PR #27258 完整报告

sgl-project/sglang

[HiSparse PD & PP]Fix HiSparse compatibility with PP decode

合并时间：2026-06-04 21:37

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/27258

执行摘要

- 一句话：修复 HiSparse 在 PP decode 路径上的兼容性
- 推荐动作：建议合并。该 PR 以极小代价修复了一个关键的兼容性缺陷，改动经过严格验证且逻辑清晰。可考虑后续跟进 process_retract_queue 中的同类问题。

功能与动机

修复 HiSparse 在 PP（Pipeline Parallelism）decode 路径下的兼容性 bug。在非 PP 路径中，KV 传输完成后请求会通过 hisparse_coordinator.admit_request_direct(req) 初始化 HiSparse 设备缓冲区元数据，但 PP decode 路径（process_decode_transfer_queue）遗漏了该调用，导致启用了 HiSparse + pp_size>1 时 GPU 缓存用量无法增长，引发错误输出。修复后 GSM8K 准确率从 0.715 提升至 0.965。

实现拆解

1. 定位问题：在 python/sglang/srt/managers/scheduler_pp_mixin.py 的 process_decode_transfer_queue 方法中，当启用了 HiSparse（self.enable_hisparse）时，未对从传输队列中取出的请求调用 hisparse_coordinator.admit_request_direct(req) 进行元数据初始化。
2. 修复方案：在 released_reqs 被 pop_transferred 取出后、加入 waiting_queue 之前，增加循环判断（if self.enable_hisparse），对每个请求执行 self.hisparse_coordinator.admit_request_direct(req)，从而与已有的非 PP 路径行为保持一致。
3. 仅涉及一个文件，改动量 3 行，无配置或部署配套变更。

关键文件：

- python/sglang/srt/managers/scheduler_pp_mixin.py（模块 调度器；类别 source；类型 core-logic；符号 process_decode_transfer_queue）：核心修复文件，在 process_decode_transfer_queue 中增加 HiSparse admit_request_direct 调用。

关键符号：process_decode_transfer_queue

关键源码片段

python/sglang/srt/managers/scheduler_pp_mixin.py

核心修复文件，在 process_decode_transfer_queue 中增加 HiSparse admit_request_direct 调用。

```
# python/sglang/srt/managers/scheduler_pp_mixin.py: process_decode_transfer_queue
# 修复后：从传输队列中取出已释放的请求后，先初始化 HiSparse 设备缓冲区元数据，再放入 waiting_queue

def process_decode_transfer_queue(
    self: Scheduler, release_rids: Optional[List[str]]
):
    if release_rids is not None:
        released_reqs = self.disagg_decode_transfer_queue.pop_transferred(
            release_rids
        )
        # 新增：若启用了 HiSparse，则对每个请求调用 admit_request_direct
        # 确保 decode 时 GPU 缓存用量能正确增长，避免因元数据缺失导致的错误输出
        if self.enable_hisparse:
            for req in released_reqs:
                self.hisparse_coordinator.admit_request_direct(req)
        self.waiting_queue.extend(released_reqs)
        return [req.rid for req in released_reqs]
    return None
```

评论区精华

Review 评论（来自 gemini-code-assist[bot]）：指出了另一个潜在相似问题：在 process_retract_queue（约 1357-1365 行）中，当请求被 retract 后其 HiSparse 资源被完全释放（request_finished），但 resume 时也只是直接加入 waiting_queue 而未调用 admit_request_direct，建议同样修复。本 PR 未对此做处理，部分 reviewers 已批准，该问题可作为后续改进。

- process_retract_queue 中可能存在同类遗漏 (correctness): 本 PR 未处理，reviewer 仅提出建议，合并者未要求必须修复。

风险与影响

- 风险：该修复仅增加了条件分支（if self.enable_hisparse），在 HiSparse 未启用时无额外开销。风险极低，但 process_retract_queue 中可能存在的同类遗漏尚未修复，可能在某些场景下仍有隐患。
- 影响：影响范围限于启用了 HiSparse 且 PP decode 的配置（enable_hisparse + direct-to-host + decode pp_size>1）。修复前该场景下 GSM8K 准确率仅 0.715，修复后恢复至 0.965，属于功能正确性的关键修复。对非 HiSparse 用户无影响。
- 风险标记：核心路径变更，无测试覆盖，同类问题可能遗漏

关联脉络

- PR #27046 [HiCache] fix PD L3 cache hit details from decode responses: 同为 HiCache/HiSparse 相关的 bugfix，涉及 decode 路径与调度器交互。

- PR #25000 Reduce mamba prefill allocation overhead: 涉及调度器 memory pool 优化，与 PP decode 路径有一定关联。