PR #19918 完整报告

sgl-project/sglang

correct allreduce fusion and dummy_run alignment in SCATTERED MLP mode (moe_dense_tp_size=1)

合并时间：2026-05-23 10:18

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/19918

执行摘要

- 一句话：修复 SCATTERED MLP 模式的 allreduce 融合与 dummy_run 对齐
- 推荐动作：值得合并。设计上，明确 SCATTERED 模式不应参与 allreduce 融合是合理的；对齐逻辑与调度器对齐避免了 warmup 崩溃。建议关注后续是否有更广泛的融合条件重构，以及测试是否需要在更多模型上验证。

功能与动机

当使用 DeepSeek-R1 等模型且设置 --moe-dense-tp-size 1 时，MLP 以 SCATTERED 模式运行。此前 allreduce 融合逻辑未排除该模式，导致调度器合成不存在的 all-reduce 操作，所有推理请求得到乱码输出；另外 _dummy_run() 绕过了调度器的对齐逻辑，当 --max-running-requests 不能被 tensor parallelism size 整除时服务器会在预热阶段崩溃。详见 Issue #18237 及 PR body 中的复现步骤。

实现拆解

1. communicator.py：在 should_fuse_mlp_allreduce_with_next_layer() 中插入 self.layer_scatter_modes.mlp_mode == ScatterMode.SCATTERED 条件，当 MLP 运行于 SCATTERED 模式时直接返回 False，避免虚假融合。
2. model_runner.py：将 require_gathered_buffer 条件替换为 require_mlp_sync（更精确反映调度器逻辑），并对 num_tokens 使用 ceil_align(num_tokens, attn_tp_size) 进行对齐，确保 warmup 时的 token 数能被 attention TP size 整除。
3. test_hybrid_dp_ep_tp_mtp.py：在 Test03 启动参数中添加 --enable-flashinfer-allreduce-fusion，使 MMLU 评估能在融合开启条件下覆盖 SCATTERED 路径，捕获回归。

关键文件：

- python/sglang/srt/layers/communicator.py（模块 通信层；类别 source；类型 core-logic；符号 should_fuse_mlp_allreduce_with_next_layer）：核心逻辑改动：新增 SCATTERED 模式短路检查，阻止虚假 allreduce 融合。
- python/sglang/srt/model_executor/model_runner.py（模块 模型执行器；类别 source；类型 core-logic；符号 _dummy_run, require_mlp_sync）：主入口修复：修改 dummy_run 对齐逻辑以匹配调度器行为，避免预热崩溃。

- test/registered/moe/test_hybrid_dp_ep_tp_mtp.py（模块 集成测试；类别 test；类型 test-coverage；符号 Test03）：测试覆盖：在 Test03 中添加融合标志以捕获回归。

关键符号：should_fuse_mlp_allreduce_with_next_layer, _dummy_run

关键源码片段

python/sglang/srt/layers/communicator.py

核心逻辑改动：新增 SCATTERED 模式短路检查，阻止虚假 allreduce 融合。

```
def should_fuse_mlp_allreduce_with_next_layer(self, forward_batch):
    # 当 MoE CP allgather 启用时，必须禁用融合，因为融合路径会跳过
    # postprocess_layer，导致形状不匹配
    if is_enable_moe_cp_allgather():
        return False

    # DP attention + Eagle 特殊处理
    if is_dp_attention_enabled() and self._speculative_algo is not None and self._speculative_algo.
    is_eagle():
        return False

    # 注意力输入已分散时跳过融合
    if get_attn_tp_context().input_scattered:
        return False

    # 关键修复：当 MLP 模式为 SCATTERED 时，MLP 运行在分散数据上，
    # 不存在 TP all-reduce，因此没有操作可以融合到下一层
    if self.layer_scatter_modes.mlp_mode == ScatterMode.SCATTERED:
        return False

    batch_size = forward_batch.input_ids.shape[0] if hasattr(forward_batch, "input_ids") else 0

    return (
        apply_flashinfer_allreduce_fusion(batch_size)
        or (
            _use_aiter
            and batch_size > 0
            and get_tensor_model_parallel_world_size() != 6
            and get_global_server_args().enable_aiter_allreduce_fusion
        )
    ) and (not self.is_last_layer) and (self._context.tp_size > 1)
```

python/sglang/srt/model_executor/model_runner.py

主入口修复：修改 dummy_run 对齐逻辑以匹配调度器行为，避免预热崩溃。

```
# 新增导入：ceil_align 和 require_mlp_sync
from sglang.srt.utils.common import ceil_align, require_mlp_sync

# ... 类方法中 ...
```

```
def _dummy_run(self, batch_size: int, run_ctx=None):
    # ... 前面的逻辑 ...
    num_tokens = batch_size * num_tokens_per_bs

    # 与调度器对齐：在需要 MLP 同步时对 num_tokens 进行 ceil-align
    # 对应 scheduler 中的 prepare_mlp_sync_batch 逻辑
    if require_mlp_sync(self.server_args):
        attn_tp_size = get_attention_tp_size()
        if attn_tp_size > 1 and num_tokens % attn_tp_size != 0:
            num_tokens = ceil_align(num_tokens, attn_tp_size)
            batch_size = num_tokens // num_tokens_per_bs
    # ... 后续创建 buffers ...
```

评论区精华

1. DP attention 与 moe-dense-tp-size 的交互：@Fridge003 提出是否只有启用 DP attention 时才应使用该模式；@weireweire 引用 lmsys 博客说明即使没有 DP，SCATTERED 模式也能减少通信量（reduce-scatter + all-gather 取代两次 all-reduce）。
2. 测试覆盖缺口：@nvpohanh 指出每周测试未捕获该 bug 是因为缺少 --enable-flashinfer-allreduce-fusion；@weireweire 确认后在 Test03 中补充该参数。
3. _dummy_run 条件的选择：@ch-wan 认为改用 require_mlp_sync 不如直接使用 require_gathered_buffer 清晰；@weireweire 解释目的是与调度器 prepare_mlp_sync_batch 中的对齐条件一致，最终被接受并简化了注释。

- DP attention 与 moe-dense-tp-size 的交互 (design): 确认 SCATTERED 模式独立有效，修复不影响该设计。
- 测试覆盖缺失 (testing): 已添加参数，MMLU 测试可覆盖融合路径。
- _dummy_run 条件选择 (correctness): weireweire 精简注释后，ch-wan 批准。

风险与影响

- 风险：影响范围局限于 SCATTERED MLP 模式（--moe-dense-tp-size 1）。核心风险在于 allreduce 融合逻辑的变更可能影响其他融合条件（如 is_enable_moe_cp_allgather、dp_attention 等），但新加的检查仅在 SCATTERED 模式下短路，不影响原有判断。_dummy_run 对齐调整使用 ceil_align，与调度器行为一致，回归风险低。测试层面对融合场景的覆盖由 MMLU 评测验证，但 64 样本的 MMLU 可能不足以捕捉所有精度退化，建议增加更多端到端测试。
- 影响：对用户：使用 DeepSeek 类模型并设置 --moe-dense-tp-size 1 且开启 --enable-flashinfer-allreduce-fusion 的用户将消除随机乱码输出和预热崩溃，推理结果正确。对系统：无性能退化，修复后通信模式与设计意图一致。对团队：修复了两个隐蔽 bug，并通过测试增强防止回归。
- 风险标记：核心路径变更，配置耦合风险

关联脉络

- PR #18237 Garbage output with moe-dense-tp-size=1 and flashinfer allreduce fusion: 该 PR 修复了 #18237 中报告的垃圾输出问题。