PR #20887 完整报告

sgl-project/sglang

CUTLASS FP8 Blockwise GEMM improvement of SM120

合并时间：2026-03-22 17:55

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/20887

执行摘要

本 PR 优化了 SM120 GPU 上的 FP8 块状 GEMM 内核，通过引入 pingpong schedule 并基于 M 大小动态选择，显著提升了小 M 场景下的性能，推理速度提升约 53%，同时保持了准确性。

功能与动机

原 SM120 fp8 blockwise GEMM kernel 使用 KernelScheduleAuto，在 SM120 上仅选择 *cooperative* schedule，导致小 M 时性能不足。pingpong schedule 相比 *cooperative* 对于小 M 快约 2 倍，因此本 PR 旨在利用这一性能机会，提升整体效率。

实现拆解

关键改动集中在 sgl-kernel/csrc/gemm/fp8_blockwise_gemm_kernel.cu 文件的 launch_sm120_fp8_blockwise_scaled_mm 函数：

- 添加 kCanUsePingpong 常量检查。
- 根据 M 大小（M≤64）选择 pingpong 路径使用 KernelTmaWarpSpecializedBlockwisePingpongSm120，否则使用 *cooperative* 路径。
- 重构 kernel setup 代码，提高可读性和维护性。

代码示例：

```
constexpr bool kCanUsePingpong = (64 % ScaleGranularityM == 0);
int m = a.size(0);
// ... 基于m选择schedule
```

评论区精华

review 过程中无实质性技术讨论，仅由 BBuf 批准。作者在 issue 评论中提供了 NCU 性能报告，进一步验证了变更的有效性，例如 pingpong schedule 相比 *cooperative* schedule 性能提升约一倍。

风险与影响

风险：

- 对于 M>64 使用 *cooperative* 路径以避免 CUTLASS 库问题，但可能存在未解决的准确性隐患。
- 新添加的 pingpong 路径可能引入回归或兼容性问题，尤其是在边缘 *case* 或不同硬件配置下。
- 代码变更集中在核心路径，测试覆盖可能不足。

影响：

- 用户受益：在 SM120 GPU 上运行小 M 形状的 FP8 GEMM 时，推理速度显著提升，基准测试显示速度从 34.14 token/s 提升至 52.11 token/s。
- 系统层面：优化了内核调度策略，可能减少延迟并提高资源利用率。

关联脉络

本 PR 性能测试中与 FlashInfer 比较，关联到 PR #20214（添加 FlashInfer 集成），表明团队在持续优化性能并集成第三方库。此外，无直接关联 Issue，但体现了对硬件特定调优的关注。