PR #26425 完整报告

sgl-project/sglang

[core] Maintain `req_pool_indices_cpu` host mirror (like `seq_lens_cpu`)

合并时间：2026-05-27 09:59

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/26425

执行摘要

- 一句话：维护 req_pool_indices 的 CPU 镜像，消除 Hisparse 每 decode 的 D2H 复制
- 推荐动作：建议团队关注此 PR 的实现思路：通过维护 CPU 镜像来避免设备 - 主机同步，是常见的性能优化模式。对后续调度系统优化有参考价值。

功能与动机

PR body 指出：维持 req_pool_indices 的 CPU 镜像，供 Hisparse 使用以消除每次 decode 的 D2H 复制，并解锁重叠调度中主机端索引的使用。

实现拆解

1. 新增属性：在 ScheduleBatch 类中增加 req_pool_indices_cpu 属性（数据类型 torch.int64 的 CPU tensor），作为 req_pool_indices 的 CPU 镜像，注释标明 schedule-path only, stale in spec draft window。
2. 分配入口修改：在 prepare_for_extend 中，原来忽略 alloc_for_extend 的第三个返回值（_），现在接收为 req_pool_indices_cpu 并赋值给 self.req_pool_indices_cpu。同时修改 alloc_for_extend 返回签名，由 list[int] 改为 torch.Tensor，直接返回已构建的 CPU tensor。
3. 其他生命周期同步：在 prepare_for_prebuilt（decode_schedule_batch_mixin）中也从 Python int list 构建并赋值；在 prepare_for_idle 中初始化为空 tensor；在 filter_batch 中按 keep_indices 切片；在 merge_batch 中 torch.cat 连接。所有操作与 seq_lens_cpu 完全对等。
4. 消费者修改：HisparseCoordinator.map_last_loc_to_buffer 增加 req_pool_indices_cpu 参数，移除函数内部的 req_pool_indices.cpu() 调用，直接使用传入的 CPU tensor。
5. 测试配套：无专门测试文件变更。作者声明此变更为 non-behavioral，且 hisparse 在 CI 中无覆盖；CI 中 test_hisparse_unit 和 test_disaggregation_basic 通过。

关键文件：

- python/sglang/srt/managers/schedule_batch.py（模块 调度器；类别 source；类型 core-logic）：核心调度批处理类，添加 req_pool_indices_cpu 属性并在所有生命周期函数中同步维护。
- python/sglang/srt/mem_cache/common.py（模块 缓存层；类别 source；类型 core-logic）：alloc_for_extend 函数修改返回类型为 CPU tensor，使得调用方能直接获得 CPU 镜像。

- python/sglang/srt/managers/hisparse_coordinator.py（模块 稀疏注意力；类别 source；类型 core-logic）：Hisparse 协调器，修改 map_last_loc_to_buffer 以接收 req_pool_indices_cpu 参数，移除内部 D2H 复制。
- python/sglang/srt/disaggregation/decode_schedule_batch_mixin.py（模块 解耦解码；类别 source；类型 core-logic）：disaggregation 混合类，在 prepare_for_prebuilt 中添加 req_pool_indices_cpu 的构建。

关键符号：alloc_for_extend, ScheduleBatch.prepare_for_extend, HisparseCoordinator.map_last_loc_to_buffer

关键源码片段

python/sglang/srt/managers/schedule_batch.py

核心调度批处理类，添加 req_pool_indices_cpu 属性并在所有生命周期函数中同步维护。

```
# python/sglang/srt/managers/schedule_batch.py

class ScheduleBatch:
    # ... 已有字段 ...
    req_pool_indices: torch.Tensor = None # shape: [b], int64
    seq_lens: torch.Tensor = None # shape: [b], int64
    seq_lens_cpu: torch.Tensor = None # shape: [b], int64
    # CPU mirror of req_pool_indices; schedule-path only, stale in spec draft window
    req_pool_indices_cpu: torch.Tensor = None # shape: [b], int64

    def prepare_for_extend(self):
        # ... 前置代码 ...
        # 分配内存：现在从 alloc_for_extend 接收 req_pool_indices_cpu
        out_cache_loc, req_pool_indices_tensor, req_pool_indices_cpu = alloc_for_extend(self)
        # ... 后续设置 ...
        self.req_pool_indices = req_pool_indices_tensor
        self.req_pool_indices_cpu = req_pool_indices_cpu # 新增：存储 CPU 镜像

    def filter_batch(self, keep_indices):
        # ... 对所有 batch tensor 应用 keep_indices ...
        self.req_pool_indices_cpu = self.req_pool_indices_cpu[keep_indices] # 新增

    def merge_batch(self, other):
        # ... 连接所有 batch tensor ...
        self.req_pool_indices_cpu = torch.cat(
            [self.req_pool_indices_cpu, other.req_pool_indices_cpu]
        ) # 新增
```

python/sglang/srt/mem_cache/common.py

alloc_for_extend 函数修改返回类型为 CPU tensor，使得调用方能直接获得 CPU 镜像。

```
# python/sglang/srt/mem_cache/common.py

def alloc_for_extend(
```

```
    batch: ScheduleBatch,
) -> tuple[torch.Tensor, torch.Tensor, torch.Tensor]:
    """
    Allocate KV cache for extend batch and write to req_to_token_pool.

    Returns:
        out_cache_loc: allocated cache locations
        req_pool_indices_device: request pool indices as a device tensor
        req_pool_indices_cpu: request pool indices as a CPU tensor (host mirror)
    """
    # ... 分配 req slots ...
    req_pool_indices = alloc_req_slots(...)
    # 从 Python list 构建 CPU tensor
    req_pool_indices_cpu = torch.tensor(req_pool_indices, dtype=torch.int64)
    req_pool_indices_device = req_pool_indices_cpu.to(batch.device, non_blocking=True)
    # ... 分配 KV cache ...
    # ... 写入 cache indices ...
    return out_cache_loc, req_pool_indices_device, req_pool_indices_cpu # 变更：返回 CPU
    tensor 而非 list
```

python/sglang/srt/managers/hisparse_coordinator.py

Hisparse 协调器，修改 map_last_loc_to_buffer 以接收 req_pool_indices_cpu 参数，移除内部 D2H 复制。

```
# python/sglang/srt/managers/hisparse_coordinator.py

class HisparseCoordinator:
    def map_last_loc_to_buffer(
        self,
        seq_lens: torch.Tensor,
        out_cache_loc: torch.Tensor,
        req_pool_indices: torch.Tensor,
        seq_lens_cpu: torch.Tensor,
        req_pool_indices_cpu: torch.Tensor, # 新增参数：已在 CPU 上
    ) -> None:
        # 之前：req_pool_indices_cpu = req_pool_indices.cpu() # 移除了 D2H 复制
        self._eager_backup_previous_token(
            seq_lens, req_pool_indices, seq_lens_cpu, req_pool_indices_cpu
        )
        # ... 后续代码不变 ...
```

评论区精华

本 PR 无公开 review 讨论。作者在 GitHub 上声明此变更是 non-behavioral（仅增加新字段，仅被 hisparse 使用，且 hisparse 在 CI 中无覆盖）。CI 中 test_hisparse_unit 和 test_disaggregation_basic 通过。

- 暂无高价值评论线程

风险与影响

- 风险：低风险。新字段仅用于调度路径，在 spec draft 窗口中标记为 stale，不影响模型前向。所有修改点与 seq_lens_cpu 对等，容易同步维护。潜在风险：后续新增 batch 操作（如 split_batch）若未同步维护 req_pool_indices_cpu，会导致不一致；hisparse 功能无 CI 覆盖，若引入 bug 可能不易察觉。
- 影响：直接影响：Hisparse 用户受益——每次 decode 步减少一次 CPU 复制操作，降低延迟。间接影响：为重叠调度功能铺平道路，使主机端可以直接访问 req_pool_indices。对其他用户无影响。
- 风险标记：Hisparse 缺少 CI 覆盖，filter/merge 同步维护风险

关联脉络

- 暂无明显关联 PR