PR #24762 完整报告

sgl-project/sglang

[AMD] fix(triton-mla): cap max_kv_splits at 256 on gfx942 (Kimi-K2.6 hang)

合并时间：2026-06-03 15:13

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/24762

执行摘要

- 一句话：限制 gfx942 上 max_kv_splits 为 256，修复 Kimi-K2.6 挂起
- 推荐动作：值得精读。设计决策：针对特定 SKU 硬编码上限是否优于动态内存预算？后续若能统一为“两倍最大上下文分割数”则更通用。此外，is_gfx942_supported 的引入为后续 AMD 特殊处理提供了范例。

功能与动机

修复 nightly-8-gpu-kimi-k26 MI325X 挂起问题，原因是 PR #20479 的 _mla_decode_kv_splits_cap() 将 max_kv_splits 提升至 512，导致 cuda_graph_attn_logits 缓冲区膨胀至 4 GiB，超出 ROCm CUDA 图重放能力（https://github.com/sgl-project/sglang/actions/runs/25513282022/job/74877480809）。

实现拆解

1. 在 python/sglang/srt/utils/common.py 中新增 is_gfx942_supported() 函数，带 @lru_cache 装饰，检测 gcnArchName 是否包含 gfx942。
2. 在 python/sglang/srt/layers/attention/triton_backend.py 中导入 is_gfx942_supported，模块级缓存 _is_gfx942。在 TritonAttentionBackend.__init__ 的 MLA 分支内追加条件：若 _is_gfx942 为真，则将 self.max_kv_splits 限制为 min(self.max_kv_splits, 256)。
3. 在 test/registered/amd/test_kimi_k2_instruct.py 中将 parallel=1319 改为 parallel=512，避免修复后剩余内存不足以支撑高并发。
4. 更新 .github/workflows/pr-test-amd.yml 和 pr-test-amd-rocm720.yml 中的 --auto-partition-size 从 3 增至 4，以容纳新增的 MI325X 测试分区。

关键文件：

- python/sglang/srt/utils/common.py（模块 工具库；类别 source；类型 core-logic；符号 is_gfx942_supported）：新增 is_gfx942_supported() 函数，作为平台检测基础设施。
- python/sglang/srt/layers/attention/triton_backend.py（模块 注意力层；类别 source；类型 core-logic；符号 _is_gfx942）：核心修复：在 MLA 初始化时限制 max_kv_splits，防止缓冲区过大。
- test/registered/amd/test_kimi_k2_instruct.py（模块 测试；类别 test；类型 test-coverage）：调整测试并发数，避免修复后因内存减少仍导致 OOM。

- .github/workflows/pr-test-amd.yml（模块 CI 配置；类别 infra；类型 infrastructure）：增加 MI325X 测试分区数，确保新增测试能被调度。
- .github/workflows/pr-test-amd-rocm720.yml（模块 CI 配置；类别 infra；类型 infrastructure）：同 pr-test-amd.yml，修改分区数以匹配新增测试需求。

关键符号：is_gfx942_supported

关键源码片段

python/sglang/srt/layers/attention/triton_backend.py

核心修复：在 MLA 初始化时限制 max_kv_splits，防止缓冲区过大。

```
# triton_backend.py 文件头部分
from sglang.srt.utils import (
    is_gfx942_supported,
)
_is_gfx942 = is_gfx942_supported() # 模块级缓存，只检测一次

# 在 __init__ 方法中，self.use_mla 分支内
if self.use_mla:
    self.max_kv_splits = _mla_decode_kv_splits_cap(
        self.max_kv_splits,
        self.device_core_count,
        self.max_context_len,
    )
    if _is_gfx942:
        # gfx942 (MI300X / MI325X) 有 304 个 CU，next_power_of_2 会得到 512，
        # 导致 cuda_graph_attn_logits 缓冲区在 Kimi-K2.6 上膨胀到 4 GiB。
        # 强制限制为 256，与 gfx950 的行为一致且经过验证。
        self.max_kv_splits = min(self.max_kv_splits, 256)
```

评论区精华

HaiShaw 在审查时指出应避免使用含糊变量名（如早期提交中的 bs），确保代码可读性；最终实现使用自解释的 self.max_kv_splits。

- 变量命名规范 (style): 最终代码未使用 bs，采用了自解释的 self.max_kv_splits。

风险与影响

- 风险：
  1. 仅针对 gfx942 限制，不影响 NVIDIA 或其他 AMD SKU；但未来若有新 SKU 的 CU 数量超过 256，需重新评估。
  2. 测试并行度降低至 512，可能未覆盖高并发场景下的内存压力。
  3. is_gfx942_supported() 基于 GPU 名称字符串匹配，若 ROCm 报告格式变化可能导致误判。 - 影响：用户：Kimi-K2.6 在 AMD MI325X 上不再 hang，恢复正常推理。系统：CUDA 图捕获阶段内存占用从 4 GiB 降至 2 GiB，图重放稳定性提升。团队：新增平台检测函数可复用，但需注意代码维护。CI 测试增加一个分区以确保覆盖。 - 风险标记

：平台特异性硬编码，测试并发度降低，字符串匹配依赖

关联脉络

- PR #20479 Support Triton MLA FP8 KV cache: 引入 _mla_decode_kv_splits_cap() 导致 max_kv_splits 过度膨胀，是本次 bug 的根因。
- PR #27004 fix(disagg): correct DSA/SWA state-page transfer mismatch in PD disaggregation: 同为 AMD 平台 bugfix，涉及 kv-cache 传输。