PR #36094 完整报告

sgl-project/sglang

[AMD][DSV4] perf: retune decode split-K heuristic for MI355X

合并时间：2026-08-29 04:48

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/36094

执行摘要

- 一句话：调优 DSv4 split-K 启发式，MI355X 解码内核提速 20.8%
- 推荐动作：值得精读。这是一个小而完整的性能调优案例：用几何平均遗憾量化启发式偏差、识别 CUDAGraph 捕获期无法感知 per-token K 的根本限制，并用契约测试固定不变量而非数值。对 AMD 性能工作和 attention 内核开发者有直接参考价值，尤其是 test_tuned_operating_points 显式传参跨运行器的做法。

功能与动机

_kv_splits_heuristic 在 MI355X 上对整个 *decode* 区间过度拆分：它在 H=128/block_h=64 时为 T=32/64/128 选择了 8/4/2 个 split，而实测 4/2/1 更快。PR body 指出 Split-K 只在基础网格未填满设备时才有收益，每次额外 split 都会增加 partial buffer 写入和 reduce 内核工作；当每个 split 的 K 变短后，这些开销不再被摊还。旧常量 (2.0, 64) 的几何平均遗憾为 33.5%（最坏 119%），新常量 (1.5, 16) 降到 3.7%（最坏 36%）。

实现拆解

1. 定位过度拆分：分析 python/sglang/kernels/ops/attention/dsv4/unified_kv_kernels/paged_decode.py 中 _kv_splits_heuristic 的填充逻辑，其用 base_ctas = T * ceil(H / block_h) 与 target_wg = target_wg_per_cu * num_cu 比较决定是否拆分；MI355X 扫描（T in 1..256 × kv_len in 128/512/1024，H=128）显示旧 (2.0, 64) 的几何平均遗憾 33.5%、最坏 119%，而 (1.5, 16) 只有 3.7% / 36%。
2. 调整常量并加入 HIP 门控：引入 from sglang.srt.utils import is_hip，定义模块级 _is_hip，_MAX_KV_SPLITS 在 HIP 上收紧为 16（CUDA 保持 64），_TARGET_WG_PER_CU 在 HIP 上降为 1.5（CUDA 保持 2.0），并让 _kv_splits_heuristic 的默认参数指向这两个常量；同步更新 docstring，记录 MI355X 测量依据与拆分开销原理。CUDA 路径不受影响。
3. 新增契约测试：新增 test/registered/unit/layers/test_dsv4_kv_splits_heuristic.py，8 组 38 个用例，通过 register_cpu_ci(est_time=5, suite="base-a-test-cpu") 注册 CPU CI。测试断言的是不变量（pow2 上限、单调不增、饱和网格不拆、CUDAGraph 安全）而非常量值，只有 test_tuned_operating_points 用显式传入的 MI355X 常量钉住六档操作点，未来重新调优只需改这一张表。
4. 配套提交：*e3ecc36* 满足 pre-commit 的 assert 换行格式；fbcc65d 注册 CPU CI（否则 test/registered/ 下的文件不会被调度）；*dc3cb4e* 修复 codespell 对 retuned 的误报；

cbfa616 按 review 要求把常量改动收进 _is_hip 门控。

关键文件：

- python/sglang/kernels/ops/attention/dsv4/unified_kv_kernels/paged_decode.py（模块 解码内核；类别 source；类型 core-logic；符号 _kv_splits_heuristic, _MAX_KV_SPLITS, _TARGET_WG_PER_CU, _is_hip）：核心参数调优位置：_kv_splits_heuristic 的默认常量 target_wg_per_cu 与 _MAX_KV_SPLITS 按平台切换，并新增 _is_hip 门控保证 CUDA 不受影响。
- test/registered/unit/layers/test_dsv4_kv_splits_heuristic.py（模块 启发式测试；类别 test；类型 test-coverage；符号 test_prev_pow2_is_largest_power_of_two_not_exceeding, test_prev_pow2_clamps_non_positive, test_result_is_positive_power_of_two_within_cap, test_splits_never_increase_with_token_count）：为之前完全没有测试的 _kv_splits_heuristic 新增 38 个契约测试，固定启发式不变量与 MI355X 调优操作点，并注册 CPU CI。

关键符号：_kv_splits_heuristic, _prev_pow2, test_tuned_operating_points, test_splits_never_increase_with_token_count, test_saturated_grid_does_not_split, test_does_not_read_tensors_only_capture_time_scalars, test_result_is_positive_power_of_two_within_cap

关键源码片段

python/sglang/kernels/ops/attention/dsv4/unified_kv_kernels/paged_decode.py

核心参数调优位置：_kv_splits_heuristic 的默认常量 target_wg_per_cu 与 _MAX_KV_SPLITS 按平台切换，并新增 _is_hip 门控保证 CUDA 不受影响。

```
# Split-K 启发式常量。 (1.5, 16) 是针对 MI355X 实测调优的组合；
# CUDA 路径没有测量，保留原有 (2.0, 64)，避免未经验证的回归。
_is_hip = is_hip()
_MAX_KV_SPLITS = 16 if _is_hip else 64 # 每个 token 的 split 硬上限
_TARGET_WG_PER_CU = 1.5 if _is_hip else 2.0


def _kv_splits_heuristic(
    T: int,
    H: int,
    block_h: int,
    num_cu: int | None = None,
    target_wg_per_cu: float = _TARGET_WG_PER_CU,
    max_kv_splits: int = _MAX_KV_SPLITS,
) -> int:
    '挑选 KV_SPLITS 来填满 GPU。 CUDAGraph 安全：只依赖捕获时标量。

    原 2.0 会在整个 decode 区间多拆一个 2 的幂次（H=128 时 T=32/64/128\n 选 8/4/2，实测 4/2/1 更快）。 Split-K 只在基础网格填不满设备时才有\n 收益，额外 split 的 partial buffer 写入与 reduce 开销会在 K 变短后\n 占据主导。\n '
```

```
    # base_ctas = T * ceil(H / block_h)：每个 token 的头部块数 × token 数
    base_ctas = T * ceil(H / block_h)
    # 目标工作项数略高于 CU 数，用来掩盖负载不均衡
    target_wg = target_wg_per_cu * num_cu
    if base_ctas >= target_wg:
        # 基础网格已饱和，再拆只会增加 reduce 开销
        return 1
    # 拆到刚好能填满设备并向下取 2 的幂次；reduce 内核按 pow2 步长
    # 索引 partial buffer，非 2 的幂次会破坏归约结果而不是变慢
    return _prev_pow2(int(min(target_wg / base_ctas, max_kv_splits)))
```

test/registered/unit/layers/test_dsv4_kv_splits_heuristic.py

为之前完全没有测试的 _kv_splits_heuristic 新增 38 个契约测试，固定启发式不变量与 MI355X 调优操作点，并注册 CPU CI。

```
@pytest.mark.parametrize(
    'tokens,expected',
    [
        # MI355X（256 CU、H=128、block_h=64）上调优的操作点。
        # base_ctas = tokens * 2；target_wg = int(1.5 * 256) = 384。
        (8, 16), # 384 // 16 = 24 -> 受上限 16 约束
        (16, 8), # 384 // 32 = 12 -> prev_pow2 -> 8
        (32, 4), # 384 // 64 = 6 -> prev_pow2 -> 4
        (64, 2), # 384 // 128 = 3 -> prev_pow2 -> 2
        (128, 1), # 384 // 256 = 1
        (256, 1), # 基础网格已饱和，不再拆分
    ],
)
def test_tuned_operating_points(tokens, expected):
    '显式传入 MI355X 常量，使该表在任何 CI 运行器上都成立；\n
    重新调优时只需更新这一张表。\n    '
    assert (
        _kv_splits_heuristic(
            tokens,
            HEADS,
            BLOCK_H,
            num_cu=NUM_CU,
            target_wg_per_cu=1.5,
            max_kv_splits=16,
        )
        == expected
    )


def test_does_not_read_tensors_only_capture_time_scalars():
    'CUDAGraph 安全：启发式必须能用纯 int 调用且不创建 CUDA 上下文。\n    若在捕获阶段读取
    kv_indices / kv_indptr，会把捕获时的值烘焙进每次回放。\n    '
    splits = _kv_splits_heuristic(32, HEADS, BLOCK_H, num_cu=NUM_CU)
    assert isinstance(splits, int)
```

评论区精华

核心讨论围绕平台安全性展开。HaiShaw 在 review 中要求：“Please make parameter change under _is_hip”，即重调参数不能影响未测量的 CUDA 路径；作者在 cbfa616 提交中实现 _is_hip 门控，并补充说明 test_tuned_operating_points 固定的是 MI355X 数字、在 CPU CI 上 is_hip() 为 False，因此测试改为显式传入 MI355X 常量，表格在任何运行器上保持有效。HaiShaw 随后批准并确认 “HIP specific”。没有遗留未解决问题。

- 将重调参数限定在 HIP 路径 (design): 已实现 is_hip() 门控，HIP 使用 (1.5, 16)，CUDA 保持 (2.0, 64)；HaiShaw 批准并确认 'HIP specific'。
- CPU CI 上测试与 HIP 常量的脱节 (testing): 测试显式传入 target_wg_per_cu=1.5 与 max_kv_splits=16，表在任何运行器上有效；属性测试继续使用模块默认值。

风险与影响

- 风险：
  1. 平台覆盖不全：新常量只在 MI355X 实测，所有 HIP 设备（包括 MI300 系列）都会走 (1.5, 16)；虽然启发式本身对任意常量都安全，但其他 ROCm 设备上可能不是最优。
  2. 性能数据噪声：端到端基准每个并发点只跑一次，存在运行噪声；不过吞吐与 TPOT 在每个并发点都同向变化（一升一降），比单一指标更可信。
  3. 测试运行环境：CPU CI 上模块默认常量是 CUDA 的 (2.0, 64)，属性测试因此没有直接覆盖 HIP 常量；但属性对两组常量都成立，操作点表单独显式传参，不会造成回归盲区。
  4. 正确性风险低：选择逻辑未改，只是两个常量；GSM8K 两次运行 0.958 / 0.946、invalid 0，且拆分归约仅引入 bf16 重关联差异。 - 影响：用户影响：MI355X + DP attention 部署 DeepSeek-V4-Pro 时端到端吞吐提升 0.8%～4.3%（并发越高越明显），TPOT 下降 0.8%～4.2%，解码内核生产形状（T=32、top-k 1024、H=128）从 55.7 us 降至 44.1 us（-20.8%）。普通 TP8 下效果持平。系统影响：改动集中于 ROCm-only 的 dsv4 unified_kv 后端，CUDA 保持原状；无 API、配置或部署变更。团队影响：为 _kv_splits_heuristic 补齐了此前缺失的测试覆盖，建立了契约测试模式，未来架构调参的成本显著降低。 - 风险标记：仅 HIP 路径生效，其他 ROCm 设备未验证，基准单次运行，CUDAGraph 捕获期静态启发式

关联脉络

- 暂无明显关联 PR