

PR #39213 完整报告

sgl-project/sglang

[Docs] GLM-5.3-Flash cookbook: fixed MTP 5/1/6, EP1 + flashinfer_trtllm on Blackwell

合并时间: 2026-09-13 03:56

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/39213>

执行摘要

本 PR 依据 4x GB300 对拍 vLLM 的 benchmark campaign (运行目录 20260911_glm53flash_vs_vllm) 全面翻新 GLM-5.3-Flash cookbook 的部署配方: Low Latency 策略从 adaptive MTP 改为固定深度 MTP 5/1/6 加自然接受 (chat C80 从 4293 提升到 8976 tok/s), Blackwell 通道删除显式 `--ep-size` 并切换 `--moe-runner-backend` 为 `flashinfer_trtllm`, Hopper 保留 EP8 + `deep_gemm`; 同时替换 gb300 的 FP8 + TRT-LLM 实测数据, 未复测单元格统一标记 in-progress 并加 TODO。PR 只改文档与配置片段 (3 个文件, +111/-73), 不触碰运行时代码, 但会直接改变用户生成的部署命令。

功能与动机

PR body 明确说明三个动因:

The adaptive policy caps draft steps to 3 at batch 8-16 and disables drafting at batch ≥ 64 ; fixed MTP 5/1/6 recovers most of the gap (chat C80 4293 $\rightarrow$ 8976 tok/s on stock source).

即 adaptive MTP 在高并发 batch 下投机深度被严重压缩 (batch 8-16 压到 3 步, batch ≥ 64 直接关闭 drafting), 导致低延迟场景吞吐明显落后; 固定深度可恢复大部分差距。第二个动因是 MoE 后端选择: trtllm-gen MoE kernels 仅 SM100 可用, 所以 Blackwell 用 `flashinfer_trtllm`、Hopper 只能保留 `deep_gemm`。第三个动因是文档准确性: 删除显式 `--ep-size` (EP1 本就是 sglang 默认), 并让未用新命令复测的 benchmark 行如实标注状态。

实现拆解

- 投机策略收敛为固定深度 MTP 5/1/6: docs/src/snippets/configs/zai-org/glm-5.3-flash.jsx 中所有 Low Latency 单元格与 EAGLE preset 统一删除 `--speculative-adaptive`, 保留 EAGLE + 5 steps + top-k 1 + 6 draft tokens; Strategy 维度副标题从 "Adaptive MTP 5/1/6" 改为 "MTP 5/1/6", DP-Attention 禁用文案同步去掉 adaptive 表述。这是本次性能结论的核心载体。
- Blackwell 通道切换 MoE 后端与 EP 默认值: gb300/gb200/b200/b300 单元格删除 `--ep-size 4` (EP1 是 sglang 默认, 依据 python/sglang/srt/arg_groups/fields/parallel.py), `--moe-runner-backend` 从 `deep_gemm` 改为 `flashinfer_trtllm`; PR body 记录了可整除性校验 (moe_intermediate_size 2048 在 TP4/TP8 下为 512/256, 均整除 128)。h100/h200 保留 EP8 + `deep_gemm`, 并新增注释说明 SM100-only 约束。

3. gb300 基准行替换与状态标记: glm-5.3-flash-benchmarks.jsx 仅 fp8-trtllm Low Latency 一行替换为 campaign 实测 (sglang b3dc0388ed、FlashInfer 0.6.18、chat 1000/1000 与 summarization 8000/1000、并发 1/16/80、自然接受、accept length 5.98/4.7) ; 其余旧命令行保留但每行上方加 TODO 注释; **verificationStatus** 重写为三支, 仅新的 fp8-trtllm + dcp off 组合为 verified。
4. 文档正文同步: GLM-5.3-Flash.mdx 更新策略描述 (固定 5 steps、top-k 1、6 draft tokens、自然接受)、DCP 验证范围 (旧配方实测、待复测) 与 encoder disaggregation 示例 (语言服务进程改用 **flashinfer_trtllm**) , 并在 encoder 段加 TODO 注明该组合尚未用新命令复验。

配套验证: **mint validate**、**mint broken-links**、pre-commit 与 jsx module 检查全部通过; campaign 的 GSM8K 自然停止门禁 (11 条 lane) 给出 90.75-92.72% 且全部自然停止。

docs/src/snippets/configs/zai-org/glm-5.3-flash.jsx

配置面板源码, 承载本次核心变更: 删除 --speculative-adaptive 与显式 --ep-size, Blackwell 切换 --moe-runner-backend flashinfer_trtllm, 重写 verificationStatus 三态逻辑, 直接影响用户生成的部署命令。

```
// ----- Speculative 卡片: Low Latency 从 adaptive MTP 改为固定深度 MTP 5/1/6 -----
// 2026-09 对拍 campaign 实测: adaptive 策略在 batch 8-16 会把 draft steps 压到 3,
// batch >= 64 时直接关闭 drafting, 导致低延迟高并发吞吐明显落后;
// 固定 5 steps / top-k 1 / 6 draft tokens + 自然接受可恢复大部分差距
// (chat c80 从 4293 tok/s 提升到 8976 tok/s, 见 gb300 基准行) 。
speculative: {
  options: [
    { id: "current", label: "Inherited from base" },
    { id: "off", label: "Off (greedy)" },
    {
      id: "eagle",
      label: "EAGLE / MTP 5-1-6",
      flags: [
        "--speculative-algorithm EAGLE", // NEXTN 老拼写已并入 EAGLE
        "--speculative-num-steps 5", // 固定深度, 不再随 batch 自适应
        "--speculative-eagle-topk 1",
        "--speculative-num-draft-tokens 6",
        // --speculative-adaptive 已移除: 接受长度不再被 SGLANG_SIMULATE_ACC_LEN 模拟钉住
      ],
      disable: [
        {
          when: { dpAttnOn: [true] },
          reason: "MTP speculative decoding with DP-Attention is not validated on this model.",
        },
      ],
    },
  ],
},
```

```
// ----- Blackwell Low Latency 单元格 (以 gb300 为例) -----
```

```

// 删除显式 --ep-size: EP1 是 sglang 默认 (见 parallel.py) ,
// 并切换 --moe-runner-backend flashinfer_trtllm;
// trtllm-gen MoE kernel 仅 SM100 可用, 因此 Hopper 保留 EP8 + deep_gemm。
{
  match: { hw: "gb300", strategy: "low-latency", quant: "fp8" },
  nnodes: 1,
  verified: true,
  verificationStatus: (s) =>
    s.bcg !== "off" || s.mmTransport !== "auto" || s.hicache !== "off" ? "unverified" :
    s.kvDsaPair === "fp8-trtllm" && s.dcp === "off" ? "verified" :
    ["bf16-tilelang", "fp8-trtllm"].includes(s.kvDsaPair) && ["off", "4"].includes(s.dcp)
    ? "in-progress" : "unverified",
  flags: [
    "--model-path {{MODEL_NAME}}",
    "--tp-size 4",
    "--dsa-prefill-backend trtllm",
    "--dsa-decode-backend trtllm",
    "--kv-cache-dtype fp8_e4m3",
    "--moe-runner-backend flashinfer_trtllm",
    "--speculative-algorithm EAGLE",
    "--speculative-num-steps 5",
    "--speculative-eagle-topk 1",
    "--speculative-num-draft-tokens 6",
    "--reasoning-parser glm45",
    "--tool-call-parser glm47",
  ],
}

```

docs/src/snippets/configs/zai-org/glm-5.3-flash-benchmarks.jsx

基准数据表: gb300 的 FP8 + TRT-LLM Low Latency 行替换为 2026-09 campaign 固定 MTP 实测 (six workload rows、自然接受、accept length 5.98/4.7) , 其余旧命令数据加 TODO 并保留, 是性能结论的量化证据。

```

// gb300 Low Latency FP8 KV + TRT-LLM DSA 行: 2026-09 campaign 固定 MTP 实测
// 运行环境: sglang main b3dc0388ed + FlashInfer 0.6.18, TP4 / EP1,
// flashinfer_trtllm MoE, 固定 MTP 5/1/6 自然接受 (不再用模拟 accept length) 。
// 协议: 每行 80 个随机 token 请求, streaming /v1/completions, ignore_eos,
// 温度 0、seed 42、prefix cache 关闭; chat 为 1000/1000, summarization 为 8000/1000,
// --context-length 16384、--max-running-requests 80、--mem-fraction-static 0.90。
{
  match: { hw: "gb300", strategy: "low-latency", kvDsaPair: "fp8-trtllm", quant: "fp8" },
  sglang_version: "b3dc0388ed",
  latencyPercentile: "Mean",
  speed: [
    // 并发 1 为单次运行, c16/c80 为两次运行中位数
    { workload: { dataset: "random", isl: 1000, osl: 1000, max_concurrency: 1, num_prompts: 80 }
    ,
      ttft_ms: 204.79, tpot_ms: 2.0, tokens_per_sec_per_gpu: 227.06 },
    { workload: { dataset: "random", isl: 1000, osl: 1000, max_concurrency: 16, num_prompts:

```

```

80 },
  ttft_ms: 367.48, tpot_ms: 4.8, tokens_per_sec_per_gpu: 1492.93 },
  { workload: { dataset: "random", isl: 1000, osl: 1000, max_concurrency: 80, num_prompts:
80 },
    ttft_ms: 1243.26, tpot_ms: 6.28, tokens_per_sec_per_gpu: 4487.92 },
  // summarization (8000/1000) 行: c1/c16/c80 的 tpot_ms 分别为 1.95/5.16/11.65
],
notes: "chat c80 聚合输出 8975.84 tok/s、accept length 4.7; 同 campaign 中固定深度 MTP
在旧命令上为 8490.07 tok/s, 而 adaptive 仅 3664.53 tok/s, EP1 + flashinfer_trtllm
又在固定深度基础上提升 4.4-13.6%。",
}

```

评论区精华

本 PR 没有实质 review 讨论: reviewer zijiexia 直接 APPROVED (空评论体), 唯一自动化反馈是 mintlify[bot] 的 preview deployment 提示。技术结论全部沉淀在 PR body 与代码注释中, 其中最有价值的两条是:

- adaptive MTP 在 batch 8-16 把 draft steps 压到 3、batch ≥ 64 完全关闭 drafting, 是本次翻新的直接动因;
- trtllm-gen MoE kernels 为 SM100-only, 因此 Hopper 无法跟随 Blackwell 切换到 flashinfer_trtllm。

风险与影响

- 数据新旧混排: benchmarks.jsx 除 fp8-trtllm Low Latency 行外, 其余行 (DCP4、NVFP4、H100、B200/B300 等) 仍是旧命令 (TP4/EP4 deep_gemm + adaptive MTP) 数据, 靠 TODO/in-progress 标记区分; 用户横向对比新旧数字时存在误读风险。
- 后端配置无自动化验证: flashinfer_trtllm 属运行时后端参数, 文档侧只有静态检查 (mint validate、broken-links), 无法验证其在非 SM100 或特定 EP 组合下的行为; 若该后端存在未覆盖的退化路径, 文档会带偏用户部署。
- encoder disaggregation 未复验: mdx 已把语言服务进程示例改为 flashinfer_trtllm, 但明确标注 238,080 visual-token 场景用的是旧命令, 且运营备注说明该后端在 PD 场景下未测试。
- 影响面: 所有通过 cookbook 部署 GLM-5.3-Flash 的用户会获得新默认命令; 低延迟 chat c80 基准吞吐提升约 2.1 倍, 但实际收益依 workload 而定 (adaptive 在高 batch 压步数的场景收益最大)。

关联脉络

- 同仓库近期 #39190 (Kimi-K3 cookbook) 与 #39104 (AMD MI355X cookbook) 是同一 cookbook 文档线的平行更新, 本 PR 延续了「按实测 campaign 翻新配方 + 标记验证状态」的维护模式。
- 39165 将 `--dcp-comm-backend` 默认解析为 `fi_a2a/a2a`, 本 PR 中 DCP4 单元格仍显式写 `--dcp-comm-backend a2a`, 两个 PR 共同推进 DCP 配置语义收敛。

- 文档正文引用的 #36507 (GLM-5.3-Flash 支持分支) 与 #36708 (hidden-state capture) 表明该模型的部分能力仍在分支上、未完全合入 main; 本 PR 的 DFlash2 段落继续依赖这两个 PR, 后续 main 合并后需同步清理 note。
- 整体看, GLM-5.3-Flash 的推理支持 (MTP 投机、trtllm MoE、DSA) 正在通过 cookbook 先行固化最佳实践, 属于「文档先行、runtime 跟进」的演进节奏。