大规模智算互联网络仿真技术和平台

Large-Scale Intelligent Computing Interconnection Network Simulation Technology and Platform

熊先奎/Xiong Xiankui，童贤慧/Tong Xianhui，
王勇/Wang Yong，张启明/Zhang Qiming
（中兴通讯股份有限公司，中国 深圳 518057）
(ZTE Corporation, Shenzhen 518057, China)

DOI：10.12142/ZTETJ.202603011
网络出版地址：https://link.cnki.net/urlid/34.1228.TN.20260626.0957.004
网络出版日期：2026-06-26
收稿日期：2026-04-25

摘要：网络仿真作为高效的评估预测手段，已成为智算基础设施发展不可或缺的工具。总结了大规模智算场景下网络仿真的三大核心挑战：规模与复杂性、效率与可扩展性、精度与真实性，并系统剖析层次化建模、混合仿真、并行仿真、硬件加速、真实数据驱动等仿真关键技术。中兴通讯探索仿真新架构，设计实现面向数据流特征驱动的网络仿真平台，并在工程实践中验证了其高效性和可扩展性。

关键词：网络仿真；仿真建模；仿真加速；仿真校准

Abstract: As an efficient evaluation and prediction method, network simulation has become an indispensable tool for the development of intelligent computing infrastructure. This paper summarizes three core challenges of network simulation in large-scale intelligent computing scenarios: scale and complexity, efficiency and scalability, accuracy and fidelity, and systematically analyzes key technologies, including hierarchical modeling, hybrid simulation, parallel simulation, hardware acceleration, and real-data-driven simulation. ZTE Corporation explores a new simulation architecture, designs and implements a network simulation platform driven by data flow characteristics, and verifies its efficiency and scalability in engineering practice.

Keywords: network simulation; simulation modeling; simulation acceleration; simulation calibration

引用格式：熊先奎，童贤慧，王勇，等．大规模智算互联网络仿真技术和平台 [J]. 中兴通讯技术，2026，32(3)：60-68. DOI：10.12142/ZTETJ.202603011

Citation: Xiong X K, Tong X H, Wang Y, et al. Large-scale intelligent computing interconnection network simulation technology and platform [J]. ZTE technology journal, 2026, 32(3): 60-68. DOI: 10.12142/ZTETJ.202603011

现代大语言模型（LLM）在Scaling Laws的驱动下持续突破规模边界。智算互联网络作为连接海量计算资源和存储资源的核心基础设施，呈现出新的特征：

1）高带宽与低延迟需求：分布式训练要求网络具备太比特每秒级聚合带宽和亚微秒级延迟能力，推动光互联技术实现突破[1]。

2）超大规模与高扩展性：算力集群中数十万节点通过多层网络拓扑结构实现互联，驱动规模组网架构技术创新[2]。

3）非对称突发流量：All-Reduce、All-to-All等集合通信操作产生独特的流量模式，推动传统流量调度与拥塞控制机制革新[3]。

4）异构资源协同耦合：中央处理器（CPU）、图形处理器（GPU）、神经网络处理器（NPU）等异构计算单元与高速网络、存储设备深度协同，促进统一资源管理、跨域调度优化等技术应用[4]。

面对如此复杂系统特性，物理部署实测的方式难以覆盖全场景、高成本、长周期的验证需求。网络仿真通过构建数字化模型复现网络行为、评估网络性能，凭借低成本、高灵活性优势，已成为支撑智算互联网络设计验证的关键手段。

1 网络仿真发展现状

网络仿真技术历经两个阶段发展，从通用网络场景逐步演进至智算网络场景。

1.1 通用网络仿真

通用网络仿真聚焦传统计算机网络、通信网络的性能模拟与分析，广泛应用于网络协议研发、架构设计与优化等场景。该类仿真主要包括开源与商业两条技术路线：

1）开源路线：以NS-3[5]为代表，具备模块化架构与高度可扩展性，但在大规模场景下仿真效率明显不足。Unison[6]、DONS[7]等项目通过并行化仿真、事件调度优化等技术提升效率。

2）商业路线：以OPNET[8]为典型代表，具备图形化交互界面、高精度模型库以及多维度性能分析能力，但其商业化属性带来使用成本高、难以快速适配特殊场景等问题。

随着网络规模扩大与复杂度提升，通用网络仿真逐渐凸显出效率低、精度不足、兼容性差等突出问题。

1.2 智算网络仿真

智算网络仿真是面向大模型训练、推理等智算场景的定制化技术，其核心目标是精准模拟算力节点与网络的协同特性、工作负载的传输规律以及异构互联架构的性能表现。目前智算网络仿真已广泛采用“以任务为中心”的新范式，其发展经历两个阶段：

1）基于通用工具扩展阶段，粗粒度集成智算模块，快速构建仿真能力

例如，ASTRA-sim[9]早期1.0版本仅支持有限集合通信，忽略了计算通信等耦合效应，精度不足；SimGrid[10]仿真引擎早期在通用仿真框架中集成了加速器网络的通信模型。

2）专用定制化框架阶段，深度适配智算场景的算网协同需求

此类框架普遍采用计算、存储、网络协同建模。例如，ASTRA-sim在2.0版本中支持层级化网络抽象和计算调度联合仿真；SimAI[11]面向大模型训练场景，集成了训练框架、计算、集合通信库和网络后端；Vidur[12]则针对推理场景，构建动态批处理仿真框架；ATLAHS[13]以应用为中心，采用通用组运算汇编语言（GOAL）格式表达AI、高性能计算（HPC）、存储三类工作负载。

目前，中国头部企业已纷纷开展定制化仿真技术研发并落地应用。例如：阿里云HPN[3]双平面两层架构设计，借助仿真平台建模训练任务，快速迭代验证通信调度优化机制，最终达成系统利用率与可靠性的双重提升；华为CloudMatrix384[14]超节点架构，针对Pangu Ultra MoE模型的昇腾亲和特性构建仿真模型，通过评估不同策略的性能表现，确定长序列场景的最优部署方案。

2 网络仿真面临的主要挑战

智算互联网络仿真面临一大核心难题：规模扩展、执行效率、高准确度三者难以兼顾。

2.1 规模与复杂性挑战

规模扩展导致复杂度呈指数级增长，传统建模表达能力已不再适配复杂场景。

1）互联拓扑复杂。万卡级集群的多层级拓扑结构难以精确建模。SlipStream[15]相关研究证实，节点间通信关系动态变化进一步加剧了建模难度。同时SplitSim[16]实验表明，传统的集中建模方式引发状态空间爆炸，显著提升了计算复杂度。

2）异构性与动态性突出。网络设备、链路带宽、流量模式的异构性，叠加网络拥塞、故障、负载均衡等动态行为，显著增加了仿真建模的适配难度[8]。

3）多维度资源耦合。智算任务中计算、存储、网络资源间存在复杂的依赖和竞争关系。每种硬件和网络技术都有其独特的性能参数和行为模式，单一模型缺乏对耦合效应的精准模拟，更难以支撑系统级优化[17]。

2.2 效率与可扩展性挑战

仿真效率与规模扩展的非线性矛盾突出，传统串行和弱并行架构已接近算力天花板。主要表现在以下3个方面：

1）资源消耗高。SplitSim[16]实验表明，10万节点的网络仅拓扑建模就需占用数十GB内存，数据包模拟计算量更是达到千亿次级别。单轮仿真占用CPU资源超过100核/h，远超单节点的承载能力。

2）仿真时间长。网络仿真的时间复杂度与节点数量的平方成正比。在Multiverse[18]测试中，模拟一次千亿参数大模型的训练过程，仅参数同步就需要处理数百万次传输事件，整体仿真耗时可达数周。

3）并行仿真瓶颈。网络事件之间存在复杂的依赖关系，事件同步、状态一致性维护等开销限制了并行效率提升，超大规模场景下更难以实现线性加速[19]。并行离散事件仿真（PDES）在高动态场景下事件回滚率超50%，反而降低有效吞吐量[6]。

2.3 精度与真实性挑战

软件层不确定性与硬件特性细节缺失，共同导致仿真结果与实际数据偏差显著。

1）软件栈影响难以量化。操作系统、设备驱动、通信库等软件层对网络性能的实际开销和影响，在仿真中难以精

准复现[20]。

2）工作负载建模不足。现有静态模型难以刻画多任务并发时计算和通信耦合引发的资源竞争与调度交互行为。实验表明[21]，在LLM训练仿真中，如果忽略通信触发的时序依赖关系，仿真误差将超过30%。

3）动态网络行为缺失。仿真模型无法复现真实网络的瞬态响应和局部性能波动；同时基于实时流量的路由调整、基于算力节点负载感知调度等动态负载均衡自适应策略未得到精准建模，导致仿真无法反映实际环境的自适应优化能力[9]。

4）硬件特性建模简化。现有的模型大多忽略芯片缓存、网卡队列管理、调度算法等底层硬件交互细节。Net-TLMSim[22]实验表明，网卡队列调度策略会显著影响流量传输的公平性与传输延迟，简化建模会导致10%～40%的性能预测误差。

3 网络仿真关键技术研究

针对前文所述挑战，学术界与产业界开展了一系列系统和硬件的改进优化工作。

3.1 规模与复杂性优化

3.1.1 分层/模块化拓扑建模

建模过程中将复杂的网络按逻辑层级或地域进行拆分：

1）层次化建模，按逻辑关系等划分层次。例如，ASTRA-sim对核心交换层、算力节点接入层进行建模，通过层级间的接口适配构建大规模拓扑。

2）模块化建模，按地域等划分多模块。例如，数据中心内采用细粒度建模，精准刻画拓扑端口带宽等细节；跨数据中心采用中粒度建模，重点关注骨干网络的路由特性与传输延迟；广域算力协同采用粗粒度建模，简化地理距离带来的延迟与带宽衰减[2]。

3.1.2 多粒度混合仿真

多粒度混合仿真通过设置差异化建模精度，兼顾精度与开销，在仿真规模与运行效率之间取得平衡。该仿真的关键在于不同粒度模型之间的协同机制。通过流量映射、状态同步等方式确保模型间的平滑切换。例如，NetTLM-Sim[22]多粒度混合策略会固定划分建模区域：对加速器网络的核心互联部分采用精细化硬件建模，对边缘网络采用抽象流量建模，通过流量状态同步实现网络高效仿真；Si-mAI[11]可动态调整建模粒度，在仿真过程中根据流量类型自动切换细粒度和粗粒度模型，既保证热点区域的仿真精度，又降低整体开销。

3.1.3 统一资源抽象

统一资源抽象技术对计算、通信、存储等多维度资源进行一体化整合。该技术的关键在于构建统一资源抽象与元数据模型。经典的单步任务执行延时量化模型[23]能够提取各类资源的核心属性，并将不同资源映射为标准化的抽象实体，具体如公式（1）所示：

$$T_{\text{step}} = \max(T_{\text{comp}}, T_{\text{mem}}, T_{\text{comm}}) + T_{\text{unhidden}} + T_{\text{sync}} \quad (1),$$

其中，T_{step}为执行总延时；计算时间$T_{\text{comp}} = \frac{F}{P \times \pi}$，$F$为单卡浮点运算量，$P$为单卡峰值算力，$\pi$为实际利用率；访存时间$T_{\text{mem}} = \frac{B}{\beta_{\text{mem}}}$，$B$为访存数据量，$\beta_{\text{mem}}$为显存带宽；通信时间$T_{\text{comm}} = (\frac{M}{\beta_{\text{net}}} + \alpha)N$，$M$为单次通信数据量，$\beta_{\text{net}}$为网络带宽，$\alpha$为网络延迟，$N$为通信次数；$T_{\text{unhidden}}$为无法重叠的时间；$T_{\text{sync}}$为系统同步开销。

该公式依赖静态参数构建，存在动态适应性不强、资源耦合刻画不足等问题。为突破静态模型局限，Meta的Arcadia[24]将不同维度的参数作为相互影响的系统组件来描述，通过协调器来同步和模拟各种组件的行为，实现状态实时同步与动态参数修正。

3.2 效率与可扩展性优化

3.2.1 基于采样的加速仿真

通过部分样本仿真结果推测整体网络性能，在保证精度损失可控的前提下，可大幅提升仿真效率。

采样策略是有效性的关键保证。合理设计采样策略（如分层采样、逐流采样、重点采样），对智算网络的流量、拓扑、任务等关键要素进行采样，确保采样数据能准确反映整体特征。Network Sampling[25]实验证明，对智算网络中的热点链路和高并发流量进行优先采样，忽略低流量的网络区域，在采样率仅为30%的情况下，可使仿真精度保持在90%以上。

采样策略的选择取决于网络类型和考量指标。目前还没有一种普遍最优的采样方法，因此，探索特定指标和应用驱动的采样策略是重要研究方向之一[25]。

3.2.2 高效事件调度与管理

针对智算网络仿真的事件密集型特征，可采用优先级调度、批量处理机制提升执行效率；同时通过合并冗余、筛选

过滤减少事件处理开销。

1）优先级调度。Unison[6]设计了分层事件调度器，将网络事件、计算事件、存储事件分别分配到独立的线程池处理，通过优先级队列确保关键事件及时响应。DONS[7]引入依赖感知机制，通过分析通信计算图来动态调整优先级。

2）事件合并与过滤。例如SplitSim[16]采用混合保真度仿真方案，将多个同类事件合并为一个批量事件，减少事件处理次数。粗粒度仿真模式也会过滤部分事件，从而缩短仿真时间。

此外，DeepQueueNet[26]提出预调度技术，可提前预测关键事件的发生时间与顺序，优化事件处理流程，进一步降低仿真时间开销。

3.2.3 并行与分布式仿真

将大规模仿真任务拆分至多个计算节点并行执行，可突破单节点的计算资源与内存限制。

该方案的核心在于合理的任务划分策略与高效的同步机制。

1）任务划分。划分过程需保证各节点负载均衡，减少跨节点通信。例如，DONS[7]将智算网络按照链路连接关系划分为多个独立的仿真区间，支持自动并行能力；SplitSim[16]通过编排机制将任务分解为多个并行逻辑单元，采用动态负载均衡策略。

2）同步机制。该机制主要包括混合同步策略、因果依赖调度等技术。例如，NSX[27]采用混合同步策略，仿真域内部采用乐观同步，域间则采用保守同步。此外，也有研究通过自动识别“低风险异步区”放宽同步约束[28]。

3.2.4 硬件加速仿真

硬件加速仿真技术利用专用硬件的并行计算能力加速计算密集型任务，主要包括两个部分：

1）硬件加速架构设计。“软件控制+硬件计算”的混合架构是普遍采用的架构方式。软件负责仿真任务调度、拓扑管理、动态场景控制等逻辑密集型操作，硬件则通过加速器实现数据包转发、延迟计算、拥塞控制等计算密集型任务的并行处理[22]。

2）硬件加速模块实现。例如Miniature[29]在现场可编程门阵列（FPGA）上对队列、内部状态和协议栈进行建模。实验表明，在仿真65 536个节点的智算集群时，相比基于软件的网络仿真，效率提升4 332倍。Multiverse[18]在GPU上通过CUDA编程实现仿真任务的大规模并行计算，多实验并行仿真的效率提升40倍。

3.3 精度与真实性优化

3.3.1 真实数据驱动仿真

真实数据驱动仿真首先采集真实智算网络中的运行数据，并利用数据挖掘技术提取关键特征。

该技术的核心流程包括3个部分：

1）数据采集。通过网络探针、设备日志、任务监控系统等手段，获取智算网络的流量特征、拓扑变化、设备性能等真实数据[26]；

2）数据预处理。通过数据清洗、异常值剔除、特征提取、归一化等操作，将原始数据转化为符合仿真模型输入要求的结构化数据[30]；

3）模型校准。利用预处理后的数据对仿真模型的关键参数进行校准，提升模型对真实场景的拟合能力[4-8]。例如，SimAI[11]通过采集数千次LLM训练的真实通信数据，将仿真精度提升至95%以上。

3.3.2 计算/通信协同建模

该技术融合计算任务特性与网络传输特性，使工作负载的任务执行顺序、调度策略选择能够真实反映算网耦合过程。

该技术的核心是要建立计算与通信的动态关联机制，具体包括两个部分：

1）时序关联模型。将通信事件与计算任务的关键节点（如层计算完成时刻、批量处理结束时刻）绑定，确保通信触发点与真实场景中一致[31]。

2）反馈调节机制。将网络传输延迟、带宽占用等结果实时反馈至计算任务模型，调整计算任务的执行进度。例如，Arcadia[24]将大模型训练中的层计算与参数同步进行动态绑定，模拟计算与通信的重叠与阻塞关系，计算过程中调整等待通信完成的空闲时间，使得仿真真实度提升25%。

3.3.3 动态场景精确建模

引入实时状态反馈、参数自适应调节等方法，有助于解决动态网络行为建模缺失问题。该技术主要分为两类实现方式：

1）控制行为建模。精准刻画拥塞窗口调整、信用值分配、流量控制帧交互等动态过程，能够模拟网络过载时的延迟增加、数据包丢失等真实现象。例如，Unison[6]对以太网远程直接内存访问（RoCE）优先级流控（PFC）帧在多级拓扑中的传播延迟与链路信用耗尽触发条件进行建模，复现了“反压级联”引发的非拥塞丢包机制。

2）异常行为建模。一方面，构建随机突发流量模型，并依托事件驱动机制触发流量突发。Vidur[12]借助该方式成

功模拟了LLM推理过程中的流量突发场景，仿真结果准确性可达90%。另一方面，引入故障注入机制[32]，支持节点故障、链路中断等多种故障类型的模拟仿真。

3.3.4 精细化硬件特性建模

通过解析关键硬件的底层工作机制和性能特征，构建贴合真实硬件行为的硬件仿真模型，精准还原硬件级别的性能差异和行为细节。

1）网络设备建模。该模块涵盖队列调度策略、缓冲分区机制以及多端口并发处理能力[26]。例如，ASTRA-sim[9]实测了不同交换机的队列调度延迟，建立非线性延迟模型，精准模拟不同负载场景下的传输延迟差异。

2）计算设备通信接口建模。该模块包括其计算能力、内存带宽、数据访问延迟，以及与网卡的直接数据交互特性，可量化硬件卸载对CPU开销、传输效率的影响[16-17]。例如，NetTLMSim[22]完整还原RDMA网卡卸载机制（如零拷贝、远程内存访问流程、缓存同步策略），应用于Transformer框架设计时，延迟预测误差降低52.8%。

3）存储设备建模。该模块覆盖传统存储与智算专用存储（如内存池化）。重点区分各类存储在输入/输出（I/O）延迟、带宽特性、随机/顺序访问性能上的差异，构建分布式存储系统的元数据管理延迟模型，精准反映存储访问对网络传输的影响[31]。例如，Arcadia[24]通过实测不同存储设备I/O性能，建立了存储访问模式与网络有效带宽的关联模型，能够准确模拟存储瓶颈导致的网络传输效率下降问题。

4 中兴通讯智算互联网络仿真的探索与实践

4.1 面向数据设计的高性能网络仿真平台

中兴通讯联合清华大学提出了一种数据流特征驱动（DOD）的仿真架构，可针对性解决万卡集群网络仿真的效率与可扩展性问题。

如图1所示，通过重构LLM训练任务中的计算图与通信依赖关系、优化时间调度逻辑，团队开发了首款面向GPU加速的专用仿真器。实测数据表明，相较于主流仿真工具ASTRA-sim，该方案的网络仿真速率提升20倍以上。

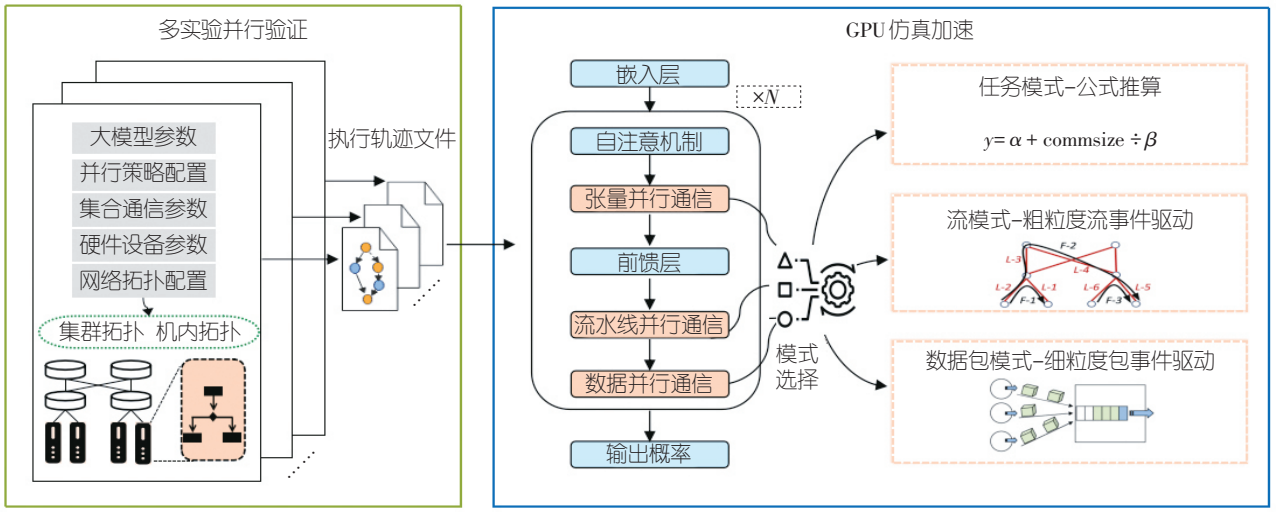
4.2 算网存联合仿真平台

中兴通讯智算前瞻团队正推进计算、网络、存储联合仿真平台建设。该平台整合专用加速器、多协议网络设备、高带宽内存等异构资源，构建标准化交互模型。核心目标是支撑三大关键应用场景：算法架构适配性评估、资源需求敏感性分析、集群扩展边界预测。

为应对建模高精度挑战，本平台采用如图2所示的架构，采集真实业务数据开展全流程模型校准。基于样本数据集，可进一步挖掘仿真数据与真实运行数据的关联规律，构建智能预测模型，实现对网络拥塞、存储访问异常、算力资源瓶颈等潜在故障的在线预警能力。

4.3 新型互联协议验证与训练部署寻优

自研仿真平台可支撑新型互联协议的快速开发与验证。



CPU：中央处理器　GPU：图形处理器

图1　高性能智算网络仿真平台

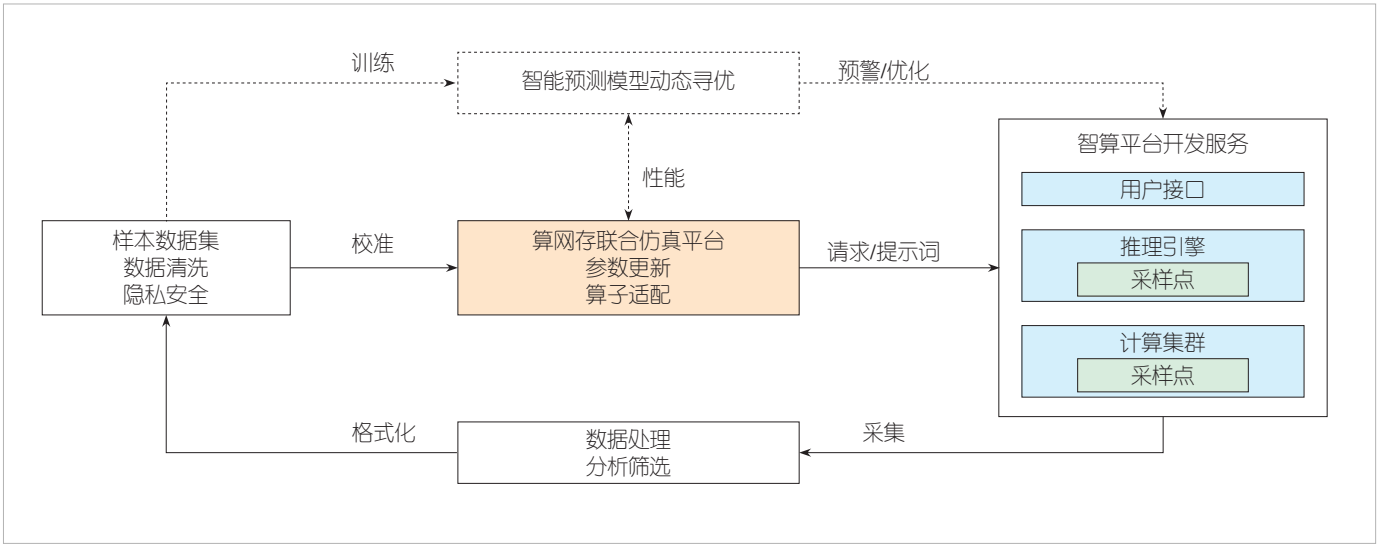


图2　大语言模型推理服务仿真校准流程

由图3灌包测试结果可知，自研逐流拥塞控制机制具备更快的拥塞响应速度、更优的理论速率逼近能力，同时收敛过程中震荡幅度显著低于传统数据中心量化拥塞通知（DCQCN）方案。

此外，该仿真平台还可支撑分布式训练部署方案寻优。如表1所示，借助仿真平台测试不同资源配置与网络拓扑，可量化评估不同并行策略对系统性能的影响，在实际部署前筛选出最优方案。未来该仿真平台将进一步拓展应用边界，覆盖更多异构分布式训练、推理平台，实现更多场景的性能量化预测与设计空间寻优。

5 趋势与展望

未来智算网络仿真将深度融入智算全栈仿真体系，需要从系统工程全局视角统筹仿真体系的架构设计与建模实现。

5.1 跨域一体化仿真

未来智算网络仿真将突破当前仿真的场景局限性，构建支持异构算力、跨域网络、多应用负载的一体化仿真平台。

要打破规模扩展性瓶颈，需要从3个方向实现技术突破：

1）跨域同步机制优化。通过轻量化通信协议等方式解决多域时序一致性问题，保障全局仿真逻辑的准确性。

2）模块化组件设计。通过标准化接口与组件化架构设计，统一仿真模块的交互规范，降低大规模仿真系统的维护成本。

3）动态负载适配。引入资源弹性分配机制，根据仿真负载动态调度系统资源，保障仿真系统稳定性与高效性。

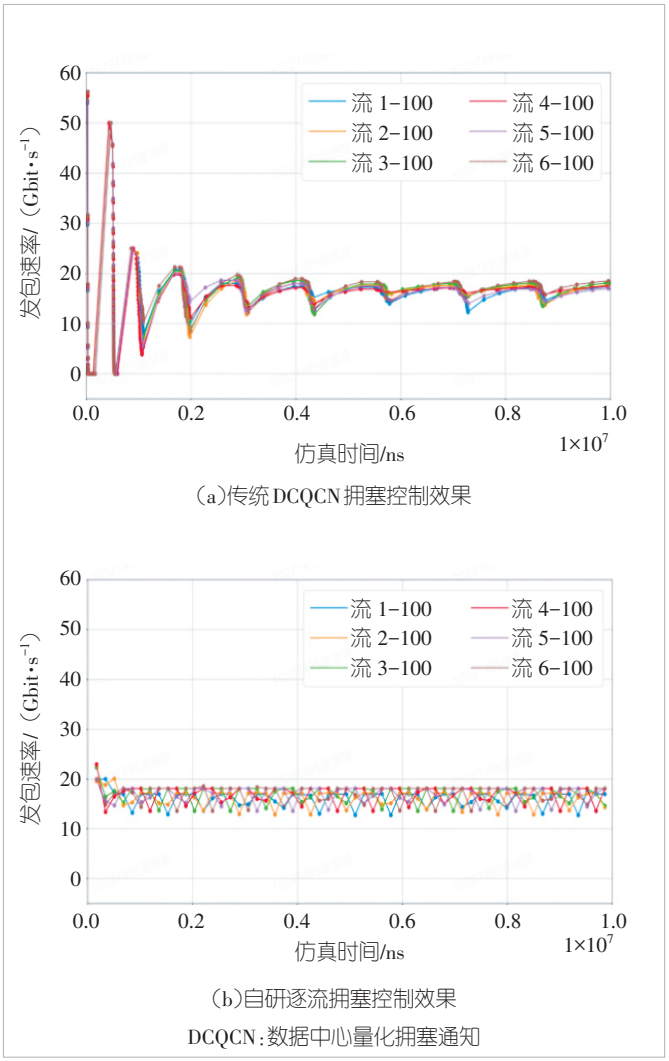


图3　传统DCQCN与自研逐流拥塞控制的灌包测试对比

表1 万卡智算集群Megatron MoE 7B模型32k长序列并行策略对比

	并行策略	计算效率/%	空泡占比/%	TP通信占比/%	EP通信占比/%	PP通信占比/%	DP通信占比/%
1	TP8PP16VPP2EP32DP96	89.38	6.24	3.06	0.90	0.39	0.03
2	TP4PP12VPP2EP64DP256	80.94	15.14	1.97	1.31	0.56	0.08
3	TP2PP16VPP4EP32DP384	73.41	22.20	1.10	2.10	0.91	0.28
4	TP2PP32VPP1EP8DP192	50.85	48.35	0.22	0.33	0.16	0.09

DP：数据并行 EP：专家并行 PP：流水线并行 TP：张量并行

ATLAHS[13]工具链的设计理念值得参考，该框架通过构建多场景的通用仿真底座，搭配定制化插件适配不同应用的负载特征。此外，还需要解决跨域数据传输延迟精准建模和跨域安全机制建模等问题，以适配更复杂的跨域智算场景[3]。

5.2 微宏协同全栈仿真

未来智算网络仿真将破除分层割裂的仿真壁垒，逐步构建覆盖“算力芯片微架构、网络传输、算法模型、系统部署”的全栈仿真体系，实现从底层硬件创新到上层系统优化的端到端支持。

相关技术探索已从多个维度开展：微观层面刻画算力芯片微架构的并行访问、指令时序等核心要素特征；宏观层面捕捉任务调度、通信交互等系统级的涌现特性。

微宏协同关键技术包括：

1）模型间自适应融合，支持全链路上模型跨粒度动态切换。多层次建模和混合仿真是实现不同模型之间的动态切换和数据交换的常用方式。例如，SplitSim[16]使用的混合保真度方法就是这一技术的早期探索。

2）跨层参数关联，捕捉全栈各层级间动态交互关系。例如，Transformer框架[33]梯度压缩算法对网络数据传输量的影响、网络拥塞对硬件算力利用率的影响、芯片通信延迟对算法效率的影响，均可通过该技术实现量化建模。

当前，该技术趋势已初见端倪。中兴通讯智算前瞻团队也自研了全栈仿真平台，一方面持续持续优化平台架构，提升平台效率；另一方面利用仿真平台对存算一体、内存池化、算力池化等新型硬件形态开展前瞻性技术探索。

5.3 实时数字孪生

未来仿真将从“离线设计验证”走向“在线实时优化”，构建“物理网络–数字孪生–决策优化”的闭环系统，最终成为在线运维的核心引擎。

数字孪生与仿真技术的深度融合已形成明确技术路径[28]：通过高效数据采集与传输实现物理网络的精准映射，依托仿真引擎完成实时分析与优化决策，最终通过执行与反馈实现闭环控制。两者融合关键在于：1）高效数据采集与传输，保障物理环境与仿真环境高可靠的数据交互；2）虚实协同决策机制，保障决策指令快速匹配和实时响应。

当前实时数字孪生技术发展迅速，持续向轻量化仿真引擎演进。仿真引擎不仅是分析工具，更是故障预判、系统自愈、性能优化等技术创新的基石。

5.4 智能化仿真新范式

未来仿真将更多使用大模型技术替代传统的解析建模与离散事件驱动，逐步推动“仿真即服务”的自动化演进。

大模型与仿真技术深度融合，有望实现两大核心突破：

1）自动化建模。利用LLM的自然语言理解和生成能力，可根据描述性文本自动生成仿真模型。例如，基于系统网络架构文档即可生成对应的网络拓扑、流量模型和参数配置[31]；也可利用物理信息神经网络（PINN）等技术，直接从数据中学习物理规律并构建模型。

2）智能化决策。依托实时数据流与预测模型，自动输出拓扑调整、资源调度等优化策略[11]，为智算服务的在线调度提供精准实时的决策支撑，从而显著提升系统运维效率。

虽然智能化仿真面临数据采集成本高、模型泛化能力不足、可解释性缺失等技术挑战，但从长远来看，智能化仍是仿真范式演进的关键技术方向。目前，m4[31]、MimicNet[34]等项目已初步验证了该范式的技术可行性和复杂场景下的应用潜力。

6 结束语

大规模智算网络的规模化部署与高性能需求，驱动着网络仿真技术从“辅助工具”向“基础设施”转变。同时，智算业务的复杂协同特性，正推动网络仿真深度融入计算、网络、存储全栈仿真体系。尽管当前智算网络仿真仍面临规模、精度、真实性、异构性和标准化等多重挑战，但通过分布式仿真、并行仿真、混合建模、数据驱动、AI辅助以及云原生等关键技术优化，这些痛点正在被逐一突破。未来，网络仿真将摆脱离线运行的局限，演进为与物理环境深度融合的设计、验证与优化核心引擎。行业竞争的焦点也将逐步

转向仿真架构的自主进化能力与集成深度。

参考文献

[1] 段晓东, 程伟强, 张昊. 智算网络发展综述[J]. 中兴通讯技术, 2025, 31(2): 53 - 62. DOI:10.12142/ZTETJ.202502008

[2] 边彦晖, 刘明远, 虞红芳. 面向多算力中心协同的广域智算网络仿真架构设计 [J]. 中兴通讯技术, 2025, 31(2): 39–46. DOI: 10.12142/ZTETJ.202502006

[3] 翟恩南, 操佳敏, 钱坤, 等. 面向大模型时代的网络基础设施研究: 挑战、阶段成果与展望 [J]. 计算机研究与发展, 2024, 61(11): 3664–3677

[4] Akram A, Sawalha L. A survey of computer architecture simulation techniques and tools [J]. IEEE access, 2019, 7: 78120–78145. DOI: 10.1109/ACCESS.2019.2917698

[5] Riley G F, Henderson T R. The ns–3 network simulator [M]// Wehrle K, Güneş M, Gross J. Modeling and Tools for Network Simulation. Berlin, Heidelberg: Springer, 2010: 15–34. DOI: 10.1007/978–3–642–12331–3_2

[6] Bai S Y, Zheng H, Tian C, et al. Unison: a parallel–efficient and user–transparent network simulation kernel [C]//Proceedings of the Nineteenth European Conference on Computer Systems. ACM, 2024: 115–131. DOI: 10.1145/3627703.3629574

[7] Gao K H, Chen L, Li D, et al. DONS: fast and affordable discrete event network simulation with automatic parallelization [C]// Proceedings of the ACM SIGCOMM 2023 Conference. ACM, 2023: 167–181. DOI: 10.1145/3603269.3604844

[8] Fan W W, Hong L J, Jiang G X, et al. Review of large–scale simulation optimization [PP/OL]. arXiv (2024–03–23) [2025–12–26]. https://arxiv.org/abs/2403.15669

[9] Won W, Heo T, Rashidi S, et al. ASTRA–sim2.0: modeling hierarchical networks and disaggregated systems for large–model training at scale [PP/OL]. arXiv (2023–03–24) [2025–12–26]. https://arxiv.org/abs/2303.14006

[10] Casanova H, Giersch A, Legrand A, et al. Lowering entry barriers to developing custom simulators of distributed applications and platforms with SimGrid [J]. Parallel computing, 2025, 123: 103125. DOI: 10.1016/j.parco.2025.103125

[11] Wang X, Li Q, Feng Y, et al. SimAI: unifying architecture design and performance tuning for large–scale large language model training with scalability and precision [C]//22nd USENIX Symposium on Networked Systems Design and Implementation (NSDI' 25). USENIX Association, 2025: 541–558. DOI: 10.5281/zenodo.14287654

[12] Agrawal A, Kedia N, Mohan J, et al. Vidur: a large–scale simulation framework for LLM inference [PP/OL]. arXiv (2024–05–08) [2025–12–26]. https://arxiv.org/abs/2405.05465

[13] Shen S Y, Bonato T, Hu Z Y, et al. ATLAHS: an application–centric network simulator toolchain for AI, HPC, and distributed storage [C]//Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis. ACM, 2025: 349–367. DOI: 10.1145/3712285.3759838

[14] Zuo P F, Lin H M, Deng J B, et al. Serving large language models on Huawei CloudMatrix384 [PP/OL]. arXiv (2025–06–15) [2025–12–26]. https://arxiv.org/abs/2506.12708

[15] Gandhi S, Zhao M, Hariharan S, et al. SlipStream: adapting pipelines for distributed training of large DNNs amid failures [PP/OL]. arXiv (2024–05–22) [2025–12–26]. https://arxiv. org/abs/2405.14009

[16] Li H J, Balasubramanian P, Meiers M, et al. SplitSim: large–scale simulations for evaluating network systems research [PP/OL]. arXiv (2024–02–07) [2025–12–26]. https://arxiv. org/abs/2402.05312

[17] Kortas N, Recker T. Large–scale space network simulator for performance–optimized DTNs [C]//Proceedings of the 15th International Conference on Satellite and Space Communications (SPACOMM 2023). 2023: 45–50. DOI: 10.5220/0012178900450050.

[18] Gui F, Gao K H, Chen L, et al. Accelerating design space exploration for LLM training systems with multi–experiment parallel simulation [C]//Proceedings of the 22nd USENIX Symposium on Networked Systems Design and Implementation. ACM, 2025: 473–488. DOI: 10.5555/3767955.3767980

[19] Wang H D, Yan H, Rong C, et al. Multi–scale simulation of complex systems: a perspective of integrating knowledge and data [PP/OL]. arXiv (2023–06–17) [2025–12–26]. https://arxiv.org/abs/2306.10275

[20] Svedas J, Watson H, Laubeuf N, et al. A survey of end–to–end modeling for distributed DNN training: workloads, simulators, and TCO [PP/OL]. arXiv (2025–06–10) [2025–12–26]. https://arxiv.org/abs/2506.09275

[21] Kumar S, Temura A, Sharma N, et al. Simulating LLM training workloads for heterogeneous compute and network infrastructure [C]//Proceedings of the 2nd Workshop on Networks for AI Computing. ACM, 2025: 105–107. DOI: 10.1145/3748273.3749212

[22] Heo J, Kim S, Shin H, et al. NetTLMSim: a virtual prototype simulator for large–scale accelerator networks [C]//The 3rd Workshop on Computer Architecture Modeling and Simulation (CAMS´23). ACM, 2023. DOI: 10.1145/3619895.3623250

[23] Rico–Gallego J A, Díaz–Martín J C, Manumachu R R, et al. A survey of communication performance models for high–performance computing [J]. ACM computing surveys, 2019, 51 (6): 1–36. DOI: 10.1145/3284358

[24] Aarwal A, Anand S, Jain A, et al. Arcadia: an end–to–end AI system performance simulator [EB/OL]. (2023–09–07) [2025–12–26]. https://engineering. fb. com/2023/09/07/data–infrastructure/arcadia–end–to–end–ai–system–performance–simulator/

[25] Nguyen Q C. Network sampling: an overview and comparative analysis [PP/OL]. arXiv (2025–04–24) [2025–12–26]. https://arxiv.org/abs/2504.17701

[26] Yang Q Q, Peng X, Chen L, et al. DeepQueueNet: towards scalable and generalized network performance estimation with packet–level visibility [C]//Proceedings of the ACM SIGCOMM 2022 Conference. ACM, 2022: 441–457. DOI: 10.1145/3544216.3544248

[27] Khashab S, Sezhiyan H, Abboud R, et al. NSX: large–scale network simulation on an AI server [C]//Proceedings of the 2nd Workshop on Networks for AI Computing. ACM, 2025: 19–25. DOI: 10.1145/3748273.3749199

[28] Yang L Y, Luo S, Cheng X, et al. Leveraging large language models for enhanced digital twin modeling: trends, methods, and challenges [PP/OL]. arXiv (2025–03–04) [2025–12–26]. https://arxiv.org/abs/2503.02167

[29] Qian Y C, Shu R, Ma R, et al. Miniature: fast AI supercomputer networks simulation on FPGAs [C]//Proceedings of the 9th Asia–Pacific Workshop on Networking. ACM, 2025: 114–120. DOI: 10.1145/3735358.3735369

[30] 田海东, 张明政, 常锐, 等. 大模型训练技术综述 [J]. 中兴通讯技术, 2024, 30(2): 21–28. DOI: 10.12142/ZTETJ.202402004

[31] Li C N, Zabreyko A A, Nasr–Esfahany A, et al. m4: a learned flow–level network simulator [PP/OL]. arXiv (2025–03–03) [2025–12–26]. https://arxiv.org/abs/2503.01770

[32] Yu G B, Tan G, Huang H J, et al. A survey on failure analysis and fault injection in AI systems [PP/OL]. arXiv (2024–06–28) [2025–

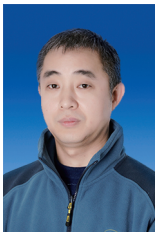
12–26]. https://arxiv.org/abs/2407.00125
[33] 朱炫鹏, 姚海东, 刘隽, 等. 大语言模型算法演进综述 [J]. 中兴通讯技术, 2024, 30(2): 9–20
[34] Zhang Q Z, Ng K K W, Kazer C, et al. MimicNet: fast performance estimates for data center networks with machine learning [C]//Proceedings of the 2021 ACM SIGCOMM 2021 Conference. ACM, 2021: 287–304. DOI: 10.1145/3452296.3472926

作 者 简 介

熊先奎，中兴通讯股份有限公司无线首席架构师、智算技术委员会前瞻组组长；长期从事计算系统和体系结构、先进计算范式以及异构计算加速器研究工作；曾主导过中兴通讯 ATCA 先进电信计算平台、服务器存储平台、智能网卡和AI加速器等系统架构设计。

童贤慧，中兴通讯股份有限公司先进计算存储系统工程师；长期从事智算系统仿真、安全可信计算等方向的技术研究。

王勇，中兴通讯股份有限公司预研工程师；长期从事网络协议、可编程网络以及加速器相关的技术研发工作。



张启明，中兴通讯股份有限公司系统架构师；长期从事智算网络及系统架构前瞻预研工作。