

PR #24739 完整报告

sgl-project/sglang

fix patch_torch test queue race

合并时间: 2026-05-09 11:25

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/24739>

执行摘要

- 一句话: 修复测试队列竞争导致的 CUDA IPC 失败
- 推荐动作: 简单的测试竞态修复, 无需深入阅读。但拆分队列的模式 (SPSC) 值得在类似跨进程通信场景中参考。

功能与动机

原始测试使用单个共享 `mp.Queue`, sender 在 put tensor 后立即 get 等待 ack, 此时 receiver 可能尚未调用 get, 而 sender 自己的 get 可能从队列中取出自己刚放入的 tensor (CUDA IPC 自重入失败), 导致测试不稳定。PR body 明确说明: "Split single shared mp.Queue into two single-producer/single-consumer queues so the sender cannot pop the CUDA tensor it just put before the receiver wakes up."

实现拆解

1. 拆分队列: 将原来的单个 `queue = mp.Queue()` 替换为 `tensor_queue = mp.Queue()` 和 `ack_queue = mp.Queue()`, 分别用于传输 tensor 和确认信号。
2. 调整子进程逻辑: sender 角色从 `queue.put(tensor); queue.get() == 'done'` 改为 `tensor_queue.put(tensor); ack_queue.get() == 'done'`; receiver 角色从 `tensor = queue.get(); queue.put('done')` 改为 `tensor = tensor_queue.get(); ack_queue.put('done')`。
3. 简化测试用例: 移除 `enable_patch` 循环 (始终为 True) 和重复的 `params` 循环 (保持一个关键场景: sender 可见设备 [0,1] `tensor_device=1`, receiver 可见设备 [1,0] `tensor_device=0`), 并删除 `_test_monkey_patch_torch_reductions_core` 方法, 将逻辑内联到测试方法中。
4. 清理注册信息: 删除 AMD CI 注册 (`register_amd_ci`) 和导入, 仅保留 CUDA CI 注册, 并将 `est_time` 从 38 缩短至 15。

关键文件:

- `test/registered/rl/test_patch_torch.py` (模块测试; 类别 test; 类型 test-coverage; 符号 `_test_monkey_patch_torch_reductions_core`, `_run_subprocess`, `test_monkey_patch_torch_reductions`): 唯一变更文件, 修复测试竞态、简化测试逻辑并清理 CI 注册。

关键符号: `test_monkey_patch_torch_reductions`, `_run_subprocess`

评论区精华

无 review 评论，仅由作者触发 CI rerun 三次并通过。

- 暂无高价值评论线程

风险与影响

- 风险：仅修改测试文件，不影响生产代码。风险极低，测试用例逻辑简化后覆盖率略有降低（不再测试 patch 禁用和同设备场景），但对于核心竞态场景的针对性更强。
- 影响：仅影响 `test/registered/rl/test_patch_torch.py` 测试用例的稳定性和执行时间（估计从 38s 降至 15s）。生产代码无影响。
- 风险标记：暂无

关联脉络

- 暂无明显关联 PR