PR #29678 完整报告

sgl-project/sglang

feat(mem_cache): unified memory pool for hybrid Mamba / SWA models

合并时间：2026-07-02 04:21

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/29678

执行摘要

- 一句话：统一内存池动态分配 KV 与 Mamba 状态缓存
- 推荐动作：建议精读。此 PR 是 SGLang 内存管理架构的重要演进，展示了如何通过统一内存池实现灵活的资源分配。特别值得关注：
  - MultiEndedAllocator 的懒压缩策略——只重映射表而不重写引用，避免数据搬迁开销。
  - cuda-graph 集成中虚拟地址预先翻译的设计，确保图中无翻译节点。
  - 调度器准入控制的字节粒度预算计算，防止过度分配。 对于希望理解推理引擎内存管理的工程师而言，此 PR 提供了很好的实践案例。

功能与动机

Hybrid Mamba/GDN 和 hybrid SWA 模型维护两种缓存：full-attention KV 缓存和 per-request Mamba conv/SSM（或 SWA）状态。当前两者有各自独立大小的池子，容量划分在启动时固定——其中一个可能耗尽而另一个还有空闲。此 PR 增加了 opt-in --enable-unified-memory 来解决此问题。

实现拆解

1. 设计统一内存池架构

在 unified_memory_pool.py 中引入 UnifiedKVPool，它管理一个单一的 uint8 字节缓冲区，并通过两个 MultiEndedAllocator 分别提供给 full-attention KV 子池和 Mamba/SWA 状态子池。每个子池的字节布局由 SubPoolSpec 抽象类及其子类 MHASubPoolSpec、MambaSubPoolSpec 定义。

2. 实现多端分配器

multi_ended_allocator.py 中的 MultiEndedAllocator 从缓冲区的两端之一向上或向下增长，管理虚拟页面到物理页面的映射表，并支持懒压缩（仅重映射表、不重写引用）。

3. 集成到模型运行器和注意力后端

在 model_runner_kv_cache_mixin.py 中添加 _init_unified_mamba_pools 和 _init_unified_swa_pools，根据模型配置创建统一池。在 triton_backend.py 和 hybrid_linear_attn_backend.py 中，所有写位置信息（KVWriteLoc）包含物理地址，池子不再持有写状态，且 cuda-graph 的虚拟地址翻译在 init_forward_metadata_out_graph 中预先

完成。

4. 调整调度准入控制

在 schedule_policy.py 的 PrefillAdder 中新增 _mamba_gap_budget_for_req，计算每个请求的 Mamba 状态在共享间隙中的字节预留，防止过度准入。alloc_req_slots 现在在无法满足时直接抛出异常（fail-loud）。

5. 添加约束和验证

在 server_args.py 的 _handle_unified_memory_pool 中验证约束：需要 Triton 注意力后端、线性注意力后端、Mamba 后端；仅支持 monolithic decode cuda-graph；拒绝 PD 分解、推测解码、层次化缓存、decode 上下文并行等。

6. 测试覆盖

新增多个单元测试：test_layout_compat.py 验证字节偏移和视图正确性；test_unified_mamba_views.py 验证 Mamba 状态视图的完整性；test_full_loc_fast_path.py 验证写路径路由；以及 test_store_cache_4d.py 的修改。

关键文件：

- python/sglang/srt/mem_cache/unified_memory_pool.py（模块 内存池；类别 source；类型 core-logic；符号 _prod, _store_dtype_for, SubPoolSpec, post_init）：统一内存池的核心实现，包含字节布局定义、子池规格、池子创建工厂函数及虚拟到物理翻译逻辑。
- python/sglang/srt/model_executor/model_runner_kv_cache_mixin.py（模块 模型运行器；类别 source；类型 data-contract；符号 _should_enable_lazy_compaction, _init_unified_mamba_pools, _init_unified_swa_pools）：添加统一内存池的初始化入口，根据模型类型（Mamba/SWA）调用对应的工厂函数，是集成关键点。
- python/sglang/srt/mem_cache/multi_ended_allocator.py（模块 分配器；类别 source；类型 core-logic；符号 MultiEndedAllocator, UnifiedMambaTokenToKVPoolAllocator, UnifiedSWATokenToKVPoolAllocator）：实现多端分配器，是统一池从两端生长的关键基础设施。
- python/sglang/srt/layers/attention/triton_backend.py（模块 注意力后端；类别 source；类型 core-logic；符号 _translate_cuda_graph_shared_pool_locs）：集成 cuda-graph 的虚拟地址翻译，以及写位置元数据的传递。
- python/sglang/srt/managers/schedule_policy.py（模块 调度策略；类别 source；类型 core-logic；符号 _mamba_gap_budget_for_req）：调整准入控制，增加 Mamba 字节预算计算，防止过度分配共享间隙。
- test/registered/unit/mem_cache/test_layout_compat.py（模块 测试；类别 test；类型 test-coverage；符号 _make_mha_spec, _make_mamba_spec, TestMHASpecLayerOffsets, test_offsets_at_page_size_1_match_envelope）：验证字节布局偏移量在 page_size=1 和 page_size>1 时的一致性，确保向后兼容。
- test/registered/unit/mem_cache/test_unified_mamba_views.py（模块 测试；类别 test；类型 test-coverage；符号 _make_pool, TestUnifiedMambaViews, _fill_and_roundtrip, test_roundtrip_falcon_like）：验证统一池中 Mamba 状态视图（conv/temporal）的字节

正确性和不对齐问题。

关键符号：_init_unified_mamba_pools, _init_unified_swa_pools, _translate_mamba_indices, _translate_cuda_graph_shared_pool_locs, _mamba_gap_budget_for_req, alloc_req_slots, set_kv_buffer, _should_enable_lazy_compaction, MultiEndedAllocator.alloc, MultiEndedAllocator.free

关键源码片段

python/sglang/srt/model_executor/model_runner_kv_cache_mixin.py

添加统一内存池的初始化入口，根据模型类型（Mamba/SWA）调用对应的工厂函数，是集成关键点。

```
def_init_unified_mamba_pools(self: ModelRunner, max_num_reqs: int): """为混合 Mamba 模型构建统一内存池：一个字节缓冲区分割给 full-attention KV 和 Mamba 状态子池"""
fromsglang.srt.mem_cache.unified_memory_poolimport init_unified_mamba_pools
config = self.mambaish_config assert config is not None assert not self.use_mla_backend, "统一池暂不支持 MLA-hybrid-Mamba" assert self.page_size >= 1 # 镜像非统一路径的 extra_max_context_len 计算 extra_max_context_len = 4 if self.server_args.speculative_num_draft_tokens is not None:
extra_max_context_len += self.server_args.speculative_num_draft_tokens # 过滤出当前 PP rank 的层 ID mamba_layer_ids = [ i for i in config.mamba2_cache_params.layers if self.start_layer <= i < self.end_layer ]
full_attention_layer_ids = [ i for i in config.full_attention_layer_ids if self.start_layer <= i < self.end_layer ] # 调用工厂函数创建统一池 bundle = init_unified_mamba_pools( device=self.device,
kv_cache_dtype=self.kv_cache_dtype, head_num=self.model_config.get_num_kv_heads(get_attention_tp_size()), head_dim=self.model_config.head_dim,
page_size=self.page_size, start_layer=self.start_layer,
end_layer=self.end_layer, is_draft_worker=self.is_draft_worker,
use_mla_backend=self.use_mla_backend, mamba_layer_ids=mamba_layer_ids,
full_attention_layer_ids=full_attention_layer_ids,
mamba2_cache_params=config.mamba2_cache_params,
model_context_len=self.model_config.context_len,
extra_max_context_len=extra_max_context_len,
max_total_num_tokens=self.max_total_num_tokens,
max_mamba_cache_size=self.server_args.max_mamba_cache_size,
max_num_reqs=max_num_reqs, enable_memory_saver=self.server_args.enable_memory_saver, enable_mamba_extra_buffer=self.server_args.enable_mamba_extra_buffer(), speculative_num_draft_tokens=self.server_args.speculative_num_draft_tokens, disable_overlap_schedule=self.server_args.disable_overlap_schedule,
need_sort=self.server_args.disaggregation_mode in ("decode", "prefi"..., ) 注意：以上为截取的核心部分，实际代码较长。
```

评论区精华

1. 设计决策：fail-loud 替代 planner-refusal

Codex 审查指出，原先的 planner-refusal + re-queue 路径掩盖了调度器会计错误。作者在 commit d48ab40477 中移除了 PlannerRefused 异常，改为 fail-loud，强制 PrefillAdder 正确收集可用大小。

2. 优化：Mamba 字节预留上取整

Codex 审查发现，Mamba 插槽大小不是完整 KV token 整数倍时向下取整会导致预留不足。作者在 commit 54ac2e6915 中改为上取整。

3. 正确性：FP8 缓存 dtype 处理

Codex 审查指出，FP8 缓存时统一池错误地传递存储 dtype（uint8）而非逻辑 dtype，导致转换错误。已在后续提交中通过 _store_dtype_for 函数修复。

4. 兼容性：流水线并行下零层子池

在 PP 配置下，一个 rank 可能仅拥有 full-attention 或仅 Mamba 层，导致子池层数为 0 而断言失败。此问题在后续提交中通过允许零层子池解决。

- FP8 缓存 dtype 处理错误 (correctness): 已在后续提交中通过 _store_dtype_for 函数修复：当 KV 缓存 dtype 为 FP8 时，返回值仍为原 dtype，而非 uint8。
- 流水线并行下零层子池断言失败 (correctness): PR 后续提交中允许零层子池（通过调整断言逻辑或为空的子池返回特殊处理）。
- Mamba 字节预留量应向上取整 (performance): 作者在 commit 54ac2e6915 中改为向上取整，保守估计预算。
- planner-refusal 路径掩盖调度错误 (design): 在 commit d48ab40477 中移除 PlannerRefused 异常，改为 RuntimeError，强制 PrefillAdder 正确计算可用预算。

风险与影响

- 风险：
  1. 约束限制较多：要求使用 Triton 注意力 / 线性注意力 /Mamba 后端；仅支持 monolithic decode cuda-graph；与 PD 分解、推测解码、层次化缓存、decode 上下文并行不兼容。用户需明确了解这些约束。
  2. 性能基准缺失：PR 说明吞吐量 / 利用率基准测试是后续任务，当前仅保证正确性。实际性能收益有待验证。
  3. cuda-graph 集成复杂：cuda-graph 的虚拟地址翻译在 replay-prep 中预先完成，若某个翻译遗漏或错误可能导致静默数据损坏。虽然已有测试，但仍是风险点。
  4. 与 HiCache 的 Mamba 路径尚未充分测试：PR 提到 HiCache Mamba offload/restore 路径已接线但未在准确性运行中测试。
  5. 与 FP8 缓存共同使用的潜在风险：Codex 审查指出过 dtype 处理问题，虽已修复，但类似的细微错误可能还存在。

- 影响：影响范围：
  - 用户：仅在显式指定 --enable-unified-memory 且满足后端约束时激活。默认行为不变，因此对现有用户无影响。
  - 系统：对于混合 Mamba/SWA 模型（如 Qwen3.5、Falcon-H1、Gemma-4、gpt-oss-20b），启用统一池后，KV 和 Mamba 状态将共享同一字节缓冲区，容量可动态调整，提高内存利用率。但需要额外的虚拟到物理翻译，可能引入少许开销（但 cuda-graph 路径预先翻译以控制开销）。
  - 团队：需维护新的 multi_ended_allocator.py 和 unified_memory_pool.py 代码，以及相关 plumbing。
  - 风险标记：约束限制多，性能基准待测，与 PD/ 推测解码不兼容，cuda-graph 翻译风险，HiCache 路径未充分测试

关联脉络

- PR #29533 feat(mem_cache): page-major (layer-major within a page) KV/state layout: 此 PR 基于 page-major 布局构建，统一池的物理布局要求页面内按层连续，因此依赖于 #29533 的布局变更。