PR #1888 完整报告

THUDM/slime

Fix(checkpoint): add resume/pause in save_model() for offload_train (fixes #1886)

合并时间：2026-05-06 10:13

原文链接：http://prhub.com.cn/THUDM/slime/pull/1888

执行摘要

- 一句话：修复 offload_train 时 checkpoint 保存崩溃问题
- 推荐动作：建议阅读本 PR 以了解 offload 训练中 checkpoint 保存的生命周期管理。设计上使用统一封装（wake_up/sleep）管理进程组和显存状态是良好实践，值得在其他类似场景复用。

功能与动机

用户报告在启用 --offload 和 --save-interval 时 checkpoint 保存失败（issue #1886），跟踪发现是 #1856 重构后 train() 自动调用 sleep() 释放 GPU 内存，但 save_model() 未在保存前 resume 模型，导致 Megatron 的 save_checkpoint 操作在模型暂停状态下访问 GPU 数据引发 CUDA error: invalid argument。此修复补全了 offload 生命周期，使保存正常进行。

实现拆解

1. 分析问题：在 slime/backends/megatron_utils/actor.py 的 save_model() 方法中，当 offload_train=True 时，原来只重建进程组（reload_process_groups()）但未恢复模型显存状态，而 train() 已在末尾通过 self.sleep() 释放了显存。
2. 补全唤起操作：将 reload_process_groups() 替换为 self.wake_up()，该封装方法依次执行 torch_memory_saver.resume() 和 clear_memory()，确保模型参数重新驻留 GPU。
3. 补全暂停操作：在保存完成后，将 destroy_process_groups() 替换为 self.sleep()，该封装方法依次执行 clear_memory(clear_host_memory=True)、destroy_process_groups() 和 torch_memory_saver.pause()，安全释放显存。
4. 测试验证：在 H200 上使用 Qwen3.5-4B TP=2 验证，每轮 rollout 后保存成功。注意无自动化测试新增。

关键文件：

- slime/backends/megatron_utils/actor.py（模块 训练后端；类别 source；类型 core-logic；符号 save_model）：核心文件，修改 save_model 方法，在 offload_train 路径添加 wake_up/sleep 以恢复模型状态

关键符号：save_model

关键源码片段

slime/backends/megatron_utils/actor.py

核心文件，修改 save_model 方法，在 offload_train 路径添加 wake_up/sleep 以恢复模型状态

```
@timer
def save_model(self, rollout_id: int, force_sync: bool = False) -> None:
    if self.args.debug_rollout_only:
        return

    # 当使用 offload_train 时，train() 已通过 self.sleep() 释放显存
    # 保存前必须通过 wake_up() 恢复模型状态（resume + clear_memory）
    if self.args.offload_train:
        self.wake_up()

    if self.args.async_save:
        from megatron.training.async_utils import maybe_finalize_async_save
        maybe_finalize_async_save(blocking=True)

    save(rollout_id, self.model, self.optimizer, self.opt_param_scheduler)

    if force_sync and self.args.async_save:
        maybe_finalize_async_save(blocking=True)

    if self.args.save_hf is not None and self.role == "actor":
        from slime.backends.megatron_utils.model import save_hf_model
        save_hf_model(self.args, rollout_id, self.model)

    # 保存完成后再次暂停模型以释放显存
    if self.args.offload_train:
        self.sleep()
```

评论区精华

Reviewer lilei199908 建议使用现有的封装方法 self.wake_up() 和 self.sleep() 替代直接调用底层 API，使代码更简洁并复用已定义的 offload 生命周期接口。作者接受建议并更新了提交。

- 使用封装方法 wake_up/sleep 替代直接调用 (style): 作者接受建议，将初始补丁升级为使用 wake_up/sleep。

风险与影响

- 风险：风险较低，改动集中在单一方法的 offload 分支，且复用了经过测试的 wake_up/sleep 封装。主要风险是 wake_up/sleep 内部实现如果存在未覆盖状态可能导致其他 offload 操作异常，但已有 update_weights 等其他方法使用相同封装，因此回归风险小。此外，本次修复未新增测试，将来 offload 相关行为变更可能遗漏此路径。
- 影响：直接影响所有启用 --offload 和 --colocate 并设置 --save-interval 的用户，修复后能够正常保存 checkpoint。对于不使用 offload 的用户无影响。团队需确保 wake_up/sleep 封装的一致性。

- 风险标记：现有接口依赖

关联脉络

- PR #1856 refactor/ppo: 引入回归：train() 中新增自动 sleep() 导致保存时模型处于暂停状态
- PR #1886 [Question] Checkpoint save fails with --colocate + --save-interval after #1856: 用户报告了此问题，本 PR 直接修复该 issue