DailyLoadFlow by OPENLF

Introduction

The following documentation gives simple instructions for users who are going to use script named dailyloadflow.py in the path I:\Konstantinos Sidiropoulos\LoadFlow_reports.

The specified script creates a comprehensive LoadFlow analysis and saves the output in .xlsx form similar to Unicorns LoadFlows reports. It uses functions of Pypowsybl library alongside with libraries of python and creates a loop where a user can have a daily or hourly LoadFlow analysis.

The code has been tested and works for igms and cgms in UCTE format. It was used in computer 10.91.100.15 with the following installations:

- Python version 3.12.0
- Pypowsybl library: User can simply install released versions of pypowsybl from PyPI using pip:

```
pip install pypowsybl
```

- Visual Studio Code: VSCodeUserSetup-x64-1.91.0.exe

Note: File_type was always FO = Day Ahead (D-1) or 2D = Two Days Ahead (D-2)
All UCTE's had name structure ucte_filename = {Date}_{hour}_{File_type}_{country_code}{number}.{format}

1. How to use

User has to enter after **running** the script the following information to console:

- **Enter the path to your UCTE files folder:** Enters the path where his hourly UCTE files are.
- **Enter the path where output reports should be saved:** Enters the path where his LoadFlow reports are going to be saved.
- **Enter the date (e.g., 20240717):** Fill in the date of the UCTE files.
- **Enter the file type (e.g., FO3):** Fill in the file type of the UCTE files.
- **Enter the country code (e.g., GR):** Fill in the code of the country for IGM UCTE files or 'UX' for CGM UCTE files.
- **Enter the format (e.g., UCT):** User enters the format of his network file.
- **Enter the hours (comma-separated ex. 0030,0130 ... , or leave blank for default 0030-2330):** Fill in the hours of UCTE files where he wants to have LoadFlow reports.
- **Enter the version numbers (comma-separated ex. 0,8... , or leave blank for default 0-20):** Enter the version of numbers (ex. 0,8 from zero to 7) or blank to check for versions from 0 to 20

The default construct of the UCTE filename is :

```
1.ucte_filename = f'{Date}_{hour}_{File_type}_{country_code}{number}.{format}'
```

Figure 1: Dynamic filename construct

If UCTE file has different structure, an error in the terminal will occur (in **line 92** of the script) and user has to change the default construct of the **ucte_filename**.

2. AC LoadFlow execution

User has to define the parameters he needs for the loadFlow execution.

Replaces **True/False** instead of **None** values for all the parameters he needs except: **balance_type, countries_to_balance, connected_component_mode, dc_power_factor**, **provider_parameters.**

All parameters description for AC LoadFlow execution are available in:

https://powsybl.readthedocs.io/projects/pypowsybl/en/stable/reference/loadflow/parameters.html#pypowsybl.loadflow.Parameters

```
p = lf.Parameters(
distributed_slack=False ,
transformer_voltage_control_on=False ,
phase_shifter_regulation_on=True ,
shunt_compensator_voltage_control_on=True ,
voltage_init_mode=None,
use_reactive_limits=None,
twt_split_shunt_admittance=None,
read_slack_bus=None,
write_slack_bus=None,
balance_type=None,
dc_use_transformer_ratio=None,
countries_to_balance=None,
connected_component_mode=None,
dc_power_factor=None,
provider_parameters={
'maxOuterLoopIterations' : str(30) ,
'lowImpedanceBranchMode ': 'REPLACE_BY_MIN_IMPEDANCE_LINE' ,
'slackBusesIds'  : 'G5MEGA14'
  })
```

Figure 2: Parameters

Provider_parameters contain a list of parameters linked to the loadflow provider. All provider parameters available in the excel named **'provider_parameters.xlsx'** in I:\Konstantinos Sidiropoulos\Documentation\LoadFlow_reports. User can add the follow orders to get anytime the provider parameters names and description:

```
It =lf.get_provider_parameters() # OPTIONAL --> GET A LIST OF AVAILABLE provider PARAMETERS TO EXECUTE AC LOADFLOW
It.to_excel('provider_parameters.xlsx')
```

Figure 3:Provider parameters

3. Results

The expected LoadFlow reports have **.xlsx** form and are saved in **output_folder** with names as (**line 143**):

```
1. output_filename = f'{Date}_{hour}_{File_type}_{country_code}_0_OPENLF_REPORT.xlsx'
```

Each excel contains 5 sheets.

1. First sheet named **‘Bus’** contains information as:

Id, voltage, theta, P generation, Q generation, reference voltage of generators , max/min Q limits of generators , voltage_regulation of generators , P load , Q load about the buses of the network.

2. Second sheet named **‘Transformers’** contains information as:

Id of the transformer, side of transformer and id of the bus that the side of the transformer is connected to. Also base Voltage , voltage , theta , I ,,I_limit , P, Q information about the transformers of the network is included in.

3. Third sheet named **‘Line’** contains information as:

Id of the line , side of the line and id of the bus that the side of the line is connected to. Also voltage, theta, I, I_limit, P, Q about the lines of the network information is included.

4. Fourth sheet named **‘X-Nodes’** contains information as:

Id of the line connecting X-Node with real boundary bus. Id, voltage and theta of the bus that is connected with the X-Node. I, I_limit, P and Q of side of line connected to bus. voltage , theta of the X-Node and P, Q of the line connected to the X-node.

5. Fifth sheet named **‘Switches’** contains information as:

Kind, state(open/close) of the switch. Also Id’s of buses switch is connected to and information about if the switch is part of the model but not of the network (fictitious) is included.