PR #26148 完整报告

sgl-project/sglang

Skip PP output communication for pure chunked prefill batches

合并时间：2026-05-26 21:59

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/26148

执行摘要

- 一句话：PP 中跳过纯 chunked prefill 批次的输出通信，释放 SM 资源提升性能
- 推荐动作：建议阅读：理解 pipeline parallelism 中 SM 占用对 kernel 延迟的影响及如何通过跳过无用通信优化。若部署 DeepSeek-V4 长输入场景（256K token），可启用该优化获得 3-7% TTFT 收益。代码实现简洁（+112 行），验证充分，值得参考。

功能与动机

在 pipeline parallelism 中，非最终 chunk 的 prefill 的 next_token_ids 在 process_batch_result_prefill 中被丢弃，通信纯属浪费。PP send/recv 各占用 1 个 SM，导致需要全部 78 SM 的内核（如 flash_fwd_splitkv_mla_fp8_sparse_kernel、deep_gemm 等）被迫分两波执行，延迟从 4.9ms 增加到 9.7ms。对于需要 32 个 chunk 的 256K 输入，每个请求有 31 次不必要的 send/recv。

实现拆解

1. 在 environ.py 中新增 SGLANG_PP_SKIP_PURE_CHUNKED_OUTPUT_COMM 环境变量，默认关闭。
2. 在 schedule_batch.py 的 ScheduleBatch 类中新增布尔属性 contains_last_prefill_chunk，默认为 True 确保安全。
3. 在 scheduler.py 的 _get_new_batch_prefill_raw 中，根据 chunked_req 和批次大小设置该标记：若不存在 chunk 请求或批次大小不为 1，则为 True（含最终 chunk），否则为 False（纯中间 chunk）。
4. 在 scheduler_pp_mixin.py 中新增模块级函数 _pp_can_skip_output_comm，检查环境变量、forward_mode=EXTEND、batch size=1、contains_last_prefill_chunk=False 且 return_logprob=False 时返回 True。
5. 在 _pp_send_output_to_next_stage 和 _do_recv 中分别检查该条件：send 路径跳过 send_dict_to_next_stage；recv 路径调用新方法 _pp_make_skip_output_result 直接返回全零 placeholder 和 GenerationBatchResult(skipped_output_comm=True)，同时将 next_pp_outputs 置为 None 使非 last rank 跳过 forward。
6. 在 utils.py 的 GenerationBatchResult 中添加 skipped_output_comm 字段。
7. 在 batch_result_processor.py 中新增 _validate_pp_skip_output_comm 静态方法：skip=True 时断言所有 active req 的 inflight_middle_chunks > 0，防止占位输出被误用；

skip=False 且无 req 消费输出时记录 warning。该方法受同一环境变量控制。

关键文件：

- python/sglang/srt/managers/scheduler_pp_mixin.py（模块 PP 通信；类别 source；类型 core-logic；符号 _pp_can_skip_output_comm, _pp_make_skip_output_result）：核心变更文件：新增 _pp_can_skip_output_comm 和 _pp_make_skip_output_result，修改 send/recv 路径逻辑，实现跳过通信的核心控制。
- python/sglang/srt/managers/scheduler_components/batch_result_processor.py（模块 结果处理；类别 source；类型 core-logic；符号 _validate_pp_skip_output_comm）：新增 _validate_pp_skip_output_comm 进行正确性校验，防止占位输出被消费；并添加 warning 记录未命中优化的情况。
- python/sglang/srt/managers/scheduler.py（模块 调度核心；类别 source；类型 core-logic；符号 contains_last_prefill_chunk）：在创建新 batch 时根据 chunked_req 状态设置 contains_last_prefill_chunk 标记，是决策判断的关键一环。
- python/sglang/srt/managers/utils.py（模块 结果定义；类别 source；类型 data-contract；符号 skipped_output_comm）：GenerationBatchResult 新增 skipped_output_comm 字段，用于传递跳过标记，供校验使用。
- python/sglang/srt/environ.py（模块 环境变量；类别 source；类型 configuration；符号 SGLANG_PP_SKIP_PURE_CHUNKED_OUTPUT_COMM）：新增环境变量 SGLANG_PP_SKIP_PURE_CHUNKED_OUTPUT_COMM，作为功能开关，默认关闭。
- python/sglang/srt/managers/schedule_batch.py（模块 批处理；类别 source；类型 data-contract；符号 contains_last_prefill_chunk）：ScheduleBatch 新增 contains_last_prefill_chunk 属性，配合跳过逻辑。

关键符号：_pp_can_skip_output_comm, _pp_make_skip_output_result, _validate_pp_skip_output_comm

关键源码片段

python/sglang/srt/managers/scheduler_pp_mixin.py

核心变更文件：新增 _pp_can_skip_output_comm 和 _pp_make_skip_output_result，修改 send/recv 路径逻辑，实现跳过通信的核心控制。

```
# 新增模块级函数，判断是否可以跳过当前 batch 的 PP 输出通信
# 条件：环境变量开启、非 prebuilt 的 EXTEND 模式、单请求批次、
# 且该请求是中间 chunk（非最后一个 chunk）、且不要求返回 logprob。
def _pp_can_skip_output_comm(batch: ScheduleBatch) -> bool:
    """Check if output send/recv can be skipped for this batch."""
    return (
        envs.SGLANG_PP_SKIP_PURE_CHUNKED_OUTPUT_COMM.get()
        and batch is not None
        and batch.forward_mode == ForwardMode.EXTEND
        and len(batch.reqs) == 1
        and not batch.contains_last_prefill_chunk
        and not batch.return_logprob
```

```
    )

# 当跳过通信时，由接收端调用，直接构造占位结果
# placeholder 是全零 int64 tensor，标记 skipped_output_comm=True
# next_pp_outputs 返回 None，使非 last rank 跳过 forward
def _pp_make_skip_output_result(
    self: Scheduler,
    batch: ScheduleBatch,
    mb_metadata: Optional[PPBatchMetadata],
):
    bs = len(batch.reqs)
    placeholder = torch.zeros(bs, dtype=torch.int64, device=self.device)
    batch.output_ids = placeholder # 非 last rank 需要占用 output_ids
    batch_result = GenerationBatchResult(
        logits_output=None,
        pp_hidden_states_proxy_tensors=None,
        next_token_ids=placeholder,
        can_run_cuda_graph=(
            mb_metadata.can_run_cuda_graph if mb_metadata else False
        ),
        skipped_output_comm=True,
    )
    d2h_event = self.device_module.Event()
    d2h_event.record(self.device_module.current_stream())
    return None, batch_result, d2h_event # next_pp_outputs = None
```

评论区精华

Reviewer ShangmingCai 提出两个关键讨论：

1. 建议使用 req level 的 inflight_middle_chunks > 0 来判断跳过条件（如 all(req.inflight_middle_chunks > 0 for req in batch.reqs)），而非新增 batch level 标记。作者最终保留了 batch level 的 contains_last_prefill_chunk，认为逻辑更简洁。第二版评论中 reviewer 表示 LGTM。
2. 担心 _validate_pp_skip_output_comm 引入的校验影响 CI，建议加 PP size 判断。作者回复已通过环境变量控制，默认关闭，不影响现有测试。

- 使用 req level 的 inflight_middle_chunks 替代 batch level 标记 (design): 保留 batch level 标记，reviewer 后续评论 LGTM，认为逻辑更简洁。
- 考虑 CI 影响，建议在 batch_result_processor 中加 PP size 保护 (testing): 通过环境变量控制，默认关闭，不影响现有测试。

风险与影响

- 风险：主要风险：若 contains_last_prefill_chunk 计算错误或环境变量误开启，可能跳过实际需要输出的批次，导致推理结果错误。但已有防御：_validate_pp_skip_output_comm 在 skip=True 时断言所有 active req 的 inflight_middle_chunks > 0，确保占位输出不会被消费；同时功能默认关闭。条件还限制 batch size=1 且 return_logprob=False，比较保守。

此外，若开启后负载导致 batch size 变化，跳过条件不满足时自动 fallback 到正常通信，不影响正确性。潜在风险是未来变更可能打破条件判断与调度逻辑的耦合（如 adding mixed batch），需要持续关注。

- 影响：对用户：开启后 TTFT 在长输入（256K token）时降低最多 6.8%，TPOT 无变化（skip 仅影响 prefill）。对 H20-3E 等 SM 资源敏感硬件效果明显。未开启时行为完全不变。部署需设置 SGLANG_PP_SKIP_PURE_CHUNKED_OUTPUT_COMM=1。默认关闭，对现有系统无风险。对团队：此优化为后续 PP 通信进一步重叠或消除提供基础。
- 风险标记：环境变量默认关闭，需显式开启，条件判断正确性依赖调度逻辑，缺少新测试文件覆盖（但已有 CI 测试通过）

关联脉络

- PR #26299 [PD] Fix top logprobs crash in prefill path: 同为 PP/PD 路径的修复和优化，涉及调度与通信逻辑。
- PR #26394 [PD] Fix cross-rank queue divergence by gating metadata readiness before all-reduce: 同为 PP 跨 rank 通信修复，与本 PR 的 PP 通信优化属于同一功能线。