PR #31080 完整报告

sgl-project/sglang

[CI] Use torch.testing.assert_close in custom-all-reduce test (~1400x faster compare)

合并时间：2026-07-14 08:53

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/31080

执行摘要

- 一句话：用 torch.assert_close 替代 triton 版本，加速约 1400x
- 推荐动作：值得立即合并，是一项高性价比的 CI 优化。建议在类似多 GPU 测试中推广使用 torch.testing.assert_close 替代 triton.testing.assert_close，以减少 CPU 拷贝开销。

功能与动机

PR body 指出 triton.testing.assert_close 会将两个张量拷贝到 CPU，做 bf16→fp32 升型后单线程 numpy 比较，每次耗时 142.9 ms (vs torch.testing.assert_close 的 0.1 ms)，16 次参数化再叠加多 rank 竞争，导致该测试文件在 8 卡环境上耗时 ~625s，成为 CI 瓶颈。替换为 torch.testing.assert_close 可在 GPU 上直接比较，且通过 atol=0, rtol=0 保持精确相等语义。

实现拆解

1. 移除不再使用的 triton 导入：在 test/registered/jit/test_custom_all_reduce.py 中删除 import triton 语句，因为整个文件内 triton.testing.assert_close 已被替换。
2. 替换断言函数调用：将第 227 行的 triton.testing.assert_close(out_ref, out_jit, atol=0, rtol=0) 替换为 torch.testing.assert_close(out_ref, out_jit, atol=0, rtol=0)。该函数在 GPU 上原地比较，避免 CPU 拷贝和类型转换，性能提升约 1400x。
3. 保持断言语义不变：torch.testing.assert_close 默认检查 dtype/device 一致性，而输入均为同一 dtype 和 GPU device，结合 atol=0, rtol=0 实现和原来一样的精确相等断言（因为测试数据为小整数，在 bf16 精度内可精确表示）。

关键文件：

- test/registered/jit/test_custom_all_reduce.py（模块 测试；类别 test；类型 test-coverage）：唯一变更文件，包含导入删除和断言函数替换，直接影响 custom all-reduce 测试的执行性能。

关键符号：未识别

关键源码片段

test/registered/jit/test_custom_all_reduce.py

唯一变更文件，包含导入删除和断言函数替换，直接影响 custom all-reduce 测试的执行性能。

```
# 变更前后的核心比较逻辑片段

# 张量生成与 NCCL 参考值计算
for _ in range(TEST_LOOP):
    # NOTE: 15 * 8 < 128, which is the precision limit for bf16
    inp = torch.randint(0, 16, (TEST_LAYERS, size), dtype=dtype, device=device)
    assert comm.should_custom_ar(inp[0])
    out_ref = inp.clone()
    dist.all_reduce(out_ref, group=nccl_group)
    out_jit = run(inp)
    # 精确相等比较，使用 torch.testing.assert_close 替代 triton 版本
    # 原代码为 : triton.testing.assert_close(out_ref, out_jit, atol=0, rtol=0)
    # torch.testing.assert_close 在 GPU 上直接比较，避免 CPU 拷贝和类型转换
    # 由于输入为 bf16 小整数，精确相等语义完整保留
    torch.testing.assert_close(out_ref, out_jit, atol=0, rtol=0)
```

评论区精华

该 PR 无公开 review 评论。BBuf 直接批准了合并。GitHub Actions 上的 rerun test 显示一次失败，作者未进一步排查。

- 暂无高价值评论线程

风险与影响

- 风险：风险极低。变更仅涉及测试文件中的断言函数替换和导入删除，未改动任何生产代码或测试逻辑语义。torch.testing.assert_close 是 PyTorch 官方推荐工具，在 CI 环境中已广泛使用，与 triton.testing.assert_close 在 atol=0, rtol=0 下行为一致。唯一潜在差异是 torch.testing.assert_close 默认检查 dtype 和 device 完全一致，而 triton 版本可能自动升型，但本测试中两侧 dtype/device 相同，因此不影响正确性。
- 影响：CI 耗时降低：custom all-reduce 测试在多 GPU 环境下耗时显著缩短，预期从 ~625s 降到约 0.5s 以内，释放 CI 资源，缩短整体流水线时间。 仅影响测试：对用户无直接影响，不改变任何推理功能、API 或性能。 团队影响：维护者无需额外适配，改动极小。
- 风险标记：测试变更

关联脉络

- PR #30835 [Tiny] Enable Full Cuda Graph with Page size = 1: 同为与 custom all-reduce 或 CI 相关的 PR，间接涉及多 GPU 测试场景。
- PR #31056 Fix MockDSV4ModelRunner missing spec_algorithm: 同为近期测试修复类 PR，体现了对 CI 稳定性的持续关注。