

# PR #33851 完整报告

sgl-project/sglang

[diffusion] validate and document Spectrum controls

合并时间: 2026-08-06 23:23

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/33851>

## 执行摘要

- 一句话: 为 Spectrum 增加参数校验并补齐公共文档
- 推荐动作: 值得快速浏览。核心看点: 1) `__post_init__` 集中校验的写法, 特别是拒绝 bool、使用 `math.isfinite`、按字段类型分组校验; 2) 文档导航注册 (`docs.json`) 与能力矩阵对齐的配套习惯。若后续要为其他采样参数加校验, 本 PR 是可直接参考的样板。

## 功能与动机

PR body 明确说明: Spectrum settings previously lacked an explicit supported-model guard and the feature was difficult to discover from public documentation。也就是说, 用户既不知道 Spectrum 支持哪些原生模型路径, 也没有参数取值范围说明; 无效控制 (如非有限浮点或越界整数) 不会被拦截, 导致运行时行为不可预期。本 PR 将“守卫”提前到参数构造阶段, 并让功能具备公开指南。

## 实现拆解

1. 变更入口: `python/sglang/multimodal_gen/configs/sample/spectrum.py` 的 `SpectrumParams` 新增 `post_init`, 在 `dataclass` 实例化时立即执行校验。校验分三批: 浮点字段需为有限数且拒绝 bool; 窗口、混合权重等字段各自限定数值域; 整数类字段区分非负与正数, `taylor_order` 限定 1/2/3。所有失败统一抛出带字段名的 `ValueError`。
2. 测试配套: `python/sglang/multimodal_gen/test/unit/test_sampling_params.py` 新增 `test_spectrum_params_reject_invalid_controls` (覆盖 9 种无效组合) 和 `test_spectrum_dict_is_validated_when_sampling_params_constructs_it` (验证 `SamplingParams` 内嵌 `spectrum_params` dict 转换路径同样触发校验), 并新增 `SpectrumParams` 导入。
3. 文档主变更: 新增 `docs/docs/sglang-diffusion/spectrum.mdx`, 包含快速开始、支持范围 (仅 native FLUX.1/Wan/HunyuanVideo/SD3, 不含 FLUX.2 与 diffusers 后端)、与 `enable-teacache` 互斥、高级控制参数表。`docs/docs.json` 将该页注册到 Caching Acceleration 导航组。
4. 配套文档对齐: `caching-acceleration.mdx` 从两种策略扩展为三种并加入 Spectrum 对比行; `index.mdx` 入口更新; `cli.mdx` 修正 `--tp-size` 说明与 Cache-DiT 启用方式; `compatibility_matrix.mdx` 补充 MiniMax-H3 的 Sage 支持; `attention_backends.mdx` 补充 SM100/B200 cuDNN SDPA 回退; MiniMax-H3 cookbook 增加 SageAttention 说明。

5. CI 触发：通过 issue 评论 /tag-and-rerun-ci 触发 lint 与额外 CI，未在本地运行测试。

关键文件：

- python/sglang/multimodal\_gen/configs/sample/spectrum.py（模块 采样参数；类别 source；类型 core-logic；符号 post\_init）：核心源码变更：新增 \_\_post\_init\_\_ 集中校验所有 Spectrum 控制参数，是本次 PR 的行为修改入口。
- python/sglang/multimodal\_gen/test/unit/test\_sampling\_params.py（模块 单元测试；类别 test；类型 test-coverage；符号 test\_spectrum\_params\_reject\_invalid\_controls, test\_spectrum\_dict\_is\_validated\_when\_sampling\_params\_constructs\_it）：新增两个测试覆盖无效控制拒绝与 SamplingParams 内嵌 dict 的校验路径，是校验逻辑的正确性保障。
- docs/docs/sglang-diffusion/spectrum.mdx（模块 文档；类别 other；类型 documentation）：新增的 Spectrum 公共指南，包含支持范围、约束和参数表，是功能可发现性的主要载体。
- docs/docs.json（模块 文档导航；类别 config；类型 configuration）：将 spectrum.mdx 注册到 Caching Acceleration 导航组，决定文档可发现性。
- docs/docs/sglang-diffusion/caching-acceleration.mdx（模块 文档；类别 other；类型 documentation）：缓存策略概览从两种扩展为三种，加入 Spectrum 对比行并指向新指南。
- docs/docs/sglang-diffusion/compatibility\_matrix.mdx（模块 文档；类别 other；类型 documentation）：能力矩阵与当前运行时对齐，补充 MiniMax-H3 的 Sage 支持并修正一项能力标记。

关键符号：post\_init, test\_spectrum\_params\_reject\_invalid\_controls, test\_spectrum\_dict\_is\_validated\_when\_sampling\_params\_constructs\_it

## 关键源码片段

[python/sglang/multimodal\\_gen/configs/sample/spectrum.py](#)

核心源码变更：新增 \_\_post\_init\_\_ 集中校验所有 Spectrum 控制参数，是本次 PR 的行为修改入口。

```
@dataclass
class SpectrumParams(CacheParams):
    """Spectrum 谱特征预测的采样参数。

    继承自 CacheParams，核心控制包括窗口大小、切比雪夫基数量、
    Taylor 融合权重等。所有字段默认值对齐论文与参考实现。
    """

    cache_type: str = "spectrum"
    window_size: float = 2.0
    flex_window: float = 0.75
    warmup_steps: int = 5
    w: float = 1.0
    lam: float = 0.1
    m: int = 4
    history_size: int = 100
```

```

tau_num_steps: int = 50
taylor_order: int = 1

def __post_init__(self) -> None:
    # 第一批：必须是有穷数。bool 是 int 子类，显式拒绝，避免 True/False 混入数值字段。
    finite_numbers = {
        "window_size": self.window_size,
        "flex_window": self.flex_window,
        "w": self.w,
        "lam": self.lam,
    }
    for name, value in finite_numbers.items():
        if (
            isinstance(value, bool)
            or not isinstance(value, (int, float))
            or not math.isfinite(value)
        ):
            raise ValueError(f"Spectrum {name} must be a finite number.")

    # 第二批：浮点字段的数值域约束。
    if self.window_size <= 0:
        raise ValueError("Spectrum window_size must be greater than zero.")
    if self.flex_window < 0:
        raise ValueError("Spectrum flex_window must be non-negative.")
    if not 0 <= self.w <= 1:
        raise ValueError("Spectrum w must be between zero and one.")
    if self.lam < 0:
        raise ValueError("Spectrum lam must be non-negative.")

    # 第三批：整数域约束，同样拒绝 bool。
    non_negative_ints = {"warmup_steps": self.warmup_steps}
    positive_ints = {
        "m": self.m,
        "history_size": self.history_size,
        "tau_num_steps": self.tau_num_steps,
    }
    for name, value in non_negative_ints.items():
        if isinstance(value, bool) or not isinstance(value, int) or value < 0:
            raise ValueError(f"Spectrum {name} must be a non-negative integer.")
    for name, value in positive_ints.items():
        if isinstance(value, bool) or not isinstance(value, int) or value <= 0:
            raise ValueError(f"Spectrum {name} must be a positive integer.")

    # taylor_order 只允许 1、2、3 三个取值。
    if self.taylor_order not in (1, 2, 3):
        raise ValueError("Spectrum taylor_order must be one of 1, 2, or 3.")

```

python/sglang/multimodal\_gen/test/unit/test\_sampling\_params.py

新增两个测试覆盖无效控制拒绝与 `SamplingParams` 内嵌 dict 的校验路径，是校验逻辑的正确性保障。

```
class TestSamplingParamsValidate(unittest.TestCase):
    def test_spectrum_params_reject_invalid_controls(self):
        # 覆盖文档中列出的每一个可调控制：越界、非法类型、非法取值。
        invalid_controls = (
            {"window_size": 0},
            {"flex_window": -0.1},
            {"w": 1.1},
            {"lam": -0.1},
            {"warmup_steps": -1},
            {"m": 0},
            {"history_size": 0},
            {"tau_num_steps": 0},
            {"taylor_order": 4},
        )
        for kwargs in invalid_controls:
            with self.assertRaises(ValueError):
                SpectrumParams(**kwargs)

    def test_spectrum_dict_is_validated_when_sampling_params_constructs_it(self):
        # 通过 SamplingParams 内嵌 dict 传入时，也必须触发底层校验。
        with self.assertRaisesRegex(ValueError, "history_size"):
            SamplingParams(
                enable_spectrum=True,
                spectrum_params={"history_size": 0},
            )
```

## 评论区精华

本 PR 没有 review 评论或讨论线程；issue 评论仅为作者三次 /tag-and-rerun-ci（含一次 extra）触发 CI。因此没有已解决 / 未解决的技术分歧。唯一值得注意的约束在 PR body：本地未跑测试，依赖 CI 验证。

- 暂无高价值评论线程

## 风险与影响

- 风险：
  - 行为变更：所有 `SpectrumParams` 构造现在都会执行硬校验，此前可接受但语义非法的值（如 NaN、Inf、越界值、bool 冒充数值）会在请求构造期直接失败；若有关键路径依赖旧行为需回归确认。
  - 构造路径覆盖：校验只在 `dataclass __post_init__` 生效，测试覆盖了 `SamplingParams` 内嵌 dict 路径；CLI 参数解析等其他入口是否必然走该构造点，PR 未明确说明（不确定性）。
  - 文档一致性：`spectrum.mdx` 声称的支持范围（原生 FLUX.1/Wan/HunyuanVideo/SD3，不含 FLUX.2、非 `serve/OpenAI` 选项）需要与实现保持同步，否则会误导用户。

- 兼容性：改动集中在构造期校验与文档，不触及调度、去噪内核路径，回归面很小。
- 影响：
  - 用户：无效或越界的 Spectrum 控制会立即得到带字段名的 ValueError，不再静默埋雷。
  - 支持用户：获得从快速开始到参数表的完整文档路径，可从导航直接发现。
  - 团队：为后续 TeaCache 等其他 cache 参数的校验与文档化提供了可复用的模式（`dataclass post_init`+ 单测 + 导航注册）。
- 风险标记：输入校验行为变更，文档与实现需保持同步，构造路径覆盖待确认

## 关联脉络

- PR #32999 [diffusion] Fix GLM-Image resolution alignment: 同样改动 `test_sampling_params.py` 并对采样参数做契约收紧，与本 PR 同属采样参数校验与文档对齐线。
- PR #33843 [diffusion] consolidate pipeline core hygiene: 同批 diffusion 维护系列之一，体现子系统从功能落地走向稳健性与可发现性阶段。
- PR #33653 Gate multimodal feature transport by model capability: 按模型能力门控特性的思路与本 PR 的 `supported-model guard` 目标一致。