PR #33100 完整报告

sgl-project/sglang

[CP]: FIx some issue for glm5.2 cp v2

合并时间：2026-08-02 04:50

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/33100

执行摘要

- 一句话：修复 GLM 5.2 DSA 模型 CP-v2 三处缺陷
- 推荐动作：值得精读。对推理引擎开发者，重点看 dsa/utils.py 的 cal_padded_tokens 与 dsa_backend.py 的 _forward_trtllm：它们清楚展示了 CP-v2（interleave 分片）与 legacy CP（split-and-rebuild）在位置语义与 padding 语义上的差异，以及如何用 is_cp_v2_active 做双路径兼容。对使用 GLM 5.2 + prefill-cp 的用户，建议在 MTP 场景观察行为并关注后续修复；对 SGLang 维护者，建议为本 PR 补充针对 per_rank_actual_token 初始化顺序的单元测试。

功能与动机

PR body 为空白模板，未提供动机文字，只能从代码变更推断：GLM 5.2 属 DSA 架构且带 index_share_for_mtp_iteration 配置，此前 server_args.py 的 _handle_context_parallelism 会强制关闭 SGLANG_ENABLE_CP_V2、回退 legacy CP 路径。本 PR 修复 CP-v2 在该模型上暴露出的 ROPE 位置切分、padding token 数计算与默认开关问题，从而删除该特判。作者通过 /rerun-test 验证了 test/registered/cp/test_glm52_cp_index_share.py、test/registered/models_e2e/test_dsa_glm52_cache_layer_split.py、test/registered/cp/test_dsa_prefill_cp.py 三个 CP 专项测试。

实现拆解

修复分三步落地，分别对应三个文件的改动：

1. 启动配置：移除 GLM 5.2 的 CP-v2 禁用特判。在 python/sglang/srt/server_args.py 的 _handle_context_parallelism 中，删除针对 hf_config.index_share_for_mtp_iteration 强制关闭 SGLANG_ENABLE_CP_V2 的分支，使 GLM 5.2（DSA 架构）与其他 DSA 模型统一走『仅 interleave 策略默认启用 CP-v2』的判定逻辑。这样在用户显式设置 --enable-prefill-cp --cp-strategy interleave 时，GLM 5.2 会真正进入 CP-v2 路径而不是回退 legacy CP。
2. 注意力后端：ROPE 位置切分按 CP 版本分流。在 python/sglang/srt/layers/attention/dsa_backend.py 的 _forward_trtllm FP8 路径中，当 dsa_use_prefill_cp 为真时新增 is_cp_v2_active 判断：CP-v2 激活时使用 get_cp_strategy().shard_position_ids 按 interleave 分片策略切分 rope_positions；否则沿用 cp_split_and_rebuild_position。原因是 CP-v2 每 rank 持有的是按位置切分的 token 子集，legacy 的 split-and-rebuild 位置序列与之不兼容，会导致 ROPE 位置错位。

3. 并行工具函数：CP-v2 下跳过 legacy padding 计算。在 python/sglang/srt/layers/attention/dsa/utils.py 的 cal_padded_tokens 中新增早返回分支：如果 is_cp_v2_active(forward_batch)，直接读取 forward_batch.attn_cp_metadata.per_rank_actual_token[attn_cp_rank]；因为 CP-v2 在 ForwardBatch 准备阶段已按 rank 物理大小完成 padding，原有基于 global_num_tokens_cpu + ceil_align + DpPaddingMode 的计算会导致重复 padding 或 token 数偏差。
4. 测试配套：本 PR 未新增单元测试，依赖现有 e2e 覆盖。CI 重跑结果显示 test_glm52_cp_index_share.py、test_dsa_glm52_cache_layer_split.py、test_dsa_prefill_cp.py 全部通过（分别在 4-gpu-b200、8-gpu-b200、8-gpu-h200），而 Fridge003 重跑的 test_dsa_glm52_tp_mtp.py 与 test_dsa_glm52_dp_mtp.py 在 8-gpu-h200 上失败，失败原因未在评论中说明。

关键文件：

- python/sglang/srt/server_args.py（模块 启动配置；类别 source；类型 core-logic；符号 _handle_context_parallelism）：启动配置核心：删除 GLM 5.2（index-share MTP）强制禁用 CP-v2 的特判，使 DSA 模型统一按 interleave 策略决定默认是否启用 CP-v2，直接改变 GLM 5.2 的 CP 路径选择。
- python/sglang/srt/layers/attention/dsa_backend.py（模块 注意力后端；类别 source；类型 core-logic；符号 _forward_trtllm）：注意力后端核心：FP8 decode 路径的 ROPE 位置切分按 CP-v2 是否激活分流，修复 CP-v2 interleave 下位置编码错位问题，影响所有 DSA + FP8 + prefill-cp 模型。
- python/sglang/srt/layers/attention/dsa/utils.py（模块 并行工具；类别 source；类型 dependency-wiring；符号 cal_padded_tokens）：并行工具函数：cal_padded_tokens 为 CP-v2 新增早返回分支，直接使用 per-rank 实际 token 数，避免 legacy 全局 padding 计算在 CP-v2 下重复 padding 或算错 token 数。

关键符号：_handle_context_parallelism, _forward_trtllm, cal_padded_tokens

关键源码片段

python/sglang/srt/server_args.py

启动配置核心：删除 GLM 5.2（index-share MTP）强制禁用 CP-v2 的特判，使 DSA 模型统一按 interleave 策略决定默认是否启用 CP-v2，直接改变 GLM 5.2 的 CP 路径选择。

```
# ServerArgs._handle_context_parallelism 节选：决定 DSA 类模型默认是否启用 CP-v2。
# 本 PR 移除了对 GLM 5.2（index-share MTP）单独禁用 CP-v2 的特判分支。
if model_arch in CP_V2_DEFAULT_MODEL_CLASSES:
    is_dsa_default_model = is_deepseek_dsa(hf_config)
    # DSA 的 CP-v2 目前只支持 interleave 策略；
    # 非 DSA 模型不受此限制，直接默认启用。
    enable_default_cp_v2 = not is_dsa_default_model or (
        self.enable_prefill_cp and self.cp_strategy == 'interleave'
    )
    if enable_default_cp_v2 and not envs.SGLANG_ENABLE_CP_V2.is_set():
        envs.SGLANG_ENABLE_CP_V2.set(True)
```

```
if (
    self.enable_prefill_cp
    and model_arch in ('MiMoV2ForCausalLM', 'MiMoV2FlashForCausalLM')
    and envs.SGLANG_ENABLE_CP_V2.get()
):
    # MiMo V2 的 CP-v2 只支持 zigzag 策略与纯文本推理，启动期直接报错拦截。
    if self.cp_strategy != 'zigzag':
        raise ValueError('MiMo V2 CP-v2 only supports --cp-strategy zigzag.')
    if model_config.is_multimodal and not self.language_only:
        raise ValueError(
            'MiMo V2 CP-v2 only supports text inference; add --language-only.'
        )
```

python/sglang/srt/layers/attention/dsa_backend.py

注意力后端核心：FP8 decode 路径的 ROPE 位置切分按 CP-v2 是否激活分流，修复 CP-v2 interleave 下位置编码错位问题，影响所有 DSA + FP8 + prefill-cp 模型。

```
# _forward_trtllm 节选：FP8 decode 路径下 ROPE 位置切分必须与当前 CP 实现一致。
rope_positions = forward_batch.positions
if dsa_use_prefill_cp(forward_batch):
    if is_cp_v2_active(forward_batch):
        # CP-v2 interleave：各 rank 持有按位置切分的 token 子集，
        # 直接用 CP 策略的 shard_position_ids 切分 ROPE 位置。
        rope_positions = get_cp_strategy().shard_position_ids(
            rope_positions, forward_batch
        )
    else:
        # legacy CP：先按 split-and-rebuild 方式重建位置序列再切分。
        rope_positions = cp_split_and_rebuild_position(
            forward_batch, rope_positions
        )

# 切分后的 ROPE 位置与 query / key 一起做 FP8 量化与 ROPE。
q, k, k_rope = mla_quantize_and_rope_for_fp8(
    q,
    q_rope,
    k.squeeze(1),
    k_rope.squeeze(1),
    rope_positions,
    cos_sin_cache,
    is_neox,
    self.kv_lora_rank,
    self.qk_rope_head_dim,
)
```

python/sglang/srt/layers/attention/dsa/utils.py

并行工具函数：cal_padded_tokens 为 CP-v2 新增早返回分支，直接使用 per-rank 实际 token 数，避免 legacy 全局 padding 计算在 CP-v2 下重复 padding 或算错 token 数。

```
def cal_padded_tokens(forward_batch: 'ForwardBatch'):
    # 与 ForwardBatch.prepare_mlp_sync_batch 的 padding 逻辑保持一致,
    # 计算 attn_tp_size > 1 或 MAX_LEN padding 模式下的实际 token 数。
    from sglang.srt.layers.cp.padding import get_cp_padding_align_size
    from sglang.srt.layers.cp.utils import is_cp_v2_active

    # CP-v2 下每个 rank 的局部 shard 已按物理大小提前 pad 好,
    # 直接使用 metadata 中记录的 per-rank 实际 token 数即可。
    if is_cp_v2_active(forward_batch):
        return forward_batch.attn_cp_metadata.per_rank_actual_token[
            get_parallel().attn_cp_rank
        ]

    # legacy CP 路径: 基于全局 token 数按 CP 对齐做 ceil_align,
    # 并复用 DP buffer 准备时选择的 padding 模式。
    global_num_tokens = forward_batch.global_num_tokens_cpu.copy()
    sync_group_size = len(global_num_tokens)
    attn_cp_size = get_parallel().attn_cp_size
    cp_align_size = get_cp_padding_align_size()
    for i in range(sync_group_size):
        global_num_tokens[i] = ceil_align(global_num_tokens[i], cp_align_size)
    dp_padding_mode = forward_batch.dp_padding_mode
    if dp_padding_mode is None:
        dp_padding_mode = DpPaddingMode.get_dp_padding_mode(
            forward_batch.is_extend_in_batch, global_num_tokens
        )
    if dp_padding_mode.is_max_len():
        tokens = max(global_num_tokens)
    elif len(global_num_tokens) > 1:
        tokens = global_num_tokens[get_parallel().attn_dp_rank]
    else:
        tokens = global_num_tokens[0]
    if can_dsa_prefill_cp_round_robin_split(forward_batch):
        tokens = ceil_div(tokens, attn_cp_size)
    return tokens
```

评论区精华

本 PR 没有正式的 review 评论（review_comments_count = 0），Fridge003 直接 approve。可见的讨论都发生在 CI 重跑流程中：

- hzh0425 发起 /rerun-test 运行 test/registered/cp/test_glm52_cp_index_share.py、test/registered/models_e2e/test_dsa_glm52_cache_layer_split.py、test/registered/cp/test_dsa_prefill_cp.py，三个任务分别在 4-gpu-b200 / 8-gpu-b200 / 8-gpu-h200 通过。
- Fridge003 发起 /rerun-test 运行 test_dsa_glm52_tp_mtp.py 与 test_dsa_glm52_dp_mtp.py，在 8-gpu-h200 上均失败；作者 hzh0425 回复了截图，但未留下文字说明，随后 PR 被批准合并。 结论：CP 专项测试通过是合入的主要依据；MTP

相关失败未在合并前闭环，属于遗留风险。

- CP-v2 专项测试重跑验证 (testing): 三个 CP 专项测试通过，成为合入依据之一。
- GLM 5.2 的 MTP 测试失败未闭环 (testing): MTP 相关失败未在本 PR 内解释或修复，属于遗留风险；可能与本 PR 修改的 prefill 路径关联较弱，但需跟踪后续 PR。

风险与影响

- 风险:
  1. 默认行为变更风险（server_args.py）：删除 index_share_for_mtp_iteration 特判后，GLM 5.2 在启用 --enable-prefill-cp --cp-strategy interleave 时会默认进入 CP-v2，不再自动回退 legacy CP。若该模型组合仍存在未覆盖场景（MTP 测试失败即为信号），用户侧行为将直接变化，且缺少显式逃生开关。
  2. 精度风险（dsa_backend.py）：改动影响所有 DSA + FP8 + prefill-cp 模型的 decode 路径。shard_position_ids 与 mla_quantize_and_rope_for_fp8 的 ROPE 语义必须完全一致，否则会出现位置编码错位，且长序列下更容易暴露。
  3. 依赖顺序风险（dsa/utils.py）：cal_padded_tokens 的 CP-v2 早返回依赖 forward_batch.attn_cp_metadata.per_rank_actual_token 已正确填充；若某条调用路径在 metadata 初始化前进入该函数，会数组越界或返回错误 token 数。同时该路径绕过了 DpPaddingMode 分支，需确认所有调用方都满足 CP-v2 前置条件。
  4. 测试缺口：没有新增单元测试，仅依赖 e2e；MTP 相关两个测试失败且未修复。
- 影响:
  - 模型支持面：GLM 5.2（DSA + index-share MTP）在 interleave 策略下正式启用 CP-v2，长 prefill 场景可从 CP-v2 的切分与 padding 优化中受益；但 tp_mtp / dp_mtp 场景仍不稳定。
  - 系统架构：DSA 后端的 FP8 decode 与 token 数计算完成 CP-v2 适配，为 DeepSeek DSA 系列模型统一接入 CP-v2 扫清障碍，也意味着 CP-v2 与 legacy CP 双路径将长期并存。
  - 团队与流程：合入者与作者通过 CI 重跑完成验证，未走强 review；MTP 失败未闭环的合入模式提示后续需要跟进修复 PR。
  - 风险标记：CP-v2 默认启用面扩大，DSA FP8 路径精度风险，MTP 测试失败未闭环，无新增单测，依赖 metadata 初始化顺序

关联脉络

- PR #33012 runtime_context: record the publishing process role: 同改 python/sglang/srt/server_args.py（CP 相关配置的进程角色发布迁移），反映 server_args 中上下文并行开关逻辑正随配置框架演进。
- PR #31987 [BCG][3/N] Enable bcg on dsa & deepep a2a backend: 同属 DSA 模型在 prefill 阶段的性能特性扩展线（BCG），与本 PR 的 dsa_backend 改动目标一致，都属于让 DSA 在 CP / 并行场景下更高效。
- PR #33168 Fix the chunked-prefix-cache gate writing config the backends never read: 同类『配置开关与实际后端行为不一致』的 bugfix，反映仓库近期在系统性对齐配置

gate 与后端实现。