信息物理系统应用。

当前的计算系统通常遵循“编程编译—调度部署—更新调试”的范式完成计算任务，云边协同计算也不例外。然而，面对复杂且动态变化的云边端信息物理系统，上述计算范式还无法完全摆脱对人工参与和经验调优的依赖。例如：编程时需要专人理解应用需求并设计软件架构；分布式逻辑的调度部署依赖手工切分或场景特定算法；系统功能与性能调试还依赖专家经验，难以实现和环境动态交互后的自我更新与提升等。这些局限性，大幅提高了构建一个符合应用需求且具有最优性能的信息物理系统的难度。因此，急需一个能够随需求变化自主调整计算任务实现、随环境变化自主完成计算调度部署、视结果反馈自主更新计算行为的新型云边端分布式协同计算架构。

随着人工智能（artificial intelligence, AI）模型的不断演进，上述目标并非遥不可及。通过海量高质量数据的训练，以 DeepSeek、千问（Qwen）等大语言模型（large language models, LLMs）为代表的人工智能模型已拥有较强的人类意图理解能力和多步计划推理能力。此外，借助参数的反向传播与多模态记忆机制，人工智能模型也拥有自主反馈更新的能力。因此，若能借助人工智能模型来表征传统“编程编译—调度部署—更新调试”计算范式中的每一个过程，利用其理解、推理、记忆与动态更新能力来降低计算过程中人在环路的参与比例，实现“自主指令生成—自主调度部署—自主演化更新”，最终达到面向云边端分布式计算过程的智能原生。

智能原生云边端协同计算的由来与愿景

在深入了解智能原生云边端协同计算之前，首先讨论其由来与愿景。随着信息物理系统需求的快速发展，云边端分布式协同系统已逐渐发展成为囊括了云计算、分布式、网络通信、嵌入式系统等多方向交叉的复杂系统。在此场景中，从编程编译到调度部署，再到更新调试的计算任务全生命周期往往过度依赖人工专家知识，不仅导致任务完成错误率高，而且在性能优化与资源调度上存在较大改进空间。针对此问题，学术界与工业界均开始探索智能原生云边端协同计算的概念，即借助人工智能理论与模型原生表征计算的各个

DOI：10.11991/cccf.202511107
基金项目：国家自然科学基金项目（62302096, 62272098, U24B20152）；江苏省自然科学基金项目（BK20230813）
通信作者：王帅，E-mail: shuaiwang@seu.edu.cn

步骤。

引入人工智能来表征计算流程有三个动因：其一，人工智能能够实现完全自主的决策与控制，减少系统对人工干预的依赖，实现“自主指令生成—自主调度部署—自主演化更新”的自主智能计算；其二，面对高度复杂的计算系统，传统的数值分析与优化方法往往难以全面刻画系统参数与整体性能之间的非线性关系，而人工智能模型能够通过大规模数据学习来捕捉这种复杂映射；其三，人工智能具备基于反向传播的自主演化能力，从而使被人工智能表征的计算系统能够不断适应动态多变的运行环境，实现持续优化与进化。

智能原生云边端协同计算的整体愿景框架如图 1 所示。与现有计算系统类似，整体框架自上而下由用户态、内核态、基础设施与硬件设备组成。用户态主要面向应用与计算任务，包含智能原生的自主计算指令生成与自主计算演化更新；内核态主要使用智能原生的方式管理底层的基础设施与硬件设备；基础设施与硬件设备中包含的设备无感知协同运行环境，能够实现云边端协同计算场景中硬件、能力皆异构的分布式硬件设备的自由计算任务部署与迁移。用户态与内核态的三大智能原生计算步骤均依托于人工智能基础模型，因此在云边端协同环境中的推理运行优化也是基础设施的重要组成部分。目前关于基础设施层面的研究已有较多相关综述[1–3]，本研究则聚焦用户态与内核态的计算全生命周期，针对智能原生计算中自主计算指令生成、自主计算演化更新、自主资源调度与部署等工作的发展研究现状及面临的挑战进行梳理与分析，并在此基础上对未来智能原生计算的发展进行展望。

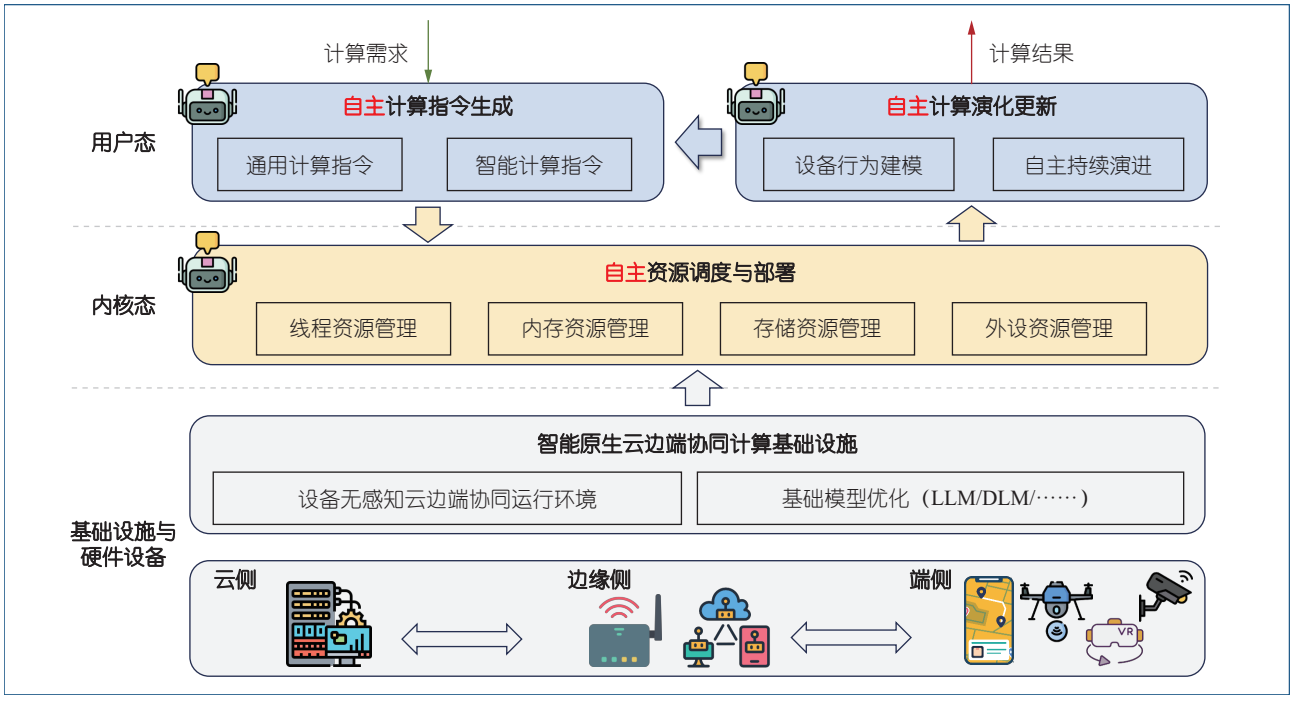

图1　智能原生云边端协同计算愿景框架

智能原生云边端协同计算中的自主计算指令生成

自主计算指令生成面临的挑战

自主指令生成指的是随用户需求自主生成所计算的内容。在智能原生云边端协同计算框架中，根据所生成的内容，自主计算指令生成可进一步划分为“通用计算指令”和“智能计算指令”的自主生成。通用计算指令生成指通过自然语言交互自动生成可执行的程序代码（如 C、Python 等），实现计算逻辑的自动化构建，其核心挑战在于准确理解用户意图并转化为结构正确、功能完整的程序。而智能计算指令生成则专注于对人工智能模型（特别是神经网络）的自动化设计与优化，涉及模型架构搜索、超参数调优等，目标是在满足精度要求的前提下高效生成适配云边端特定硬件约束

的模型。当用户使用日常语言描述需求时，系统须精准理解其语义并将其映射为可执行的计算流程。然而，自然语言本身固有的歧义性易导致用户意图的解析偏差；同时，在云边端协同计算场景中，软硬件的异构性与耦合性更为自主指令生成增加难度。**因此，如何实现软硬耦合下的计算指令的可控生成仍存挑战**。

通用计算指令自主生成

在已有的通过拖拉拽模块进行编程的低代码开发范式的基础上，借助生成式人工智能技术的快速发展，学术界和工业界已经开始积极探索“零代码”应用生成的实现路径，也即通过自然语言描述或简单交互，由 AI 自动完成从需求分析到代码生成、测试部署的全流程应用开发。这些研究工作充分利用了 LLMs 强大的自然语言理解能力和分析能力，以及其内置的海量专业知识储备，目前已经能够实现部分应用场景的零代码开发。例如，斯坦福大学研究团队开发的 TextGrad[4] 框架创新性地采用 LLM 生成的文本反馈作为“梯度”，通过模拟神经网络的反向传播机制来自动优化编程代码，从而实现了代码调试和性能改进的自动化过程。浙江大学研究团队提出的 ChatIoT[5] 系统则利用 LLM 来解析用户的自然语言指令，自动生成适用于智能家居场景的触发–动作规则代码。这两个典型系统都充分发挥了 LLMs 在自然语言理解方面的独特优势，使得非专业用户仅需通过简单的语言描述就能直接获得可执行的应用代码，完全跳过了传统编程的复杂流程。

然而，上述工作并未充分考虑不同物联网设备的特征差异性和软硬件调用的耦合制约问题。因此，本研究提出基于软–硬件耦合特征的零代码应用开发架构。如图 2 中代码构建部分所示，将硬件的物理功能和约束，以及软件的应用功能和约束表征为应用抽象。该应用抽象是一种通用且格式统一的文本表达形式，智能体能够根据用户需求组合生成新的应用抽象。这种方式有助于智能体理解软硬件的特征，提高应用生成准确性。

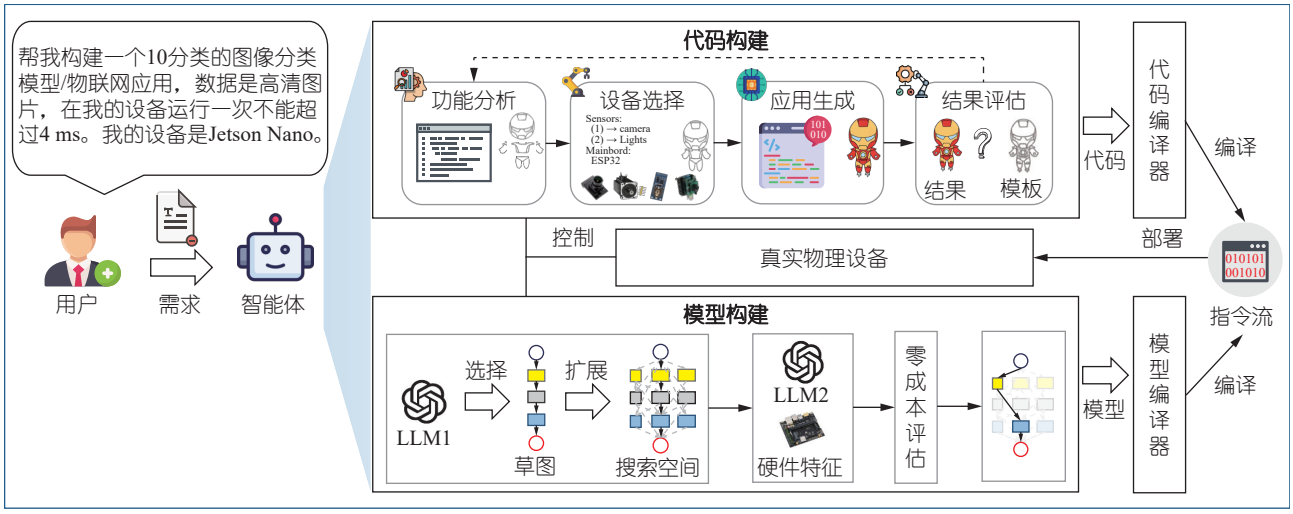

图2　面向自然语言的智能原生计算指令生成

智能计算指令自主生成

在自主模型生成方面，模型架构的设计质量不仅直接影响模型的推理精度，还会显著影响应用系统的执行效率，进而对最终的服务质量产生决定性影响。传统上，要快速设计出满足用户需求的高质量 AI 模型，往往需要开发者具备深厚的模型设计专业知识，并投入大量的训练和评估计算资源。

随着自动化机器学习（automated machine learning, AutoML）技术[6]和神经网络架构搜索（neural architecture search, NAS）技术[7]的持续发展，AI 模型的设计门槛正在逐步降低。例如，阿里巴巴研究团队开发的 Zen-NAS[8] 采用创新的“零样本评估”方法，仅需 0.5 个图形处理器（graphics processing unit, GPU）天即可完成神经网络架构搜索。该技术的核心突破在于通过随机前向推理直接评估模型潜力，完全规避了传统方法中耗时的训练环节。韩国科学技术院研究团队提

出的 AutoML-Agent[9] 则构建了一个智能体协作系统，该系统通过规划引擎和验证机制实现了从数据准备到模型部署的全流程自动化，将多个专业 AI 模块进行有机整合。这两项突破性技术分别从架构搜索效率提升和开发流程自动化两个关键维度入手，显著提高了 AI 模型的开发效率，为降低 AI 应用开发门槛作出了重要贡献。

然而，上述工作主要面向单一且通用的计算平台，并不适用于资源受限且平台异构的云边端协同场景。因此，本研究提出面向物理设备特性的个性化 AI 模型生成技术。如图 2 中模型构建部分所示，首先将物联网设备的 AI 计算性能提取为高维向量，从而充分表征计算能力，并基于物理世界知识和已有的 AI 模型库为用户需求构建模型草图；然后利用模型架构和梯度更新健壮性评估算法来评估模型推理能力；最终利用 LLMs 的分析和理解能力不断对草图模型修改与优化，提高模型推理精度和服务效率。

智能原生云边端协同计算中的自主计算演化更新

自主计算演化更新面临的挑战

在上述计算指令生成的基础上，为实现计算系统能力的持续演化，需要构建“任务执行—效果评估—能力优化”的闭环反馈演化机制，以实现真正意义上的动态感知、自主优化与持续成长。自主计算演化更新的目标是通过运行反馈持续优化计算策略，其中既包含计算任务自身的自主演进，也包含对计算生成与管理方式进行自主反馈与演进。例如，支持 ChatGPT、DeepSeek-R1 等表现优异的大模型的关键之一在于使用了基于人类反馈的强化学习（reinforcement learning from human feedback, RLHF）[10] 与基于 AI 反馈的强化学习（reinforcement learning from ai feedback, RLAIF）[11]。然而，信息物理系统面向软硬耦合、多设备协同的现实环境，面临更高复杂度与动态性。**由于信息物理系统的复杂性，通过直接部署计算任务获取反馈的方式开销大、不可扩展，如何面向信息物理系统实现自主的计算行为的评估与演进，是一个亟待解决的问题**。为此，首先要构建高保真、强泛化的智能原生仿真评估方法，用于模拟任务过程、生成行为反馈与语义评价；另一方面，系统还须具备从即时响应到持续学习的多尺度更新机制，将反馈转化为策略调整与能力成长，支撑智能系统的持续优化。

智能原生的物联网设备行为建模

传统的云边端协同应用在迭代升级过程中，普遍依赖真实设备和物理环境进行功能测试与性能验证。这种方式不仅流程繁琐、成本高昂，还存在反馈粒度粗、复现性差等问题，难以支持大规模、复杂场景下的快速演化更新需求。特别是在多设备、多协议、多策略协同的复杂应用中，传统测试评估机制已逐渐成为系统智能进化的“瓶颈”。

为破解上述难题，近年来学术界与工业界纷纷将仿真系统与智能语义评估机制引入物联网系统测试流程，探索低成本、高通用性的演化验证新路径。例如，TinySim[12] 系统可支持上万节点级别的通信与能耗仿真，并借助 Unity 引擎构建三维可视化环境，实现智能任务的全流程模拟验证；μEMU[13] 与 sEMU[14] 则通过符号执行与手册建模方式实现芯片仿真，实现物联网外设硬件的深入分析验证。在此基础上，为分析 LLMs 所生成的代码是否满足用户需求，通过基于 LLMs 的语义裁判机制实现类人化、多维度、可解释的语义评估与反馈，显著增强了评估的智能性和表现力[15−17]。例如，Prometheus[18] 和 CodeJudge[19] 通过自然语言解释、偏好排序和策略比较等方式，实现复杂交互任务中的主观意图理解与代码行为评判。

基于多尺度记忆的计算行为演进

在上述智能仿真与评估机制的支持下，可获得所生成的云边端应用代码在任务完成、策略优劣与行为合理性等多维度的反馈。真正实现智能演化的关键在于如何将这些反馈高效转化为系统能力的持续演进。因此，更新机制不仅要快速适应短期任务变化，还须具备长期知识积累与能力迁移能力，形成“即时响应—持续学习”的闭环调优体系。

在短期适配层面，系统可基于反馈快速刷新键值（key value, KV）对缓存、修改提示上下文等，实现毫秒级响应，提升对突发任务和环境扰动的处理能力[20]；在长期演化层面，系统则基于跨轮任务评估、行为追踪与

策略建模，触发如模型微调、低秩适配（low-rank adaptation, LoRA）增量学习、策略迁移等深度更新。借助历史任务数据、失败示例与环境上下文，系统逐步积累可泛化的“能力模型”，实现从“即时反馈调整”向“长期能力成长”的进步。

尽管双尺度策略具备良好的适应性与扩展性，在实际部署中仍面临挑战：短期适配可能导致系统过度依赖即时响应、抑制长期能力积累；频繁更新带来算力和通信负担，尤其在边缘环境中影响性能；长期演化则易陷入局部最优，且缺乏透明性与可解释性。为此，系统可引入任务感知的权重调控机制，动态平衡短期与长期更新频率[21]，并结合进化算法与因果追踪提升更新路径的稳定性与可解释性[22]，从而构建既具有响应速度又能持续演进的智能能力体系。

智能原生云边端协同计算中的自主计算资源调度与部署

自主计算资源调度与部署面临的挑战

自主计算资源调度与部署指的是，面向生成应用指令及云边端分布式系统，基于人工智能技术实现资源自主调度和部署。在计算系统的发展历程中，资源调度与部署始终是核心问题。传统计算系统的资源调度主要依赖操作系统完成，操作系统又依赖调度器、内存管理与输入/输出（in/out, I/O）子系统维持资源供需平衡，其背后是基于数学建模或人工规则设定的静态策略。然而随着操作系统的多年发展，其复杂程度与海量可调参数远远超出了可数学建模的量级，例如 Linux 仅内核就有 42 万行代码，加之系统中存在大量的启发式算法，导致系统整体性能难以建模与优化。随着系统规模与复杂度的持续提升，特别是云–边–端协同带来的异构硬件与动态负载环境，这套机制的局限性日益凸显：其内部庞大的参数空间与策略组合在全局范围内表现出不可预测性，且缺乏根本的自适应能力。借助人工智能模型理论上能够拟合任意高维函数的特性，将计算任务对资源的使用与硬件对资源的供给模式通过人工智能模型表征出来，进而实现资源的高效、可优化管理。**然而，面对云边端复杂协同应用与异构分布式设备，如何实现资源使用与供给模式的高效表征，并在此基础上实现联合优化也是一个亟待解决的问题。**

尽管学术界和工业界探索了诸如多内核架构 Barrelfish[1]、统一资源抽象[23] 等分布式方案来扩展跨节点调度能力，但这些改进本质上仍依赖于预设的启发式策略，未能从根本上解决动态环境下的自治优化问题。

自主计算资源调度与部署解决思路

为突破这一瓶颈，近期研究开始尝试利用 LLMs 的泛化能力与 AI 模型对高维问题的刻画能力来增强系统调度。例如，LSFS[24] 将 LLM 融入文件系统接口，通过语义索引与自然语言交互改进文件管理。NetLLM[25] 则借助 AI 模型逼近高维非线性函数的能力，以数据驱动方式刻画网络资源的复杂调度关系。然而，这类方法往往将任务与设备视为黑盒，忽略了其内在特征，从而限制了模型的可解释性与迁移性。究其根本，上述研究仅是在现有操作系统的抽象边界之上进行增量式的智能增强，并未触及资源管理内核的核心架构，未能实现跨资源、跨节点的全局协同与自主管理，其优化效果是局部且有限的。

智能原生计算提出了一种新的视角，即将资源管理抽象为供给侧与使用侧的双向表征与建模：计算供给侧对应计算、存储和网络等资源的性能表征，计算使用侧则对应应用和任务的动态需求模式。对供给侧而言，传统的基准测试和静态配置方式已难以捕捉硬件在实际运行环境下的动态表现，而基于 AI 的供给建模能够结合实时监测数据，生成更为细致的资源画像。对使用侧而言，过去以“CPU 密集型”或“I/O 密集型”为代表的粗粒度分类方式无法适应新型应用的复杂性，而机器学习模型能够对任务在不同阶段的资源消耗规律进行更精准的预测。与传统方式不同，这一思路旨在构建一个由 AI 模块驱动的分布式智能原生操作系统，使自主的资源管理与调度成为系统的内生能力。

总结与展望

本研究系统阐述了面向云边端协同的智能原生计

[1] BarrelfishOS. https://github.com/BarrelfishOS/barrelfish, 2025−11−20

算框架，旨在通过人工智能技术实现“自主指令生成—自主调度部署—自主演化更新”的全流程自动化，以降低传统计算范式对人工经验的依赖。在分析其愿景与层次架构基础上，重点指出了三大关键挑战：软硬耦合下的可控指令生成、低开销自主演化更新机制及异构资源联合优化，为构建自适应、高性能的云边端分布式智能计算系统提供了研究方向和理论参考。现有 LLMs 能支撑智能原生计算各步骤以构建原型，其具备自然语言理解、工具调用与任务拆解能力，可实现需求解析、算力调度、模型优化等核心流程的自主运行，能满足系统从交互到演化的基础功能验证需求。但替代现有体系仍有差距，现有 LLMs 在软硬协同的精准控制、异构算力的高效适配及动态场景的低耗响应上存在不足，且幻觉问题与上下文局限难以应对生产级的可靠性要求，未来须使 LLMs 在技术能力上持续突破。

展望未来，智能原生计算仍存在广阔的研究空间与发展潜力。首先，如何让智能体进一步自主设计和生成模型架构，而不是根据已有配置去构建和调整模型架构，仍要深入研究。在演化更新方面，目前的仿真与反馈还停留在表层的设备整体行为方面，如何进一步在系统层，甚至硬件层实现细粒度的自主仿真，仍待进一步解决。在资源管理与调度方面，完全基于人工智能的操作系统仍处于愿景阶段，其中的决策效率问题与一致性问题都亟待解决。未来，人工智能算力的跨越式发展，模型训练时与测试时的规模化法则（training-time and test-time scaling law）进一步生效，将会推动人工智能在计算方面的能力跃迁，走向真正的智能原生计算。 ■

李博睿

CCF 专业会员。东南大学助理教授、至善青年学者。主要研究方向为智能物联网与边缘计算系统、设备无感知计算。libr@seu.edu.cn

王　帅

CCF 高级会员、普适计算专委会和网络与数据通信专委会执行委员，《计算》编委。东南大学青年首席教授。主要研究方向为智能原生物联网、大数据分析。shuaiwang@seu.edu.cn

王维龙

CCF 学生会员。东南大学博士研究生。主要研究方向为设备无感知计算、计算指令生成。wang_wl@seu.edu.cn

刘佳伟

东南大学博士研究生。主要研究方向为智能系统演化更新。liujoy@seu.edu.cn

刘天恩

东南大学博士研究生。主要研究方向为设备无感知计算、资源调度。toliutianen@seu.edu.cn

参考文献

[1] 王帅, 李博睿, 王维龙, 等. 设备无感知边缘模型协同推理[J]. 中国计算机学会通讯, 2025, 21(3): 28−34.

[2] Yupeng Chang, Xu Wang, Jindong Wang, et al. A Survey on Evaluation of Large Language Models[J]. *ACM Transactions on Intelligent Systems and Technology*, 2024, 15(3): 1−45.

[3] Xupeng Miao, Gabriele Oliaro, Zhihao Zhang, et al. Towards Efficient Generative Large Language Model Serving: A Survey from Algorithms to Systems[J]. *ACM Computing Surveys*, 2025, 58(1): 1−37.

[4] Mert Yuksekgonul, Federico Bianchi, Joseph Boen, et al. Optimizing Generative AI by Backpropagating Language Model Feedback[J]. *Nature*, 2025, 639(8055): 609−616.

[5] Yi Gao, Kaijie Xiao, Fu Li, et al. ChatIoT: Zero-Code Generation of Trigger-action Based IoT Programs[J]. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 2024, 8(3): 1−29.

[6] Xin He, Kaiyong Zhao, Xiaowen Chu. AutoML: A Survey of the State-of-the-Art[J]. *Knowledge-Based Systems*, 2021, 212: 106622.

[7] Pengzhen Ren, Yun Xiao, Xiaojun Chang, et al. A Comprehensive Survey of Neural Architecture Search: Challenges and Solutions[J]. *ACM Computing Surveys (CSUR)*, 2021, 54(4): 1−34.

[8] Ming Lin, Pichao Wang, Zhenhong Sun, et al. Zen-NAS: A Zero-Shot NAS for High-Performance Image Recognition[C]//*Proceedings of the IEEE/CVF International Conference on Computer Vision*. Red Hook: Curran

Associates Inc. , 2021: 337−346
[9] Trirat, Patara, Wonyong Jeong, Sung Ju Hwang. AutoML-Agent: A Multi-Agent LLM Framework for Full-Pipeline AutoML[EB/OL]. (2024−10−03)[2025−11−20]. https://doi.org/10.48550/arXiv.2410.02958.
[10] Shreyas Chaudhari, Pranjal Aggarwal, Vishvak Murahari, et al. RLHF Deciphered: A Critical Analysis of Reinforcement Learning from Human Feedback for LLMs[J]. *ACM Computing Surveys*, 2025, 58(2): 1−37.
[11] Harrison Lee, Samrat Phatale, Hassan Mansoor, et al. RLAIF vs. RLHF: Scaling Reinforcement Learning from Human Feedback with AI Feedback[EB/OL]. (2023−09−01)[2025−11−20]. https://arxiv.org/abs/2309.00267.
[12] Gonglong Chen, Wei Dong, Fujian Qiu, et al. Scalable and Interactive Simulation for IoT Applications with TinySim[J]. *IEEE Internet of Things Journal*, 2023, 10(23): 20984−20999.
[13] Wei Zhou, Le Guan, Peng Liu, et al. Automatic Firmware Emulation through Invalidity-Guided Knowledge Inference[C]//*30th USENIX Security Symposium*. Berkeley: USENIX Association, 2021: 2007−2024.
[14] Wei Zhou, Lan Zhang, Le Guan, et al. What Your Firmware Tells You Is Not How You Should Emulate It: A Specification-Guided Approach for Firmware Emulation[C]//*Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*. New York: ACM, 2022: 3269−3283.
[15] Jiawei Gu, Xuhui Jiang, Zhichao Shi, et al. A Survey on LLM-as-a-Judge[EB/OL]. (2024−11−23)[2025−11−20]. https://doi.org/10.48550/arXiv.2411.15594.
[16] Mingchen Zhuge, Changsheng Zhao, Dylan R. Ashley, et al. Agent-as-a-Judge: Evaluating Agents with Agents[J]. *Proceedings of the 42nd International Conference on Machine Learning*, 2025, 267: 80569−80611.
[17] Jiaju Chen, Yuxuan Lu, Xiaojie Wang, et al. Multi-Agent-as-Judge: Aligning LLM-Agent-Based Automated Evaluation with Multi-Dimensional Human Evaluation [EB/OL]. (2025−07−28)[2025−11−20]. https://doi.org/10.48550/arXiv.2507.21028.
[18] Seungone Kim, Jamin Shin, Yejin Cho, et al. Prometheus: Inducing Fine-grained Evaluation Capability in Language Models[C]//The Twelfth International Conference on Learning Representations. Vienna: ICLR, 2024: 1−36.
[19] Weixi Tong, Tianyi Zhang. CodeJudge: Evaluating Code Generation with Large Language Models[C]//*Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*. Stroudsburg: ACL, 2024: 20032−20051.
[20] Dongfang Li, Zetian Sun, Xinshuo Hu, et al. CMT: A Memory Compression Method for Continual Knowledge Learning of Large Language Models[J]. *Proceedings of the AAAI Conference on Artificial Intelligence.*, 2025, 39(23): 24413−24421.
[21] Qizheng Zhang, Ali Imran, Enkeleda Bardhi, et al. Caravan: Practical Online Learning of In-Network ML Models with Labeling Agents[C]//*Proceedings of the 3rd Workshop on Practical Adoption Challenges of ML for Systems*. New York: ACM, 2024: 17−20.
[22] Anpeng Wu, Kun Kuang, Minqin Zhu, et al. Causality for Large Language Models[EB/OL]. (2024−10−20)[2025−11−20]. https://doi.org/10.48550/arXiv.2410.15319.
[23] Philipp Moritz, Robert Nishihara, Stephanie Wang, et al. Ray: A Distributed Framework for Emerging AI Applications[C]//*13th USENIX Symposium on Operating Systems Design and Implementation*. Berkeley: USENIX Association, 2018: 561−577.
[24] Zeru Shi, Kai Mei, Mingyu Jin, et al. From Commands to Prompts: LLM-based Semantic File System for AIOS[C]//The Thirteenth International Conference on Learning Representations. Singapore, ICLR, 2025: 1−24.
[25] Duo Wu, Xianda Wang, Yaqi Qiao, et al. NetLLM: Adapting Large Language Models for Networking[C]//*Proceedings of the ACM SIGCOMM 2024 Conference*. New York: ACM, 2024: 661−678.

AI-Native Computing for Cloud-Edge-Device Distributed Collaboration

Borui Li, Shuai Wang, Weilong Wang, Jiawei Liu, Tianen Liu
Southeast University

Abstract: With the increasing complexity and intelligence of cyber-physical systems (CPS), traditional centralized computing architectures struggle to meet the demands for low latency and high computational power, making cloud-edge collaboration a promising alternative. However, the current "programming-deployment-debugging" paradigm relies heavily on manual expertise and cannot adapt to dynamic environments. This article proposes an "AI-native cloud-edge-device collaborative computing" framework, which leverages artificial intelligence (e.g., large language models) to achieve autonomous instruction generation, autonomous scheduling and deployment, and autonomous computation evolution, reducing human intervention and enhancing system autonomy. The article focuses on the three challenges of AI-native computing, i.e., controllable instruction generation under software-hardware constraints, scalable computing behavior evolution, and the modeling of complicated heterogeneous resources. This article also surveyed the existing related works and outlines future research directions on AI-native computing.

Keywords: cyber-physical systems; cloud-edge-device collaboration; instruction generation; resource management; self-evolution

摘　要: 随着信息物理系统（cyber-physical systems, CPS）复杂性和智能化需求的提升，传统集中式计算架构难以满足实时与算力需求，云边协同分布式架构逐渐成为主流。然而，现有"编程—部署—调试"范式依赖人工经验，无法适应动态复杂环境。本研究

提出“智能原生云边端协同计算”框架，通过人工智能（如大语言模型）实现自主计算指令生成、自主调度部署、自主演化更新，以降低人为干预，提升系统自治能力；深入分析并探讨了上述云边端协同智能原生计算的三大挑战：软硬耦合下的指令生成、高可扩展的计算行为演化、复杂异构系统资源的性能表征，整理了现有工作，以期为未来研究提供思路。

关键词：信息物理系统；云边端协同；指令生成；资源管理；自主演进

中图分类号：TP39

中文引用格式：李博睿, 王帅, 王维龙, 等. 面向云边端分布式协同的智能原生计算 [J]. 计算, 2025, 1(8): 38−45.

英文引用格式：Borui Li, Shuai Wang, Weilong Wang, et al. AI-Native Computing for Cloud-Edge-Device Distributed Collaboration[J]. *Computing Magazine of the CCF*, 2025, 1(8): 38−45.

CCF 秀湖会议学术委员会 2025 年第二次工作会议召开

2025 年 10 月 25 日，CCF 秀湖会议学术委员会（AC）2025 年第二次工作会议在哈尔滨召开。本次会议由 CCF 会士、CCF 监事长、CCF 秀湖会议 AC 主席、北京大学教授**金芝**主持，围绕秀湖会议 2025 年上半年论坛情况总结、未来选题规划、组织形式优化等核心议题展开深度研讨，为秀湖会议下一阶段的高质量发展明确路径、凝聚共识。

会议期间，金芝明确了本次会议对厘清秀湖会议发展方向、提升学术影响力的关键意义，为整场会议奠定务实高效的研讨氛围。秀湖会议 AC 学术秘书、北京航空航天大学教授**汪淼**汇报了 2025 年上半年会议整体组织情况、学术成果及参会反馈。全体 AC 委员围绕会议组织形式、成果转化、国际影响力提升等展开热烈讨论，提出加强国际交流、增设并行论坛深化议题讨论、提升青年学者参与度等多项建设性建议。

秀湖会议是 CCF 打造的小型精品国际学术讨论会品牌，借鉴德国达堡研讨会（Dagstuhl Seminars）、日本湘南会议模式，旨在深入探讨计算机相关领域的科学、技术、应用、教育和产业等问题，为未来计算技术的发展和应用提供新思路和新建议。每期研讨会均针对某一个具体的前沿问题讨论交流，仅限发起人邀请的一线专家参与，不对外开放，会期 3 天以上，要求参会者全程参会，不能中途离会，以引导科学家、企业技术专家及教育专家在浮躁的社会中沉下心来钻研学术。

秀湖会议面向全界学者开放申请。CCF 鼓励有新想法的业界同仁聚集一处，深入讨论前沿和跨领域问题的新观点，打造一个计算机领域的高端交流平台。有意申办秀湖会议的专家可访问秀湖会议网站 https://bls.ccf.org.cn 了解申办方式。联系：bls@ccf.org.cn。