

PR #34203 完整报告

sgl-project/sglang

[AMD] Fix AITER custom reduce-scatter CUDA-graph capture crash under torch_memory_saver

合并时间: 2026-08-11 06:36

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/34203>

执行摘要

- 一句话: TMS 下 AITER reduce-scatter 改用未注册路径, 修复 CUDA 图捕获崩溃
- 推荐动作: 值得快速浏览: 修复点虽小但切中一个真实崩溃场景, 且与已有 all-gather 修复模式一致。建议后续补充 TMS + ROCm 组合的单元测试或回归用例, 避免该路径再次漂移。

功能与动机

PR 描述指出: `SGLANG_MEMORY_SAVER_CUDA_GRAPH=1` 时 TMS 用 HIP VMM 分配 CUDA-graph 池, AITER 的 registered 自定义集合路径在捕获后尝试对调用方张量做 IPC 注册, 但 HIP IPC 不支持这些 VMM 分配, 导致 CUDA 图捕获以 `invalid device pointer` 中止。此前 all-reduce (#19162 / #20155) 与 all-gather (#30557) 已改走未注册路径, reduce-scatter 是剩余缺口: `_maybe_aiter_reduce_scatter()` 仍在捕获期间硬编码 `registered=True`。

实现拆解

1. 变更入口: `python/sglang/srt/distributed/parallel_state.py` 的 `_maybe_aiter_reduce_scatter()` 方法, 该方法在 ROCm 且满足相等块约束时调用 AITER 自定义 reduce-scatter, 否则返回 False 让调用方回退 RCCL。
2. 控制流调整: 在 `torch.cuda.is_current_stream_capturing()` 捕获分支内, 先检查 `envs.SGLANG_MEMORY_SAVER_CUDA_GRAPH.get()`: 为真时改用 `registered=False` (未注册路径, 避免对 VMM 分配做 IPC 注册); 为假时保持 `registered=True`, 性能行为不变。这与 `_all_gather_into_tensor` 在 TMS 下的既有处理方式对齐。
3. 保留其他分支: TC piecewise CUDA graph 分支仍使用 `registered=False`, 纯 CUDA graph 预热分支仍以 `output.zero_()` 占位, 避免在捕获期间引入不同的 host collective。
4. 测试与 CI 配套: 本次未新增测试文件, 仅 1 个源码文件的 +4/-1 改动; PR body 中 CI 状态显示失败, 但从现有材料无法判断失败是否与本改动相关。

关键文件:

- `python/sglang/srt/distributed/parallel_state.py` (模块 并行通信; 类别 source; 类型 core-logic; 符号 `_maybe_aiter_reduce_scatter`): 唯一变更文件; 在 CUDA 图捕获分支内为 TMS 场景切换到未注册 AITER reduce-scatter 路径, 修复 HIP VMM 分配下 IPC 注册导致的崩溃。

关键符号: `_maybe_aiter_reduce_scatter`

关键源码片段

python/sglang/srt/distributed/parallel_state.py

唯一变更文件；在 CUDA 图捕获分支内为 TMS 场景切换到未注册 AITER reduce-scatter 路径，修复 HIP VMM 分配下 IPC 注册导致的崩溃。

```
def _maybe_aiter_reduce_scatter(
    self, output: torch.Tensor, input: torch.Tensor
) -> bool:
    # AITER 自定义 reduce-scatter (ROCm 专用)。镜像 `_all_gather_into_tensor` 的
    # 自定义 all-gather 路径：等分块（无变长）reduce-scatter，使用已注册的对称内存
    # 缓冲，对小型、延迟敏感的 decode collective 比通用 RCCL 内核更快。
    # 由 `SGLANG_DP_USE_REDUCE_SCATTER` 控制；非 ROCm、形状 / 大小 / 拓扑不支持时
    # 返回 False，由调用方回退到 RCCL。
    if not (
        is_hip()
        and envs.SGLANG_DP_USE_REDUCE_SCATTER.get()
        and self._has_aiter_custom_reduce_scatter()
        and input.is_contiguous()
        and output.is_contiguous()
        and input.dtype in (torch.float32, torch.float16, torch.bfloat16)
    ):
        return False

    ca_comm = self.ca_comm
    # `input` 是归约前的完整缓冲，`should_custom_ar` 限制其大小。
    if not ca_comm.should_custom_ar(input):
        return False

    # 仅支持等分块：`input` 的行数必须能平均分成 `world_size` 块，
    # 且与每 rank 的 `output` 行数匹配。
    if input.shape[0] != output.shape[0] * self.world_size:
        return False

    if getattr(ca_comm, "_IS_CAPTURING", False):
        if torch.cuda.is_current_stream_capturing():
            if envs.SGLANG_MEMORY_SAVER_CUDA_GRAPH.get():
                # 关键修复：TMS 下 CUDA 图池由 HIP VMM 分配，AITER 的 registered
                # 路径尝试 IPC 注册张量会被拒绝（invalid device pointer），因此走
                # 未注册路径，避免捕获中止。
                ca_comm.reduce_scatter(input, output, registered=False)
            else:
                # 非 TMS 场景维持既有 registered=True 路径，保持性能特性。
                ca_comm.reduce_scatter(input, output, registered=True)
        elif is_in_tc_pieewise_cuda_graph():
            # TC pieewise 图场景本就使用未注册路径，保持不变。
            ca_comm.reduce_scatter(input, output, registered=False)
        else:
            # 真正的 CUDA graph 预热：避免在捕获分支内引入不同的 host collective。
```

```
    output.zero_()
    return True
```

```
# 非捕获场景走未注册路径，与原有行为一致。
ca_comm.reduce_scatter(input, output, registered=False)
return True
```

评论区精华

该 PR 没有产生评论线程。唯一审阅反馈来自合并者 HaiShaw 的批准评论：> AITER path only。这表明变更被有意限定在 AITER 自定义集合路径，不触碰 RCCL 等其他后端，降低了风险面。

- AITER-only 范围确认 (other): 变更被批准，且限定在 AITER 路径，不涉及 RCCL 等其他后端。

风险与影响

- 风险：
 - 缺少测试覆盖：没有对应单元测试覆盖 SGLANG_MEMORY_SAVER_CUDA_GRAPH=1 + CUDA 图 + AITER reduce-scatter 组合，回归风险主要依赖 CI 和真机验证。
 - AMD 专用路径：改动只在 is_hip() 分支内生效，不影响 CUDA 平台；但 TMS 用户会失去 registered 缓冲的潜在性能优势，不过原先该场景根本无法工作。
 - 环境变量交互：SGLANG_MEMORY_SAVER_CUDA_GRAPH 只在 is_current_stream_capturing() 分支内生效，is_in_tc_pieewise_cuda_graph() 等其余分支逻辑不变，交互风险有限。
 - 影响面控制：默认服务行为完全不变，仅在 TMS + AITER reduce-scatter 的 ROCm 组合下改变行为。
 - 影响：影响用户：在 AMD (ROCm) 平台上启用 SGLANG_MEMORY_SAVER_CUDA_GRAPH=1 且走 AITER 自定义 reduce-scatter 的用户，CUDA 图捕获不再因 invalid device pointer 崩溃。影响系统：默认路径不变，非 TMS 与 RCCL 回退路径不受影响。影响团队：补全了 TMS 下 AITER all-reduce / all-gather / reduce-scatter 三类自定义集合的处理一致性，是 AMD 后端兼容性维护的连续工作。影响程度低，单文件小改动。
- 风险标记：缺少测试覆盖，AMD 专用路径，环境变量门控变更

关联脉络

- 暂无明显关联 PR