PR #24359 完整报告

sgl-project/sglang

Minor scheduler fixes

合并时间：2026-05-05 02:01

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/24359

执行摘要

- 一句话：优化混合 SWA 预 fill 及容量检查
- 推荐动作：值得合并的微小修复，逻辑清晰。建议在合并前添加针对混合 SWA 及非 PP 场景下 get_num_allocatable_reqs 行为的单元测试，尤其是边界 case（token pool 接近耗尽）。

功能与动机

PR body 明确指出两个动机：

1) get_num_allocatable_reqs 未考虑某些情况下 slot 被保留，需要始终用 req_to_token_pool.available_size() 做上限； 2) 混合 SWA 在 decode 阶段滑动窗口淘汰会释放页面，但 batch_is_full 只在请求结束时才清除，导致即使有空余也无法加入新请求，造成预 Fill 停滞。

实现拆解

在 python/sglang/srt/managers/scheduler.py 中做了两处微小调整：

1. get_num_allocatable_reqs 方法（第 2566-2569 行）：移除 if self.pp_size > 1: 条件判断，改为无条件将 req_to_token_pool.available_size() 与 pp_max_micro_batch_size - running_bs 取最小值。这一改动确保无论是否启用流水线并行，可分配的请求数都不会超过 token pool 的可用容量，避免因遗漏检查导致调度器认为有空位却实际无法分配。
2. _get_new_batch_prefill_raw 方法（第 2601-2603 行）：在 if self.enable_priority_preemption: 条件中增加 or self.is_hybrid_swa，使得混合 SWA 模型也能在每个调度周期开始时将 self.running_batch.batch_is_full 重置为 False。这是因为混合 SWA 的 sliding window eviction 可能在 decode 过程中释放页面，但原来的 batch_is_full 仅在请求结束时才清除，导致预 Fill 一直无法加入新请求。注意 batch_is_full 的完整条件是 self.get_num_allocatable_reqs <= 0 且 chunked_req is None 且非 priority preemption 模式（第 2617-2624 行），修改后混合 SWA 不会因该标志而提前拒绝。

关键文件：

- python/sglang/srt/managers/scheduler.py（模块 调度器；类别 source；类型 core-logic；符号 get_num_allocatable_reqs, _get_new_batch_prefill_raw）：调度器核心文件，包含 get_num_allocatable_reqs 和 _get_new_batch_prefill_raw 两个关键方法的修改，直接

影响请求准入和预 Fill 行为。

关键符号：get_num_allocatable_reqs，_get_new_batch_prefill_raw

关键源码片段

python/sglang/srt/managers/scheduler.py

调度器核心文件，包含 get_num_allocatable_reqs 和 _get_new_batch_prefill_raw 两个关键方法的修改，直接影响请求准入和预 Fill 行为。

```
# python/sglang/srt/managers/scheduler.py

def get_num_allocatable_reqs(self, running_bs):
    res = get_global_server_args().pp_max_micro_batch_size - running_bs
    # 始终以 token pool 可用容量为上限，避免因保留 slot 而错误地拒绝请求
    res = min(res, self.req_to_token_pool.available_size())
    return res

def _get_new_batch_prefill_raw(self, prefill_delayer_single_pass):
    # ... 前置检查代码 ...

    # 对于混合 SWA 模型，滑动窗口可能在 decode 中途释放页面，
    # 因此需要在每个调度周期重置 batch_is_full，允许新增请求
    if self.enable_priority_preemption or self.is_hybrid_swa:
        self.running_batch.batch_is_full = False

    if (
        self.running_batch.batch_is_full or len(self.waiting_queue) == 0
    ) and self.chunked_req is None:
        return None
    # ... 后续逻辑 ...
```

评论区精华

只有一次来自 gemini-code-assist[bot] 的自动 review，确认了变更内容且没有反馈意见。没有人工 review 或争议讨论。

- 暂无高价值评论线程

风险与影响

- 风险：风险较低。两处改动均为条件放宽或限制收紧：
  - get_num_allocatable_reqs 的变更始终引入 token pool 容量约束，在非 PP 场景下可能略微降低最大并发数，但符合资源安全预期。
  - batch_is_full 重置条件增加 is_hybrid_swa，只影响混合 SWA 模型；可能带来的副作用是 prefill 尝试更频繁，但每次依然受 get_num_allocatable_reqs 和 batch_is_full 的其他条件限制，不会导致无限制加入。 未引入测试，回归风险依赖已有 CI 覆盖。

- 影响：影响范围：仅 SGLang 调度器核心逻辑，面向所有使用混合 SWA 或非 PP 部署的用户。对混合 SWA 模型（如 DeepSeek V2/V3 等）的预 Fill 吞吐有正面影响，修复了之前可能出现的预 Fill 长期停滞问题；对非 PP 场景的 request admission 增加了 token pool 容量检查，提升了资源使用的安全性。影响程度较低——5 行代码变更，两者都是条件分支调整。
- 风险标记：缺少测试覆盖

关联脉络

- PR #23882 原 PR（本 PR 从中拆分）：本 PR 明确标注为从 #23882 拆分出的调度器修复，涉及同一文件 scheduler.py 的相关逻辑。
- PR #24334 extract adjust_hybrid_swa_layers_for_pp: 同为混合 SWA 相关的重构，与本 PR 的 is_hybrid_swa 条件修改处于同一功能线。