PR #35044 完整报告

sgl-project/sglang

Stabilize GB300 nightly tests

合并时间：2026-08-17 18:30

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/35044

执行摘要

- 一句话：固定 GB300 NCCL 端口并拆分测试文件，消除夜测偶发失败
- 推荐动作：值得精读，尤其是对负责 CI/ 测试基础设施的工程师。本 PR 展示了两个可复用的工程决策：一是对 ephemeral 端口冲突的根因分析（端口范围 + 残留连接占用），二是把文件级超时细化为配置级超时以改善失败归因。建议合并后观察 1-2 轮 GB300 nightly 实际运行结果，确认端口假设与时间校准是否成立；若稳定，可考虑把 GB300_NCCL_PORT 的用法沉淀为文档约定，并评估是否用共享 fixture 减少 8 个文件间的配置复制。

功能与动机

PR body 明确了两个独立失败源：其一，分布式服务器启动时动态选择的端口落在 runner 的 ephemeral 客户端端口范围（10240-65535）内，尽管任务串行执行，刚关闭的客户端连接仍可能占用一个 ephemeral 端口，导致下一次 TCPStore 绑定出现 EADDRINUSE；其二，Qwen3.5 FP8 测试把 TP4 和 DP4 两个配置放在同一文件，共享 10800 秒文件级超时，TP4 消耗的时间挤占了 DP4 的预算，最终在 DP4 精度评估仍健康处理请求时整体超时；Kimi K2.5 以 10008 秒完成，处于同样的结构性风险之下。

实现拆解

本 PR 按以下 5 个步骤完成 GB300 nightly 测试套件的稳定化改造：

1. 固定 NCCL rendezvous 端口：新增 python/sglang/test/gb300_utils.py，定义 GB300_NCCL_PORT = "10000"；所有 GB300 nightly 测试文件的启动参数末尾追加 --nccl-port + GB300_NCCL_PORT。根因是 runner 的 ephemeral 客户端端口范围为 10240-65535，动态绑定可能撞上刚关闭连接残留的端口，固定到 10000 后端口可预测、可排查。
2. 按配置拆分测试文件：删除 test/registered/gb300/test_deepseek_v4_pro_fp4.py、test_kimi_k25_nvfp4.py、test_qwen35_fp8.py 三个合并文件，新增 8 个单配置文件（DeepSeek 的 low_latency / balanced / high_throughput，Kimi 的 tp / dp，Qwen 的 tp / dp）。每个文件只包含一个 ModelLaunchSettings 变体，独立调用 register_cuda_ci 注册自己的 est_time，使超时、失败与 CI 结果展示都能归因到具体配置。
3. 统一基类与参数保真：拆分后的测试类从 unittest.TestCase 改为继承 CustomTestCase，获得一致的清理行为。精度参数（如 mmmu_pro_vision 的 temperature 0.7、

max_tokens 32768、seed 0，gsm8k 的 baseline 0.935）、模型启动参数、环境变量（如 SGLANG_DEEPEP_NUM_MAX_DISPATCH_TOKENS_PER_RANK）均原样保留；唯一例外是 Qwen TP 的性能批次从 [1, 4] 调整为 [1, 8]，以保留 batch-size-8 的 TP 覆盖。

4. 迁移 TP1 确定性测试：把 test/registered/attention/test_glm4_moe_lite_deterministic.py 的 runner 池从 4-gpu-gb300 改为 4-gpu-b200（修改 register_cuda_ci 的 runner_config 参数）。理由是该测试为 TP1 且走相同的 SM100 Blackwell 后端路径，不必占用 GB300 机器。
5. 校准时间估计并验证：按日志重建 8 个配置的实测耗时并向上取整注册（如 Kimi DP 6940s→7200s、Qwen DP >6116s→7200s、DeepSeek low-latency 688s→720s），同步校准 GLM-5.2 与 GLM4 deterministic 的时间。最后通过 pre-commit、AST 解析、CI 注册收集与套件验证，确认最终 GB300 nightly 套件包含 8 个启用测试文件，两路 LPT 分区估计低于 6 小时上限；但未在真实 GB300/B200 硬件上执行。

关键文件：

- python/sglang/test/gb300_utils.py（模块 测试工具；类别 test；类型 configuration；符号 GB300_NCCL_PORT）：新增 GB300_NCCL_PORT 常量，是本 PR 端口冲突修复的核心载体；所有拆分后的 GB300 测试都从该模块导入端口值并追加到启动参数。
- test/registered/gb300/test_deepseek_v4_pro_fp4.py（模块 GB300 测试；类别 test；类型 deletion；符号 TestDeepSeekV4ProFp4, test_deepseek_v4_pro_fp4）：删除的合并文件：原文件在一个测试方法内串行运行 low-latency / balanced / high-throughput 三个配置，共享文件级超时；拆分后三个配置各自独立注册。
- test/registered/gb300/test_kimi_k25_nvfp4.py（模块 GB300 测试；类别 test；类型 deletion；符号 TestKimiK25Nvfp4, test_kimi_k25_nvfp4）：删除的合并文件：原文件同时承载 TP4+EAGLE3 与 TP4+DP4+DPA+EAGLE3 两个配置，Kimi 实测 10008 秒接近 10800 秒上限，存在与 Qwen 相同的超时误伤风险。
- test/registered/gb300/test_qwen35_fp8.py（模块 GB300 测试；类别 test；类型 deletion；符号 TestQwen35Fp8, test_qwen35_fp8）：删除的合并文件：TP4 与 DP4 共用一个 10800 秒文件级超时，TP4 挤占 DP4 预算导致误报失败，是本 PR 拆分最直接的起因。
- test/registered/gb300/test_deepseek_v4_pro_fp4_balanced.py（模块 GB300 测试；类别 test；类型 test-coverage；符号 TestDeepSeekV4ProFp4Balanced, test_deepseek_v4_pro_fp4_balanced）：拆分产物：balanced 配置（DP4+deepep+EAGLE1）独立注册 est_time=600s，继承 CustomTestCase，并追加 --nccl-port 固定端口。
- test/registered/gb300/test_kimi_k25_nvfp4_dp.py（模块 GB300 测试；类别 test；类型 test-coverage；符号 TestKimiK25Nvfp4Dp, test_kimi_k25_nvfp4_dp）：拆分产物：DP4+DPA+EAGLE3 配置独立注册 est_time=7200s，保留 mmmu_pro_vision 精度参数与 temperature 0.7 基线。
- test/registered/gb300/test_kimi_k25_nvfp4_tp.py（模块 GB300 测试；类别 test；类型 test-coverage；符号 TestKimiK25Nvfp4Tp, test_kimi_k25_nvfp4_tp）：拆分产物：TP4+EAGLE3 配置独立注册 est_time=4800s，性能批次保持 [1, 8]。

- test/registered/gb300/test_qwen35_fp8_dp.py（模块 GB300 测试；类别 test；类型 test-coverage；符号 TestQwen35Fp8Dp, test_qwen35_fp8_dp）：拆分产物：DP4+DPA+MTP 配置独立注册 est_time=7200s，性能批次保持 [16]。
- test/registered/gb300/test_qwen35_fp8_tp.py（模块 GB300 测试；类别 test；类型 test-coverage；符号 TestQwen35Fp8Tp, test_qwen35_fp8_tp）：拆分产物：TP4+MTP 配置独立注册 est_time=4800s，性能批次从 [1, 4] 调整为 [1, 8] 以保留 batch-size-8 的 TP 覆盖。
- test/registered/gb300/test_deepseek_v4_pro_fp4_low_latency.py（模块 GB300 测试；类别 test；类型 test-coverage；符号 TestDeepSeekV4ProFp4LowLatency, test_deepseek_v4_pro_fp4_low_latency）：拆分产物：low-latency 配置（flashinfer_mxfp4+EAGLE3）独立注册 est_time=720s。
- test/registered/gb300/test_deepseek_v4_pro_fp4_high_throughput.py（模块 GB300 测试；类别 test；类型 test-coverage；符号 TestDeepSeekV4ProFp4HighThroughput, test_deepseek_v4_pro_fp4_high_throughput）：拆分产物：high-throughput 配置（DP4+megamoe）独立注册 est_time=600s。
- test/registered/gb300/test_glm52_nvfp4.py（模块 GB300 测试；类别 test；类型 test-coverage；符号 TestGlm52Nvfp4）：GLM-5.2 NVFP4 测试的时间估计被校准，与 GLM deterministic 时间校准同属本次时间重校准配套。
- test/registered/attention/test_glm4_moe_lite_deterministic.py（模块 模型测试；类别 test；类型 test-coverage）：TP1 的 GLM4 MoE Lite 确定性测试从 GB300 池迁移到 B200 池（runner_config 改动），释放 GB300 资源且不损失 Blackwell 后端路径覆盖。

关键符号：test_deepseek_v4_pro_fp4, test_kimi_k25_nvfp4, test_qwen35_fp8, test_deepseek_v4_pro_fp4_low_latency, test_deepseek_v4_pro_fp4_balanced, test_deepseek_v4_pro_fp4_high_throughput, test_kimi_k25_nvfp4_tp, test_kimi_k25_nvfp4_dp, test_qwen35_fp8_tp, test_qwen35_fp8_dp

关键源码片段

test/registered/gb300/test_qwen35_fp8_tp.py

拆分产物：TP4+MTP 配置独立注册 est_time=4800s，性能批次从 [1, 4] 调整为 [1, 8] 以保留 batch-size-8 的 TP 覆盖。

```
import unittest

from sglang.test.accuracy_test_runner import AccuracyTestParams
from sglang.test.ci.ci_register import register_cuda_ci
from sglang.test.gb300_utils import GB300_NCCL_PORT
from sglang.test.performance_test_runner import PerformanceTestParams
from sglang.test.run_combined_tests import run_combined_tests
from sglang.test.test_utils import CustomTestCase, ModelLaunchSettings

# 拆分后的每个配置独立注册，获得独立的文件级超时与失败归因
register_cuda_ci(est_time=4800, stage="nightly", runner_config="4-gpu-gb300")

MODEL_PATH = "Qwen/Qwen3.5-397B-A17B-FP8"
```

```
COMMON_ARGS = [
    "--trust-remote-code",
    "--reasoning-parser=qwen3",
    "--tool-call-parser=qwen3_coder",
    "--enable-flashinfer-allreduce-fusion",
    "--attention-backend=trtllm_mha",
    "--mem-fraction-static=0.8",
    "--mamba-scheduler-strategy=extra_buffer",
    "--enable-multimodal",
    "--enable-metrics",
    # 固定 NCCL 端口，避免 ephemeral 范围内与残留连接端口冲突
    "--nccl-port",
    GB300_NCCL_PORT,
]

TP_MTP_ARGS = [
    "--speculative-algorithm=EAGLE",
    "--speculative-num-steps=3",
    "--speculative-eagle-topk=1",
    "--speculative-num-draft-tokens=4",
]


class TestQwen35Fp8Tp(CustomTestCase):
    """Qwen3.5-397B FP8 TP4+MTP on GB300（4x GB300 NVL4）。"""

    def test_qwen35_fp8_tp(self):
        # 精度参数与旧合并文件完全一致（temperature 0.7 而非 greedy），
        # 迁移时不得改动，以免破坏 baseline 可比性
        run_combined_tests(
            models=[
                ModelLaunchSettings(
                    MODEL_PATH,
                    tp_size=4,
                    extra_args=COMMON_ARGS + TP_MTP_ARGS,
                    variant="TP4+MTP",
                )
            ],
            test_name="Qwen3.5-397B-FP8 (TP4+MTP)",
            accuracy_params=AccuracyTestParams(
                dataset="mmmu_pro_vision",
                baseline_accuracy=0.76,
                repeat=1,
                max_tokens=32768,
                temperature=0.7,
                seed=0,
                sgl_eval_thinking=False,
            ),
            performance_params=PerformanceTestParams(
```

```
                # TP 批次从 [1, 4] 调整为 [1, 8]，保留 batch-size-8 的 TP 覆盖
                batch_sizes=[1, 8],
                result_dir="performance_results_gb300",
            ),
        )


if __name__ == "__main__":
    unittest.main()
```

评论区精华

该 PR 没有实质性的 review 评论线程（review_comments_count 为 0），唯一的 PR 评论是作者 mmangkad 贴出的 CI 运行链接（Run #31960672891）作为验证依据，Fridge003 直接 APPROVED 合并。PR body 对几个关键决策给出了自证逻辑：

> 端口选择 10000 的理由：低于 runner 的 ephemeral 范围下限 10240，避开刚关闭连接的残留端口。 Qwen TP 批次调整理由：保留 batch-size-8 的 TP 覆盖，DP 性能保持 [16]。 GLM deterministic 迁移理由：TP1 且走相同的 SM100 Blackwell 后端路径，不依赖 GB300 机器。

这些自证说明方案在团队内没有引发设计争议，但也没有得到 reviewer 的独立挑战。

- CI 验证运行链接 (other): 无实质技术讨论；Fridge003 直接 APPROVED 合并。

风险与影响

- 风险：
  1. 端口固定依赖环境假设：python/sglang/test/gb300_utils.py 中固定 10000，依赖 runner 上无其他服务占用该端口；若未来有服务绑定 10000，将从偶发 EADDRINUSE 变为确定冲突。
  2. 验证缺口：PR 自述未在真实 GB300/B200 硬件上运行，nightly 首次实际执行是唯一验证手段，若端口或超时假设不成立会直接暴露在正式套件中。
  3. 时间估计偏差：8 个配置的 est_time 基于单次日志重建（如 Qwen DP 仅观测到 >6116s 即超时，Kimi DP 6940s 接近上限），若模型权重、网络或 runner 负载波动，过紧的估计会导致误报超时。
  4. 配置漂移：8 个新文件各自复制 COMMON_ARGS 等大量配置常量，后续模型参数或端口约定变更需要同步修改多个文件，存在漂移风险（如新测试作者绕过 GB300_NCCL_PORT 约定）。
  5. 覆盖变化：Qwen TP 性能批次从 [1, 4] 调整为 [1, 8]，移除了 batch-size-4 的 TP 覆盖，虽然保留了 batch-size-8，但覆盖组合发生变化。 - 影响：影响范围集中在 CI/ 测试基础设施：GB300 nightly 套件的失败归因粒度从文件级细化到配置级，单个配置失败不再拖垮整个文件；分布式启动的端口选择从随机变为固定，降低偶发抖动；B200 池增加了一个 TP1 确定性测试的负载。对用户侧无功能影响，对推理性能无影响。测试维护者需要遵守新的端口约定与单配置文件约定，否则会退回到旧结构；同时新增的 8 个文件增加了测试套件的文件数量与维护成本。 - 风险标记：端口固定依赖环境假设，未在真

实硬件验证，测试时间估计偏差，配置常量多处复制

关联脉络

- PR #34856 fix tpot by adjusting the sliding max-prefill-size window size: 同属通过调整测试参数与时间窗口来稳定 nightly/ 性能指标的工程实践，与本 PR 的 est_time 校准动机一致。
- PR #33480 [AMD] Support prefill context parallel two batch overlap for DeepSeek V4: DeepSeek-V4-Pro 测试线（test_deepseek_v4_pro_fp4_* 命名族）的配套演进，本 PR 拆分的 DeepSeek 测试文件与其共享模型与测试主题。