PR #34844 完整报告

sgl-project/sglang

[Spec] Support MegaMoE for DSpark under dp attention

合并时间：2026-08-15 08:20

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/34844

执行摘要

- 一句话：DSpark 在 DP attention 下支持 MegaMoE 后端
- 推荐动作：值得精读，但重点不在代码量，而在于它展示了一个典型的「先论证 lockstep 前提、再放开参数限制」的工程范式：用 _fill_dp_moe_sync_metadata 的 all-gather 与 run_idle_participation 的参与机制证明 EP 下每 rank MoE 调用次数一致，再以 SGLANG_RAGGED_VERIFY_MODE=static 兜底图 tier 一致性。PR body 附带的正确性 / 性能双验证也值得作为后续同类 gate 变更的参考模板。

功能与动机

PR body 明确指出：_handle_dspark 在 --enable-dp-attention 开启时拒绝了所有非 none 的 --moe-a2a-backend，导致 DSpark 无法与 EP all-to-all 分发（如 MegaMoE）同时使用。作者论证该限制已过时：DSV4-MoE draft 路径已在 _fill_dp_moe_sync_metadata 中 all-gather global_num_tokens，且无本地生成请求的 rank 仍通过 run_idle_participation 进入 draft forward，因此每个 EP rank 每步调用 MoE 的次数一致，MegaMoE 的 fused kernel 已能容纳零 token rank，lockstep 前提已满足。

实现拆解

实现共两步：

1. 放宽 _handle_dspark 的 a2a 后端校验（python/sglang/srt/arg_groups/speculative_hook.py）：
   - 将原来的 moe_a2a_backend != "none" 拒绝条件改为 not in ("none", "megamoe")，即除内置 TP MoE 外，只放行 megamoe，deepep、pplx 等其他后端仍按原名报错。
   - 新增第二道校验：当 moe_a2a_backend != "none" 时，读取 SGLANG_RAGGED_VERIFY_MODE 环境变量，非 static 模式直接 raise，原因是 per-rank 图 tier 只有在 _dp_tier_gather_enabled（依赖 require_mlp_tp_gather，随 a2a 后端配置变化）成立时才能达成一致，token 维度 tier 不一致会导致各 rank 重放不同图形状。import 采用函数内延迟导入，避免模块级循环依赖。
2. 新增门禁契约测试（test/registered/spec/dspark/test_dspark_draft_path_default.py）：
   - 新增 TestDsparkDpAttentionMoeA2aGate 测试类，通过 _dp_server_args 构造 --enable-dp-attention --enable-dp-lm-head dp_size=2 tp_size=2 的参数组合。

- test_only_megamoe_is_admitted 在 static 模式下验证 megamoe 通过、deepep/pplx 按名抛出 ValueError。
- test_a2a_backend_with_compact_verify_mode_raises 验证 compact 模式下即使 megamoe 也会被拒绝。
- 新增 from sglang.srt.environ import envs 导入，用于测试中覆盖环境变量。

3. CI 与测试配套：无新增配置文件，测试注册为 CPU CI（register_cpu_ci），无需 GPU 即可跑通参数校验逻辑；PR 附带了 8x GB300 双节点 TP8/DP8/EP8 的速度与 profile 数据。

关键文件：

- python/sglang/srt/arg_groups/speculative_hook.py（模块 参数校验；类别 source；类型 dependency-wiring；符号 _handle_dspark）：核心变更文件：_handle_dspark 的 a2a 后端白名单从仅 none 扩展为 none/megamoe，并新增 SGLANG_RAGGED_VERIFY_MODE=static 的强制校验，是 DSpark + DP attention + EP 组合能否启动的 gate 所在。
- test/registered/spec/dspark/test_dspark_draft_path_default.py（模块 测试；类别 test；类型 test-coverage；符号 TestDsparkDpAttentionMoeA2aGate, _dp_server_args, test_only_megamoe_is_admitted, test_a2a_backend_with_compact_verify_mode_raises）：新增 TestDsparkDpAttentionMoeA2aGate 测试类，覆盖白名单两侧（megamoe 通过 / deepep、pplx 拒绝）与 compact 模式拒绝，是本次 gate 变更的回归保障。

关键符号：_handle_dspark

关键源码片段

python/sglang/srt/arg_groups/speculative_hook.py

核心变更文件：_handle_dspark 的 a2a 后端白名单从仅 none 扩展为 none/megamoe，并新增 SGLANG_RAGGED_VERIFY_MODE=static 的强制校验，是 DSpark + DP attention + EP 组合能否启动的 gate 所在。

```
# python/sglang/srt/arg_groups/speculative_hook.py
# DSpark + dp attention 下允许的 MoE a2a 后端白名单
def _handle_dspark(server_args: ServerArgs) -> None:
    _is_npu = server_args.device.startswith("npu")
    if not server_args.device.startswith("cuda") and not _is_npu:
        raise ValueError("DSpark speculative decoding only supports CUDA and NPU devices.")

    # dp_size == 1 且开 dp_attention 是 DSV4 CP 下的退化组合，跳过 DP 专属检查
    if server_args.enable_dp_attention and server_args.dp_size > 1:
        if not server_args.enable_dp_lm_head:
            raise ValueError("DSpark with dp attention requires --enable-dp-lm-head.")

        # 白名单：内置 TP MoE（none）或 MegaMoE；其余后端保持拒绝并点名
        if not _is_npu and server_args.moe_a2a_backend not in ("none", "megamoe"):
            raise ValueError(
                "DSpark with dp attention supports moe_a2a_backend 'none' "
                "(built-in TP MoE) or 'megamoe', got "
```

```
                f"{server_args.moe_a2a_backend!r}."
            )

        # MegaMoE 等 a2a 后端依赖 per-rank 图 tier 一致:
        # _dp_tier_gather_enabled 依赖 require_mlp_tp_gather（随 a2a 配置变化），
        # 若 rank 间 token 维度 tier 不一致会重放不同图形状，因此强制 static 模式。
        # 函数内延迟 import，避免模块级循环依赖。
        if not _is_npu and server_args.moe_a2a_backend != "none":
            from sglang.srt.speculative.ragged_verify import (
                RaggedVerifyMode,
                read_ragged_verify_mode,
            )

            if read_ragged_verify_mode() is not RaggedVerifyMode.STATIC:
                raise ValueError(
                    "DSpark with dp attention + "
                    f"moe_a2a_backend={server_args.moe_a2a_backend!r} requires "
                    "SGLANG_RAGGED_VERIFY_MODE=static."
                )

        if server_args.attn_cp_size > 1:
            raise ValueError(
                "DSpark with dp attention does not support context parallel "
                f"(attn_cp_size={server_args.attn_cp_size})."
            )

        # spec 后端必须与目标后端一致，否则 EP 集合通信会错位
        if (
            not _is_npu
            and server_args.speculative_moe_a2a_backend is not None
            and server_args.speculative_moe_a2a_backend != server_args.moe_a2a_backend
        ):
            raise ValueError(
                "DSpark ignores --speculative-moe-a2a-backend; with dp attention it "
                f"must match the target moe_a2a_backend={server_args.moe_a2a_backend!r} "
                f"(got {server_args.speculative_moe_a2a_backend!r})."
            )
```

test/registered/spec/dspark/test_dspark_draft_path_default.py

新增 TestDsparkDpAttentionMoeA2aGate 测试类，覆盖白名单两侧（megamoe 通过 / deepep、pplx 拒绝）与 compact 模式拒绝，是本次 gate 变更的回归保障。

```
# test/registered/spec/dspark/test_dspark_draft_path_default.py
# DSpark + dp attention + MoE a2a 后端的门禁契约测试
class TestDsparkDpAttentionMoeA2aGate(CustomTestCase):
    """Gate contract for DSpark + dp attention + MoE a2a backends."""

    def _dp_server_args(self, *, moe_a2a_backend: str) -> ServerArgs:
        # 构造 DSpark + DP attention 的典型参数组合
```

```
        server_args = _make_dspark_server_args(
            model_path=_BUNDLED_MODEL_PATH, hf_config=_bundled_hf_config()
        )
        server_args.enable_dp_attention = True
        server_args.enable_dp_lm_head = True
        server_args.dp_size = 2
        server_args.tp_size = 2
        server_args.moe_a2a_backend = moe_a2a_backend
        return server_args

    def test_only_megamoe_is_admitted(self):
        """Both sides of the allowlist: megamoe passes, others raise by name."""
        # static 模式下 megamoe 应被放行，deepep/pplx 按名抛错
        with envs.SGLANG_RAGGED_VERIFY_MODE.override("static"):
            _handle_dspark(self._dp_server_args(moe_a2a_backend="megamoe"))
            for backend in ("deepep", "pplx"):
                with self.assertRaisesRegex(ValueError, backend):
                    _handle_dspark(self._dp_server_args(moe_a2a_backend=backend))

    def test_a2a_backend_with_compact_verify_mode_raises(self):
        # compact 模式无法保证 per-rank 图 tier 一致，megamoe 也必须拒绝
        server_args = self._dp_server_args(moe_a2a_backend="megamoe")
        with envs.SGLANG_RAGGED_VERIFY_MODE.override("compact"):
            with self.assertRaisesRegex(ValueError, "static"):
                _handle_dspark(server_args)
```

评论区精华

PR 无 review 评论，唯一 Discussion 是作者触发的 /tag-and-rerun-ci（恰好对应 Extra CI 失败后的重跑操作），reviewer hnyls2002 直接 APPROVED。PR body 中作者以数据自证了关键设计判断：gsm8k 400 题两臂均为 0.9775，8x GB300 下 640/640 请求成功、无 CUDA/NCCL/barrier 错误，并指出 target-verify 与 draft-verify CUDA graph 在 8 个 EP rank 上完全一致，以排除 phase-flip barrier 反同步风险。

- CI extra 运行失败与重跑 (other): 通过重跑命令重试，reviewer hnyls2002 最终 APPROVED。

风险与影响

- 风险：
  1. 图形状一致性风险：SGLANG_RAGGED_VERIFY_MODE=static 是硬性前置条件，若用户遗漏该环境变量，_handle_dspark 直接报错，属预期防护；但若运行环境与参数校验时的环境不一致（如 worker 进程环境被改动），理论上仍可能出现 rank 间图 tier 分歧。
  2. 依赖面扩大：megamoe 路径涉及的 fused kernel 与 DeepEP 类后端不同，当前仅在 moe_a2a_backend != "none" 分支内做校验，后续若新增其他兼容后端，需同步扩展 _dp_tier_gather_enabled 判定。

3. 覆盖缺口：测试只覆盖参数门禁，未覆盖真实 CUDA 图验证；PR body 的 8 卡数据是一次性验证，仓库 CI 不保证该组合持续回归。
4. 兼容性：NPU 设备沿用 _is_npu 分支跳过 *a2a* 校验，行为不变，但 *megamoe* 在 NPU 上的表现未验证。 - 影响：影响范围集中在 DSpark + DP attention 的部署组合：此前必须退回内置 TP MoE，现在可选用 *MegaMoE*（EP）获得更优的 MoE 扩展性。对现有使用 *none* 后端的用户零影响；对其他 *a2a* 后端用户，报错信息更明确（按名提示）。团队侧新增了一个参数组合的门禁契约测试，后续改动 _handle_dspark 时会被该测试约束。整体影响面小且正向。 - 风险标记：参数门禁变更，环境变量强依赖，缺少端到端回归，仅限 CUDA 验证

关联脉络

- PR #34816 [Perf] Publish the WAR *read-done event* at DSPARK verify: 同为 DSpark verify 阶段的调度与同步优化，关注 DSPARK verify 路径的 rank 间同步事件。
- PR #33006 fix(*dsa*): use FlashInfer fused top-k for packed PAGED rows: 同为 DeepSeek 系 MoE 路径的后端选型与 fallback 移除，涉及 DSA 下 MoE 相关 kernel 选择。
- PR #34810 fix(*qwen3*): support DeepEP-class backends and *early* EPLB state: 同属 MoE *a2a* 类后端兼容性修复，关注 DeepEP/EP 后端与调度器的配合。
- PR #34019 [SM12x] Default the fused MHC post+pre path on: 同为 DeepSeek 系在 DP/EP 配置下的性能路径默认化变更，关注跨 rank 图同步。