PR #32040 完整报告

sgl-project/sglang

[NPU] ascend fuseep use moe ep group

合并时间：2026-07-23 14:07

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/32040

执行摘要

- 一句话：Ascend FuseEP 改用 MoE EP 通信组
- 推荐动作：PR 逻辑清晰、改动极小，风险低。建议合并，并推荐后续添加测试覆盖。

功能与动机

PR body 指出，通过切换到 MoE EP group，可以借助 DEEPEP_HCCL_BUFFSIZE 环境变量独立配置 MoE 集合通信的 HCCL buffer，实现更灵活的通信资源控制。

实现拆解

1. 修改导入语句：在 python/sglang/srt/hardware_backend/npu/moe/fuseep.py 中，将 get_tp_group 替换为 get_moe_ep_group。
2. 修改 buffer 获取逻辑：在 _get_fuseep_buffer 函数中，将 get_tp_group().device_group 替换为 get_moe_ep_group().device_group，使 DeepEP 通信 buffer 使用 MoE EP 组的设备组。
3. 无其他改动：本次变更仅涉及上述两处，逻辑和接口保持不变。

关键文件：

- python/sglang/srt/hardware_backend/npu/moe/fuseep.py（模块 MoE 执行；类别 source；类型 dependency-wiring；符号 _get_fuseep_buffer, forward_fuseep）：核心变更文件，仅修改两行：导入函数和 device_group 来源。

关键符号：_get_fuseep_buffer, forward_fuseep

关键源码片段

python/sglang/srt/hardware_backend/npu/moe/fuseep.py

核心变更文件，仅修改两行：导入函数和 device_group 来源。

```
"""Ascend FuseEP fused dispatch+GEMM+combine forward path."""

from __future__ import annotations
from typing import TYPE_CHECKING
import torch

# 变更：从导入 get_tp_group 改为导入 get_moe_ep_group
```

```
from sglang.srt.distributed import get_moe_ep_group
from sglang.srt.environ import envs
from sglang.srt.hardware_backend.npu.utils import npu_format_cast
from sglang.srt.layers.moe.token_dispatcher.deepep import DeepEPBuffer
from sglang.srt.layers.moe.utils import DeepEPMode
from sglang.srt.runtime_context import get_exec

if TYPE_CHECKING:
    from sglang.srt.layers.moe.fused_moe_triton.layer import FusedMoE
    from sglang.srt.layers.moe.topk import TopKOutput

_PARAMS_BYTES = 2 # bf16 — Ascend's Dispatch & Combine does not support fp16

def _get_fuseep_buffer(layer: FusedMoE):
    DeepEPBuffer.set_dispatch_mode_as_low_latency()
    return DeepEPBuffer.get_deepep_buffer(
        # 变更：使用 MoE EP group 的设备组，而非 TP group
        get_moe_ep_group().device_group,
        layer.hidden_size,
        _PARAMS_BYTES,
        DeepEPMode.LOW_LATENCY,
        envs.SGLANG_DEEPEP_NUM_MAX_DISPATCH_TOKENS_PER_RANK.get(),
        layer.num_experts,
    )

def forward_fuseep(
    layer: FusedMoE,
    hidden_states: torch.Tensor,
    topk_output: TopKOutput,
) -> torch.Tensor:
    buf = _get_fuseep_buffer(layer)
    hidden_states, _ = buf.fused_deep_moe(
        hidden_states,
        topk_idx=topk_output.topk_ids,
        topk_weights=topk_output.topk_weights,
        gmm1_permuted_weight=layer.w13_weight,
        gmm1_permuted_weight_scale=layer.w13_weight_scale,
        gmm2_weight=layer.w2_weight,
        gmm2_weight_scale=layer.w2_weight_scale,
        num_max_dispatch_tokens_per_rank=(
            envs.SGLANG_DEEPEP_NUM_MAX_DISPATCH_TOKENS_PER_RANK.get()
        ),
        num_experts=layer.num_experts,
        fuse_mode=get_exec().moe.fuseep_mode,
    )
    return hidden_states
```

评论区精华

PR 无 review 讨论。sglang-npu-bot 在评论中确认“Only modify the NPU-related parts. After passing the NPU test cases, merge it.”，表明变更范围限定在 NPU 模块，且需通过 NPU 测试。

- 暂无高价值评论线程

风险与影响

- 风险：风险极低。变更仅替换分布式 group 引用，若 MoE EP group 不存在或初始化顺序不正确，可能导致运行时错误。但 NPU 后端已确保 MoE EP group 在 FuseEP 前向前已创建。
- 影响：影响范围仅限于 NPU 后端且使用 ascend_fuseep MoE 后端的场景。用户可通过 DEEPEP_HCCL_BUFFSIZE 环境变量独立调优 MoE 通信 buffer，提升调度灵活性和潜在性能。
- 风险标记：缺少测试覆盖

关联脉络

- PR #31863 [NPU]remove duplicate code: 同一文件（NPU 后端）的另一重构变更，涉及 MoE 相关逻辑。
- PR #32113 [Bugfix] [NPU] Fix w4a8 MoE performance degradation: 同一文件（NPU 后端）的 MoE 性能修复，展示了 NPU MoE 模块的持续演进。