PR #35064 完整报告

sgl-project/sglang

docs: fix Qwen3.8-27B mamba ratio calculator for speculative decoding

合并时间：2026-08-17 13:54

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/35064

执行摘要

本 PR 修复了 Qwen3.8-27B cookbook 页面内嵌 mamba 缓存比例计算器（docs/src/snippets/_qwen38_mamba_ratio_calculator.jsx）在推算 --mamba-full-memory-ratio 与 --max-mamba-cache-size 时的 4 个行为偏差。该计算器派生自 Kimi K3 版本时丢失了 ReplaySSM、DSPARK、pin 公式与 S 槽位调制 4 项语义，导致所有 spec 配置下输出错误参数（RTX 5090 实测 ReplaySSM 场景 KV 池缩小 38%）。改动仅限文档站点的 2 个文件，不涉及服务端代码，但直接影响用户从页面复制部署参数的准确性。

功能与动机

PR body 明确指出计算器 was derived from the Kimi K3 one but lost four behaviours along the way, so it emitted the wrong --mamba-full-memory-ratio and --max-mamba-cache-size for every speculative configuration。四个缺陷分别是：

1. 忽略 --enable-linear-replayssm-spec：ReplaySSM 把验证中间态放到固定 ring 上，D 应为 0，旧代码却按 D=4 计算，使 r 从真实的 1.04 变成 2.08，状态池 2 倍过度预留，挤压 KV 池。
2. DSPARK 草稿数计算错误：DSPARK 不产生 --speculative-num-draft-tokens，验证窗口应为 --speculative-dspark-block-size（gamma）+ 1，旧代码 Number(null) || 4 得到 4，而实际是 8，导致状态池欠配。
3. pin 重复计入 D：引擎只按 S 划分主状态池并单独分配 spec 缓冲，pin 应为 C * S。
4. S 硬编码 5/4/3/1：忽略 SGLANG_OPT_MAMBA_SKIP_DECODE_LOCK 与 overlap/--pp-size 调制。

PR body 特别指出缺陷 1 与 2 误差方向相反，which is why no single spot-check surfaced them，这也是此前抽查未能发现问题的主要原因。

实现拆解

1. 定位问题入口：核心改动集中在 _qwen38_mamba_ratio_calculator.jsx 的 derive() 函数，它是页面实时计算 --mamba-full-memory-ratio 与等价 pin 值的核心逻辑，输出会直接进入用户复制的部署命令。
2. 重写 S 槽位计算：将硬编码 5/4/3/1 改为镜像服务端 kv_cache_configurator._calculate_mamba_ratio() 的表达式——基线 3，SGLANG_OPT_MAMBA_SKIP_DECODE_LOCK=1 时减 1，叠加 ping-pong track buffer（overlap 调度开启时 +2，extra_buffer_lazy 或 overlap 关闭时 +1），no_buffer 恒为 3，radix cache 禁用时 S 退化为 1。derive() 签名

新增 env 参数以读取环境变量，调用处 eff 与 bs 分别传入所属命令的 env，避免 base 命令继承 Playground 的 overlay env。

3. 重写 D 槽位计算：按 spec 算法分派——--enable-linear-replayssm-spec 时 D 恒为 0（验证中间态移入固定 ring）；DSPARK 时 D = --speculative-dspark-block-size + 1，缺省 fallback 为 7（RadixArk/Qwen3.8-27B-DSpark 默认 checkpoint 的 block_size）；其他 EAGLE/MTP 场景沿用 --speculative-num-draft-tokens（缺省 4）。
4. 修正 pin 公式与 env 传递：pin 从 C * (S + D) 改为 C * S，依据是引擎 kv_cache_configurator.py 中 mamba_cap = max_mamba_cache_size // _calculate_mamba_ratio() 只按 S 划分主状态池、spec 缓冲独立分配；base 命令 env 选择改为 cfg.baseFlags.length ? cfg.baseEnv : cfg.env，与 K3 计算器语义一致。
5. 配套文档与验证：同步更新 Qwen3.8-27B.mdx 中 Accordion 的 S/D 语义、pin 公式与 DSPARK/ReplaySSM 说明，并重写 jsx 注释只描述当前语义。作者在 RTX 5090 上复现了 10 个 servable 配置的比率，并验证 no_buffer、radix-off、skip-lock、overlap-off 与显式 --speculative-dspark-block-size 7 各分支。

docs/src/snippets/_qwen38_mamba_ratio_calculator.jsx

核心改动文件，重写 derive() 的 S 与 D 计算、修正 pin 公式并新增 env 参数传递，直接影响用户从页面复制的部署参数准确性。

```
// 核心：derive() 从配置 flag 列表与运行环境推导缓存比例，语义对齐服务端
// kv_cache_configurator._calculate_mamba_ratio() 与 DSpark/EAGLE 的 spec 行为。
const derive = (flags, env) => {
  const flagArg = (name) => {
    for (const f of flags) {
      const parts = f.split(/\s+/);
      if (parts[0] === name) return parts[1];
    }
    return null;
  };
  const hasFlag = (name) => flags.some((f) => f.split(/[\s=]/)[0] === name);

  // S = 每请求占用主状态池的槽位数（镜像引擎实现）：
  // 基线 3；SGLANG_OPT_MAMBA_SKIP_DECODE_LOCK=1 时减 1；
  // 叠加 ping-pong track buffer：overlap 调度开启时为 2，
  // extra_buffer_lazy 或 overlap 关闭时为 1，no_buffer 恒为 3，
  // radix cache 禁用时 S 退化为 1。默认组合恰好等于旧硬编码 5/4/3/1。
  const skipLock = env.some((e) =>
    e.startsWith("SGLANG_OPT_MAMBA_SKIP_DECODE_LOCK=1"));
  const overlapOff =
    hasFlag("--disable-overlap-schedule") ||
    (Number(flagArg("--pp-size")) || 1) > 1;
  const slots = radixOff
    ? 1
    : strategy === "no_buffer"
      ? 3
      : 3 - (skipLock ? 1 : 0) +
```

```
        (overlapOff || strategy === "extra_buffer_lazy" ? 1 : 2);

  // D = 每条请求的 spec 验证中间状态数（不计入主池 pin）。
  const algo = (flagArg("--speculative-algorithm") || "").toUpperCase();
  // ReplaySSM 把验证中间态移到固定 ring 上，D 恒为 0。
  const replaySpec = hasFlag("--enable-linear-replayssm-spec");
  // DSPARK 不读 --speculative-num-draft-tokens：窗口 = gamma + 1，
  // gamma 缺省时为 draft checkpoint 的 block_size（RadixArk 默认 7）。
  const dsparkBlock = Number(flagArg("--speculative-dspark-block-size")) || 7;
  const drafts = !specOn || replaySpec
    ? 0
    : algo === "DSPARK"
      ? dsparkBlock + 1
      : Number(flagArg("--speculative-num-draft-tokens")) || 4;

  // 引擎只用 S 划分状态池：
  // mamba_cap = max_mamba_cache_size // _calculate_mamba_ratio()，
  // spec 缓冲独立分配，所以 pin = C x S 而非 C x (S + D)。
  const ratio = ((slots + drafts) * stateBytesPerSlot) / (kvBytesPerToken * L);
  return { ratio, slots, drafts, specOn /* 其余派生几何常量省略 */ };
};

// 两处调用：原命令与 base 命令各自携带自己的 env，
// base 命令的 env 不再继承 Playground overlay，语义与 K3 计算器一致。
const eff = derive(cfg.flags, cfg.env);
const bs = derive(
  cfg.baseFlags.length ? cfg.baseFlags : cfg.flags,
  cfg.baseFlags.length ? cfg.baseEnv : cfg.env,
);
const pin = Math.ceil(C * slots);
```

评论区精华

> "If we fallback to the default DSpark checkpoint shouldn't it be 7 here?" —— zijiexia 指向默认 checkpoint 的 config.json block_size，Jiminator 以 checkpoint 配置、服务端 boot log 与 K3 一致性三重确认后修正为 7。
>
> "There's no observable difference today — but the semantics are wrong, and this is the one line where the body claims K3 parity." —— zijiexia 抓住 env 选择逻辑与 K3 的唯一分歧点，Jiminator 随即修正。
>
> "Can you also update the note in the mdx about how to calculating --max-mamba-cache-size? And please clean up the comments, some seems not accurate." —— zijiexia 的总体意见，推动文档与代码同步收敛。

风险与影响

风险：本次改动仅涉及文档站点文件，无运行时风险，但存在一致性维护风险——derive() 中的 S 表达式是服务端 _calculate_mamba_ratio() 的手工镜像，DSPARK 的 block_size=7

fallback 是硬编码默认值，若服务端逻辑或未来 checkpoint 的 block_size 变化，计算器需人工同步；当前无自动化测试或跨文件校验防止再次漂移。

影响：修复前用户从页面复制的参数在 spec 场景存在 2 倍偏差——ReplaySSM 场景 KV 池收缩 38%（19,469 → 11,992 tokens），DSPARK 场景状态池欠配（r=2.08 对 3.12）。修复后文档推荐配置与引擎实际划分一致，对长上下文与并发 >1 的部署尤其重要。团队维护成本体现为 jsx、mdx、服务端三处需保持一致。

关联脉络

该 PR 是 speculative-decoding 功能线在文档配套侧的收敛：上游 #34696 为 DSpark 增加 logprobs 支持、#35058 简化 spec logprob 接口，本 PR 则让 Qwen3.8-27B 的部署计算器与 DSpark/EAGLE worker 的实际行为（gamma 窗口、验证缓冲分配）对齐。其方法与 Kimi K3 计算器保持 parity 的约定，意味着未来新模型派生计算器时应以 K3 为基线并逐项核对 S/D 语义，避免再次出现同类丢失。