PR #52419 完整报告

vllm-project/vllm

[Bugfix][Spec Decode] Keep EAGLE cache registration on the partial-hash-hit path

合并时间：2026-08-16 12:03

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/52419

执行摘要

- 一句话：修复 EAGLE 分支在部分哈希命中路径上丢失缓存注册
- 推荐动作：值得精读。虽然只有 2 个文件，但 PR 展示了高质量 bugfix 的完整闭环：精确的因果定位（#50062 重写分支时丢失豁免）、可达性论证（哪些配置组合触发）、量化性能验证（60 分钟端到端 ladder + 命中率）以及防止复发的结构手段（统一 _align_cacheable 入口 + 针对性回归测试）。review 中关于"豁免是 load-bearing"的讨论也值得留意。

功能与动机

50062 在重写 HybridKVCacheCoordinator.cache_blocks 的 EAGLE 分支时，用 num_finalized_computed_tokens 无条件执行 // scheduler_block_size * scheduler_block_size 重新推导注册边界，把原本已在 partial hash hits 路径上移除的向下取整带了回来。结果 EAGLE 组的注册上限从 n 变成 floor(n / scheduler_block_size) * scheduler_block_size + manager.block_size，未对齐尾部 (n % scheduler_block_size) - manager.block_size 个 token 每次调用都不再注册。PR body 给出实测：Kimi-K3 + DSpark 在并发 16 时吞吐下降 13.8%、命中率从 86.3% 掉到 77.4%；可达性条件为 Mamba align 组 + prefix caching + dcp_world_size == 1，任何 hybrid Mamba 模型配 EAGLE 系列 drafter 都会命中该路径。

实现拆解

1. 抽取统一舍入入口：在 vllm/v1/core/kv_cache_coordinator.py 新增私有方法 _align_cacheable(num_tokens)，并从 vllm.utils.math_utils 引入 round_down。enable_partial_hash_hits 为真时返回原始 token 数（不取整），否则返回

round_down(num_tokens, self.scheduler_block_size)。这一步把 " 开启 partial hashhits 时禁止一切向下取整 " 的规则收敛到单一函数，避免后续在别的分支里再次丢失。

2. 重构 cache_blocks 主路径：原来外层 aligned_num_computed_tokens 与 EAGLE 分支内 aligned_num_finalized_computed_tokens 各写了一遍取整逻辑；现在外层先算 cached_num_computed_tokens = self._align_cacheable(num_computed_tokens)，EAGLE 分支对 num_finalized_computed_tokens 也调用同一 helper。变量名从 aligned_* 改为 cached_*，因为 partial hash hits 下结果并不对齐，命名更准确；min(num_finalized_computed_tokens, cached_num_finalized_computed_tokens + manager.block_size) 的 EAGLE lookahead 语义保持不变。
3. 新增回归测试：在 tests/v1/core/prefix_cache/test_partial_prefix_cache_hits.py 添加 test_eagle_group_registers_unaligned_tail_under_partial_hash_hits，构造 full-attention（block_size = 2）与 Mamba align（block_size = 8）两组、scheduler_block_size = 8 的配置，把 full-attention 组标记为 EAGLE 并断言其 block_size < scheduler_block_size；用 spy 包装每个 manager 的 cache_blocks 记录实际传入的 num_tokens_to_cache，通过 allocate_slots 走真实路径后断言两组都注册完整 22 个 token。
4. 配套验证：无配置、schema、部署改动。作者在父提交 acb0f1dcdb 上确认新测试以 [18, 22] != [22, 22] 失败，修复后 tests/v1/core/prefix_cache/ 25 个测试全过，tests/v1/core/ 全量对比仅新增 1 个通过；端到端 60 分钟 ladder 给出前后对照。

关键文件：

- vllm/v1/core/kv_cache_coordinator.py（模块 缓存协调；类别 source；类型 core-logic；符号 cache_blocks, _align_cacheable）：核心修复文件：新增 _align_cacheable 统一注册边界舍入逻辑，修复 #50062 在 EAGLE 分支重新引入向下取整导致的 prefix-cache 注册遗漏。
- tests/v1/core/prefix_cache/test_partial_prefix_cache_hits.py（模块 前缀缓存；类别 test；类型 test-coverage；符号 test_eagle_group_registers_unaligned_tail_under_partial_hash_hits, spy）：新增回归测试，用 spy 捕获每个 manager 收到的 num_tokens_to_cache，验证 EAGLE 组在 partial hash hits 下注册完整未对齐尾部；该测试在父提交上失败、修复后通过。

关键符号：cache_blocks, _align_cacheable, test_eagle_group_registers_unaligned_tail_under_partial_hash_hits

关键源码片段

vllm/v1/core/kv_cache_coordinator.py

核心修复文件：新增 _align_cacheable 统一注册边界舍入逻辑，修复 #50062 在 EAGLE 分支重新引入向下取整导致的 prefix-cache 注册遗漏。

```
# SPDX-License-Identifier: Apache-2.0
# SPDX-FileCopyrightText: Copyright contributors to the vLLM project

    def _align_cacheable(self, num_tokens: int) -> int:
        """返回未来缓存命中最大可匹配的 token 前缀。
```

```
        未开启 fine-grained partial hash hits 时，命中总是落在
        ``scheduler_block_size`` 边界上（见 ``find_longest_cache_hit``），
        需要向下取整；开启后不得取整 —— 即使按 ``hash_block_size``
        取整，也会把 CoW 私有化的 Mamba 尾部重新注册进 hash map。
        """
        if self.enable_partial_hash_hits:
            return num_tokens
        return round_down(num_tokens, self.scheduler_block_size)

    def cache_blocks(self, request: Request, num_computed_tokens: int) -> None:
        # 统一入口：先算一次可注册边界，EAGLE 分支不再自行推导舍入，
        # 避免 partial hash hits 路径上把已移除的取整重新引入（#50062 回归）。
        cached_num_computed_tokens = self._align_cacheable(num_computed_tokens)
        for manager in self.single_type_managers:
            num_tokens_to_cache = cached_num_computed_tokens
            # EAGLE 组在每个对齐边界后多匹配一个块并丢弃它，
            # 因此让这个 lookahead 块也有资格被缓存。
            if manager.use_eagle and cached_num_computed_tokens > 0:
                # 只缓存 KV 已 finalize 的 token：最后
                # num_reprefillable_tokens 个 token 在 multi-module MTP
                # 中可能被重新 prefill。
                num_finalized_computed_tokens = max(
                    0, num_computed_tokens - self.num_reprefillable_tokens
                )
                cached_num_finalized_computed_tokens = self._align_cacheable(
                    num_finalized_computed_tokens
                )
                num_tokens_to_cache = min(
                    num_finalized_computed_tokens,
                    cached_num_finalized_computed_tokens + manager.block_size,
                )
            manager.cache_blocks(
                request,
                num_tokens_to_cache,
                retention_interval=self.retention_interval,
            )
```

评论区精华

njhill 在 review 中提出三条风格建议并全部被采纳：(1) 用已有的 round_down 工具函数替代手写取整；(2) 精简内部方法 docstring；(3) 将 aligned_num_computed_tokens 重命名为 cached_num_computed_tokens，因为 partial hash hits 下该值并不总是对齐的。mispa-ms 在回复中特别解释了保留 docstring 中一句关键说明的原因：在 enable_partial_hash_hits 下即使按 hash_block_size 取整也会让 test_hybrid_mamba_partial_tail_owner_uses_cow_on_continue 失败——CoW 私有化的 tail block 被重新注册后 block_hash 不再是 None，因此这条"禁止取整"的豁免是 load-bearing 的，作者实测验证而非假设。两位 reviewer（njhill、TheEpicDolphin）均批准。

- 复用 round_down 工具函数 (style): 已采纳，引入 vllm/utils/math_utils.py 的 round_down。
- docstring 精简与变量命名 (style): mispa-ms 全部采纳：删除 Args/Returns，保留一句关键说明；变量名统一为 *cached_**。
- partial hash hits 下禁止取整的语义依据 (correctness): 该豁免被确认为 load-bearing，通过新测试固化，文档明确记载。

风险与影响

- 风险：
  1. 影响面收敛：修复只作用于 *enable_partial_hash_hits* 为真（Mamba align 组 + prefix caching + dcp_world_size == 1）且使用 EAGLE 系列 drafter 的组合；普通模型与无投机解码场景不受影响。
  2. 正确性保持：变更只扩大已计算 token 的注册范围，不改变计算与命中返回语义；num_reprefillable_tokens 仍排除可重 prefill 尾部，EAGLE last-block drop 逻辑未动，输出 bit-level 不变。
  3. 遗留风险：scheduler_block_size >= num_prefill_lookahead 守卫的注释在 partial hash hits 开启时与 *_cache_hit_alignment_tokens* 实际返回 hash_block_size 的行为不一致，作者明确留给 reviewer 处理；该组合（multi-module MTP + hybrid Mamba）当前无人运行。
  4. 维护风险：*_align_cacheable* 的“禁止取整”语义是隐性契约，后续若有人把 hash_block_size 舍入加回来会重新引入 CoW tail 注册 bug，新测试提供了防护。
- 影响：
  - 用户：Kimi-K3 等 hybrid Mamba 模型 + DSpark/EAGLE 投机解码在 partial hash hits 下吞吐恢复 7-14%（并发 16 时 6,611 → 7,654 tok/s/GPU），prefix-cache 命中率 77.4% → 86.2%；无投机解码的 ladder 全程在 ±1.8% 内。
  - 系统：改动集中于 HybridKVCacheCoordinator.cache_blocks，每条请求注册范围变宽，hash map 内存占用略增但量级很小。
  - 团队：PR body 提供了完整的可达性分析与性能前后对照，可作为后续 prefix-cache 相关回归修复的参考模板。
  - 风险标记：影响面集中于部分哈希命中路径，*_align_cacheable* 语义为隐性契约，prefill lookahead 守卫注释待澄清

关联脉络

- PR #46384 Partial prefix-cache hit feature: PR body 明确指出 partial hash hits 功能由 #46384 引入，本修复作用的 *enable_partial_hash_hits* 路径正是它的产物。
- PR #49125 Bugfix in the same area: stale partial hash revival: 同一区域的另一个 bugfix（stale partial hash revival），PR body 中列为邻居以说明本 PR 不是重复。
- PR #50438 RFC: lookahead-aware prefix-cache hashing for EAGLE-style drafters: PR body 对照的最接近主题的提案，主张不同 hash 方案，与本 PR 的注册边界修复互补。

- PR #49793 [Spec Decode][Perf] Fuse the MTP trailing all-reduce; local-argmax draft tokens: 同属 EAGLE/MTP 投机解码 + KV 缓存路径的持续优化，历史 PR 分析中与 deepseek_v32/mtp.py 相关。
- PR #52288 [Bugfix][Spec Decode] DSpark: inherit the target's attention backend when the speculative config names none: 同为 DSpark 投机路径修复，与本 PR 端到端验证的 Kimi-K3/DSpark 场景直接相关。