PR #45917 完整报告

vllm-project/vllm

[Bugfix] Pass TP group to FlashInfer all-reduce fusion

合并时间：2026-06-17 23:24

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/45917

执行摘要

- 一句话：修复 DP+TP 时 FlashInfer all-reduce 融合死锁
- 推荐动作：建议尽快合入并标记为 bugfix。变更虽小但涉及并行启动的关键路径，且已有清晰的根因分析。后续可考虑添加持续集成测试覆盖 TP+DP 或 TP+PP 组合的启动验证。

功能与动机

使用 --tensor-parallel-size > 1 和 --data-parallel-size > 1 运行 Nemotron-3-Ultra-NVFP4 时，服务器在 CUDA-graph 捕获 / 预热阶段卡死，EngineCore 反复日志 No available shared memory broadcast block found in 60 seconds。该问题源自 vLLM 调用 create_allreduce_fusion_workspace 时未传入 group 参数，FlashInfer 默认使用 torch.distributed.group.WORLD，而 DP > 1 时 WORLD 包含多个独立的 TP 子组，导致各 TP 组在 rendezvous 时相互等待，形成死锁。

实现拆解

1. 恢复 PP 场景下的 all-reduce 融合：在 vllm/config/vllm.py 的 enable_allreduce_rms_fusion 函数中，移除了 cfg.parallel_config.pipeline_parallel_size == 1 这一限制条件，并精简了 docstring。这意味着当 PP > 1 时，融合也将被启用。
2. 传入 TP 进程组：在 vllm/distributed/device_communicators/flashinfer_all_reduce.py 的 _create_workspace 函数中，将 group 参数传递给 flashinfer_comm.create_allreduce_fusion_workspace。该参数在调用链中已被传入（值为 TP 进程组），确保 FlashInfer 的 symm-mem rendezvous 仅在 TP 子组内进行，避免与 DP 组冲突。
3. 回滚 PP 禁用补丁：第三个 commit 显式 revert 了之前针对 PP 的禁用提交（#43616），与第一步配合，使 PP > 1 时也能安全使用融合。

关键文件：

- vllm/config/vllm.py（模块 配置层；类别 source；类型 core-logic；符号 enable_allreduce_rms_fusion）：移除了 all-reduce RMS 融合的 PP 限制条件（pipeline_parallel_size == 1），并简化 docstring，从而允许 PP > 1 时也启用融合。
- vllm/distributed/device_communicators/flashinfer_all_reduce.py（模块 通信层；类别 source；类型 core-logic；符号 _create_workspace）：调用 create_allreduce_fusion_workspace 时显式传入 group=group，将 FlashInfer 的

symm-mem rendezvous 限定在 TP 子组内，修复死锁根因。

关键符号：*enable_allreduce_rms_fusion*, *_create_workspace*

关键源码片段

vllm/distributed/device_communicators/flashinfer_all_reduce.py

调用 *create_allreduce_fusion_workspace* 时显式传入 *group=group*，将 FlashInfer 的 symm-mem rendezvous 限定在 TP 子组内，修复死锁根因。

```
def _create_workspace(
    backend: str,
    world_size: int,
    rank: int,
    max_token_num: int,
    hidden_dim: int,
    dtype: torch.dtype,
    group: ProcessGroup,
):
    """Create a flashinfer allreduce workspace, returning None on failure."""
    comm_backend = TorchDistBackend(group=group)
    rng_state = random.getstate()
    try:
        random.seed(int.from_bytes(os.urandom(16), byteorder="big"))
        workspace = flashinfer_comm.create_allreduce_fusion_workspace(
            backend=backend,
            world_size=world_size,
            rank=rank,
            max_token_num=max_token_num,
            hidden_dim=hidden_dim,
            dtype=dtype,
            comm_backend=comm_backend,
            group=group, # 新增：显式传入 TP 进程组，避免 FlashInfer 默认使用 WORLD 导致死锁
        )
    except Exception as e:
        # 异常处理逻辑不变
        ...
```

评论区精华

Reviewer tomeras91 询问该修复是否也能解决 PP 问题，建议如果可行则移除 PP 的自动禁用。作者确认后，tomeras91 批准了 PR。此外，tomeras91 建议删除 *_create_workspace* 中关于传递 group 的冗余注释，作者随即移除。

- 修复是否也解决 PP 问题 (correctness): 作者确认修复同样适用于 PP，并移除了 PP 禁用条件。
- 代码注释冗余 (style): 作者同意并删除了该注释。

风险与影响

- 风险：
  1. 回归风险：移除 PP 禁用条件后，PP > 1 时融合被重新启用，但本 PR 的根因修复（传入正确 group）同样适用于 PP 场景（PP 也是子组），理论上不会引入死锁。但 PP 场景的详细测试可能不充分（PR 仅报告了 GB200 上的 TP2/DP2 测试）。
  2. FlashInfer 版本依赖：修复依赖 FlashInfer 0.6.10.post1+（支持 group 参数），若用户使用旧版本则会报错。但旧版本不会产生死锁（因为旧版本不使用 symm-mem rendezvous）。
  3. 缺少测试覆盖：本次变更未添加回归测试，后续若 FlashInfer 接口变动可能不易察觉。 - 影响：影响范围：修复了 DP+TP（或 PP+TP）组合下 all-reduce + RMSNorm 融合的死锁问题，使这些配置能够正常启动。性能影响：融合本身可减少 GPU 内核启动开销，恢复融合后性能有望提升。用户影响：所有使用 TP > 1 且 DP > 1 或 PP > 1 的用户均受益；之前因死锁而依赖 --fuse-allreduce-rms false 变通方案的用户可移除该选项。 - 风险标记：启动死锁修复，恢复 PP 融合调用，缺少回归测试

关联脉络

- PR #43616 [Bugfix] Disable allreduce_rms_fusion when pipeline_parallel_size > 1: 该 PR 通过禁用 PP 场景下的融合来规避死锁，本 PR 回滚了该禁用并修复了根因。
- PR #41458 Re-enable allreduce rms fusion for DP / PP: 该 PR 移除了 DP 的融合禁用，使得 DP+TP 场景暴露了 FlashInfer 的 group 缺失问题，本 PR 修复了该问题。