

PR #34963 完整报告

sgl-project/sglang

Fix dsv4 kl test timeout

合并时间: 2026-08-16 01:57

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/34963>

执行摘要

- 一句话: 修复 dsv4 KL 测试超时, 上调 `est_time` 并新增文件级超时
- 推荐动作: 值得快速阅读, 因为它展示了一个典型的 CI 超时诊断思路: 跟踪默认超时、`est_time` 和分片算法之间的联动。若要深入学习, 可进一步查看 `run_suite.py` 和 `compute_partitions.py` 的实现。推荐评分 4/10。

功能与动机

标题和 PR body 说明: `test_unified_radix_cache_kl_dsv4.py` 在每次定时运行时都被杀掉 (最近一次见 <https://github.com/sgl-project/sglang/actions/runs/31849115802>)。原因是 stage 未设置 `timeout_per_file`, `run_suite.py` 回退到 1200s 扁平默认值, 低于该文件注册的 `est_time=1500`。作者实测 4 卡 GPU 上 16 个测试需要 2055s 才能通过, 远超默认超时, 因此必须调整。

实现拆解

1. 定位超时链路: `run_suite.py` 在 stage 未设置 `timeout_per_file` 时使用 1200s 扁平默认值, 而该测试 `est_time` 注册为 1500s, 导致文件被杀。同时 `compute_partitions.py` 按 `est_time` 打包分片, 因此不能只调文件超时而不调 `est_time`。
2. 上调测试 `est_time`: 在 `test/registered/radix_cache/unified_radix_tree/test_unified_radix_cache_kl_dsv4.py` 中, 将 `register_cuda_ci(est_time=1500, ...)` 改为 `register_cuda_ci(est_time=2400, ...)`, 使分片排队和超时预估均按真实耗时 2055s 合理放大。
3. 新增文件级超时: 在 `.github/workflows/pr-test-extra.yml` 的对应 job 中增加 `timeout_per_file: '3600'`, 确保即使默认值变化也不会再误杀该测试, 并为同类长测试预留余量。
4. 说明未采用备选方案: `--timeout-from-est-time` 本可推导 3300s, 但该开关受 `scheduled` 输入门控, 且开启会让所有预定 stage 的 `max-parallel` 变为 1, 代价过大, 因此采用显式配置。

关键文件:

- `test/registered/radix_cache/unified_radix_tree/test_unified_radix_cache_kl_dsv4.py` (模块 缓存测试; 类别 test; 类型 test-coverage): 该测试文件是超时问题的直接对象, `est_time` 从 1500 上调至 2400, 使 CI 超时与分片估算对齐真实运行时长。

- .github/workflows/pr-test-extra.yml (模块 workflows配置; 类别 infra; 类型 infrastructure)
: 为 extra 阶段显式添加 timeout_per_file: 3600, 覆盖 run_suite.py 的 1200s 默认值,
是本次修复的另一半。

关键符号: register_cuda_ci

关键源码片段

test/registered/radix_cache/unified_radix_tree/test_unified_radix_cache_kl_dsv4.py

该测试文件是超时问题的直接对象, `est_time` 从 1500 上调至 2400, 使 CI 超时与分片估算对齐真实运行时长。

```
# 本文件注册为 CUDA CI 测试: est_time 必须覆盖真实运行耗时,  
# 否则 run_suite.py 会按 1200s 扁平默认值杀进程。  
# 实测 16 个用例在 4 卡 H100 上约需 2055s, 因此 est_time 从 1500  
# 上调到 2400, 同时避免 compute_partitions.py 按旧值分片导致作业超时。  
register_cuda_ci(  
    est_time=2400, # 覆盖实测耗时, 留约 15% 余量  
    stage="extra-b",  
    runner_config="4-gpu-h100",  
)
```

评论区精华

PR 无 review 评论; 作者在合并前通过 `/rerun-test` 命令单独重跑该测试, 机器人回复在 `4-gpu-h100` 上 [通过](https://github.com/sgl-project/sglang/actions/runs/31895403726) (<https://github.com/sgl-project/sglang/actions/runs/31895403726>), 确认超时调整有效。

- 重跑验证测试通过 (testing): 确认在提高超时后测试可以稳定跑完。

风险与影响

- 风险:
 1. `est_time` 上调至 2400 会影响 `compute_partitions.py` 的分片打包, 该测试可能被分到更晚的批次或占用更多时间预算, 可能挤压同 stage 其他测试。
 2. `timeout_per_file` 设为 3600 意味着若测试真正挂起, runner 最多会等 1 小时才失败, 资源占用风险提高。
 3. 该修复仅覆盖 `pr-test-extra.yml`, 其他 workflow (如 `pr-test.yml` 或 `scheduled` 流程) 若同样运行此测试, 仍可能受 1200s 默认超时影响。
 - 影响: 影响范围集中在 CI 基础设施: 解决了 `test_unified_radix_cache_kl_dsv4.py` 在 `extra-b` 阶段的持续超时失败, 提升定时运行的成功率和信号可靠性。对运行时的功能逻辑无影响。团队后续在新增长时间测试时, 需要同时考虑 `est_time` 与 `timeout_per_file` 的一致性。
 - 风险标记: CI 配置变更, 超时上限过高, 未覆盖所有 workflow

关联脉络

- PR #34913 [CI] Move the static ratchets back to CPU unit tests: 同为 CI 基础设施调整，涉及 test/registered 与 workflow 变更，与本 PR 同属 CI 执行机制优化。