# PR #24298 完整报告

sgl-project/sglang

[codex] Optimize LTX2.3 HQ denoising split passes

合并时间：2026-05-03 16:37

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/24298

## 执行摘要

- 一句话：优化 LTX2.3 HQ 去噪分裂策略
- 推荐动作：值得精读，尤其是 _ltx2_res2s_sde_step 的 terminal 参数设计展示了如何用 Python 层标量判断避免 CUDA bool 同步，是 GPU 优化的一般技巧。_apply_ltx2_guidance_pass_kwargs 的注入模式也适用于其他扩散模型的 split-pass 场景。建议配合 PR#23148 和 #23938 阅读，了解完整的扩散性能优化链。

## 功能与动机

PR body 指出 pass_specs 定义了模型所需的去噪结果（cond、neg、perturbed、modality），代码构建了大小为 batch_size_local * len(pass_specs) 的 expanded batch。每个 expanded-batch item 对应一个 pass spec。在 HQ 路径中，_split_ltx2_model_kwargs 使得每个模型调用只含一个 expanded-batch item，因此可以直接将 disable-attention 选项作为普通参数传递，避免使用 perturbation_configs 列表。而在 TI2V/non-HQ 中，一次模型调用可能包含多个带有不同 perturbation config 的 item，必须保持 perturbation_configs 以保证精度。

## 实现拆解

1. 消除 res2s SDE 终端路径的 CUDA bool 同步：_ltx2_res2s_sde_step 新增 terminal: bool = False 参数。调用方在进入 helper 前已通过 Python 标量判断终端条件（如 sigma_val == 0.0），直接传递 terminal=True 即可提前返回，避免在 GPU 上计算 (sigma_up == 0).any() 所需的同步开销。所有现有调用点同步更新为显式传递 terminal=False。
2. 新增 perturbation 配置辅助函数：
   - _ltx2_guidance_perturbation_config：将单个 LTX2GuidancePassSpec 对象映射为 disable-attention 选项字典。
   - _build_ltx2_guidance_perturbation_configs：对一组 pass_specs 展开并重复 batch_size 次，生成适用于 expanded batch 的配置元组（TI2V 路径使用）。
   - _apply_ltx2_guidance_pass_kwargs：将单条 pass_spec 的 disable-attention 字段（skip_video_self_attn_blocks、skip_audio_self_attn_blocks、disable_a2v_cross_attn、disable_v2a_cross_attn）直接注入 model_kwargs。

3. HQ 路径使用直接参数：在 evaluate_stage1_guided_x0 中，当检测到 HQ 模式（每个模型调用仅一个 item）时，调用 _apply_ltx2_guidance_pass_kwargs 将配置写为顶层模型参数，不再传递 perturbation_configs。TI2V/non-HQ 路径保持不变，继续使用 perturbation_configs 列表。
4. 注释与文档增强：6 个提交中有 5 个专注于澄清注释（如阐明 split-pass 同步选择、perturbation 设计意图），帮助后续维护者区分 HQ 与 TI2V 路径的行为差异。

关键文件：

- python/sglang/multimodal_gen/runtime/pipelines_core/stages/ltx_2_denoising.py（模块扩散管道；类别 source；类型 core-logic；符号 _ltx2_res2s_sde_step, _ltx2_guidance_perturbation_config, _build_ltx2_guidance_perturbation_configs, _apply_ltx2_guidance_pass_kwargs）：唯一修改文件，核心扩散管道；新增三个辅助函数并修改 _ltx2_res2s_sde_step 以消除 CUDA 同步，实现 HQ 路径的拆分优化。

关键符号：_ltx2_res2s_sde_step, _ltx2_guidance_perturbation_config, _build_ltx2_guidance_perturbation_configs, _apply_ltx2_guidance_pass_kwargs

## 关键源码片段

### python/sglang/multimodal_gen/runtime/pipelines_core/stages/ltx_2_denoising.py

唯一修改文件，核心扩散管道；新增三个辅助函数并修改 _ltx2_res2s_sde_step 以消除 CUDA 同步，实现 HQ 路径的拆分优化。

```
# python/sglang/multimodal_gen/runtime/pipelines_core/stages/ltx_2_denoising.py

@classmethod
def _ltx2_res2s_sde_step(
    cls,
    *,
    sample: torch.Tensor,
    denoised_sample: torch.Tensor,
    sigma: torch.Tensor,
    sigma_next: torch.Tensor,
    noise: torch.Tensor,
    eta: float = 0.5,
    terminal: bool = False, # 新增参数：由调用方基于 Python 标量决定是否需要提前返回
) -> torch.Tensor:
    # The caller decides terminal steps from Python scalars before entering
    # this helper. Keep that branch on host to avoid a CUDA bool sync in
    # every res2s SDE update.
    if terminal:
        # 直接返回 denoised_sample，避免进入 GPU 上的条件检查（如 sigma_up == 0）
        return denoised_sample.to(dtype=sample.dtype)
    alpha_ratio, sigma_down, sigma_up = cls._ltx2_get_sde_coeff(
        sigma_next,
        sigma_up=sigma_next * eta,
```

```
        )
        # 原代码在此处有 if bool((sigma_up == 0).any()) ... ，导致 CUDA 同步
        # 现在已移除，改为由调用方通过 terminal 参数控制
        eps_next = (sample - denoised_sample) / (sigma - sigma_next)
        denoised_next = sample - sigma * eps_next
        x_noised = (
            alpha_ratio * (denoised_next + sigma_down * eps_next) + sigma_up * noise
        )
        return x_noised.to(dtype=sample.dtype)

    @staticmethod
    def _apply_ltx2_guidance_pass_kwargs(
        model_kwargs: dict[str, object],
        pass_spec: LTX2GuidancePassSpec,
    ) -> None:
        """将单个 pass_spec 的 disable-attention 选项直接注入 model_kwargs。
        用于 HQ 路径（每调用一个 expanded-batch item），替代 per-item 的 perturbation_configs
        列表。
        """
        if pass_spec.skip_video_self_attn_blocks:
            model_kwargs["skip_video_self_attn_blocks"] = pass_spec.skip_video_self_attn_blocks
        if pass_spec.skip_audio_self_attn_blocks:
            model_kwargs["skip_audio_self_attn_blocks"] = pass_spec.skip_audio_self_attn_blocks
        if pass_spec.disable_a2v_cross_attn:
            model_kwargs["disable_a2v_cross_attn"] = True
        if pass_spec.disable_v2a_cross_attn:
            model_kwargs["disable_v2a_cross_attn"] = True

    @staticmethod
    def _ltx2_guidance_perturbation_config(
        pass_spec: LTX2GuidancePassSpec,
    ) -> dict[str, object]:
        """将单个 pass_spec 映射为 disable-attention 配置字典。"""
        return {
            "skip_video_self_attn_blocks": pass_spec.skip_video_self_attn_blocks,
            "skip_audio_self_attn_blocks": pass_spec.skip_audio_self_attn_blocks,
            "skip_a2v_cross_attn": pass_spec.disable_a2v_cross_attn,
            "skip_v2a_cross_attn": pass_spec.disable_v2a_cross_attn,
        }

    @classmethod
    def _build_ltx2_guidance_perturbation_configs(
        cls,
        pass_specs: list[LTX2GuidancePassSpec],
        batch_size: int,
    ) -> tuple[dict[str, object], ...]:
        """构建用于 TI2V/non-HQ 路径的 perturbation_configs 元组。
        每个 pass_spec 重复 batch_size 次以匹配 expanded batch 布局。
        """
```

```
    return tuple(
        cls._ltx2_guidance_perturbation_config(pass_spec)
        for pass_spec in pass_specs
        for _ in range(batch_size)
    )
```

## 评论区精华

无实质性 review 讨论；唯一的人工评论来自 gemini-code-assist[bot]，仅总结了变更内容未提出疑问。所有 commits 均由作者独自迭代完成（主要由注释优化组成）。设计决策（如保留 TI2V 路径的 perturbation_configs）在 PR body 和代码注释中已有充分说明。

- 无外部审查讨论 (other): 无

## 风险与影响

- 风险：精度回归风险：HQ 路径通过一致性验证（min clip 0.8043, min SSIM 0.4844, min PSNR 12.1561, max mean abs diff 46.4682），TI2V 路径未被改动，但交叉退化未测试。性能稳定性：平均 denoise step 加速 3.1%，但 LTX2AVDecodingStage 的已知性能阈值失败（不相关）。CUDA 同步消除的隐式依赖：_ltx2_res2s_sde_step 的 terminal 参数要求调用方确保 sigma 比较准确；当前所有调用点均为合法终端条件。单一文件风险：修改集中在单个文件（130 行），回滚成本低。
- 影响：用户：LTX2.3 HQ 管道用户将获得约 3% 的 denoise 步骤加速，E2E 提升 1.78%；TI2V/non-HQ 用户无感知。系统：无配置 / 部署变更。团队：新增的辅助函数和注释降低了 HQ 与 TI2V 路径的维护混淆度，但需注意在 future 引入多 item HQ 调用时不能跳过 perturbation_configs。
- 风险标记：核心路径变更，精度敏感，缺少测试配套

## 关联脉络

- PR #23148 [codex] diffusion: enable group norm silu fuse by default: 同属 diffusion 模块的性能优化链，默认启用 GroupNorm+SiLU 融合以加速 HunyuanVideo VAE 解码。
- PR #23938 Optimize large GroupNorm SiLU apply: 同一扩散模块的另一个性能优化 PR，优化大形状 GroupNorm SiLU，VAE 解码加速 18x。