**负荷能力评估算法接入规范**

**V1.0**

**西安交通大学**

目 录

[1. 接口说明 3](#_Toc514961825)

[1.1变压器热点温度计算方法 4](#_Toc514961826)

[1.2变压器正常周期性负荷能力评估算法 5](#_Toc514961827)

[1.3变压器长期急救负荷能力评估算法 6](#_Toc514961828)

[1.4变压器短期急救负荷能力评估算法 8](#_Toc514961829)

[1.5变压器负荷状态评估方法 9](#_Toc514961830)

[1.6冷却系统控制算法 10](#_Toc514961831)

[1.7 电网风险评估直流潮流法 11](#_Toc514961832)

[1.8电网风险评估PQ分解法 11](#_Toc514961833)

[1.9电网负荷消减法 12](#_Toc514961834)

[2. 输入参数说明 14](#_Toc514961835)

[2.1基本信息（IFhnlpgBase） 14](#_Toc514961836)

[2.2变压器绕组直流电阻及电压变比（IFhnlpgResistance） 14](#_Toc514961837)

[2.3变比信息（IFhnlpgRatio） 15](#_Toc514961838)

[2.4变压器温升（IFhnlpgTRise） 15](#_Toc514961839)

[2.5 变压器初始数据（IFhnlpgInitial） 16](#_Toc514961840)

[2.6 在线数据（IFhnlpgOnload） 17](#_Toc514961841)

[2.7 限制信息（IFhnlpgLimit） 18](#_Toc514961842)

[2.8 负荷状态（IFhnlpgTStatus） 18](#_Toc514961843)

[2.9 冷却信息（IFhnlpgCool） 20](#_Toc514961844)

[2.10电网风险评估（IDwfxpgDW） 21](#_Toc514961845)

[3. 输出参数说明 24](#_Toc514961846)

[3.1 热点温度（OHST） 24](#_Toc514961847)

[3.2变压器正常周期性负荷能力评估结果（OZcfhnlpg） 24](#_Toc514961848)

[3.3变压器长期急救负荷能力评估结果（OZcfhnlpg） 25](#_Toc514961849)

[3.4变压器短期急救负荷能力评估结果（ODqfhnlpg） 25](#_Toc514961850)

[3.5变压器负荷状态评估结果（OFhzt） 25](#_Toc514961851)

[3.6变压器冷却系统评估结果（OCool） 26](#_Toc514961852)

[3.7电网风险评估直流潮流法和PQ法结果（OZLCL） 26](#_Toc514961853)

[3.8电网风险评估负荷消减结果（OFHXJ） 28](#_Toc514961854)

# 接口说明

所有算法采用统一调用方式：

UniParameter uniParameter = new UniParameter();

//TODO 设置uniParameter  
UniSolver uniSolver = new UniSolver();  
UniResult uniResult = uniSolver.solve(uniParameter, "COOL");

统一输入参数对象：

public class UniParameter {  
 private IFhnlpgBase iFhnlpgBase;  
 private IFhnlpgInitial iFhnlpgInitial;  
 private IFhnlpgOnLoad iFhnlpgOnLoad;  
 private IFhnlpgTRise iFhnlpgTRise;  
 private IFhnlpgResistance iFhnlpgResistance;  
 private IFhnlpgRatio iFhnlpgRatio;  
 private IFhnlpgLimit iFhnlpgLimit;  
 private IFhnlpgCool iFhnlpgCool;  
 private IFhnlpgTStatus iFhnlpgTStatus;  
 private IDwfxpgDW iDwfxpgDW;  
 private IDwfxpgPQ iDwfxpgPQ;

}

统一输出对象：

public class UniResult {  
 public int errcode = 0; //0：成功；-1:失败  
 public String errmsg; //失败原因  
 public String mode; //算法缩写：HST,ZCFHNLPG等  
 public OCool oCool;  
 public OHst oHst;  
 public OZcfhnlpg oZcfhnlpg;  
 public OCqfhnlpg oCqfhnlpg;  
 public ODqfhnlpg oDqfhnlpg;  
 public OFhzt oFhzt;  
 public OZLCL oZlcl;  
 public OFHXJ oFhxj;  
 public OPQ oPq;  
}

## 1.1变压器热点温度计算方法

**1）算法模型**

|  |  |
| --- | --- |
| 算法名称 | 变压器热点温度计算方法 |
| 输入数据 | 基本要求：   1. 变压器设备台账信息   冷却方式、台账数据、出厂温升试验数据   1. 油色谱数据   负载电流（高压侧）、环境温度 |
| 输出数据 | 1. 变压器顶层油温 2. 变压器绕组温度 3. 变压器热点温度 |

**2）算法接口**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 接口名称 | 类名 | UniSolver | | |
| 接口名 | Solve（） | | |
| 接口原型 | | UniResult uniResult = uniSolver.solve(uniParameter, "HST"); | | |
| 参数列表 | | 说明 | 参数类型 | 备注 |
| iFhnlpgBase | | 基本信息 | 对象 | [详见2.1](#_设备基本信息_EquipmentData) |
| iFhnlpgResistance | | 变压器绕组直流电阻及电压变比 | 对象 | 详见2.2 |
| iFhnlpgRatio | | 变比信息 | 对象 | 详见2.3 |
| iFhnlpgTRise | | 变压器温升 | 对象 | 详见2.4 |
| iFhnlpgInitial | | 初始数据 | 对象 | 详见2.5 |
| iFhnlpgOnLoad | | 在线数据 | 对象 | 详见2.6 |
| 返回值 | | 说明 | 参数类型 | 备注 |
| oHst | |  | 对象 | 详见3.1 |

## 1.2变压器正常周期性负荷能力评估算法

**1）算法模型**

|  |  |
| --- | --- |
| 算法名称 | 正常周期性负荷能力评估算法 |
| 输入数据 | 基本要求：   1. 变压器设备台账信息   冷却方式、台账数据、出厂温升试验数据   1. 变压器运行数据   负载电流（高压侧）、环境温度、故障诊断结果 |
| 输出数据 | 1. 变压器相对寿命损失 2. 变压器正常周期性负荷能力 |

**2）算法接口**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 接口名称 | 类名 | UniSolver | | |
| 接口名 | Solve（） | | |
| 接口原型 | | UniResult uniResult = uniSolver.solve(uniParameter, "ZCFHNLPG"); | | |
| 参数列表 | | 说明 | 参数类型 | 备注 |
| iFhnlpgBase | | 基本信息 | 对象 | [详见2.1](#_设备基本信息_EquipmentData) |
| iFhnlpgResistance | | 变压器绕组直流电阻及电压变比 | 对象 | 详见2.2 |
| iFhnlpgRatio | | 变比信息 | 对象 | 详见2.3 |
| iFhnlpgTRise | | 变压器温升 | 对象 | 详见2.4 |
| iFhnlpgInitial | | 初始数据 | 对象 | 详见2.5 |
| iFhnlpgOnLoad | | 在线数据 | 对象 | 详见2.6 |
| iFhnlpgLimit | | 限制数据 | 对象 | 详见2.7 |
| 返回值 | | 说明 | 参数类型 | 备注 |
| OZcfhnlpg | |  | 对象 | 详见3.2 |

## 1.3变压器长期急救负荷能力评估算法

**1）算法模型**

|  |  |
| --- | --- |
| 算法名称 | 长期急救负荷能力评估算法 |
| 输入数据 | 基本要求：   1. 变压器设备台账信息   冷却方式、台账数据、出厂温升试验数据   1. 变压器运行数据   负载电流（高压侧）、环境温度、故障诊断结果 |
| 输出数据 | 1. 变压器相对寿命损失 2. 变压器长期急救负荷能力 |

**2）算法接口**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 接口名称 | 类名 | UniSolver | | |
| 接口名 | Solve（） | | |
| 接口原型 | | UniResult uniResult = uniSolver.solve(uniParameter, " CQFHNLPG "); | | |
| 参数列表 | | 说明 | 参数类型 | 备注 |
| iFhnlpgBase | | 基本信息 | 对象 | [详见2.1](#_设备基本信息_EquipmentData) |
| iFhnlpgResistance | | 变压器绕组直流电阻及电压变比 | 对象 | 详见2.2 |
| iFhnlpgRatio | | 变比信息 | 对象 | 详见2.3 |
| iFhnlpgTRise | | 变压器温升 | 对象 | 详见2.4 |
| iFhnlpgInitial | | 初始数据 | 对象 | 详见2.5 |
| iFhnlpgOnLoad | | 在线数据 | 对象 | 详见2.6 |
| iFhnlpgLimit | | 限制数据 | 对象 | 详见2.7 |
| 返回值 | | 说明 | 参数类型 | 备注 |
| OZcfhnlpg | |  | 对象 | 详见3.3 |

## 1.4变压器短期急救负荷能力评估算法

**1）算法模型**

|  |  |
| --- | --- |
| 算法名称 | 短期急救负荷能力评估算法 |
| 输入数据 | 基本要求：   1. 变压器设备台账信息   冷却方式、台账数据、出厂温升试验数据   1. 变压器运行数据   负载电流（高压侧）、环境温度、故障诊断结果 |
| 输出数据 | 1. 变压器相对寿命损失 2. 变压器长期急救负荷能力 |

**2）算法接口**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 接口名称 | 类名 | UniSolver | | |
| 接口名 | Solve（） | | |
| 接口原型 | | UniResult uniResult = uniSolver.solve(uniParameter, " DQFHNLPG "); | | |
| 参数列表 | | 说明 | 参数类型 | 备注 |
| iFhnlpgBase | | 基本信息 | 对象 | [详见2.1](#_设备基本信息_EquipmentData) |
| iFhnlpgResistance | | 变压器绕组直流电阻及电压变比 | 对象 | 详见2.2 |
| iFhnlpgRatio | | 变比信息 | 对象 | 详见2.3 |
| iFhnlpgTRise | | 变压器温升 | 对象 | 详见2.4 |
| iFhnlpgInitial | | 初始数据 | 对象 | 详见2.5 |
| iFhnlpgOnLoad | | 在线数据 | 对象 | 详见2.6 |
| 返回值 | | 说明 | 参数类型 | 备注 |
| ODqfhnlpg | |  | 对象 | 详见3.4 |

## 1.5变压器负荷状态评估方法

**1）算法模型**

|  |  |
| --- | --- |
| 算法名称 | 变压器负荷状态评估 |
| 输入数据 | 基本要求：   1. 变压器设备台账信息   台账数据   1. 变压器运行信息   热点温度、顶层油温、负荷电流（高压、中压、低压）、负荷电压（高压、中压、低压） |
| 输出数据 | 1. 变压器负荷状态 2. 变压器过负荷优化建议   （1）变压器过热引起的过负荷  调大变压器冷却效率（比如增加冷却风扇的投入）、降低或转移变压器负荷量。  （2）变压器高压、中压和低压侧过负荷  降低或转移变压器负荷量。  （3）变压器重载  建议降低或转移变压器负荷量。  （4）变压器轻载  调小变压器冷却效率（比如减少冷却风扇的投入，节约能源）、可适量转入负荷量，优化变压器负荷状态。 |

**2）算法接口**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 接口名称 | 类名 | UniSolver | | |
| 接口名 | Solve（） | | |
| 接口原型 | | UniResult uniResult = uniSolver.solve(uniParameter, "FHZT"); | | |
| 参数列表 | | 说明 | 参数类型 | 备注 |
| IFhnlpgTStatus | | 负荷状态 | 对象 | 详见2.8 |
| 返回值 | | 说明 | 参数类型 | 备注 |
| OFhzt | |  | 对象 | 详见3.5 |

## 1.6冷却系统控制算法

**1）算法模型**

|  |  |
| --- | --- |
| 算法名称 | 冷却系统控制算法 |
| 输入数据 | 基本要求：   1. 变压器运行数据   顶层油温、热点温度、变压器负荷率（高压侧） |
| 输出数据 | 1. 冷却器投入比例 |

**2）算法接口**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 接口名称 | 类名 | UniSolver | | |
| 接口名 | Solve（） | | |
| 接口原型 | | UniResult uniResult = uniSolver.solve(uniParameter, "COOL"); | | |
| 参数列表 | | 说明 | 参数类型 | 备注 |
| IFhnlpgCool | | 冷却信息 | 对象 | [详见2.](#_设备基本信息_EquipmentData)9 |
| 返回值 | | 说明 | 参数类型 | 备注 |
| oCool | |  | 对象 | 详见3.6 |

## 1.7 电网风险评估直流潮流法

**1）算法模型**

|  |  |
| --- | --- |
| 算法名称 |  |
| 输入数据 |  |
| 输出数据 |  |

**2）算法接口**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 接口名称 | 类名 | UniSolver | | |
| 接口名 | Solve（） | | |
| 接口原型 | | UniResult uniResult = uniSolver.solve(uniParameter, "ZLCL"); | | |
| 参数列表 | | 说明 | 参数类型 | 备注 |
| IDwfxpgDW | | 电网风险评估输入参数 | 对象 | [详见2.](#_设备基本信息_EquipmentData)10 |
| 返回值 | | 说明 | 参数类型 | 备注 |
| oZlcl | | 直流潮流法输出结果 | 对象 | 详见3.7 |

## 1.8电网风险评估PQ分解法

**1）算法模型**

|  |  |
| --- | --- |
| 算法名称 |  |
| 输入数据 |  |
| 输出数据 |  |

**2）算法接口**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 接口名称 | 类名 | UniSolver | | |
| 接口名 | Solve（） | | |
| 接口原型 | | UniResult uniResult = uniSolver.solve(uniParameter, "PQ"); | | |
| 参数列表 | | 说明 | 参数类型 | 备注 |
| IDwfxpgDW | | 电网风险评估输入参数 | 对象 | [详见2.](#_设备基本信息_EquipmentData)10 |
| 返回值 | | 说明 | 参数类型 | 备注 |
| oZlcl | | PQ分解法输出结果 | 对象 | 详见3.7 |

## 1.9电网负荷消减法

**1）算法模型**

|  |  |
| --- | --- |
| 算法名称 |  |
| 输入数据 |  |
| 输出数据 |  |

**2）算法接口**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 接口名称 | 类名 | UniSolver | | |
| 接口名 | Solve（） | | |
| 接口原型 | | UniResult uniResult = uniSolver.solve(uniParameter, "FHXJ"); | | |
| 参数列表 | | 说明 | 参数类型 | 备注 |
| IDwfxpgDW | | 电网风险评估输入参数 | 对象 | [详见2.](#_设备基本信息_EquipmentData)10 |
| 返回值 | | 说明 | 参数类型 | 备注 |
| oZlcl | | 负荷消减法输出结果 | 对象 | 详见3.7 |

# 输入参数说明

## 2.1基本信息（IFhnlpgBase）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **构造函数** | public IFhnlpgBase(int y\_text, int y\_temdata, int y\_cool, int interval) {  Y\_text = y\_text;  Y\_temdata = y\_temdata;  Y\_cool = y\_cool;  this.interval = interval; } | | | |
| **参数名称** | **代码值** | **必填** | **默认值** | **备注** |
|  | **y\_text** | Y | 1 |  |
|  | **y\_temdata** | Y | 1 |  |
|  | **y\_cool** | Y | 1 |  |
|  | **interval** | Y | 10 | 单位：分钟 |

## 2.2变压器绕组直流电阻及电压变比（IFhnlpgResistance）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **构造函数** | public IFhnlpgResistance(ArrayList<ResistanceABC> d\_H, ArrayList<ResistanceABC> d\_M, ArrayList<ResistanceABC> d\_L, int tap) {  D\_H = d\_H;  D\_M = d\_M;  D\_L = d\_L;  Tap = tap; } | | | |
| **参数名称** | **代码值** | **必填** | **默认值** | **备注** |
|  | **D\_H** | Y |  |  |
|  | **D\_M** | Y |  |  |
|  | **D\_L** | Y |  |  |
|  | **Tap** | Y |  |  |
| **构造函数** | public ResistanceABC(double a, double b, double c) {  A = a;  B = b;  C = c; } | | | |
| **参数名称** | **代码值** | **必填** | **默认值** | **备注** |
|  | **A** | Y |  |  |
|  | **B** | Y |  |  |
|  | **C** | Y |  |  |

## 2.3变比信息（IFhnlpgRatio）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **构造函数** | public IFhnlpgRatio(ArrayList<RatioHML> ratio) {  this.ratio = ratio; } | | | |
| **参数名称** | **代码值** | **必填** | **默认值** | **备注** |
|  | ratio | Y |  |  |
| **构造函数** | public RatioHML(double HM, double ML, double HL) {  this.HM = HM;  this.ML = ML;  this.HL = HL; } | | | |
| **参数名称** | **代码值** | **必填** | **默认值** | **备注** |
|  | **HM** | Y |  |  |
|  | **ML** | Y |  |  |
|  | **HL** | Y |  |  |

## 2.4变压器温升（IFhnlpgTRise）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **构造函数** | public IFhnlpgTRise(TRiseV rise100, TRiseV rise70) {  this.rise100 = rise100;  this.rise70 = rise70; } | | | |
| **参数名称** | **代码值** | **必填** | **默认值** | **备注** |
|  | rise100 | Y |  |  |
|  | rise70 | Y |  |  |
| **构造函数** | public TRiseV(double t\_top\_r, double t\_oil\_r, double t\_wnd\_r, double t\_amb\_r, double p\_fe\_r, double p\_cu\_r, double i\_H\_DC, double i\_M\_DC, double i\_L\_DC) {  T\_top\_r = t\_top\_r;  T\_oil\_r = t\_oil\_r;  T\_wnd\_r = t\_wnd\_r;  T\_amb\_r = t\_amb\_r;  P\_fe\_r = p\_fe\_r;  P\_cu\_r = p\_cu\_r;  I\_H\_DC = i\_H\_DC;  I\_M\_DC = i\_M\_DC;  I\_L\_DC = i\_L\_DC; } | | | |
| **参数名称** | **代码值** | **必填** | **默认值** | **备注** |
|  | T\_top\_r | Y |  |  |
|  | T\_oil\_r | Y |  |  |
|  | T\_wnd\_r | Y |  |  |
|  | T\_amb\_r | Y |  |  |
|  | P\_fe\_r | Y |  |  |
|  | P\_cu\_r | Y |  |  |
|  | I\_H\_DC | Y |  |  |
|  | I\_M\_DC | Y |  |  |
|  | I\_L\_DC | Y |  |  |

## 2.5 变压器初始数据（IFhnlpgInitial）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **构造函数** | public IFhnlpgInitial(double t\_top\_0, double t\_oil\_0, double t\_wnd\_0) {  T\_top\_0 = t\_top\_0;  T\_oil\_0 = t\_oil\_0;  T\_wnd\_0 = t\_wnd\_0; } | | | |
| **参数名称** | **代码值** | **必填** | **默认值** | **备注** |
|  | T\_top\_0 | Y |  |  |
|  | T\_oil\_0 | Y |  |  |
|  | T\_wnd\_0 | Y |  |  |

## 2.6 在线数据（IFhnlpgOnload）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **构造函数** | public IFhnlpgOnLoad(ArrayList<Load> load) {  for (Load l : load) {  T\_amb.add(l.getT\_amb());  I\_H\_current.add(l.getI\_H\_current());  I\_M\_current.add(l.getI\_M\_current());  I\_L\_current.add(l.getI\_L\_current());  T\_top\_C.add(l.getT\_top\_C());  } } | | | |
| **参数名称** | **代码值** | **必填** | **默认值** | **备注** |
|  | load | Y |  |  |
| **构造函数** | public Load(double t\_amb, double i\_H\_current, double i\_M\_current, double i\_L\_current, double t\_top\_C) {  T\_amb = t\_amb;  I\_H\_current = i\_H\_current;  I\_M\_current = i\_M\_current;  I\_L\_current = i\_L\_current;  T\_top\_C = t\_top\_C; } | | | |
| **参数名称** | **代码值** | **必填** | **默认值** | **备注** |
|  | T\_amb | Y |  |  |
|  | I\_H\_current | Y |  |  |
|  | I\_M\_current | Y |  |  |
|  | I\_L\_current | Y |  |  |
|  | T\_top\_C | Y |  |  |

## 2.7 限制信息（IFhnlpgLimit）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **构造函数** | public IFhnlpgLimit(double t\_amb, double i\_H\_current, double i\_H\_current\_coef) {  T\_amb = t\_amb;  I\_H\_current = i\_H\_current;  I\_H\_current\_coef = i\_H\_current\_coef; } | | | |
| **参数名称** | **代码值** | **必填** | **默认值** | **备注** |
|  | T\_amb | Y |  |  |
|  | I\_H\_current | Y |  |  |
|  | I\_H\_current\_coef | Y |  |  |

## 2.8 负荷状态（IFhnlpgTStatus）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **构造函数** | public IFhnlpgTStatus(double tlimit\_top, double tlimit\_hs, Nameplate nameplate, Operation operation) {  Tlimit\_top = tlimit\_top;  Tlimit\_hs = tlimit\_hs;  this.nameplate = nameplate;  this.operation = operation; } | | | |
| **参数名称** | **代码值** | **必填** | **默认值** | **备注** |
|  | Tlimit\_top | Y |  |  |
|  | Tlimit\_hs | Y |  |  |
|  | nameplate | Y |  |  |
|  | operation | Y |  |  |
| **构造函数** | public Nameplate(double c\_H\_r, double c\_M\_r, double c\_L\_r, double v\_H\_r, double v\_M\_r, double v\_L\_r, double i\_H\_r, double i\_M\_r, double i\_L\_r) {  C\_H\_r = c\_H\_r;  C\_M\_r = c\_M\_r;  C\_L\_r = c\_L\_r;  V\_H\_r = v\_H\_r;  V\_M\_r = v\_M\_r;  V\_L\_r = v\_L\_r;  I\_H\_r = i\_H\_r;  I\_M\_r = i\_M\_r;  I\_L\_r = i\_L\_r; } | | | |
| **参数名称** | **代码值** | **必填** | **默认值** | **备注** |
|  | C\_H\_r | Y |  |  |
|  | C\_M\_r | Y |  |  |
|  | C\_L\_r | Y |  |  |
|  | V\_H\_r | Y |  |  |
|  | V\_M\_r | Y |  |  |
|  | V\_L\_r | Y |  |  |
|  | I\_H\_r | Y |  |  |
|  | I\_M\_r | Y |  |  |
|  | I\_L\_r | Y |  |  |
| **构造函数** | public Operation(double TH\_hs, double TH\_top, double v\_H\_C, double v\_M\_C, double v\_L\_C, double i\_H\_C, double i\_M\_C, double i\_L\_C) {  this.TH\_hs = TH\_hs;  this.TH\_top = TH\_top;  V\_H\_C = v\_H\_C;  V\_M\_C = v\_M\_C;  V\_L\_C = v\_L\_C;  I\_H\_C = i\_H\_C;  I\_M\_C = i\_M\_C;  I\_L\_C = i\_L\_C; } | | | |
| **参数名称** | **代码值** | **必填** | **默认值** | **备注** |
|  | TH\_hs | Y |  |  |
|  | TH\_top | Y |  |  |
|  | V\_H\_C | Y |  |  |
|  | V\_M\_C | Y |  |  |
|  | V\_L\_C | Y |  |  |
|  | I\_H\_C | Y |  |  |
|  | I\_M\_C | Y |  |  |
|  | I\_L\_C | Y |  |  |

## 2.9 冷却信息（IFhnlpgCool）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **构造函数** | public IFhnlpgCool(double a1, double b1, double a2, double b2, double a3, double b3, double t1, double TH1, double k1, double t2, double TH2, double k2) {  this.a1 = a1;  this.b1 = b1;  this.a2 = a2;  this.b2 = b2;  this.a3 = a3;  this.b3 = b3;  T1 = t1;  this.TH1 = TH1;  K1 = k1;  T2 = t2;  this.TH2 = TH2;  K2 = k2; } | | | |
| **参数名称** | **代码值** | **必填** | **默认值** | **备注** |
|  | A1 | Y |  |  |
|  | B1 | Y |  |  |
|  | A2 | Y |  |  |
|  | B2 | Y |  |  |
|  | A3 | Y |  |  |
|  | B3 | Y |  |  |
|  | TH1 | Y |  |  |
|  | K1 | Y |  |  |
|  | T2 | Y |  |  |
|  | K2 | Y |  |  |

## 2.10电网风险评估（IDwfxpgDW）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **构造函数** | public IDwfxpgDW(ArrayList<CBus> bus, ArrayList<CLoad> load, ArrayList<CGenerator> generator, ArrayList<CLine> line) {  Bus = bus;  Load = load;  Generator = generator;  Line = line; } | | | |
| **参数名称** | **代码值** | **必填** | **默认值** | **备注** |
|  | Bus | Y |  |  |
|  | Load | Y |  |  |
|  | Generator | Y |  |  |
|  | Line | Y |  |  |
| **构造函数** | public CBus(int id, double capacity, int type) {  this.id = id;  this.capacity = capacity;  this.type = type; } | | | |
| **参数名称** | **代码值** | **必填** | **默认值** | **备注** |
|  | **id** | Y |  |  |
|  | Capacity | Y |  |  |
|  | Type | Y |  |  |
| **构造函数** | public CLoad(int id, double pl) {  this.id = id;  this.pl = pl; } | | | |
| **参数名称** | **代码值** | **必填** | **默认值** | **备注** |
|  | **id** | Y |  |  |
|  | **PL** | Y |  |  |
| **构造函数** | public CGenerator(int id, double pg) {  this.id = id;  this.pg = pg; } | | | |
| **参数名称** | **代码值** | **必填** | **默认值** | **备注** |
|  | **id** | Y |  |  |
|  | **Pg** | Y |  |  |
| **构造函数** | public CLine(int sid, int eid, int sn, double resistance) {  this.sid = sid;  this.eid = eid;  this.sn = sn;  this.resistance = resistance; } | | | |
| **参数名称** | **代码值** | **必填** | **默认值** | **备注** |
|  | **Sid** | Y |  |  |
|  | **eid** | Y |  |  |
|  | **Sn** | Y |  |  |
|  | **resistance** | Y |  |  |

# 输出参数说明

## 3.1 热点温度（OHST）

|  |  |  |
| --- | --- | --- |
| **类** | public class OHst {  public ArrayList<Double> T\_hs\_G;  public ArrayList<Double> T\_top\_C;  public double HST;  public double TOPT; } | |
| **属性名称** | **代码值** | **备注** |
|  | T\_hs\_G |  |
|  | T\_top\_C |  |
|  | HST |  |
|  | TOPT |  |

## 3.2变压器正常周期性负荷能力评估结果（OZcfhnlpg）

|  |  |  |
| --- | --- | --- |
| **类** | public class OZcfhnlpg extends OHst {  public double K;  public double Flimit\_IC;  public String YXTJ; } | |
| **属性名称** | **代码值** | **备注** |
|  | K |  |
|  | Flimit\_IC |  |
|  | YXTJ |  |
|  | T\_hs\_G |  |
|  | T\_top\_C |  |
|  | HST |  |
|  | TOPT |  |

## 3.3变压器长期急救负荷能力评估结果（OZcfhnlpg）

说明：输出结果同3.2

## 3.4变压器短期急救负荷能力评估结果（ODqfhnlpg）

|  |  |  |
| --- | --- | --- |
| **类** | public class ODqfhnlpg extends OHst {  public double Flimit\_IC;  public String YXTJ;  public ArrayList<Double> K2;  public ArrayList<Double> T2; } | |
| **属性名称** | **代码值** | **备注** |
|  | Flimit\_IC |  |
|  | YXTJ |  |
|  | K2 |  |
|  | T2 |  |

## 3.5变压器负荷状态评估结果（OFhzt）

|  |  |  |
| --- | --- | --- |
| **类** | public class OFhzt {  public String result; } | |
| **属性名称** | **代码值** | **备注** |
|  | result |  |

## 3.6变压器冷却系统评估结果（OCool）

|  |  |  |
| --- | --- | --- |
| **类** | public class OCool {  public String ratio; } | |
| **属性名称** | **代码值** | **备注** |
|  | ratio |  |

## 3.7电网风险评估直流潮流法和PQ法结果（OZLCL）

|  |  |  |
| --- | --- | --- |
| **类** | public class OZLCL {  public Matrix Y;  public ArrayList<Double> Delta;  public ArrayList<CLinePIJ> PIJ; } | |
| **属性名称** | **代码值** | **备注** |
|  | Y |  |
|  | Delta |  |
|  | PIJ |  |
| **类** | //矩阵说明  public class Matrix {  private double[][] A;  private int m;  private int n;  public int getRowDimension() {  return this.m;  }   public int getColumnDimension() {  return this.n;  }  public double get(int var1, int var2) {  return this.A[var1][var2];  }  } | |
| **属性名称** | **代码值** | **备注** |
|  | A | 矩阵二维数组 |
|  | M | 行数 |
|  | N | 列数 |
| **类** | public class CLinePIJ {  public int sid;  public int eid;  public double pij;   public CLinePIJ(int sid, int eid, double pij) {  this.sid = sid;  this.eid = eid;  this.pij = pij;  }   public int getSid() {  return sid;  }   public int getEid() {  return eid;  }   public double getPij() {  return pij;  } } | |
| **属性名称** | **代码值** | **备注** |
|  | Sid |  |
|  | Eid |  |
|  | pij |  |

## 3.8电网风险评估负荷消减结果（OFHXJ）

|  |  |  |
| --- | --- | --- |
| **类** | public class OFHXJ extends OZLCL {  public int status; //1:不符合要求需引入松弛变量, 2:已找到最优解!, 3:此问题不存在最优解  public ArrayList<Double> X;  public double f;  } | |
| **属性名称** | **代码值** | **备注** |
|  | Y | 同3.7 |
|  | Delta | 同3.7 |
|  | PIJ | 同3.7 |
|  | status |  |
|  | X |  |
|  | f |  |