Matlab script to prepare the input data for Task 3

Merge of the pre-contingency load flow results (provided by the iPST platform) of the SN, FO and MCLA states with the same timestamp into a single file

INESC TEC

José Meirinhos, Helena Vasconcelos

19 October 2018

1. Objective

This MATLAB script aims to prepare the input data for task 3 (evaluation of the uncertainty model used by the MCLA).

For a set of pre-selected transmission lines (defined by a pre-specified transmission area and nominal voltage), this scrip imports the values of pre-contingency active power flow ("P", in MW), pre-contingency electric current ("I", in A) and the adopted maximum permanent current limits ("Imax", in A), from the following three types of files:

- A csv file containing the pre-contingency load flow results for each SN (snapshot) [created by the online workflow, by the itools command `export-online-workflow-states`];
- A csv file containing the pre-contingency load flow results for each FO (forecast state) and the corresponding MCLA states [created by the online workflow, by the itools command `export-online-workflow-states`];
- A IIDM (iTesla Internal Data Model) SN file with the characteristics of the network for each timestamp, namely to obtain the adopted "Imax" values.

The script merges, into a single output csv file, the pre-contingency load flow results for the SN, FO + MCLA states with the same timestamp, including the adopted "Imax" limits. An output csv file is created for each timestamp (defined by the timestamp of the considered FO states).

Warning: This script requires MATLAB R2016b.

2. Input Data

2.1 Pre-contingency load flow results

Each input csv file (containing the pre-contingency load flow results for a SN or a FO with the corresponding MCLA states) must be named "`7.workflow_states.csv`" and obtained in the following way:

- "SN" files: Run the online workflow for each SN without uncertainty (i.e. without MCLA states) and then get the csv file provided by the itools command `export-online-workflow-states`;
- "FO" files: Run the online workflow for each FO with MCLA states and then get the csv file provided by the itools command `export-online-workflow-states`.

The "SN" files must be stored into a single SN folder. The "FO" files must also be stored into a single FO folder. Inside each SN and FO folder, a different folder must exist for each timestamp to store the corresponding "`7.workflow_states.csv`" file. These folders must be named "`workflow_`" + "`ID of the online workflow defined by the platform`". An example is "`workflow_20130227_1930_20160116091728215`".

A bash language script (named "`export_data.sh`") was created to automatize the creation of these "`7.workflow_states.csv`" files for a set of pre-specified online workflows (see annex for more details).

2.2 CE file (list of substations with the corresponding transmission area)

In the main folder, a "`lienPostesCE.csv`" file must exist (which was provided by RTE) to define the "transmission area", named CE ("Centre d'Exploitation"), for each substation of the French transmission system. The first lines of this file are presented below:

```
CODNAT;CE
MANDA;Lille
MUHLB;Nancy
LOUIS;Nantes
MARSI;Toulouse
```

For each substation (identified by its code name) in the first column, the second column defines its corresponding "transmission area" (i.e. CE). This information is used to filter all substations associated to a user defined "transmission area".

2.3 IIDM SN files

For each timestamp, an associated IIDM SN file is required with the characteristics of the network (needed in order to get the "Imax" value of the transmission lines). These files must be stored in a IIDM SN folder.

3. Steps to execute the MATLAB script

The MATLAB script comprises the following files:

- `MAIN_Import.m` (Main file, which should be executed)
- `READ_VARIABLES.m` (Function to filter the branches by removed)
- `READ_WORKFLOW_STATES_SN.m` (Function to read the pre-contingency load-flow FO states)
- `READ_WORKFLOW_STATES_FO.m` (Function to read the pre-contingency load-flow SN states)

To execute the Matlab script, the following steps must be performed:

1. Open the MATLAB file "`MAIN_Import.m`"

2. Inside the code, the following configuration must be specified (see example in figure below):
   - Path for the FO folder (in "`path_FO =`");
   - Path for the SN folder (in "`path_SN =`");
   - Path for the IIDM SN folder (in "`base_case_path =`"). Note: In the current version, the month path must be specified, meaning that no more than one month can be included in the input data when running the script.
   - Path for the main folder with the "`lienPostesCE.csv`" CE file and where the output files are going to be stored (in "`CE_path=`").
   - Path where the output files are going to be stored (in "`out_path=`").
   - Specify the transmission area (in "`CE =`") to only include the transmission line records with the specified "transmission area". If not specified, this filter is not performed and therefore all the line records are considered.
   - Specify the nominal voltage (in "`Voltage =`") to only include the transmission line records with the specified nominal voltage. If not specified, no voltage filter is performed.

3. Run the script.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Configuration
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
path_FO = 'D:\INESC\iPST\Server\FEA 2018jan_feb_nS_50_UniMod\2018_Mar_FO'; % Path of the FO data
path_SN = 'D:\INESC\iPST\Server\FEA 2018jan_feb_nS_50_UniMod\2018_Mar_SN'; % Path of the SN data
base_case_path = 'C:\Users\jlmm\INESC\3. Projects\iPST\Server\caserepo\IIDM\SN\2018\03'; % Path for the base cases
CE_path = 'D:\INESC\iPST\Server\FEA 2018jan_feb_nS_50_UniMod'; % Path for CE and output files
out_path = 'D:\INESC\iPST\Server\FEA 2018jan_feb_nS_50_UniMod\Task3\2018_Mar_2sides'; % Path for output files

flow = 2; % Specify 1 for keep variables just in one side of the transmition line.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Comment if not to consider
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Specify a Region to filter the variables.
% CE = 'Lille';
% CE = 'Lyon';
% CE = 'Marseille';
% CE = 'Nancy';
% CE = 'Nanterre';
% CE = 'Nantes';
% CE = 'Toulouse';

% Specify "voltage" as follow:
% voltage = {'1'};  % '1' -> 20 kV (and below);
% voltage = {'2'};  % '2' -> 45 kV;
% voltage = {'3'};  % '3' -> 63 kV;
% voltage = {'4'};  % '4' -> 90 kV;
% voltage = {'5'};  % '5' -> 150 kV;
% voltage = {'6'};  % '6' -> 225 kV;
% voltage = {'7'};  % '7' -> 380 kV;

voltage = {'6','7'}; % Speficy multiple voltage levels
```

Warning: This script is based on the FO base cases (defined by its timestamps). If the SN base case does not exist, the corresponding timestamp is not imported.

4. Outputs

For each matched FO/SN, the script saves, in the user defined main folder, a csv file with the name "`WF_YearMonthDay_Hour.csv`".

- In each csv file, the output results are organized as follows:

	A	B	C	D	E	F
1	state	AGASSL71JONQU__TO__AGASSP7_P	AGASSL71JONQU__TO__AGASSP7_I	AGASSL71TAVEL__TO__AGASSP7_P	AGASSL71TAVEL__TO__AGASSP7_I	B.CARL71BIANC__TO__B.CARP7_P
2	-2	2383.9	3622	1740.2	2644	1442.7
3	-1	214.64	308.05	-214.64	308.05	-147.21
4	0	-124.82	173.89	-124.84	173.93	-274.55
5	1	-92.934	129.08	-92.954	129.1	-221.19
6	2	-124.41	173.23	-124.44	173.27	-272.8
7	3	-132	184.49	-132.03	184.52	-285.11
8	4	-125.46	174.77	-125.49	174.81	-276.02
9	5	-131.6	183.19	-131.62	183.22	-283.26
10	6	-129.76	181.06	-129.79	181.09	-283.26
11	7	-111.77	155.24	-111.8	155.28	-255.9
12	8	-132.17	184.33	-132.2	184.37	-284.41
13	9	-129.73	180.35	-129.75	180.38	-278.65
14	10	-126.26	175.87	-126.29	175.9	-277.1

a. The first column identifies the state ID where:
   - **State -2** identifies the data line with the permanent maximum limit values of each recorded pre-contingency operating condition ("Imax", in A, for electric currents and "Smax", in MVA, for active power flows).
   - **State -1** identifies the data line with the SN pre-contingency load-flow results.
   - **State 0** identifies the data line with the FO pre-contingency load-flow results.

- **The remaining states** (state 1, state 2, …) identify the ensemble members pre-contingency load-flow results [the ensemble members are the MCLA states created for the FO base case].

b. The remaining columns identify the recorded pre-contingency operating conditions, which can be the active power flow (P in MW) or the electric current (I in A) for each pre-selected transmission line.

Note 1: For each transmission line, if a "transmission area" is specified, the file only comprises the active power flow and the electric current for one substation bus of the transmission line. This will avoid performing a statistical analyzing for operating conditions that usually have similar values.

Note 2: The codification used for these operating conditions is the one obtained from the input files.

Note 3: The maximum permanent limit for active power flows ("Smax", in MVA) is calculated from the maximum permanent limit for the electric current ("Imax", in A) and by assuming the line nominal voltage ("Un", in kV). Namely, by assuming that Smax = sqrt(3)*Un*Imax/1000.

5. ANNEX

5.1 Pseudocode for the developed algorithm

Main Algorithm

Step 1: *Specify the input data*

Step 2: *Get the "substation code names" for the specified "transmission area".*

Step 3: ***For*** *each timestamp:*

Step 4: *From the SN input file, get the "variable header" of the active power flow and electric current in the transmission lines defined by the specified "transmission area" and voltage level;*

Step 5: *From the IIDM SN input file, get the limits of each "variable header" (State -2);*

Step 6: *From the SN input file, get the record of each "variable header" from the SN base case (State -1);*

Step 7: *From the FO input file, get the record of each "variable header" from the FO base case (State 0) and from the remaining states corresponding to the ensemble forecast (from State 1 to State m-1, being m the number of ensemble forecasts provided from the MCLA including the FO base case).*

Step 8: ***End***

Step 9: *Save the results in .csv files (one file for each timestamp).*

In the follow pseudocode, **step 4** of the main algorithm is detailed, namely how to get, from the SN input file, the "variable header" of the active power flow and electric current in the transmission lines defined by the specified "transmission area" and voltage level.

Step 4 - Variables filter

1: *Filter all the "variable headers" that are active power flow and electric current in branches (variables with "_TO_" **AND** ("_I" **OR** "_P") at the end of the variable code name).*

2: *From the last set of "variable headers", retain the ones associated to transmission lines (transformers operating conditions are excluded) of the user defined voltage level (retain variables with "L" in the 6th character **AND** where the 7th defines the specified voltage level – see voltage level codification in *).*

3: ***For** each "substation code name" obtained from step 2 of the main algorithm:*

4: ***If** flow =1*

5: *Retain the "variable headers" associated to the specified "substation code name" **AND** just including the operating conditions for one of the substation buses of each transmission line (retain variables that start with " substation code name" **AND** contain "_TO_" + "substation code name"*

6: ***Else***

7: *Retain the "variable headers" associated to the specified "substation code name" including all operating conditions of the substation buses of each transmission line.*

5: ***End***

** Voltage Level Codification:*

"1" –> 20 kV (and below);
"2" –> 45 kV;
"3" –> 63 kV;
"4" –> 90 kV;
"5" –> 150 kV;
"6" –> 225 kV;
"7" –> 380 kV.

In the follow pseudocode, **step 5** of the main algorithm is detailed, namely to get the maximum permanent limit of electric current and active power flow for the specified transmission lines.

Step 5 - Get the limits of each "variable header" (State -2)

1: *Read the IIDM SN file related to the evaluated timestamp*

2: ***For** each "variable header" obtained from step 4 of the main algorithm:*

3: *Find, in the IIDM file, the correspondent "line ID" (i.e. all the characters before "_TO_" in the "variable header" - see example in **).*

4: *Get content between "<line id="+"line ID" **AND** "</line>"*

5: ***IF** empty*

6: *Get content between "<twoWindingsTransformer id=" + "line ID" **AND** "</twoWindingsTransformer >"*

7: ***End***

8: *Get the value after "currentLimits1 permanentLimit="*

9: ***IF** empty*

10: *Get the value after "currentLimits2 permanentLimit="*

11: ***End***

12: *IF empty*

13: *Remove "variable header" from the current timestamp*

14: *End*

15: *End*

*** Example of a "variable header":*

- "variable header": "B.CARL71BIANC__TO__B.CARP7_P"
- "line ID": "B.CARL71BIANC"
- "substation code name": "B.CAR"

5.2 Script to obtain the csv files for pre-specified online workflows (export_data.sh)

To automatize the creation of the "`7.workflow_states.csv`" files for a set of pre-specified online workflows already in the server, a bash language script, named "`export_data.sh`", must be copy to the server and the command line must be called in the server.

Before calling this script, the ID of the pretended online workflows must be included inside the batch file, for example, like in the following line code:

```
for i in 20170401_0030_20171127164817978 20170416_0030_20171127170150804
20170416_0130_20171127280150521
```

In this case, the script will create a folder for each workflow "`20170401_0030_20171127164817978`", "`20170416_0030_20171127170150804`" and "`20170416_0130_20171127280150521`", and inside the folder will store the csv files outputted by the specified itools commands.