

PR #56695 完整报告

vllm-project/vllm

[CI] Move smaller H200 workloads to 18GB MIG slices

合并时间: 2026-09-14 12:30

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/56695>

执行摘要

- 一句话: H200 CI 负载下沉 18GB MIG, 大内存用例拆分保留 35GB。
- 推荐动作:

功能与动机

PR body 明确说明目标: 将 CUDAGraph 和 V1 Executor + Worker 迁到 h200_18gb, 并把 Samplers 和 E2E Core 拆开, 让较小测试使用 18GB 切片, 而 Qwen2-Audio 多模态 beam search、Gemma-3n KV-sharing fast-prefill 全六种变体、Mamba 两种 MRV2 变体继续留在 h200_35gb。核心动机是 35GB MIG 切片是稀缺资源, 大量测试实际内存占用远低于切片上限, 下沉到 18GB 可以释放 35GB 切片给真正需要大内存的用例; PR 同时明确声明不改变任何模型、上下文长度或内存预算设置。

实现拆解

变更入口是 `.buildkite/test_areas/` 下的 4 个 CI 步骤定义文件, 按四步完成:

1. 直接队列切换: `.buildkite/test_areas/cuda.yaml` 的 CUDAGraph 步骤 (key: cudagraph) 和 `.buildkite/test_areas/misc.yaml` 的 V1 Executor + Worker 步骤 (key: v1-executor-worker) 仅将 `device: h200_35gb` 改为 `h200_18gb`。这两组在 18GB 上验证通过 (CUDAGraph 102 个通过; V1 Executor + Worker 694 个通过、1 个跳过), 说明内存需求远低于切片上限, 下沉无风险。
2. Samplers 拆分: `.buildkite/test_areas/samplers.yaml` 主组迁入 18GB 后, 在两条命令中追加 `-k "not test_beam_search_passes_multimodal_data"` 排除多模态 beam search 用例; 文件末尾新增 `samplers-multimodal-beam-search` 步骤 (35GB), 以 `VLLM_USE_FLASHINFER_SAMPLER=0/1` 两种模式分别运行该用例, 排除原因在注释中说明: Qwen2-Audio-7B 在 16 GiB 切片上不剩 KV cache 空间。
3. E2E Core 拆分 (最复杂): `.buildkite/test_areas/engine.yaml` 主组迁入 18GB, 并用 `--ignore v1/e2e/general/test_kv_sharing_fast_prefill.py` 和 `-k "not test_mamba_prefix_cache_mrv2"` 过滤大内存用例; 新增 `e2e-core-large-memory` 步骤 (35GB), 显式列出 Gemma-3n 六个 KV-sharing fast-prefill 变体和 Mamba MRV2 同步 / 异步两个用例。关键细节是 AMD mirror 块被加上显式 `commands` 覆盖, 防止 NVIDIA 侧的过滤条件传播到 AMD 侧导致覆盖收缩。

4. 验证闭环与回退：第一次 targeted run (#88636) 在 18GB 上发现 Mamba MRV2 两个用例真实 OOM (FlashInfer 请求 512 MiB, 切片仅剩 208-366 MiB), 第二次提交将这两个用例保留到 35GB 后, 最终 run (#88716) 在 PR head [ebbe149](#) 上 6 个 GPU 任务全部通过。验证采用 [\[skip ci\]](#) 抑制自动全量 CI, 并用 [VLLM_CI_ONLY_STEP_KEYS](#) 将分支构建钉到指定步骤; 另外还用 ci-infra [5aeb315](#) 的生成器验证了恰好 6 个 NVIDIA 步骤加前置依赖、AMD 命令未丢失。

关键文件:

- [.buildkite/test_areas/engine.yaml](#) (模块 [.buildkite/test_areas](#); 类别 config; 类型 configuration) : 配置 / 契约; 包含 配置键调整; +17/-3
- [.buildkite/test_areas/samplers.yaml](#) (模块 [.buildkite/test_areas](#); 类别 config; 类型 configuration) : 配置 / 契约; 包含 配置键调整; +19/-4
- [.buildkite/test_areas/cuda.yaml](#) (模块 [.buildkite/test_areas](#); 类别 config; 类型 configuration) : 配置 / 契约; 包含 配置键调整; +2/-2
- [.buildkite/test_areas/misc.yaml](#) (模块 [.buildkite/test_areas](#); 类别 config; 类型 configuration) : 配置 / 契约; 包含 配置键调整; +2/-2

关键符号: 未识别

评论区精华

本 PR 没有实质人工 review 讨论: 0 条 review 评论、0 条普通评论, 仅 [claude\[bot\]](#) 自动提示该仓库配置了人工 review 模式。PR body 承担了全部审计职责, 包括构建结果表格、执行 / 跳过覆盖对照 (与原始 35GB 任务一致、无测试被丢弃)、生产管线生成器校验以及重复工作排查 (确认无既有 PR 做相同队列迁移)。

- 暂无高价值评论线程

风险与影响

- 风险:
 1. 过滤条件误伤风险: [engine.yaml](#) 中 `-k "not test_mamba_prefix_cache_mrv2"` 是 `pytest` 子串匹配, 会连带排除 `test_mamba_prefix_cache_mrv2_async` 等所有名称含该子串的用例; 当前 35GB 步骤显式列名兜底, 但未来新增 `_mrv2_*` 用例不会自动纳入 35GB 步骤, 存在静默丢覆盖风险。
 2. MIG 18GB 内存余量小: 实测切片空闲内存仅约 208-366 MiB 量级, Mamba MRV2 已触发一次 OOM; 未来新增大内存用例极易再次踩线, 需要依赖 #56747 的 MIG 内存遥测持续监控。
 3. AMD/NVIDIA 命令分离: [engine.yaml](#) 中 AMD mirror 显式覆盖 `commands`, 后续修改 NVIDIA 侧过滤规则时若忘记同步 AMD 块, 会导致两侧行为不一致。
 4. 覆盖依赖新步骤触发条件: 被排除用例的覆盖完全依赖新增步骤的 `source_file_dependencies` 命中, 若未来相关源码目录调整导致依赖不匹配, 新步骤可能不触发而无人察觉。- 影响: 对最终用户无感知, 不涉及任何模型、运行时或推理逻辑变更; 影响集中在 CI 基础设施: 35GB 切片负载下降、18GB 切片利用率提升, 释放稀

缺的大内存切片资源。对团队而言，CI 维护者需要理解和维护 - 风险标记：暂无

关联脉络

- 暂无明显关联 PR