

算例处理规范与模块接口协议

适配“大型山地光伏电站设计中的建模与算法研究”全模块

第一部分 算例处理规范

一、文档目的

- (1) 统一算例的预处理、存储、调用标准，避免各模块单独处理数据导致格式冲突、字段缺失；
- (2) 明确算例验证指标，确保不同模块的测试结果可对比、可追溯；
- (3) 衔接“公开简化算例库”与“实际算例库”，为分模块开发、集成验证提供标准化输入。

二、算例分类及来源

算例类型	来源	适用场景	核心特征
公开简化算例	Luo 开源项目（17 个标准算例）	模块一/二的基础功能测试	地形简单、约束清晰
实际算例	甘肃东亚 200MW 项目脱敏数据	模块三集成优化	地形复杂、含实际工程约束
自定义测试算例	人工构造	模块边界条件测试	针对性强、约束单一明确

三、算例预处理流程（统一执行，输出标准化 JSON）

3.1 数据清洗

- 剔除无效数据：删除坐标异常（如超出场址范围）、参数缺失（如无坡度数据）的记录；
- 单位统一：所有长度单位换算为“米（m）”、功率单位换算为“千瓦（kW）”、面积单位换算为“平方米（m²）”；
- 格式规范：日期、布尔值统一为 JSON 标准格式（日期用字符串“2025-01-01”，布尔值用 true/false）。

3.2 标准化补充（按《最终版数据字典》补全字段）

必补字段类别	具体字段示例	补充规则
地形基础字段	grid_size、slope_matrix、buildable_matrix	网格尺寸默认 10m×10m，坡度矩阵缺失值按 0 填充
设备参数字段	q、Q_box、c3	按数据字典默认值填充（q=320kW、c3=200 元/m）
约束标注字段	cut_constraint、trench_max_cables	统一标注为数组[“整数切割”，“分区连通性”]

3.3 约束显性化标注

- 为每个算例添加 constraint_info 字段，明确约束类型、取值范围、优先级（高/中/低）；
- 示例：“constraint_info”: [{"type": “切割约束”, “value”: “2×整数列”, “priority”: “高”}, ...]

3.4 脱敏处理（仅针对实际算例）

- 坐标脱敏：将真实经纬度转化为相对坐标（以场址左下角为原点(0,0)），保留相对位置关系；
- 敏感信息隐藏：删除项目名称、具体地址、涉密设备参数，仅保留通用技术参数；
- 数据缩放：地形高程按比例缩放（缩放系数记录在 desensitization_info 字段），不影响坡度、相对高差计算。

3.5 网格化处理

- 所有算例统一转化为网格矩阵格式，网格尺寸 grid_size 按数据字典默认 10m×10m；
- 地形数据（DEM、坡度、可建设区域）均以二维数组存储，数组索引(row, col)与相对坐标映射：x=col×grid_size，y=row×grid_size。

四、算例存储规范

4.1 文件格式

- 双格式存储：
 - JSON 格式 (instance_xxx.json)：供模块直接调用（字段严格对应数据字典）；
 - Excel 格式 (instance_xxx.xlsx)：含算例说明、字段注释、约束详情，便于人工查看。
- 编码格式：UTF-8，避免中文乱码。

4.2 目录结构

instance_lib/	
public/	# 公开简化算例（按难度分级）
easy/	# 简单算例（1-5 个）
medium/	# 中等算例（6-12 个）
hard/	# 复杂算例（13-17 个）
actual/	# 实际算例
sub_instance1/	# 东乡项目分区 1 算例
sub_instance2/	# 东乡项目分区 2 算例
full_instance/	# 东乡项目完整算例
custom/	# 自定义测试算例
README.md	# 算例库说明

4.3 命名规则

- 格式：[类型缩写]_[难度/分区]_[序号].json
- 示例：public_easy_01.json（公开简单算例 1）、actual_full_01.json（实际完整算例 1）、custom_slope_01.json（自定义高坡度算例 1）。

五、算例调用规则

5.1 调用脚本（utils/load_instance.py）

- 功能：支持按“算例 ID、类型、难度”调用，自动校验字段完整性；

- 核心函数：
 - load_instance(instance_id)：按 ID 加载单个算例；
 - load_instance_by_type(type="public", difficulty="easy")：按类型+难度批量加载；
 - validate_instance(instance)：校验算例字段是否符合数据字典规范。

5.2 调用参数与返回格式

- 输入参数：算例 ID（如“public_easy_01”）或筛选条件（如 type="actual", zone="sub1"）；
- 返回格式：标准化 JSON（含 instance_info（算例基本信息）、terrain_data（地形数据）、equipment_params（设备参数）、constraint_info（约束信息）四大核心字段）。

六、算例验证指标（统一评估标准）

6.1 通用指标（所有模块适用）

指标名称	计算方式	合格标准
字段完整性	（实际含有的必选字段数/数据字典必选字段数）×100%	≥95%
约束满足度	（满足的约束数/总约束数）×100%	≥98%（高优先级约束 100%满足）
数据一致性	算例中关联参数的逻辑一致性	无逻辑矛盾

6.2 模块专用指标

模块	专用指标	计算方式
模块一	覆盖面积利用率	（切割后面板覆盖可建设区域面积/可建设区域总面积）×100%
模块二	共沟成本优化率	（非共沟方案成本-共沟方案成本）/非共沟方案成本×100%

模块	专用指标	计算方式
模块三	全生命周期成本降低率	$(\text{传统方案总成本} - \text{集成优化方案总成本}) / \text{传统方案总成本} \times 100\%$

七、版本管理

- 算例更新需标注版本号（如 v1.0），记录更新内容（如“补充实际算例的道路约束字段”）；
- 旧版本算例保留在 instance_lib/history/目录，避免误删导致历史测试结果不可复现。

第二部分 模块接口协议

一、协议目的

- (1) 定义模块间数据传递的 JSON 格式、字段约束、校验规则，避免接口适配返工；
- (2) 明确数据传递方向（模块一输出→模块二输入，模块二输出→模块三输入）；
- (3) 规范错误码与异常处理，便于调试模块间衔接问题。

二、适用范围

本协议适用于“模块一（光伏面板切割及分区）”“模块二（电气设备选型+电缆共沟）”“模块三（集成优化）”的端到端数据传递，不涉及模块内部函数调用。

三、接口清单

接口名称	数据方向	用途	核心字段来源
M1-Output	模块一→模块二	传递面板切割与分区结果	模块一决策变量
M2-Input	模块二→模块二	模块二接收模块一数据	M1-Output 完整字段
M2-Output	模块二→模块三	传递设备选型与电缆路由结果	模块二决策变量
M3-Input	模块三→模块三	模块三接收模块二数据	M2-Output+损耗参数

四、接口详细定义

4.1 接口 M1-Output（模块一输出→模块二输入）

接口用途：向模块二传递光伏面板的切割方案、分区归属、逆变器配置等核心结果，为设备选型与电缆路由提供基础。

字段名	类型	单位	必填	说明
-----	----	----	----	----

字段名	类型	单位	必填	说明
instance_id	string	-	是	关联的算例 ID
cut_result	array[object]	-	是	光伏面板切割结果
cut_result[].material_id	string	-	是	原材料 ID
cut_result[].is_used	bool	-	是	该原材料是否被使用
cuts[].spec_l	float	m	是	切割长度（需满足 2.0 的整数倍）
cuts[].quantity	int	块	是	该规格的切割数量
partition_result	array[object]	-	是	面板分区结果
partition_result[].panel_id	string	-	是	面板唯一 ID
partition_result[].grid_coordinate	tuple	(row,col)	是	面板所在网格坐标
partition_result[].zone_id	string	-	是	所属分区 ID
zone_summary	array[object]	-	是	分区汇总信息
zone_summary[].pva_count	int	块	是	分区内面板数量（18-26）
zone_summary[].perimeter	float	m	是	分区周长
constraint_satisfaction	object	-	是	约束满足情况

校验规则：

- 必选字段不得缺失，否则返回错误码 E101；
- cuts[].spec_l 必须是 2.0 的整数倍（如 2.0、4.0），否则返回错误码 E102；
- zone_summary[].pva_count 必须在 18-26 之间，否则返回错误码 E103；
- 所有 panel_id、zone_id、inverter_id 必须唯一，否则返回错误码 E104。

4.2 接口 M2-Output (模块二输出→模块三输入)

接口用途：向模块三传递箱变/逆变器选型、电缆路由、共沟方案等结果，为成本+损耗集成优化提供基础。

字段名	类型	单位	必填	说明
instance_id	string	-	是	关联的算例 ID
module1_output	object	-	是	完整的 M1-Output 结果
equipment_selection	array[object]	-	是	设备选型结果
equipment_selection[].transformer_id	string	-	是	箱变 ID
equipment_selection[].Q_box	int	kVA	是	箱变容量 (1600/3200)
equipment_selection[].install_coord	tuple	(x,y)	是	安装坐标
cable_routes	array[object]	-	是	电缆路由结果
cable_routes[].route_id	string	-	是	路由 ID
cable_routes[].cable_length	float	m	是	电缆总长度
trench_summary	array[object]	-	是	管沟汇总信息
trench_summary[].cable_count	int	根	是	管沟内电缆数量 (≤4 根)

校验规则：

- equipment_selection[].Q_box 仅支持 1600 或 3200kVA，否则返回错误码 E201；
- trench_summary[].cable_count 不得超过 4，否则返回错误码 E202；
- cable_routes[].install_coord 的 x/y 必须是 grid_size 的整数倍，否则返回错误码 E203；

- connected_inverters 中的逆变器 ID 必须在 M1-Output 的 inverter_id 中存在，否则返回错误码 E204。

4.3 接口 M3-Input（模块三输入）

接口用途：模块三接收模块二输出结果+损耗相关参数，用于全生命周期成本（建设成本+电力损耗）优化。

字段名	类型	单位	必填	说明
instance_id	string	-	是	关联的算例 ID
module2_output	object	-	是	完整的 M2-Output 结果
loss_params	object	-	是	电力损耗相关参数
loss_params.r_c	float	m	是	电缆导体截面半径（0.015）
loss_params.lambda	float	-	是	运行成本权重（0.4）
loss_params.I_max	float	A	是	电缆最大允许电流（200）
loss_params.K_segments	int	段	是	电流分段数量（3）
loss_params.I_segments	array[array]	A	是	电流分段区间

校验规则：

- loss_params 字段必须与数据字典一致，取值超出范围（如 lambda>1）返回错误码 E301；
- module2_output 必须通过 M2-Output 的校验，否则返回错误码 E302。

五、错误码定义

错误码	错误类型	处理建议
E101	M1-Output 必选字段缺失	检查模块一输出代码，补充缺失字段
E102	切割长度不符合整数列约	修正模块一切割算法，确保 spec_l 为 2.0 的整

错误码	错误类型	处理建议
	束	数倍
E201	箱变容量不支持	检查模块二设备选型逻辑，仅允许1600/3200kVA
E202	管沟电缆数量超限	调整模块二共沟策略，确保单沟电缆数≤4 根
E301	损耗参数取值异常	按数据字典修正 loss_params 字段
E302	模块二输出校验失败	先修复模块二的接口错误，再重新传递数据

六、接口调用方式

- (1) 模块间通过 JSON 文件传递数据，文件命名格式：[接口名]_[instance_id].json（如 M1-Output_public_easy_01.json）；
- (2) 模块接收数据后，先调用 validate_interface()函数校验（基于本协议规则），校验通过再进行后续计算；
- (3) 校验失败时，输出错误码+错误描述，终止后续流程，待修复后重新调用。

第三部分 接口示例 JSON 文件

1. M1-Output 示例 (public_easy_01.json)

```
{
  "instance_id": "public_easy_01",
  "cut_result": [
    {
      "material_id": "mat_001",
      "is_used": true,
      "cuts": [
        { "spec_l": 2.0, "quantity": 5 },
        { "spec_l": 4.0, "quantity": 3 }
      ]
    },
    {
      "material_id": "mat_002",
      "is_used": false,
      "cuts": []
    }
  ],
  "partition_result": [
    {
      "panel_id": "pva_00123",
      "grid_coord": [15, 28],
      "cut_spec": [2.0, 3.0],
      "zone_id": "zone_045",
      "inverter_id": "inv_045"
    },
    {
      "panel_id": "pva_00124",
      "grid_coord": [15, 29],
      "cut_spec": [2.0, 3.0],
      "zone_id": "zone_045",
      "inverter_id": "inv_045"
    }
  ],
  "zone_summary": [
    {
      "zone_id": "zone_045",
      "inverter_id": "inv_045",
      "pva_count": 22,
      "perimeter": 78.5,
      "total_power": 66.0
    }
  ],
  "constraint_satisfaction": {
    "整数切割": "100%",
    "分区连通性": "100%",
    "逆变器容量约束": "100%",
    "分区周长约束": "98%"
  }
}
```

```
}  
}
```

2. M2-Output 示例 (public_easy_01.json)

```
{  
  "instance_id": "public_easy_01",  
  "module1_output": { ... },  
  "equipment_selection": [  
    {  
      "transformer_id": "box_023",  
      "Q_box": 3200,  
      "install_coord": [1180, 750],  
      "connected_inverters": ["inv_045"],  
      "cost": {  
        "purchase": 50.0,  
        "installation": 3.0  
      }  
    }  
  ],  
  "cable_routes": [  
    {  
      "route_id": "route_089",  
      "inverter_id": "inv_045",  
      "transformer_id": "box_023",  
      "substation_id": "sub_01",  
      "edges": [  
        { "u": "v1_001", "v": "v2_002", "is_trench": true },  
        { "u": "v2_002", "v": "v3_003", "is_trench": true }  
      ],  
      "cable_length": 850.2,  
      "cost": {  
        "cable": 29757.0,  
        "trenching": 170040.0  
      }  
    }  
  ],  
  "trench_summary": [  
    {  
      "trench_id": "trench_034",  
      "substation_id": "sub_01",  
      "length": 850.2,  
      "cable_count": 3,  
      "cost": 170040.0  
    }  
  ],  
  "constraint_satisfaction": {  
    "共沟约束": "100%",  
    "箱变容量": "100%",  
    "路由连续性": "100%",  
    "电缆载流量": "100%"  
  }  
}
```

3. M3-Input 示例 (public_easy_01.json)

```
{
  "instance_id": "public_easy_01",
  "module2_output": {
    ...
  },
  "loss_params": {
    "r_c": 0.015,
    "lambda": 0.4,
    "I_max": 200.0,
    "K_segments": 3,
    "I_segments": [[0, 20], [20, 35], [35, 50]]
  }
}
```