# PR #28302 完整报告

sgl-project/sglang

[Mamba][GDN] Deduplicate spec conv-window intermediate cache via sliding window layout

合并时间：2026-06-18 11:42

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/28302

## 执行摘要

- 一句话：使用滑动窗口布局去重 Mamba spec 卷积中间缓存，内存减半
- 推荐动作：值得精读：展示了巧妙的滑动窗口去重思路、Triton kernel 的正确性验证流程（布局测试 + bit-exact + E2E md5）。设计决策（条件启用、参数透传）和实践（as_strided 视图的谨慎使用）有参考价值。建议关注 max_mamba_cache_size 的 follow-up 优化。

## 功能与动机

当前 spec decode 中的中间卷积窗口缓存为每个 draft token 存储一个独立的 [dim, K-1] 窗口，但相邻 token 的窗口重叠 K-2 个元素，密集存储浪费约一半空间。本 PR 通过滑动窗口布局消除冗余，为更大并发留出显存，降低 OOM 风险。

## 实现拆解

1. 新增 Triton scatter kernel（mamba_state_scatter_triton.py）：新增 _fused_conv_window_scatter_with_mask_kernel 和 fused_conv_window_scatter_with_mask。与原有 fused_mamba_state_scatter_with_mask 不同，新 kernel 从非连续的 as_strided 视图中逐 (dim, win) 元素索引，而非 flat-copy。
2. 内存池支持去重布局（memory_pool.py）：新增 conv_window_dedup_enabled 函数（仅在 CUDA + topk ≤ 1 时返回 True）。MambaPool.__init__ 根据此标志分配去重布局（[dim, D+K-2] 物理 buffer + as_strided 视图）或原密集布局；同时新增 speculative_eagle_topk 参数以获取 topk 值。
3. 透传 topk 参数：在 model_runner_kv_cache_mixin.py 和 disaggregation/decode.py 中，将 speculative_eagle_topk 从 server_args 传递到 MambaPool 构造函数。
4. 调用点适配（hybrid_linear_attn_backend.py）：在 update_mamba_state_after_mtp_verify 中，将对 conv 中间状态的 scatter 调用从 fused_mamba_state_scatter_with_mask 替换为 fused_conv_window_scatter_with_mask。
5. 测试验证：新增 test/srt/mamba/test_conv_window_dedup.py，包括 CPU/numpy 布局正确性验证（窗口重构、切片等价、重叠一致性，比例精确 0.5）、GPU 数值 bit-exact 校验（max_abs_diff = 0.0）、以及 E2E md5 字节一致性测试（Qwen3.5-0.8B GDN 模型，

固定 prompt，temperature=0）。

关键文件：

- python/sglang/srt/layers/attention/mamba/mamba_state_scatter_triton.py（模块 Triton 内核；类别 source；类型 core-logic；符号 _fused_conv_window_scatter_with_mask_kernel, fused_conv_window_scatter_with_mask）：核心 Triton kernel 实现：新增 _fused_conv_window_scatter_with_mask_kernel 和非连续源的 scatter 函数，是去重布局的关键读写路径。
- python/sglang/srt/mem_cache/memory_pool.py（模块 内存池；类别 source；类型 core-logic；符号 conv_window_dedup_enabled, MambaPool）：内存池核心逻辑：新增 conv_window_dedup_enabled 决定是否启用去重布局，修改 MambaPool.__init__ 分配物理 / 逻辑 buffer，并接受 speculative_eagle_topk 参数。
- python/sglang/srt/model_executor/model_runner_kv_cache_mixin.py（模块 运行时；类别 source；类型 data-contract）：将 speculative_eagle_topk 从 server_args 透传到 pool 构造，使池能根据 topk 决定布局。
- python/sglang/srt/layers/attention/hybrid_linear_attn_backend.py（模块 注意力后端；类别 source；类型 core-logic）：调用点修改：使用新的 fused_conv_window_scatter_with_mask 替换原有的 fused_mamba_state_scatter_with_mask 来处理 conv 中间状态。
- python/sglang/srt/disaggregation/decode.py（模块 解耦解码；类别 source；类型 core-logic）：为 HybridMambaDecodeReqToTokenPool 添加 speculative_eagle_topk 参数传递，支持去重布局在解耦解码模式下的正确配置。

关键符号：_fused_conv_window_scatter_with_mask_kernel, fused_conv_window_scatter_with_mask, conv_window_dedup_enabled, MambaPool.init, update_mamba_state_after_mtp_verify

## 评论区精华

- 正确性：树验证别名冲突：BBuf 指出在 EAGLE 树验证场景下，两个逻辑窗口元素可能映射到同一物理列但需要不同值。yuan-luo 确认仅线性链有效，并添加了 speculative_eagle_topk <= 1 的条件守卫。
- 简化建议：BBuf 建议将合法性检查简化为 raise ValueError(...) 和 (speculative_eagle_topk or 0) <= 1，yuan-luo 均采纳。
- mem_usage 统计：BBuf 询问 MambaPool.mem_usage_bytes() 是否应使用物理缓冲区而非逻辑视图。该问题未完全闭环，但 PR 已合并，表明当前追踪逻辑视图的影响可控。
  - 去重布局是否仅适用于线性 draft 链 (correctness): 添加条件 speculative_eagle_topk <= 1 确保仅在线性链生效。
  - 树验证时别名列冲突风险 (correctness): 增加 topk > 1 时回退密集布局的检查，已在 conv_window_dedup_enabled 中添加。
  - 代码风格简化建议 (style): 接受简化：异常消息明确指定函数名，条件改为 (speculative_eagle_topk or 0) <= 1。

- mem_usage_bytes 应使用物理还是逻辑缓冲区 (design): 未完全解决，但最终 PR 合并，当前仍统计逻辑视图（即原大小），不影响容量判断（因为 max_mamba_cache_size 未变）。

# 风险与影响

- 风险：
  1. 仅适用于线性 draft 链：若未来启用 topk > 1（树验证）而未正确禁用此优化，会因别名列冲突产生错误结果。当前通过 conv_window_dedup_enabled 严格守卫，风险较低。
  2. 非 CUDA 平台回退：NPU/CPU 保留密集布局，但需确保所有调用路径都检查了标志；当前通过硬编码 _is_npu / _is_cpu 分支保证。
  3. 新 kernel 性能：_fused_conv_window_scatter_with_mask_kernel 中逐元素 stride 读取比 flat-copy 慢，但该 kernel 仅在验证阶段每步触发一次，且 footprint 减半带来的 cache 友好性可缓解。基准测试未报告回归。
  4. 测试覆盖：仅覆盖了线性链和顶部 ≤ 1 的情况，缺少对 topk>1 路径的回归测试（会被自动跳过）。 - 影响：影响范围：适用于所有使用 Mamba2/GDN 混合线性注意力模型并启用 NEXTN 推测解码的用户。显存节省比例与 linear 层数、conv_dim、并发请求数成正比（如 Qwen3.5-35B-A3B 每 rank 减少 0.04 GB）。不影响纯 Transformer 模型、NPU/CPU 后端或首次解码（prefill）。释放的内存可降低 OOM 概率，但 max_mamba_cache_size 未改变，吞吐量不变。团队需注意后续若支持树验证需重新设计或回退。 - 风险标记：条件分支风险，仅支持线性 draft 链，新 kernel 性能开销

# 关联脉络

- 暂无明显关联 PR