**Residential EV Charging Workflow**

Step0: Monte-Carlo Simulation for Markov Behavior Model

* Inputs/scritpsMCS.py: set the statistical parameters and obtain a Markov csv

**Alternative 1: Optimizing EV first**

Step1: Setting the parameters

* Set scenario parameters such as “*ev\_bat\_capacity”* etc.
* Don’t forget to check assumption and penalization parameters (unitconsumption, drop penalty)
* Input files for forecast, dso command etc (they have to be given to MaximizePV class as dictionaries)

FIND in AllApproaches\EVFirst\ RealCapacity\_maxPV\_independent.py

* Run import\_statistic() function with optimization starting hour

This function imports markov model from csv file according to the hour to be optimized

FIND in AllApproaches\function.py

Step2: Optimize the EV charging power with Dynamic Programming

* MaximizePV .solve\_dynamicprogram(): Computes optimal values and decisions for each node in the DP map. States are combination of EV SoC- EV Position

Step3: Obtain EV position and SoC trajectory for the full horizon

* MaximizePV .estimate\_ev\_charging() : random numbers are drawn subsequently for each hour

Step4: Use the estimated trajectories to optimize ESS

* optimize\_full\_EM()

**Alternative 2: Simultaneous optimization**

Step1: Setting the parameters

* Set scenario parameters such as “*ev\_bat\_capacity”* etc.
* Set DP parameters ev\_soc\_domain, ev\_decision\_domain, ess\_soc\_domain, ess\_decision\_domain
* Don’t forget to check assumption and penalization parameters (unitconsumption, drop penalty)
* Input files for forecast, dso command etc (they have to be given to MaximizePV class as dictionaries)
* FIND in AllApproaches\Simultaneous\ RealCapacity\_maxPV\_independent.py
* Run import\_statistic() function with optimization starting hour

This function imports markov model from csv file according to the hour to be optimized

FIND in AllApproaches\function.py

Step2: Optimize the EV and ESS simultaneously

* MaximizePV .solve\_dynamicprogram(): Computes optimal values and decisions for each node in the DP map. States are combination of ESS SoC- EV SoC- EV Position