PR #30557 完整报告

sgl-project/sglang

[AMD] Fix AITER custom all-gather CUDA-graph capture crash under torch_memory_saver

合并时间：2026-07-09 16:30

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/30557

执行摘要

- 一句话：修复 ROCm AITER all-gather 在 torch_memory_saver 下的崩溃
- 推荐动作：值得合入。这是一个针对特定平台环境组合的小而关键的修复，遵循了与已合并的 all-reduce 修复相同的模式，逻辑清晰且风险极低。

功能与动机

修复 ROCm 上 AITER 自定义 all-gather 在 torch_memory_saver 启用时的进程崩溃问题。PR body 指出：当 CUDA graph 内存池由 torch_memory_saver 管理时，其缓冲区为 HIP VMM 分配，通过 hipIpcGetMemHandle 注册会在捕获结束时失败并中止进程。

实现拆解

1. 修改文件：python/sglang/srt/distributed/parallel_state.py 中的 _all_gather_into_tensor 方法。
2. 核心逻辑：在 CUDA graph 捕获（_IS_CAPTURING 为 True 且流正在捕获）分支中，增加对 envs.SGLANG_MEMORY_SAVER_CUDA_GRAPH 的判断：
   - 若启用，调用 ca_comm.all_gather_unreg（不注册缓冲区）
   - 否则，保持原有的 ca_comm.all_gather_reg
3. 备注更新：同步更新了方法注释，明确说明 all_gather_reg 用于正常捕获，all_gather_unreg 用于 torch_memory_saver 及其他路径。
4. 行为不变性：当 torch_memory_saver 未启用时，代码逻辑与之前完全相同，无回归风险。

关键文件：

- python/sglang/srt/distributed/parallel_state.py（模块 通信层；类别 source；类型 core-logic；符号 _all_gather_into_tensor）：本次变更唯一文件，修改了 _all_gather_into_tensor 方法中的 CUDA graph 捕获逻辑，新增对 SGLANG_MEMORY_SAVER_CUDA_GRAPH 的判断以避开 all_gather_reg 的崩溃。

关键符号：_all_gather_into_tensor

关键源码片段

python/sglang/srt/distributed/parallel_state.py

本次变更唯一文件，修改了 _all_gather_into_tensor 方法中的 CUDA graph 捕获逻辑，新增对 SGLANG_MEMORY_SAVER_CUDA_GRAPH 的判断以避开 all_gather_reg 的崩溃。

```
# python/sglang/srt/distributed/parallel_state.py
# _all_gather_into_tensor 方法中的 CUDA graph 捕获分支
# 当流正在捕获 CUDA graph 时，根据 torch_memory_saver 状态选择注册方式
if getattr(ca_comm, "_IS_CAPTURING", False):
    if torch.cuda.is_current_stream_capturing():
        # torch_memory_saver 管理的内存池是 HIP VMM 分配，
        # all_gather_reg 会尝试通过 hipIpcGetMemHandle 注册并崩溃，
        # 因此使用 all_gather_unreg 避免注册。
        if envs.SGLANG_MEMORY_SAVER_CUDA_GRAPH.get():
            ca_comm.all_gather_unreg(input, out=output, dim=0)
        else:
            ca_comm.all_gather_reg(input, out=output, dim=0)
    elif is_in_tc_piecewise_cuda_graph():
        ca_comm.all_gather_unreg(input, out=output, dim=0)
    else:
        # True CUDA graph warmup: avoid a different host collective.
        output.zero_()
    return
```

评论区精华

无 review 讨论。HaiShaw 直接批准了 PR。

- 暂无高价值评论线程

风险与影响

- 风险：风险很低。改动仅 4 行核心逻辑，且条件守卫于 SGLANG_MEMORY_SAVER_CUDA_GRAPH 环境变量，默认不启用时行为不变。潜在风险包括：若 envs.SGLANG_MEMORY_SAVER_CUDA_GRAPH 未正确导入或在某些非 ROCm 平台意外设置，但此类环境变量通常有平台检测，风险可控。
- 影响：影响范围仅限于 ROCm 平台且使用 AITER 自定义 all-gather 的场景，尤其是启用 torch_memory_saver 的用户。修复了进程崩溃的严重 bug，提升了 ROCm 上的稳定性。对其他平台（CUDA、Intel 等）无影响。
- 风险标记：平台特定（ROCm），缺少测试覆盖（未添加对应测试）

关联脉络

- PR #19162 Fix custom all-reduce under torch_memory_saver: 同一模式的修复，为自定义 all-reduce 添加了类似的 torch_memory_saver 判断。
- PR #20155 Another fix for custom all-reduce under torch_memory_saver: 针对 torch_memory_saver 下自定义 all-reduce 的另一修复，延续同一模式。