PR #7633 完整报告

verl-project/verl

[ci, trainer] refactor: switch fsdp_turbo e2e test from GPU to NPU

合并时间：2026-09-01 10:00

原文链接：http://prhub.com.cn/verl-project/verl/pull/7633

执行摘要

- 一句话：FSDP-Turbo e2e 测试从 GPU 迁移至 NPU，清理空梯度同步上下文
- 推荐动作：值得精读，尤其是想了解“如何将一个后端的 e2e 测试从 GPU 迁移到 NPU”以及“如何处置历史 workaround”的读者。关注点：删除 _gradient_sync_context 是否真的安全，建议在 PR 描述或后续提交中补充原因；同时留意是否有计划为 GPU 侧补充单元测试或恢复轻量 e2e，避免 fsdp_turbo 在 CUDA 上失去回归保护。

功能与动机

PR body 仅简要说明“switch fsdp_turbo e2e test from GPU to NPU”，未附 Issue。从变更内容推断，主要动机是让 fsdp_turbo 的 e2e 验证平台与上游库（FSDPTurbo 来自 Ascend）保持一致，将 CI 资源从 GPU 释放给 NPU，同时删除早期为规避 OOM 而遗留的空 _gradient_sync_context 覆盖，让实现更贴近基类默认行为。文档 fsdp_turbo_backend.rst 也更新了“Source of truth”，明确 NPU 是主要支持平台。

实现拆解

实现拆解按以下步骤进行：

1. 删除 GPU e2e 工作流：整体移除 .github/workflows/e2e_ppo_trainer_fsdp_turbo_vllm.yml（-156 行），该工作流原在 8x L20 上运行 Qwen3.5-0.8B 的 fsdp_turbo 训练测试，删除后 GPU 侧不再有 fsdp_turbo 的端到端 CI 覆盖。
2. 迁移并改造测试脚本：将 tests/special_e2e/run_ppo_trainer_fsdp_turbo.sh 重命名为 tests/special_npu/run_qwen3_5_2b_fsdp_turbo.sh，模型从 Qwen3.5-0.8B 换成 Qwen3.5-2B，并行配置由 FSDP_SIZE=8, SP_SIZE=1 调整为 FSDP_SIZE=4, SP_SIZE=2，新增 HCCL 相关环境变量（HCCL_CONNECT_TIMEOUT、端口范围等）以及 RAY_EXPERIMENTAL_NOSET_ASCEND_RT_VISIBLE_DEVICES=1；新增 SP_SIZE > 1 时生效的 Ulysses 序列并行补丁（ULYSSES_FUNCTION_PATCHES、LOSS_FUNCTION_PATCHES、MODULE_PATCHES），用于 Qwen3.5 的 attention 与 loss 计算；rollout 配置中增加 mm_processor_cache_gb=0 等 vLLM/Ascend 相关参数。
3. 在 Ascend CI 工作流中接入新脚本：在 .github/workflows/e2e_ascend.yml 的 vlm_rl_job 下新增步骤，克隆 https://gitcode.com/Ascend/FSDPTurbo.git 到 /FSDPTurbo 并导出 PYTHONPATH，执行新脚本，跑 1 个训练步。

4. 清理源码 workaround：在 verl/workers/engine/fsdp/fsdp_turbo_impl.py 中删除 from contextlib import contextmanager 导入以及空的 _gradient_sync_context 上下文管理器方法（其唯一作用是注释“To avoid OOM for fsdp_turbo backend”），删除后该类直接继承 FSDPEngineWithLMHead 基类的对应实现，不再对梯度同步步调做特殊处理。
5. 同步更新文档：docs/advance/fsdp_turbo_backend.rst 将 GPU 冒烟测试描述替换为 NPU CI 测试，更新脚本路径与工作流引用（e2e_ascend.yml 替代 e2e_ppo_trainer_fsdp_turbo_vllm.yml），并调整了两个脚本的并行配置说明。

关键文件：

- verl/workers/engine/fsdp/fsdp_turbo_impl.py（模块 引擎层；类别 source；类型 core-logic；符号 _gradient_sync_context）：删除 _gradient_sync_context 空实现及 contextmanager 导入，使 fsdp_turbo 引擎回归基类梯度同步语义，是本次唯一的训练代码变更，直接影响 CUDA/NPU 上的梯度同步行为。
- .github/workflows/e2e_ppo_trainer_fsdp_turbo_vllm.yml（模块 CI 工作流；类别 infra；类型 deletion）：整个 GPU e2e 工作流被删除（-156 行），标志着 fsdp_turbo e2e 测试平台从 GPU 迁移到 NPU 的关键动作，使 GPU 侧不再有该后端的 CI 覆盖。
- tests/special_npu/run_qwen3_5_2b_fsdp_turbo.sh（模块 测试脚本；类别 test；类型 rename-or-move）：原 GPU 脚本重命名并迁移到 special_npu，模型从 0.8B 改为 2B，并行配置从 FSDP_SIZE=8/SP=1 改为 FSDP_SIZE=4/SP=2，新增 Ulysses 补丁和 HCCL 环境变量，是 NPU 迁移后的实际执行脚本。
- .github/workflows/e2e_ascend.yml（模块 CI 工作流；类别 infra；类型 infrastructure）：在 Ascend e2e 工作流的 vlm_rl_job 中新增 fsdp_turbo 执行步骤，是 NPU 迁移的落地入口，负责克隆 FSDPTurbo 并运行新脚本。
- docs/advance/fsdp_turbo_backend.rst（模块 文档；类别 docs；类型 documentation）：同步更新 fsdp_turbo 后端文档，将 GPU 冒烟测试说明替换为 NPU CI 测试，更新 Source of truth 文件清单，帮助用户定位正确脚本。

关键符号：_gradient_sync_context, optimizer_step

关键源码片段

verl/workers/engine/fsdp/fsdp_turbo_impl.py

删除 _gradient_sync_context 空实现及 contextmanager 导入，使 fsdp_turbo 引擎回归基类梯度同步语义，是本次唯一的训练代码变更，直接影响 CUDA/NPU 上的梯度同步行为。

```
# verl/workers/engine/fsdp/fsdp_turbo_impl.py（head 版本节选）
# 本 PR 删除了原本在此处定义的空 _gradient_sync_context 上下文管理器，
# 该空实现早期用于“避开 fsdp_turbo 后端的 OOM”，现在直接继承基类
# FSDPEngineWithLMHead 的默认实现，以对齐 NPU 上的梯度同步语义。

@EngineRegistry.register(model_type="language_model", backend="fsdp_turbo", device=["cuda",
"npu"])
class FSDPTurboEngineWithLMHead(FSDPEngineWithLMHead):
    def _init_device_mesh(self):
        super()._init_device_mesh()
```

```
        self._init_parallel_state()

    def _init_parallel_state(self):
        # 通过 fsdp_turbo 库初始化并行状态，并读取引擎配置中的 turbo_config
        from fsdp_turbo.distributed.parallel_state import get_parallel_state, init_parallel_state
        from fsdp_turbo.fsdp_turbo_config import FSDPTurboConfig, _dict_to_dataclass
        self.fsdp_turbo_config = _dict_to_dataclass(FSDPTurboConfig, self.engine_config.turbo_
        config)
        self.fsdp_turbo_config.distributed.fsdp_plan.cpu_offload = self.engine_config.offload_
        policy
        init_parallel_state(self.fsdp_turbo_config)
        self._parallel_state = get_parallel_state()
        if self._is_ulysses_enabled():
            self._process_ulysses_config()

    def optimizer_step(self):
        """裁剪梯度、跳过非有限更新并执行优化器 step。"""
        assert self.optimizer_config.clip_grad is not None
        scaler = getattr(self, "scaler", None) # 兼容绕过 FSDPEngine.__init__ 的子类
        if scaler is not None:
            scaler.unscale_(self.optimizer) # 先 unscale，再按真实梯度幅值裁剪
        from fsdp_turbo.training.clip_grads import clip_grad_norm
        grad_norm_value = clip_grad_norm(model=self.module, max_norm=self.optimizer_config.
        clip_grad)
        grad_norm = torch.tensor(grad_norm_value, device=next(self.module.parameters()).device,
        dtype=torch.float32)
        if scaler is not None:
            scaler.step(self.optimizer)
            scaler.update()
        else:
            if not torch.isfinite(grad_norm):
                print(f"WARN: grad_norm is not finite: {grad_norm}")
                self.optimizer.zero_grad()
```

tests/special_npu/run_qwen3_5_2b_fsdp_turbo.sh

原 GPU 脚本重命名并迁移到 special_npu，模型从 0.8B 改为 2B，并行配置从 FSDP_SIZE=8/SP=1 改为 FSDP_SIZE=4/SP=2，新增 Ulysses 补丁和 HCCL 环境变量，是 NPU 迁移后的实际执行脚本。

```
# tests/special_npu/run_qwen3_5_2b_fsdp_turbo.sh 中的关键新增逻辑
# 当 SP_SIZE > 1 时，启用 Ulysses 序列并行所需的函数补丁与模块替换
# 这些补丁来自 fsdp_turbo 库，用于 Qwen3.5 的 attention 与 loss 计算

ULYSSES_FUNCTION_PATCHES="[{target_functions:['transformers.models.qwen3_5.modeling_
qwen3_5.eager_attention_forward'],"\
"type:full_attention},"\
"{target_functions:['transformers.models.qwen3_5.modeling_qwen3_5.Qwen3_5GatedDeltaNet.
forward'],"\
"type:gated_delta_net}]"
```

```
LOSS_FUNCTION_PATCHES="[{target_functions:['transformers.loss.loss_utils.
ForCausalLMLoss'],"\
"type:causal_lm_loss}]"

MODULE_PATCHES="[{target:transformers.models.qwen3_5.modeling_qwen3_5.Qwen3_5Model.
forward,"\
"replacement:fsdp_turbo.models.qwen.qwen3_5.qwen3_5_model_forward}]"

if [ "${SP_SIZE}" -gt 1 ]; then
    ACTOR+=(
        "+${ACTOR_TURBO}.distributed.cp_plan.ulysses_function_patches=${ULYSSES_FUNCTION_
        PATCHES}"
        "+${ACTOR_TURBO}.distributed.cp_plan.loss_function_patches=${LOSS_FUNCTION_
        PATCHES}"
        "+${ACTOR_TURBO}.module_patches=${MODULE_PATCHES}"
    )
    REF+=(
        "+${REF_TURBO}.distributed.cp_plan.ulysses_function_patches=${ULYSSES_FUNCTION_
        PATCHES}"
        "+${REF_TURBO}.distributed.cp_plan.loss_function_patches=${LOSS_FUNCTION_PATCHES}
        "
        "+${REF_TURBO}.module_patches=${MODULE_PATCHES}"
    )
fi
```

评论区精华

该 PR 没有任何 review comments，合并者 wuxibin89 直接 APPROVED，未留下文字讨论。唯一隐含的设计决策是删除 _gradient_sync_context 空实现：原注释表明它是为"避免 fsdp_turbo 后端 OOM"而设，删除后 CUDA 与 NPU 路径统一使用基类实现，但 PR 中未解释为何现在可以安全删除（可能依赖 FSDPTurbo 库的新版本行为），这一点在讨论区没有展开。

- 暂无高价值评论线程

风险与影响

- 风险：
  1. CUDA 回归风险：_gradient_sync_context 在 fsdp_turbo_impl.py 中被删除，而 GPU e2e 工作流同步移除，导致 CUDA 上 fsdp_turbo 不再有端到端测试覆盖。如果基类实现会在特定 micro-batch 边界触发同步或内存操作，可能在 GPU 上引入 OOM 或挂起，但当前无 CI 能捕获。
  2. NPU 脚本参数变化风险：脚本从 0.8B 换成 2B，FSDP_SIZE=4, SP_SIZE=2，新增 Ulysses 补丁与 mm_processor_cache_gb 等配置，若 Ascend 910B 显存或驱动版本不满足要求，CI 可能失败；新增的 CP patches 依赖 fsdp_turbo 库对应版本，存在版本兼容隐患。
  3. CI 稳定性风险：e2e_ascend.yml 新增步骤需要从 gitcode.com 克隆外部仓库，网络波动可能导致 CI 不稳定；HCCL 超时参数（1500s）虽已设置，但大规模并行下仍有超时

可能。

4. 文档与脚本路径一致性：文档中引用的脚本路径已更新，但若其他文档或脚本仍引用旧路径 tests/special_e2e/run_ppo_trainer_fsdp_turbo.sh，会产生断链。 - 影响：对 CI 体系：fsdp_turbo 的 e2e 验证平台从 GPU 完全切换到 NPU，GPU 侧不再有该后端的 e2e 测试，NPU 侧新增一个稳定运行的 smoke test（1 训练步）。对开发者：需要 NPU 环境才能复现该 e2e 测试；对 GPU 用户，fsdp_turbo 仍受支持但缺少 CI 背书，回归风险需自行控制。对文档读者：脚本路径与建议命令已更新，不会再指向已删除的 GPU 脚本。整体影响集中在 CI 与测试覆盖层面，训练代码本身的运行时行为仅因删除一个空上下文管理器而改变，影响面有限。 - 风险标记：删除 workaround 无说明，GPU e2e 覆盖移除，依赖外部仓库克隆，NPU 并行配置变化

关联脉络

- PR #7584 [ci] fix: drop stale enable_chunked_prefill=False from Ascend NPU scripts: 同为 Ascend NPU CI 相关调整，清理 NPU 脚本中的过期配置，与本 PR 一样在整理 NPU 上的 e2e 测试链路。
- PR #7605 [fsdp] feat: add Qwen3.5-2B on-policy distillation FSDP script: 与本 PR 都涉及 Qwen3.5-2B 在 NPU/FSDP 上的脚本，可相互印证 NPU 上适配 Qwen3.5-2B 的实践。
- PR #7629 [ci] chore: fix ci failure: 同为 CI 稳定性修复，涉及 e2e 脚本与 workflow 调整，与本 PR 的 CI 迁移属于同一维护方向。