PR #27468 完整报告

sgl-project/sglang

dflash piecewise cuda graphs support

合并时间：2026-06-10 06:44

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/27468

执行摘要

- 一句话：DFLASH 推测解码支持分段 CUDA 图
- 推荐动作：建议合并。该 PR 代码简洁，改动量小（+55/-4），有明确的动机和测试覆盖。值得关注的设计决策是在初始化阶段提前决定 capture_hidden_mode，而不是在运行时动态判断，保证了 capture 和 replay 的一致性。

功能与动机

DFLASH 推测解码在 prefill/extend 阶段需要目标的辅助隐藏状态（aux hidden states）来物化草稿 KV 缓存。原有的分段 CUDA 图不支持此场景，导致 DFLASH 无法与 PCG 共存。该 PR 从 #23000 拆分出来，专门解决此兼容性问题。

实现拆解

1. 判断 DFLASH 算法并启用 FULL 捕获：在 piecewise_cuda_graph_runner.py 的初始化方法中，原有逻辑仅在 enable_return_hidden_states 为 True 时设置 capture_hidden_mode = CaptureHiddenMode.FULL。修改后，当 model_runner.spec_algorithm.is_dflash() 返回 True 时也启用 FULL 模式，从而在 capture 阶段就生成完整的隐藏状态。
2. 传递 capture_hidden_mode 到 capture 和 replay 调用：原先 _capture 和 _replay 方法中硬编码 capture_hidden_mode=CaptureHiddenMode.NULL，这会导致即使初始化时设置了 FULL，实际执行时仍使用 NULL。修改后使用 self.capture_hidden_mode 替代，确保 capture 和 replay 的行为一致。
3. 新增集成测试：创建 test_pcg_with_speculative_decoding_dflash.py，继承 PCGSpecBase 和 CustomTestCase，配置 Llama-3.1-8B-Instruct 作为目标模型，DFLASH 作为推测算法，启用强制 PCG，并验证 accuracy_threshold >= 0.75 和 speedup_threshold >= 2.8。

关键文件：

- python/sglang/srt/model_executor/piecewise_cuda_graph_runner.py（模块 模型执行器；类别 source；类型 data-contract）：核心改动文件：在 __init__中增加 DFLASH 检测以启用 FULL 捕获模式，并在 _capture 和 _replay 中传递正确的 capture_hidden_mode。
- test/registered/piecewise_cuda_graph/test_pcg_with_speculative_decoding_dflash.py（模块 分段图测试；类别 test；类型 test-coverage；符号 TestPCGWithDFlash）：新增

集成测试，验证 DFLASH + PCG 组合的正确性和性能。

关键符号：init, _capture, _replay, is_dflash

关键源码片段

test/registered/piecewise_cuda_graph/test_pcg_with_speculative_decoding_dflash.py

新增集成测试，验证 DFLASH + PCG 组合的正确性和性能。

```
"""Test piecewise CUDA graph coexisting with speculative decoding (DFLASH).

PCG 处理 prefill/extend 路径，而 DFLASH 需要从 prefill 获取目标的辅助隐藏状态
来物化草稿 KV 缓存。该测试验证 PCG 在启用 DFLASH 隐藏状态变体时能正确捕获该路径。
"""

import unittest

from sglang.test.ci.ci_register import register_cuda_ci
from sglang.test.server_fixtures.pcg_spec_fixture import PCGSpecBase
from sglang.test.test_utils import (
    DEFAULT_DRAFT_MODEL_DFLASH,
    DEFAULT_TARGET_MODEL_DFLASH,
    CustomTestCase,
)

register_cuda_ci(est_time=531, stage="base-b", runner_config="1-gpu-small")


class TestPCGWithDFlash(PCGSpecBase, CustomTestCase):
    """PCG + DFLASH 在 Llama-3.1-8B-Instruct 上的测试。"""

    model = DEFAULT_TARGET_MODEL_DFLASH
    # 启动参数：启用强制分段 CUDA 图，选择 DFLASH 推测算法，指定草稿模型，
    # 并设置 cuda-graph-bs 覆盖 1 到 64 的批量大小以充分测试。
    server_args = [
        "--trust-remote-code",
        "--attention-backend", "flashinfer",
        "--enforce-piecewise-cuda-graph",
        "--speculative-algorithm", "DFLASH",
        "--speculative-draft-model-path", DEFAULT_DRAFT_MODEL_DFLASH,
        "--page-size", "1",
        "--max-running-requests", "64",
        "--cuda-graph-bs", *[str(i) for i in range(1, 65)],
    ]
    server_env = {"SGLANG_ALLOW_OVERWRITE_LONGER_CONTEXT_LEN": "1"}
    accuracy_threshold = 0.75 # 准确率门限
    speedup_threshold = 2.8 # 加速比门限

if __name__ == "__main__":
```

```
    unittest.main()
```

评论区精华

Reviewer Qiaolin-Yu 仅给出 'lgtm. could you add a related test?' 的反馈，作者随后添加了测试文件。无其他争议或讨论。

- 添加相关测试 (testing): 作者随后新增了测试文件 test_pcg_with_speculative_decoding_dflash.py。

风险与影响

- 风险：低风险。改动集中在两个条件判断和参数传递，逻辑清晰。主要风险在于：若 DFLASH 算法检测 (is_dflash()) 存在误判，可能导致其他推测算法误用 FULL 模式，略微增加显存和开销。此外，新增测试对 accuracy 和 speedup 的硬编码阈值可能因环境差异而波动。
- 影响：影响范围有限，仅涉及 DFLASH + PCG 的联合使用场景。对于使用 DFLASH 的用户，可以透明地启用 PCG 获得 prefill 加速；不使用 DFLASH 的用户无影响。测试验证了 Llama-3.1-8B-Instruct + DFLASH 组合的准确性和加速比。
- 风险标记：低风险，新功能特性，测试覆盖完整

关联脉络

- PR #23000 dflash speculative decoding (base PR): 该 PR 是从 #23000 拆分出来的，专注于 PCG 支持部分。