专栏

# 面向AI算力场景的多元异构混合训练系统研究

李攀攀[1]，牛红韦华[1]，赵万龙[1]，马华伟[2]，王艳辉[1]，江伟[3]，张雯欣[1]，陆一鸣[1]，赵峰[1]
（1. 中移（苏州）软件技术有限公司，江苏 苏州 215123；
2. 中国移动通信集团设计院有限公司，北京 100080；
3. 中国移动通信集团设计院有限公司安徽分公司，安徽 合肥 230041）

**摘　要：**大语言模型训练是人工智能（artificial intelligence，AI）发展的核心场景，在算力多元化和异构化趋势下，跨生态异构算力协同能力将成为十万卡级训练的关键支撑。基于此背景，设计了一套异构AI算力混合训练系统，该系统能够主动检测、适配异构AI芯片，实现异构算力间的集合通信。基于该原型系统，在一个由3种异构算力组成的RoCEv2网络互通集群实现了多种异构算力组合的混训。在异构流水线并行（pipeline parallelism，PP）混训场景下，英伟达与壁仞的最优混训效率达到99.77%，英伟达、天数智芯、壁仞的最优混训效率可达99.03%。在异构数据并行（data parallelism，DP）混训场景下，英伟达与壁仞的最优混训效率达到92.88%。

**关键词：**大语言模型；集合通信；异构并行；异构混合训练
**中图分类号：**TP393
**文献标志码：**A
**doi:** 10.11959/j.issn.1000−0801.2025164

# Research on multi-heterogeneous hybrid training system for AI computing power scenarios

LI Panpan[1], NIU Hongweihua[1], ZHAO Wanlong[1], MA Huawei[2], WANG Yanhui[1], JIANG Wei[3], ZHANG Wenxin[1], LU Yiming[1], ZHAO Feng[1]
1. China Mobile (Suzhou) Software Technology Co., Ltd., Suzhou 215123, China
2. China China Mobile Group Design Institute Co., Ltd., Beijing 100080, China
3 Anhui Branch of China Mobile Group Design Institute Co., Ltd., Hefei 230041, China

**Abstract:** Large language model training is a pivotal scenario in AI development. Under the trend of diversified and heterogeneous computing power, the cross-ecosystem heterogeneous computing power collaboration capability will become the key support for training at the hundred-thousand-card-scale. Based on this background, a heterogeneous AI computing power mixed training system was designed, which could automatically detect and adapt to heterogeneous AI chips, enabling collective communication among heterogeneous computing powers. Based on the prototype system, heterogeneous training was implemented using three types of AI chips in a RoCEv2-interoperable cluster. In the heteroge-

收稿日期：2025−03−21；修回日期：2025−07−02
通信作者：李攀攀，lipanpan@cmss.chinamobile.com

neous pipeline parallelism (PP) training scenario, peak training efficiency reached 99.77% using NVIDIA and Biren GPU, and 99.03% using NVIDIA, Iluvatar, and Biren GPU. For heterogeneous data parallelism (DP) training, the optimal mixed training efficiency between NVIDIA and Biren GPU reached 92.88%.

**Key words:** large language model, collective communication, heterogeneous parallelism, heterogeneous hybrid training

## 0 引言

人工智能（artificial intelligence，AI）已成为21世纪的关键技术之一，其中Transformer架构[1]成为大语言模型的基础。规模化法则[2]指出，模型性能会随模型参数、训练数据的增加而提升，出现“涌现能力”[3]，但随之而来的训练成本也极大提升，如Llama 3.1-405B仅预训练就使用了16 384张H100图形处理器（graphics processing unit，GPU），并耗时约54天完成[4]。

由于海外对华高端AI算力接连禁售，构建自主可控的国产化算力生态势在必行。目前，国内华为、壁仞、天数智芯等已推出AI芯片，然而，各厂商AI芯片在硬件架构、配套软件生态等方面互不兼容，形成了壁垒分明的“生态竖井”现象。为解决这一问题，面向AI异构芯片的混合训练技术成为整合当前多元生态、提升异构AI资源利用率的关键。本文设计了一个具备主动兼容能力的异构混合训练系统，实现了主流开源模型在3类异构AI芯片上的规模化混合训练，为国产AI算力生态建设提供了可行的方案。

本文的主要创新点如下。

（1）本文设计了一种支持异构AI芯片在同一系统中互相感知与兼容的方法，能够主动检测、适配异构AI芯片，实现基础异构远程直接内存访问（remote direct memory access，RDMA）点对点（peer-to-peer，P2P）原子通信能力，能主动实现异构芯片间的通信兼容，确保可以基于统一深度学习框架、分布式框架互相识别与协作。

（2）本文设计了一种异构分布式场景下进程组初始化逻辑与并行机制，支持异构数据并行（data parallelism，DP）、异构流水线并行（pipeline parallelism，PP），能够根据异构芯片计算能力，完成模型或数据的非均匀分配，实现异构算力利用效率的最大化。

## 1 异构混训关键基础技术及业内研究现状

### 1.1 集合通信

在大模型分布式训练场景中，优化分布式通信逻辑，实现训练过程的中间态数据高效交互非常关键。消息传递接口（message passing interface，MPI）[5]作为一种信息传递接口标准与规范，基于其实现的OpenMPI、MPICH等通信库可提供标准的应用程序接口（application program interface，API），从而可以被C、C++、Python等多类语言实现系统级、应用级程序调用。集合通信库（collective communication library）针对特定异构硬件与网络架构进行优化，如英伟达集合通信库（NVIDIA collective communication library，NCCL）、超威半导体（Advanced Mircro Devices，AMD）Radeon集合通信库RCCL、华为HCCL，具备全局通信拓扑感知、路径规划等能力，在AI并行计算场景中可达到更低时延、更高带宽，但目前各类集合通信库只针对自家AI芯片与硬件平台进行了优化适配，缺乏跨平台兼容性，限制了异构算力的协同使用。

### 1.2 分布式并行

DP、张量并行（tensor parallelism，TP）和PP是大模型分布式并行训练系统中常见的方式，也被称为3D并行。

在DP场景中，分布式系统包含多个相同模型

副本，数据集被拆分为多份，并由不同计算节点上的模型副本完成处理。在前向传播、反向传播完成后，通过allreduce操作全局同步模型梯度。

TP将单层模型的多维张量计算拆分为多个子张量运算，并分配到不同的AI芯片上执行，各AI芯片在完成计算后通过allreduce、broadcast等集合通信操作进行结果同步。

PP则是将模型按层数分割为不同的PP Stage，并分配到不同的AI节点上进行计算，其核心是将全局批次拆分为多个按序计算的微批次，实现阶段间计算重叠。这种方案需要权衡激活值内存开销和流水线气泡问题，主流方法如PipeDream-1F1B[6]通过对流水线计算优化，实现内存与计算气泡之间的平衡。

### 1.3　异构混合训练

早期异构混合训练研究多基于CPU-GPU协同计算，如文献[7]提出一种分布式异构深度神经网络（deep neural network，DNN）训练框架BytePS，通过对参数优化器拆分，实现CPU梯度聚合与GPU模型计算的异构加速。在GPU-GPU异构场景，文献[8]设计了HetPipe框架，将PP与DP进行结合，通过分区算法在包含多个GPU的PP Stage内自动划分DNN层，并通过波同步并行（wave synchronous parallel，WSP）提供异构GPU集群间的参数同步，实现在NVIDIA不同代际GPU上的混合训练。Whale是由Jia等[9]提出的一个适配异构集群场景的分布式训练框架，通过引入硬件感知并行策略，实现在V100-P100异构GPU集群上混合训练ResNet50、BertLarge等模型。文献[10]提出了一个名为HETHUB的大模型混合并行分布式训练系统，通过引入一个新的无限集合通信库（infinity collective communication library，ICCL），实现了异构AI芯片间的通信，并集成分布式性能预测和自动并行规划器的功能，实现了包括AMD以及NVIDIA GPU等在内的共计6种异构AI芯片的两两组合混训。

### 1.4　研究现状分析

现有的异构混合训练研究（如BytePS、HetPipe、Whale）主要针对CPU-GPU异构、同厂商不同代际GPU异构的混合训练，聚焦训练任务在异构场景下的并行调度逻辑优化，缺乏跨生态AI芯片的高效通信兼容与优化方案。文献[11]提出了一种基于弹性联邦学习和多智能体深度强化学习的方案，解决了异构AI节点间的通信兼容与任务调度问题，相比传统异构混合训练框架，该方案能自适应不同算力差异和通信瓶颈，提升多元算力场景下异构芯片间的协同效率。

## 2　基于异构算力集群的大模型训练系统设计及实现

### 2.1　混训系统架构

本文以异构通信兼容库及异构分布式并行兼容库为核心，设计了一套面向AI算力场景的异构混合训练系统。其中，异构通信兼容库主要实现异构通信技术，包括异构芯片互相感知逻辑、异构通信流程设计与优化、异构通信方法实现，支持构建、管理全局统一的异构通信拓扑，支持异构集合通信；异构分布式并行兼容库主要实现异构并行技术，基于异构通信兼容能力，区分异构DP、PP场景，分别实现异构进程组初始化编排、模型与数据集非均匀切分，从而匹配异构AI芯片的计算能力。异构混合训练系统架构如图1所示。

### 2.2　异构通信技术

#### 2.2.1　异构集群拓扑感知

集群拓扑感知与构建是实现分布式训练的前提条件，需要让分布式系统中每个GPU进程（rank）都能感知到其他GPU进程的IP地址、服务端口号、设备类型等基础信息，并基于此信息进一步规划并行与通信逻辑。异构拓扑感知方法整体流程如图2所示，本文所涉及的异构集群的拓扑感知及构建方法主要包含以下3个阶段。

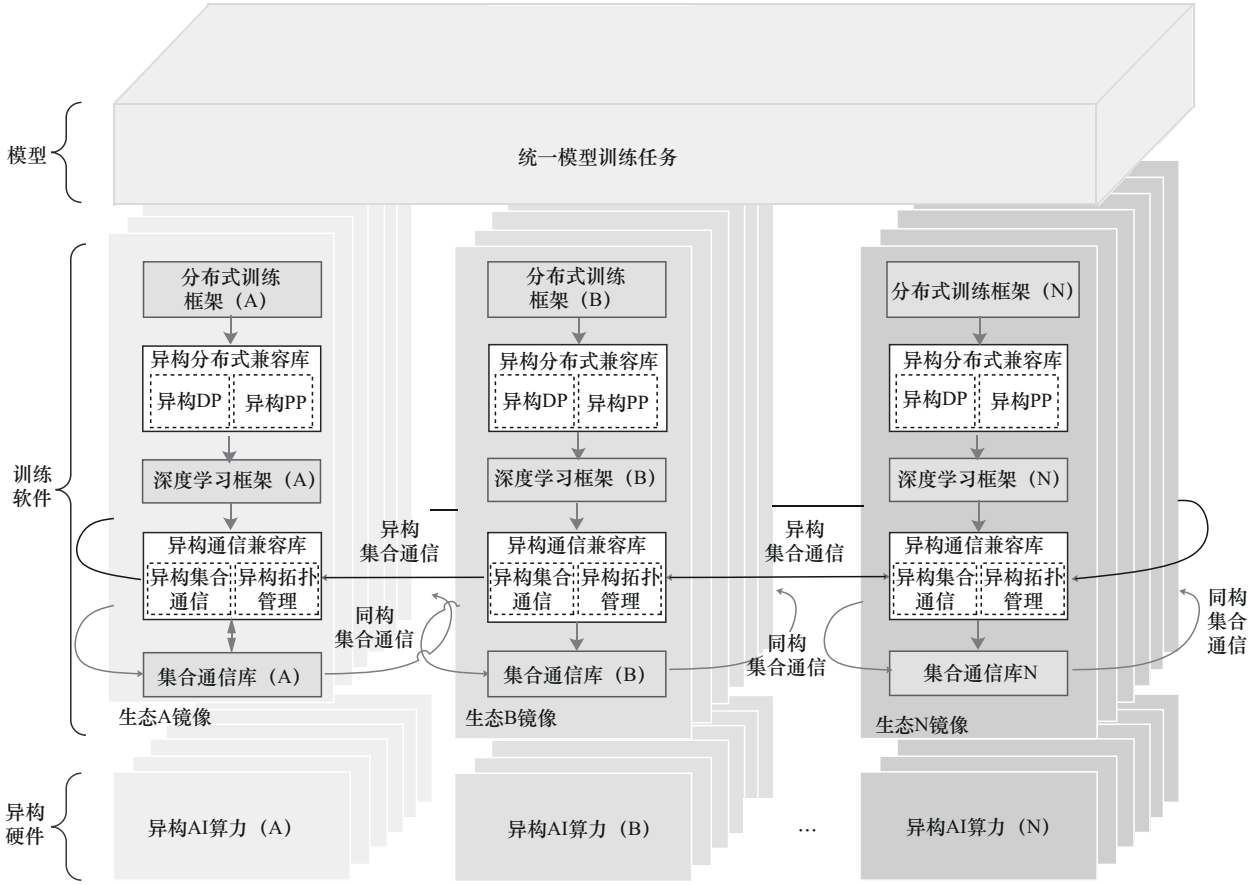


图1　异构混合训练系统架构

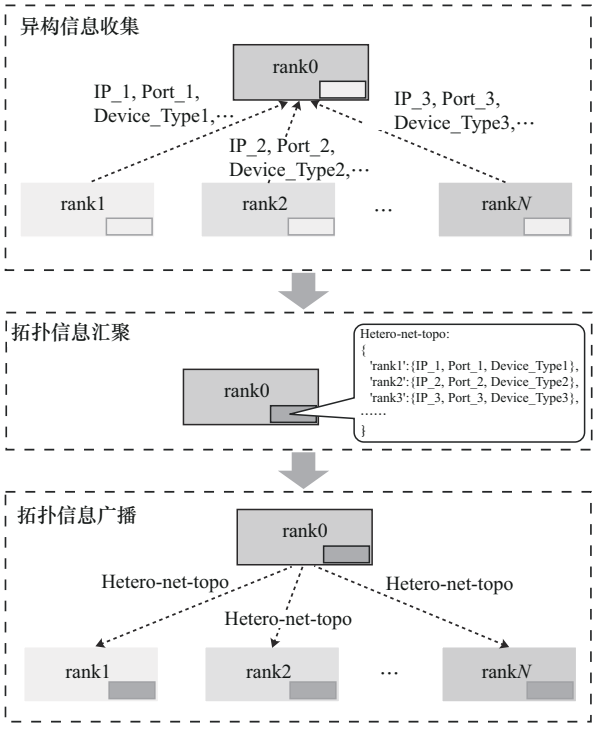


图2　异构拓扑感知方法整体流程

（1）异构信息收集阶段：各异构节点在启动任务时配置全局rank0的IP地址与端口，完成初始化后，通过TCP Socket将自身IP地址、服务端口及设备类型等信息发送至rank0。

（2）拓扑信息汇聚阶段：rank0负责信息汇聚与异构拓扑编排，在接收所有rank的IP地址、服务端口及设备类型等信息后，依据配置的异构并行模式构建异构拓扑，以rank号为索引键，对应值为｛IP, port, device_type｝。

（3）拓扑信息广播阶段：rank0记录所有rank的IP地址与端口后，通过TCP Socket将汇总构建的异构拓扑信息广播至集群内所有rank，从而实现全局rank间异构拓扑信息的同步。

### 2.2.2　异构通信总体流程

通过异构拓扑感知技术，异构集群的各个rank已经获取到全局异构拓扑、设备类型等信

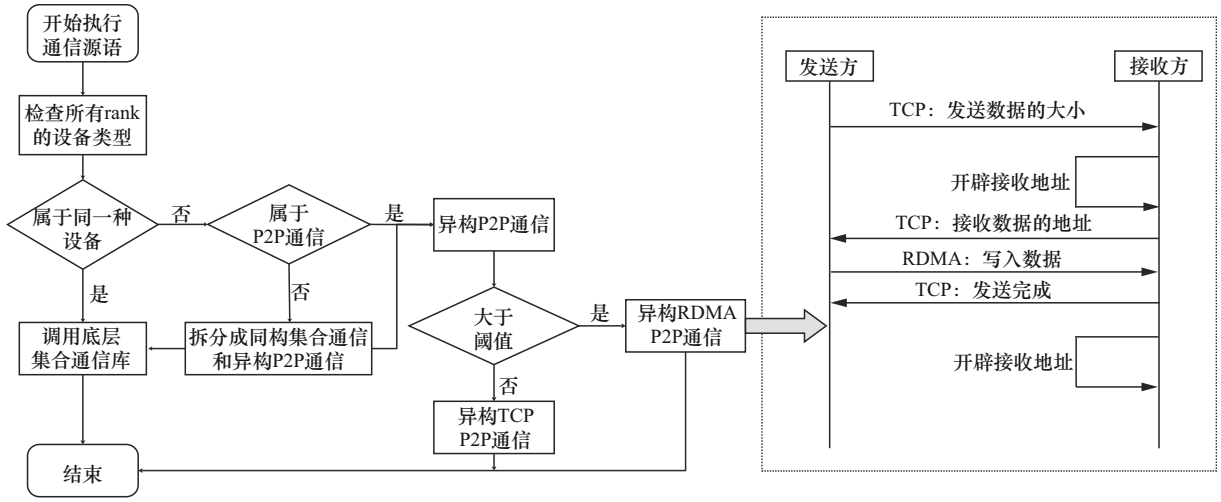


图3　异构通信流程

息。基于此特性，本文设计了异构混合训练场景下的基础通信流程。异构通信流程如图3所示。

（1）在发生节点间的通信行为前，检测待通信的rank是否属于同一类设备，若属于，则直接调用同构集合通信库完成指定集合通信操作，否则定义为异构通信场景。

（2）检测通信类型是否属于P2P通信，若属于，则进入异构P2P通信流程，否则进入异构集合通信流程。对于异构P2P通信以及异构集合通信的技术实现，在本文的2.2.3节和2.2.4节详细说明。

### 2.2.3　异构P2P通信

异构通信仅存于机间交互，典型方式包括TCP与RDMA。传统RDMA进程间数据传输依赖TCP交换控制信息，小批量数据场景效率低于TCP，且异构GPU场景下控制与传输资源分配待优化。本文通过优化传统RDMA P2P传输机制，实现控制与通信传输资源复用，提升异构场景传输效率。

首先，明确差异化数据流传输策略，针对特定阈值以下的小批量数据直接走TCP以减少控制流损耗，大批量数据则采用RDMA。其次，针对RDMA P2P通信控制3次握手过程实现优化，流程如下。

首次握手，发送方告知接收方数据包大小，以便分配内存。本文对TCP Socket（控制）与RDMA（传输）资源仅初始化一次，建立通道后复用，同时对单节点不同rank亲和性分组，均匀绑定至多个RDMA通信网络接口，保障带宽负载均衡，节约资源并提效。

二次握手，接收方反馈接收数据地址，发送方启动数据发送。本文定义发送方直接利用待发送数据地址开辟内存区域（memory region，MR），若支持GDR（GPU direct RDMA），可直接从GPU搬运数据至RDMA，提高传输效率。

三次握手，发送方通知数据发送完毕，接收方读取数据。本文定义接收方依据张量维度最大化复用MR，规避新建MR的耗时开销，进一步提升效率。

### 2.2.4　异构集合通信

以异构P2P通信机制为原子能力，本文设计了自定义异构集合通信方法。针对send、isend、recv、irecv等P2P通信方法，本文直接复用异构P2P通信流程实现。对于高阶集合通信方法如allreduce、allgather，则拆分成同构集合通信和异构P2P通信方法进行实现。以异构allreduce为例，通过将具体的通信逻辑拆分成同构reduce、异构P2P和同构broadcast这3部分操作来实现数据规约。异构all-reduce场景下集合通信实现流程如图4所示。

在异构allreduce场景下，所有待通信rank构成全局异构通信组，各类同构设备形成多个同构

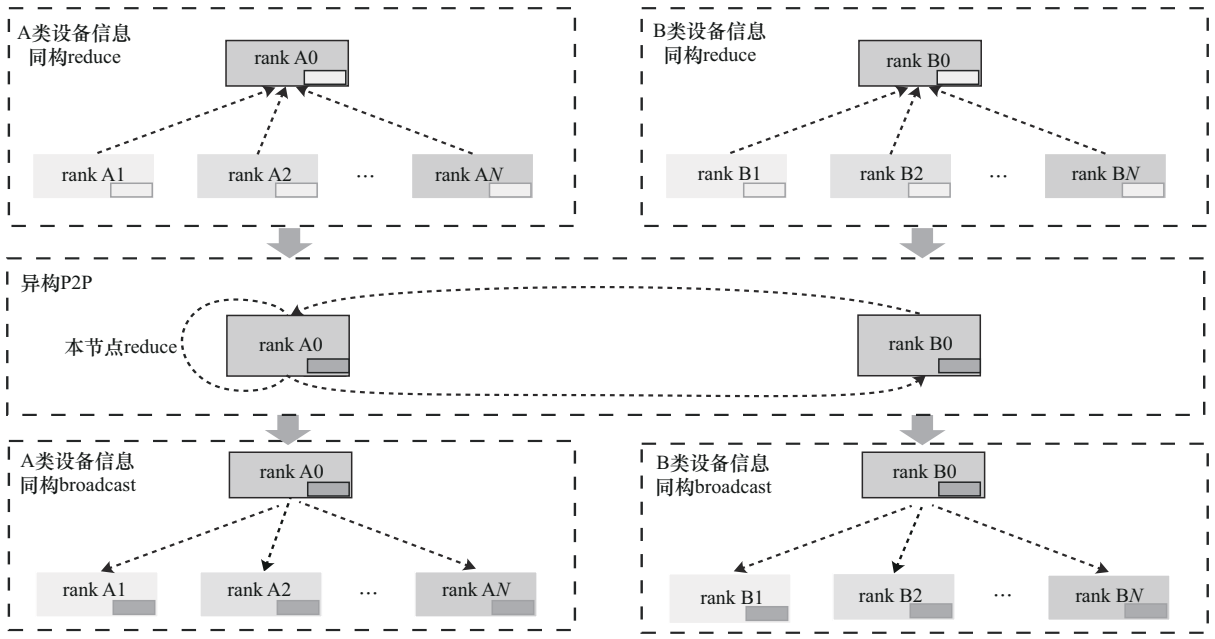


图4　异构allreduce场景下集合通信实现流程

子通信组（如A组rank编号为rankA0、rankA1、…、rankA*N*，B组为rankB0、rankB1、…、rankB*N*），具体异构allreduce通信流程如下。

（1）同构子通信组利用自有集合通信库执行组内reduce操作，数据汇聚于各子组首节点。

（2）各子组首节点将数据传输至rankA0，rankA0完成全局reduce后，通过P2P通信将最终规约数据回传至各同构子通信组首节点。

（3）各首节点借助原生集合通信库在子通信组内进行broadcast操作，使全局异构节点均获取最终规约数据，至此，异构allreduce通信完成。

### 2.3　异构并行技术

在同构集群的并行训练中，TP由于通信量较大，一般在机器内部使用，而机器内部通常不存在异构AI芯片场景，因此，本文主要对异构PP、异构DP进行了设计与实现。

#### 2.3.1　异构PP

本文实现了一种适配异构算力场景的PP方法，通过对异构PP流程中的模型参数初始化、进程组初始化、模型构建、前向训练通信过程等进行重构，实现对模型副本的非均匀切分，并通过异构isend、irecv方法实现异构PP通信。异构PP、异构DP在分布式训练场景中涉及的关键流程如图5所示。

（1）异构PP参数校验：选择异构PP模式时，具体参数设计如下。

- mixed-mode：异构混合训练模式，异构PP配置为pp，异构DP配置为dp。
- device-type：标识当前节点使用的设备类型，如nvidia、biren等。
- hetero-pipeline-stages：定义各类型设备的阶段拆分配置，格式应为$n_0L_0^0L_0^1,\cdots,n_iL_i^0L_i^1,\cdots$，每个代表对应设备类型的PP Stage数，其中每一个阶段对应的模型层数为$L_i^0L_i^1,\cdots$，总PP大小为$\sum n_i$，模型总层数为$\sum L$。

（2）异构PP进程组初始化：在异构PP场景中，TP组和DP组内的所有设备需要属于相同类型，PP组内则由异构AI芯片设备分别构成PP Stage，为保障通信兼容，同一PP Stage内所有计算和通信操作应在相同类型设备上执行。

根据异构PP场景的最终进程组划分目标，

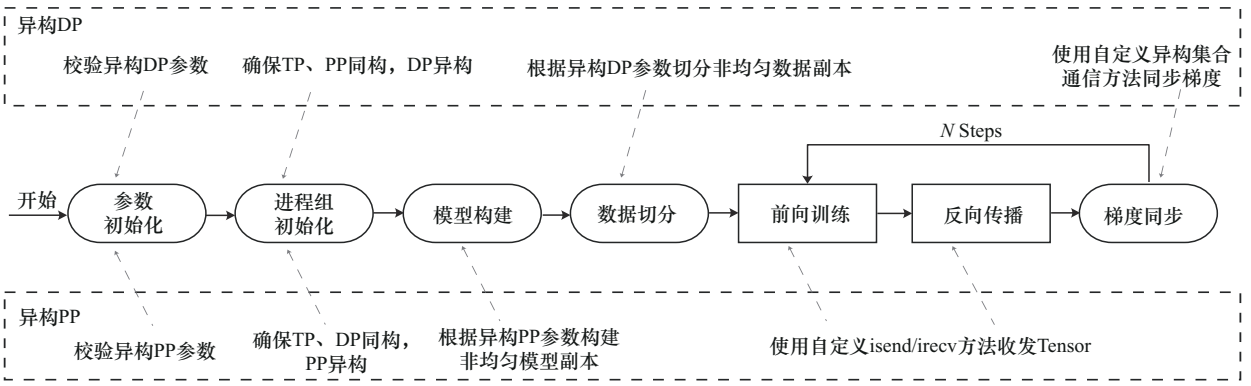


图5　异构PP、异构DP在分布式训练场景中涉及的关键流程

在初始化阶段，遵循TP-DP-PP顺序。假设给定world_size个节点，TP、DP、PP大小分别为tp_size、dp_size、pp_size，应满足tp_size×dp_size×pp_size=world_size，初始化逻辑如下。

**步骤1**　根据tp_size将相邻节点分配到同一TP组，共有$G_{tp}$=world_size/tp_size个TP组。

**步骤2**　按照间隔tp-size设置DP组，即在每个TP组之间跳过相应数量的节点来形成DP组，共有$G_{dp}$=world_size/dp_size个DP组。

**步骤3**　按照间隔tp_size×dp_size设置PP组，共有$G_{pp}$=world_size/pp_size个PP组。

（3）异构PP模型构建：根据异构AI芯片的计算能力来分配不同数量的网络层，具有较大显存的异构设备更适合承担PP靠前Stage的任务，从而保障异构PP场景下算力在整个任务系统中的负载均衡。

综上，本文进行异构PP混训模型构建的算法如下。

**算法1**　异构PP混训模型构建

输入：

num_layers：模型总层数

pp_size：pp并行度

pp_rank：当前节点所属的pp阶段

hetero_pipeline_stages：各类型设备的阶段拆分配置

输出：

model：按照拆分配置构建的当前节点维护的模型

**步骤1**　异构切分配置校验与解析

初始化stage_counter←0,stage_splits←[],layer_splits←[]

for item in hetero_pipeline_stages:

if stage_counter == 0 then

stage_splits追加item

stage_counter←item

else

layer_splits追加item

stage_counter ← stage_counter－1

end if

end for

if SUM(stage_splits)≠pp_size then

校验失败，抛出异常，停止运行

end if

if SUM(layer_splits)≠num_layers then

校验失败，抛出异常，停止运行

end if

**步骤2**　模型构建

num_layer←layer_splits[pp_rank]

pre_process←pp_rank==0

post_process←pp_rank == pp_size−1

model←构建TransformerLanguageModel(

embedding_layer←if pre_process then构建Embedding

transformer_layer←ParallelTransformer(num_layer)

output_layer←if post_process then构建VocabParallelCrossEntropy)

（4）异构PP数据交互：在异构PP的前向传播和反向传播过程中，涉及的异构设备传输会直接使用本文第2.2.4节中异构通信兼容库示例的isend、irecv方法来完成异构RDMA P2P通信，其他非异构设备间的集合通信仍保持调用原有集合通信库逻辑。

2.3.2 异构DP

本文实现了一种异构DP方法，并对进程组初始化与全局通信拓扑构建逻辑进行优化，根据异构AI芯片计算能力实现对小批量数据的非均匀切分与定向分配。异构芯片间的allreduce等高阶集合通信操作均通过异构通信兼容库中的自定义异构集合通信方法来实现，以确保异构DP组内的高效同步。下文将分流程进行阐述。

（1）异构DP参数校验：异构DP新增了特殊参数，在初始化阶段对该部分参数进行校验，并确定最终数据切分与分配逻辑，关键参数配置如下。

mixed-micro-batch-sizes：定义了不同设备类型指定不同的微批次大小，格式为$n_0$mbs$_0$,$n_1$mbs$_1$,…,$n_i$mbs$_i$。每个$n_i$代表对应设备类型的模型副本数量，总DP大小为$\sum n_i$，对于每一个种类$i$，mbs$_i$表示其设备类型的微批次大小。

（2）异构DP进程组初始化：异构DP的目标是在DP组间允许存在不同类型的异构设备，同时确保TP组、PP组内的设备类型一致，因此，异构DP混训基于TP-PP-DP的顺序完成整个系统的进程组初始化。假设给定world_size个节点，TP、DP、PP大小分别为tp_size、pp_size、tp_size，满足tp_size×dp_size×pp_size=world_size。初始化流程如下。

**步骤1** 根据tp_size将相邻节点分配到同一TP组，共有$G_{tp}$=world_size/tp_size个TP组。

**步骤2** 按照间隔tp_size设置PP组，共有$G_{pp}$=world_size/pp_size个PP组。

**步骤3** 按照间隔tp_size×pp_size设置DP组，共有$G_{dp}$=world_size/dp_size个DP组。

上述流程在TP组、PP组内计算一致性的前提下，实现了仅DP组间异构，简化了全局异构通信逻辑。

（3）异构DP数据切分：在异构分布式DP场景中，需要根据不同设备的计算能力对大模型训练输入的批次数据进行非均匀切分。为了确保数据的合理分配和负载均衡，本文进行异构DP数据切分的算法如下所示。

**算法2** 异构DP数据切分

输入：

global_batches：全局批次

mixed_micro_batch_sizes= [mbs1, mbs2, …,mbs$n$]：混合微批次大小数组

device_type：设备类型列表

micro_batch：微批次

gbs：全局批次大小

输出：

split_data：切分后的数据，分配到各设备

**步骤1** 初始化解析与微批次数量计算

解析：device_type和mixed_micro_batch_sizes

微批次数量计算：mixed_micro_batch_total=($n\_i$×mbs_$i$)的求和

验证：if (mixed_micro_batch_total%gbs==0)

否则，报告错误终止

**步骤2** 批次划分与数据分配

mixed_micro_batch_$i$= global_batches/mixed_micro_batch_total

micro_batch_$i$=mixed_micro_batch_$i$/$n\_i$,

for each device

数据分配：

• 同构设备：micro_batch_size=uniform

• 异构设备：micro_batch_size_*i*≠micro_batch_size_*j*

返回：split_data=[data_on_device1, data_on_device2,…]

（4）异构DP梯度同步：所有模型副本在完成前向计算与反向传播后会通过集合通信操作互相同步梯度，在异构DP场景中，本文采用2.2.4节提出的异构allreduce方法传递异构DP间的梯度数据。

## 3　混训系统验证结果及分析

本文使用百卡规模、RoCEv2参数网互联、3种芯片设备的异构集群进行混训验证，多元异构集群参数网络互联拓扑如图6所示。

异构集群硬件设备配置信息见表1。

表1　异构集群硬件设备配置信息

| 设备名 | 规格 | 机间RDMA互联带宽 | 本文使用设备数/台 |
|---|---|---|---|
| 英伟达 | A800-PCIe-80G×8 | 2×100 Gbit/s | 6 |
| 壁仞 | 104P-32G×8 | 2×100 Gbit/s | 6 |
| 天数智芯 | BI-V150-32G×16 | 2×100 Gbit/s | 3 |

### 3.1　异构混合训练效率评估标准

对于异构混合训练场景下的训练效率，本文以HETHUB等工作为基础进一步提出两种混训效率评估标准。

• 标准1：直接合并异构集群并等倍调整DP规模，此操作会扩大集群规模，进而改变集群总进程数（world_size）、DP规模及Global Batch Size。

• 标准2：保持合池后训练集群资源规模不变，维持3D并行策略，此方式排除了集群规模扩大与并行策略变更带来的线性比损耗，可更真实地评估混训效率损失。

两种标准遵循统一的混训效率计算式，如式（1）所示。假定混训效率为$E$，TGS（Token GPU Second）表示合池后每张卡的吞吐，TGS$i$为单一集群每张卡吞吐，共计有$n$个集群，每个集群有$m$张卡。

$$E = n \times m \times \mathrm{TGS} \div \sum_{i}^{n} m \times \mathrm{TGS}_i \tag{1}$$

对于评估标准1和标准2，其他超参如重计算、梯度累计等均保持不变，以确保调优变量对混训效率产生直接影响。

### 3.2　双芯异构PP混合训练

双芯异构PP混合训练场景使用48卡A800与48卡104P混合训练Llama 2-13B模型，其中混合训练的基线分别来自48卡A800及48卡104P训练Llama 2-13B的TGS数据。A800+104P PP混训基线数据见表2。其中，GBS表示全局批次大小，MBS表示微批次大小。

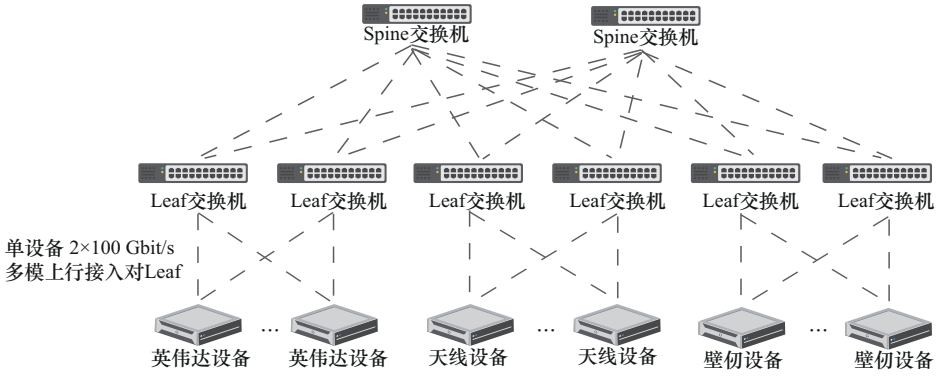


图6　多元异构集群参数网络互联拓扑

表2　A800+104P PP混训基线数据

| 基线规模 | GBS | MBS | TP | PP | DP | TGS |
|---|---|---|---|---|---|---|
| A800 48卡 | 192 | 4 | 4 | 4 | 3 | 619.67 |
| 104P 48卡 | 192 | 4 | 4 | 4 | 3 | 637.86 |

评估标准1验证时采用A800 48卡+104P 48卡，评估标准2验证时采用A800 24卡+104P 24卡。A800+104P PP混训结果见表3。

A800+104P异构集群在两种评估标准下的Loss和TGS趋势分别如图7、图8所示。

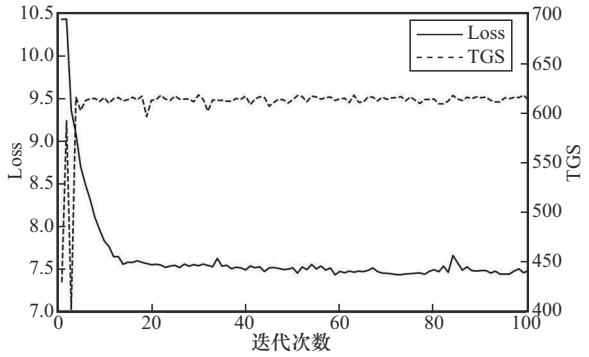


图7　评估标准1的Loss和TGS趋势

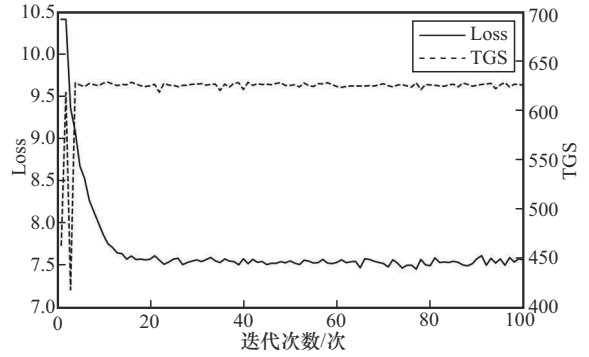


图8　评估标准2的Loss和TGS趋势

由图7和图8可知，A800+104P异构PP混训，使用标准1评估的混训效率为97.37%，使用标准2评估的混训效率为99.77%。在整个混训阶段，TGS性能整体稳定，Loss曲线正常收敛。

### 3.3　三芯异构PP混合训练

该场景使用48卡A800、48卡104P与48卡BI-V150混合训练Llama 2-13B模型，其中混合训练的基线分别来自48卡A800、48卡104P及48卡BI-V150训练Llama 2-13B的TGS数据。A800+104P+BI-V150混训基线数据见表4。

表4　A800+104P+BI-V150混训基线数据

| 基线规模 | GBS | MBS | TP | PP | DP | TGS |
|---|---|---|---|---|---|---|
| A800 48卡 | 256 | 4 | 4 | 3 | 4 | 647.84 |
| 104P 48卡 | 256 | 4 | 4 | 3 | 4 | 650.68 |
| BI-V150 48卡 | 256 | 4 | 4 | 3 | 4 | 406.96 |

评估标准1验证时采用A800 48卡+104P 48卡+BI-V150 48卡，评估标准2验证时采用A800 16卡+104P 16卡+BI-V150 16卡。A800+104P+BI-V150混训结果见表5。

表5　A800+104P+BI-V150混训结果

| 验证方式 | GBS | MBS | TP | PP | DP | 模型切分 | TGS | 混训效率 |
|---|---|---|---|---|---|---|---|---|
| 评估标准1 | 768 | 4 | 4 | 3 | 12 | 16,16,8 | 546.59 | 96.15% |
| 评估标准2 | 256 | 4 | 4 | 3 | 4 | 16,16,8 | 562.99 | 99.03% |

A800+104P+BI-V150异构集群在两种评估标准下的Loss和TGS趋势分别如图9、图10所示。

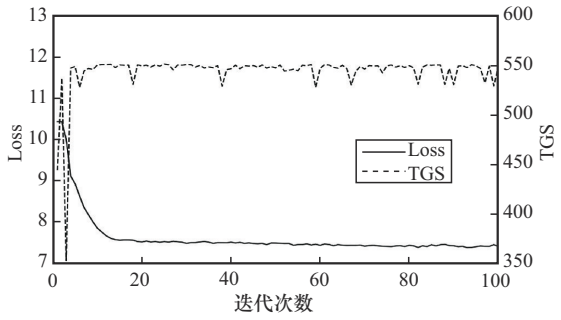


图9　评估标准1的Loss和TGS趋势

表3　A800+104P PP混训结果

| 验证方式 | GBS | MBS | TP | PP | DP | 模型切分 | TGS | 混训效率 |
|---|---|---|---|---|---|---|---|---|
| 评估标准1 | 384 | 4 | 4 | 4 | 6 | 10,10,10,10 | 612.21 | 97.37% |
| 评估标准2 | 192 | 4 | 4 | 4 | 3 | 10,10,10,10 | 627.29 | 99.77% |

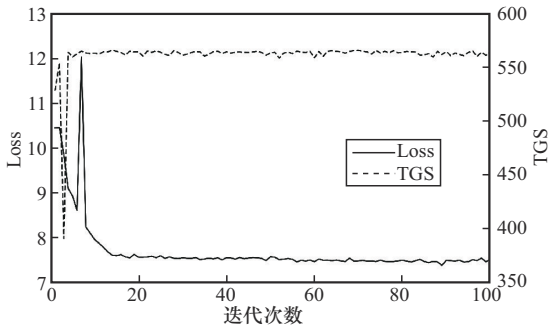


图10　评估标准2的Loss和TGS趋势

由图9和图10可知，A800+104P+BI-V150异构PP混训，使用标准1评估的混训效率为96.15%，使用标准2评估的混训效率达99.03%。

### 3.4　双芯异构DP混合训练

双芯异构DP混合训练场景使用8卡A800与8卡104P混合训练Llama 2-7B模型。A800+104P DP混训基线数据见表6，分别来自A800 16卡及104P 16卡训练Llama 2-7B的TGS数据。

表6　A800+104P DP混训基线数据

| 基线规模 | GBS | MBS | TP | PP | DP | TGS |
|---|---|---|---|---|---|---|
| A800 8卡 | 192 | 16 | 8 | 1 | 2 | 685.47 |
| 104P 8卡 | 192 | 16 | 8 | 1 | 2 | 623.20 |

评估标准2验证时采用A800 8卡+104P 8卡，A800+104P DP混训结果见表7。

A800+104P异构集群在评估标准2下的Loss和TGS趋势如图11所示。

表7　A800+104P DP混训结果

| 验证方式 | GBS | MBS | TP | PP | DP | 数据切分 | TGS | 混训效率 |
|---|---|---|---|---|---|---|---|---|
| 评估标准2 | 192 | 16 | 8 | 1 | 2 | 8,8 | 607.78 | 92.88% |

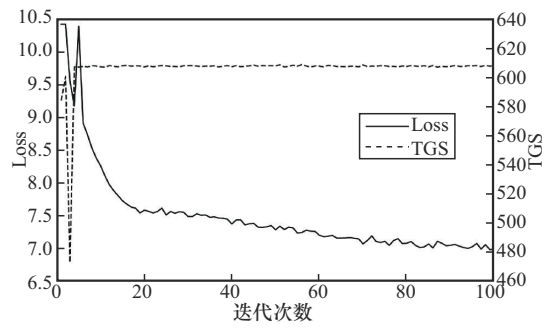


图11　评估标准2的Loss和TGS趋势

由图11可知，A800+104P异构DP混训，使用评估标准2时的混训效率达92.88%。

## 4　结束语

为打破异构算力生态壁垒，支撑多芯大规模混合训练场景，本文设计了一套面向AI算力场景的异构混合训练系统，支持多元异构算力设备主动感知、拓扑构建及异构集合通信等功能，实现了异构PP、异构DP混训能力。在异构PP混训场景下，“英伟达与壁仞”“英伟达、天数智芯与壁仞”的最优混训效率分别达到了99.77%、99.03%；在异构DP混训场景下，“英伟达+壁仞”的最优混训效率达到了92.88%。

多个混训场景的实验结果，也揭示了现阶段工作仍存在一些不足。在大规模场景下，TGS存在偶尔波动的情况，原因可能有两点：一是异构集合通信仍存在优化空间；二是用于实验的AI设备的互联带宽限制导致集群拓展后通信成为瓶颈。后续将继续研究通过该系统兼容更多生态AI算力的方法，通过优化通信算子进一步提升异构通信效率，探索多维混合并行与自动异构策略搜索等能力，以实现面向更大规模、更多生态场景的混合训练效率优化。

## 参考文献：

[1] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[J]. Advances in Neural Information Processing Systems, 2017: 5998-6008.

[2] KAPLAN J, MCCANDLISH S, HENIGHAN T, et al. Scaling laws for neural language models[J]. arXiv preprint, 2020: 2001.08361.

[3] WEI J, TAY Y, BOMMASANI R, et al. Emergent abilities of large language models[J]. arXiv preprint, 2022: 2206.07682.

[4] GRATTAFIORI A, DUBEY A, JAUHRI A, et al. The Llama 3 herd of models[J]. arXiv preprint, 2024: 2407.21783,.

[5] BARKER B. Message passing interface (MPI)[C]//Workshop: high performance computing on stampede. Houston: Cornell

University Publisher, 2015: 262.

[6] HARLAP A, NARAYANAN D, PHANISHAYEE A, et al. PipeDream: fast and efficient pipeline parallel DNN training[J]. arXiv preprint, 2018: 1806.03377.

[7] JIANG Y M, ZHU Y B, LAN C, et al. A unified architecture for accelerating distributed DNN training in heterogeneous GPU/CPU clusters[C]//Proceedings of the 14th USENIX Symposium on Operating Systems Design and Implementation (OSDI), 2020: 471-487.

[8] PARK J H, YUN G, YI C M, et al. HetPipe: enabling large DNN training on (whimpy) heterogeneous GPU clusters through integration of pipelined model parallelism and data parallelism[J]. arXiv preprint, 2020: 2005.14038.

[9] JIA X Y, JIANG L, WANG A, et al. Whale: efficient giant model training over heterogeneous GPUs[J]. arXiv preprint, 2020: 2011.09208.

[10] XU S, HUANG Z X, ZENG Y, et al. HETHUB: a distributed training system with heterogeneous cluster for large-scale models[J]. arXiv preprint, 2024: 2405.16256.

[11] WU Q, WANG W H, FAN P Y, et al. Cooperative edge caching based on elastic federated and multi-agent deep reinforcement learning in next-generation networks[J]. IEEE Transactions on Network and Service Management, 2024, 21(4): 4179-4196.

[作者简介]

**李攀攀**（1990– ），男，现就职于中移（苏州）软件技术有限公司，主要研究方向为云计算、智算中心、人工智能技术等。

**牛红韦华**（1989– ），女，中移（苏州）软件技术有限公司计划建设部副总经理，主要研究方向为大规模智算中心、公有云资源池架构、云管理平台等。

**赵万龙**（1994– ），男，现就职于中移（苏州）软件技术有限公司，主要研究方向为智算基础设施、人工智能技术等。

**马华伟**（1976– ），男，中国移动通信集团设计院有限公司高级工程师，主要研究方向为云计算、人工智能和数据业务网等。

**王艳辉**（1989– ），男，现就职于中移（苏州）软件技术有限公司，主要研究方向为智算基础设施、人工智能技术等。

**江伟**（1980– ），男，中国移动通信集团设计院有限公司高级工程师，主要研究方向为云计算、IT网络等。

**张雯欣**（1998– ），女，现就职于中移（苏州）软件技术有限公司，主要研究方向为智算基础设施、人工智能技术等。

**陆一鸣**（1998– ），男，现就职于中移（苏州）软件技术有限公司，主要研究方向为智算基础设施、人工智能技术等。

**赵峰**（1991– ），男，现就职于中移（苏州）软件技术有限公司，主要研究方向为智算基础设施、人工智能技术。