PR #26646 完整报告

sgl-project/sglang

[core] Make overlap-schedule WAR barrier CUDA-only

合并时间：2026-05-29 16:02

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/26646

执行摘要

- 一句话：将 WAR 屏障设为仅 CUDA 启用，修复 AMD 回归
- 推荐动作：值得精读，尤其是平台差异处理的决策过程。

功能与动机

PR #26380 添加的 WAR 屏障在 AMD/HIP 平台上引发了性能回归：MTP 测试超时、优先级调度断言失败、DSv4 非 MTP 吞吐量下降约 20%。该屏障在 CUDA 上免费，因此需要平台条件化以避免影响 AMD。

实现拆解

1. 在 scheduler.py 的 run_event_loop 中导入 is_cuda 并设置 self._war_barrier_enabled = is_cuda()。
2. 在 event_loop_overlap、event_loop_overlap_disagg_decode、event_loop_overlap_disagg_prefill 中，只在启用屏障时执行 schedule_stream.wait_stream(forward_stream)。
3. 在 overlap_utils.py 中，将 D2H 流和固定内存缓冲区的创建条件从 _is_cuda or _is_hip 改为 _is_cuda，因为该补偿与 CUDA 屏障的占用损失对应，其他平台不需要。
4. 无新增测试，但通过 AMD CI 验证。

关键文件：

- python/sglang/srt/managers/scheduler.py（模块 调度器；类别 source；类型 core-logic；符号 _war_barrier_enabled）：核心调度文件，添加了 _war_barrier_enabled 标志并条件化屏障调用。
- python/sglang/srt/managers/overlap_utils.py（模块 重叠工具；类别 source；类型 core-logic；符号 new_seq_lens_cpu_pinned, fwd_prepare_d2h_stream）：重叠缓冲区工具，将 D2H 流和固定内存缓冲区创建条件收紧为仅 CUDA。
- python/sglang/srt/disaggregation/decode.py（模块 分离解码；类别 source；类型 core-logic）：分离解码循环中的屏障条件化。
- python/sglang/srt/disaggregation/prefill.py（模块 分离预填；类别 source；类型 core-logic）：分离预填循环中的屏障条件化。

关键符号：Scheduler.run_event_loop, Scheduler.event_loop_overlap, Scheduler.event_loop_overlap_disagg_decode, Scheduler.event_loop_overlap_disagg_prefill, RelayBuffer.init

关键源码片段

python/sglang/srt/managers/scheduler.py

核心调度文件，添加了 _war_barrier_enabled 标志并条件化屏障调用。

```
def run_event_loop(self) -> None:
    """运行调度器事件循环，设置 schedule_stream 并分发到对应事件循环"""
    if use_mlx():
        dispatch_event_loop(self)
        return

    self.schedule_stream = self.device_module.Stream(priority=0)
    if self.device == "cpu":
        self.schedule_stream.synchronize = lambda: None # CPU 上无操作

    # WAR 屏障仅针对 CUDA 启用；其他平台（HIP/NPU/CPU）保持无屏障行为
    self._war_barrier_enabled = is_cuda()

    with self.device_module.StreamContext(self.schedule_stream):
        dispatch_event_loop(self)
```

python/sglang/srt/managers/overlap_utils.py

重叠缓冲区工具，将 D2H 流和固定内存缓冲区创建条件收紧为仅 CUDA。

```
# 用于 forward 准备的 D2H 流和 CPU 固定内存缓冲区，仅 CUDA 需要：
# 该缓冲区用于补偿 CUDA 上 WAR 屏障带来的占用损失；
# 其他平台没有屏障，直接使用 .cpu() 回退路径。
if _is_cuda:
    self.new_seq_lens_cpu_pinned = torch.empty(
        (self.req_pool_size,), dtype=torch.int64, pin_memory=True
    )
    self.fwd_prepare_d2h_stream = torch.get_device_module(self.device).Stream()
else:
    self.new_seq_lens_cpu_pinned = None
    self.fwd_prepare_d2h_stream = None
```

评论区精华

reviewer yctseng0211 已批准并确认 deepseekv4 在 AMD 上的性能验证通过。

- AMD 性能验证 (other): 屏障修改已通过 AMD CI 验证，无性能回归。

风险与影响

- 风险：风险较低。CUDA 平台保留屏障，数据竞争保护不变；非 CUDA 平台恢复为之前无屏障行为，但该屏障仅修复一个特定竞争，之前未报告问题。主要文件均为核心调度路径，修改仅是添加条件判断，引入新 bug 可能性低。
- 影响：影响范围：仅影响非 CUDA 平台（AMD、NPU、CPU），CUDA 无变化。用户：AMD 用户将看到性能回归被修复。系统：调度器重叠运行时的并发安全性在非 CUDA 平台上恢复至之前状态。团队：已完成 AMD 环境验证。影响程度：中等。
- 风险标记：核心路径变更，平台依赖，缺少新增测试覆盖

关联脉络

- PR #26380 [core] WAR barrier for overlap schedule buffer writes, without fwd occupancy cost: 引入 WAR 屏障导致 AMD 回归，本 PR 将其条件化为 CUDA-only。