PR #33448 完整报告

sgl-project/sglang

[DCP] Bound a request by the aggregate KV pool, not one rank's share

合并时间：2026-08-04 10:36

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/33448

执行摘要

- 一句话：DCP 按聚合 KV 池限制请求长度
- 推荐动作：值得精读：这是理解 SGLang DCP 计费模型（per-rank 池 vs 聚合池）的一个极佳切入点，scheduler.py 中与 PrefillAdder 一致性相关的注释值得注意。建议合入后补充一个 dcp_size > 1 的准入与 max_new_tokens 预算单元测试，并顺手核对 PrefillAdder 的准入预算是否同样基于聚合池，防止未来出现新的口径分裂。

功能与动机

PR body 指出：DCP 下序列跨 rank 分片，按 len / dcp_size 对每 rank KV 池计费，但 tp_worker.get_worker_info 的 max_req_len 和 Scheduler 的 max_new_tokens clamp 都拿原始长度与单 rank 池比较。前者导致长请求被直接拒绝，后者导致被准入的请求获得零生成预算并返回 finish_reason="length" 的空内容；两处必须一起修，否则只改一方会把显式拒绝变成静默空响应。

实现拆解

1. 修复准入上限（tp_worker.py）：alloc_memory_pool 中的内存池校验与 get_worker_info 对外报告的 max_req_len，都从 effective_max_total_num_tokens 改为乘以 model_runner.dcp_size 后再减 1。原因是 DCP 下序列分片后每个 rank 只保存 1 / dcp_size 的 KV，但单条请求的原始长度上限应基于整个聚合池；该改动让长请求不再在准入阶段被 Input length exceeds 直接拒绝。
2. 修复生成预算（scheduler.py）：init_req_max_new_tokens 的 max_new_tokens 截断公式中，max_total_num_tokens * self.server_args.dcp_size 替换原来的 max_total_num_tokens，再减 paged_input_len、page_size 与 1。这样与 PrefillAdder 的准入预算保持一致，被准入的请求会拿到非零生成预算，避免 finish_reason="length" 的空响应；同时保留对 min_new_tokens 不变量的恢复逻辑。
3. 验证配套：本 PR 未新增单元测试，作者通过 /rerun-test test/registered/dcp/* 重跑 DCP 测试矩阵（H200 8 卡 dcp8 与 B200 4 卡 dcp4 用例）全部通过，并在本地用 tp8/dcp8 + DSPARK 复现 128k/256k 长请求正常输出。

关键文件：

- python/sglang/srt/managers/scheduler.py（模块 调度器；类别 source；类型 core-logic；符号 init_req_max_new_tokens）：修正 init_req_max_new_tokens 的

max_new_tokens 预算，使其基于聚合 KV 池，避免被准入请求拿到零生成预算并返回空响应

- python/sglang/srt/managers/tp_worker.py（模块 工作器；类别 source；类型 core-logic；符号 alloc_memory_pool, get_worker_info）：修正 alloc_memory_pool 与 get_worker_info 中 max_req_len 的推导，基于聚合 KV 池，避免长请求在准入阶段被直接拒绝

关键符号：init_req_max_new_tokens, alloc_memory_pool, get_worker_info

关键源码片段

python/sglang/srt/managers/tp_worker.py

修正 alloc_memory_pool 与 get_worker_info 中 max_req_len 的推导，基于聚合 KV 池，避免长请求在准入阶段被直接拒绝

```
# python/sglang/srt/managers/tp_worker.py
# 向调度器广播 worker 信息时，max_req_len 需基于聚合 KV 池推导（DCP 修复后版本）
def get_worker_info(self):
    # DCP 下序列跨 rank 分片，每个 rank 只保存 1 / dcp_size 的 KV；
    # 若拿单 rank 的 effective_max_total_num_tokens 当上限，
    # 长上下文请求会在准入阶段被直接拒绝。
    max_req_len = min(
        self.model_config.context_len - 1,
        self.model_runner.effective_max_total_num_tokens
        * self.model_runner.dcp_size
        - 1,
    )
    return (
        self.model_runner.max_total_num_tokens,
        get_schedule().max_prefill_tokens,
        self.model_runner.max_running_requests,
        get_schedule().max_queued_requests,
        max_req_len,
        max_req_len - 5, # max_req_input_len，预留少量余量
        self.random_seed,
        self.device,
        self.model_runner.forward_stream,
        self.model_runner.req_to_token_pool.size,
        self.model_runner.req_to_token_pool.max_context_len,
        self.model_runner.token_to_kv_pool.size,
    )
```

评论区精华

本 PR 没有任何实质性 review 评论，评审人 hnyls2002 直接批准。核心论证全部在 PR body 自述中：作者强调 tp_worker 和 Scheduler 两处必须一起修复，并给出只修一方的中间态复现（准入成功但 completion_tokens: 0、finish_reason: length）。Issue 中作者触发 /rerun-test 后 DCP 测试全部通过，随后确认 “Should be safe to merge”。这种“自证 + 集成

测试覆盖”的轻讨论模式表明风险主要靠复现矩阵兜底。

- 暂无高价值评论线程

风险与影响

- 风险：
  1. 核心路径变更：init_req_max_new_tokens 是每个请求初始化的必经逻辑，任何乘数错误都会影响所有请求的生成预算；不过 dcp_size == 1 时乘法为 no-op，非 DCP 部署不受影响。
  2. 一致性依赖：scheduler.py 的 clamp 与 PrefillAdder 的准入预算必须保持同一种计费模型。PR 声称上方已有同款 dcp_size 缩放，但本次改动未附带对 PrefillAdder 的验证，若后续该处被改动，可能出现“clamp 放行但调度仍拒绝”的不一致。
  3. 测试缺口：没有针对 dcp_size > 1 的单元测试，回归完全依赖 test/registered/dcp/* 的集成用例；这些用例需要真实多卡环境，普通 CI 无法覆盖。
  4. 内存放大：放开长上下文准入后，单请求可能占用聚合池的大量 KV（如 256k prompt），在池较小时可能加剧排队甚至 OOM，需要关注 mamba-full-memory-ratio 等显存配比的调参。 - 影响：用户侧：DCP 部署下长上下文请求（128k、256k）从“被拒绝”或“空响应”变为正常服务，PR 实测 128k prompt TTFT 3.04 s、256k 16.10 s，直接解锁 Kimi-K3 等长上下文模型在 tp8/dcp8 下的使用。系统侧：准入与生成预算统一到聚合池口径，避免请求进入等待队列却永远无法被调度的情况，降低健康检查误报风险；对 dcp_size == 1 的普通部署完全透明。团队侧：改动仅 2 文件、净增 7 行，属于低风险高收益的核心路径修正，但需要后续补齐单元测试。 - 风险标记：核心路径变更，缺少直接测试覆盖，DCP 专属逻辑

关联脉络

- PR #32880 Bound prefill delayer all-branch delay and decay the max_prefill_bs high-watermark: 同属调度器 admission 边界修复，与本次 max_new_tokens 预算截断同处 scheduler.py 的请求预算逻辑。
- PR #33118 [PD] Fix false health-503 during decode retraction re-admission: 同为调度器 admission 边界问题修复，与本次 DCP 准入问题都涉及队列阻塞与健康检查风险。
- PR #33038 Updated b200 kimi-k3 cookbook: 本 PR 的复现场景是 Kimi-K3 在 tp8/dcp8 + DSPARK 下的长上下文服务，与 Kimi-K3 部署配方形成同一模型的配套演进。