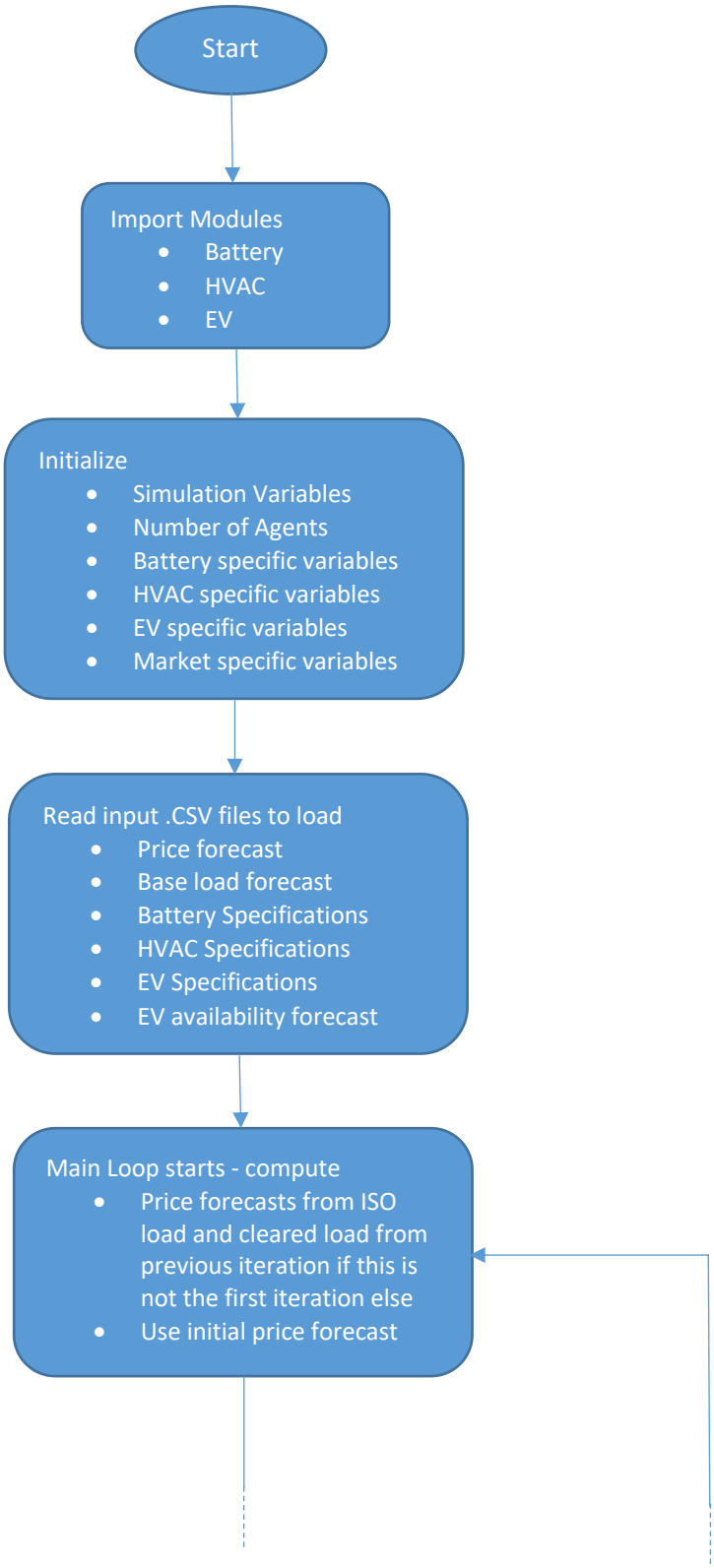
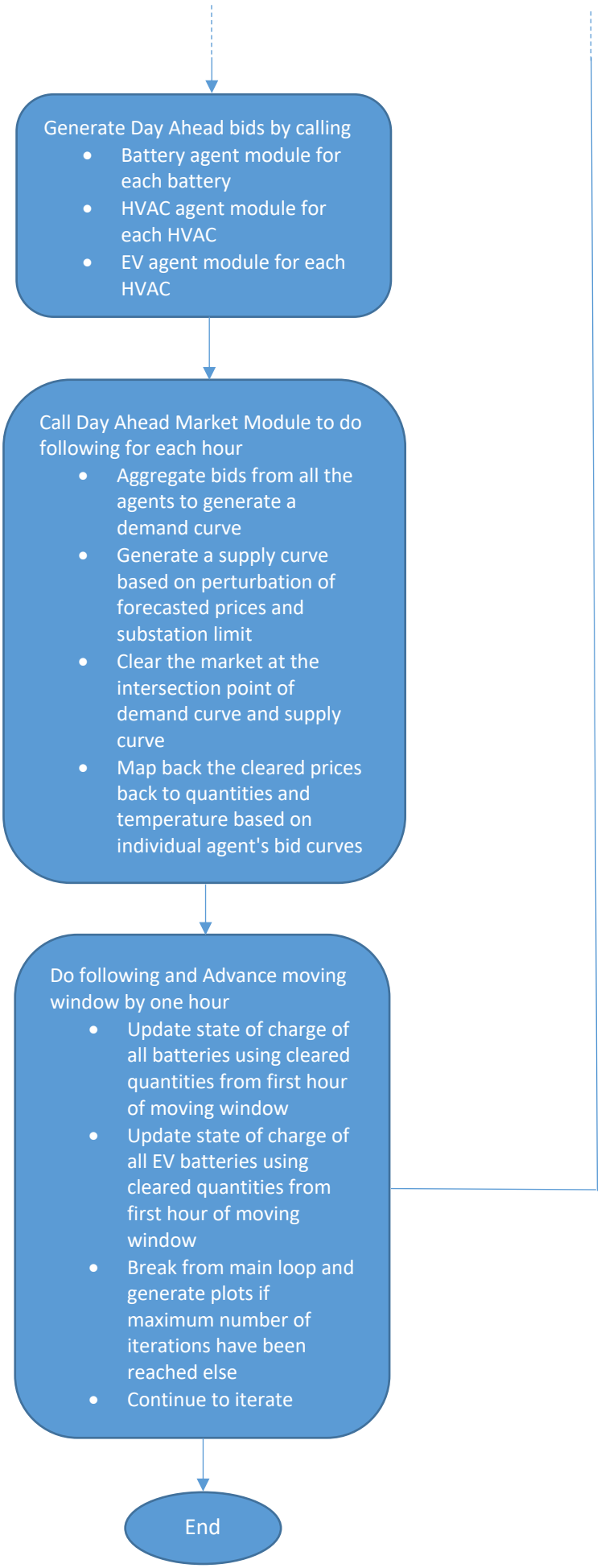
Master Script Documentation

v1.0 11/21/2018

This document explains master script for day ahead market simulation by first presenting code flow and then the interaction of master script with each individual module.

Flow chart explaining code Flow




Generate Day Ahead bids by calling
Battery agent module for each battery
HVAC agent module for each HVAC
EV agent module for each HVAC
Call Day Ahead Market Module to do following for each hour
Aggregate bids from all the agents to generate a demand curve
Generate a supply curve based on perturbation of forecasted prices and substation limit
Clear the market at the intersection point of demand curve and supply curve
Map back the cleared prices back to quantities and temperature based on individual agent's bid curves
Do following and Advance moving window by one hour
Update state of charge of all batteries using cleared quantities from first hour of moving window
Update state of charge of all EV batteries using cleared quantities from first hour of moving window
Break from main loop and generate plots if maximum number of iterations have been reached else
Continue to iterate
End

Data exchange between master script and individual modules

The interaction of Master script with individual modules is shown in Figure 1. The data exchange back and forth between master script and individual modules is listed in Tables 1, 2, 3 and 4 given in Appendix. Note that all variables are named as seen from the side of each individual module.

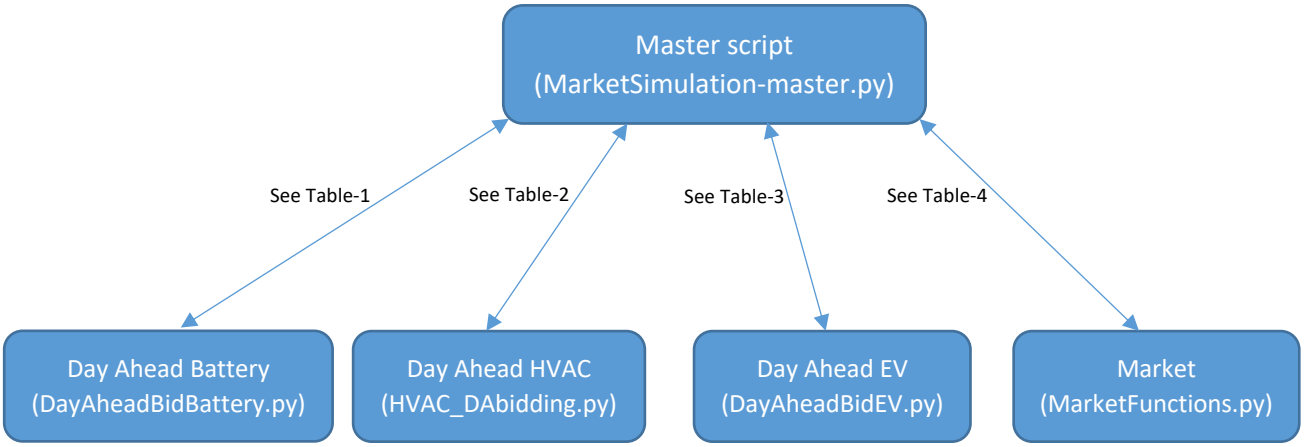


Figure 1. Interaction of Master Script with individual modules

Appendix

Table 1 Data Exchange between Master script and Day Ahead Bid Battery module

From Master Script to Day Ahead Bid Battery module			
Parameter Name	**Function involved**	**Parameter Type and size**	**Parameter Description**
R_c	DayAheadBid	Float (1x1)	Rated charging power in kW for the battery
R_d	DayAheadBid	Float (1x1)	Rated discharging power in kW for the battery
L_{in}	DayAheadBid	Float (1x1)	Battery charging loss in %
L_{out}	DayAheadBid	Float (1x1)	Battery discharging loss in %
C_{min}	DayAheadBid	Float (1x1)	Minimum allowable stored energy in kWh (state of charge lower limit)
C_{max}	DayAheadBid	Float (1x1)	Maximum allowable stored energy in kWh (state of charge upper limit)
C_{init}	DayAheadBid	Float (1x1)	Initial stored energy in the battery in kWh
ProfitMargin	DayAheadBid	Float (1x1)	Specified in % and used to modify slope of bid curve. Set to 0 to disable
BindingObjFunc	DayAheadBid	Boolean (1x1)	If True, then optimization considers cleared price, quantities from previous iteration in the objective function
f_{DA}	DayAheadBid	List of Floats (1 x *windowLength*)	Forecasted prices in $/kWh for all the hours in the duration of *windowLength*
prev_clr_Quantity	DayAheadBid	List of Floats (1 x *windowLength*)	Cleared quantities (kWh) from previous market iteration for all hours
prev_clr_Price	DayAheadBid	List of Floats (1 x *windowLength*)	Cleared prices ($/kWh) from previous market iteration
batteryLifeDegFactor	DayAheadBid	Float (1x1)	Constant to model battery degradation
bidSpread	DayAheadBid	Integer (1x1)	This can be used to spread out bids in multiple hours. When set to 1 hour (recommended), it's effect is none
windowLength	DayAheadBid	Integer (1x1)	Length of day ahead optimization period in hours (e.g. 48 hours)

dayAheadCapacity	DayAheadBid	Float (1x1)	% of battery capacity reserved for day ahead bidding
From Day Ahead Bid Battery module to Master Script			
Parameter Name	**Function Involved**	**Parameter Type and size**	**Parameter Description**
CurveSlope	DayAheadBid	List of Floats (1 x *windowLength*)	Slopes of bid curves for all hours of the window specified by *windowLength*
yIntercept	DayAheadBid	List of Floats (1 x *windowLength*)	y-intercept of bid curves for all hours of the window specified by *windowLength*
bidTrack	DayAheadBid	List of Integers (1 x *windowLength*)	Specified for all hours of the window = -1 in case of no bid = 0 in case of charging bid = 1 in case if discharging bid
bidMade	DayAheadBid	List of Floats (1 x *windowLength*)	Optimal quantity from optimization for all hours of the window specified by *windowLength*

Table 2 Placeholder for HVAC table

Parameter Output	Parameter Description	

Table 3 Data Exchange between Master script and Day Ahead Bid EV module

From Master Script to Day Ahead Bid EV module			
Parameter Name	**Function Involved**	**Parameter Type and size**	**Parameter Description**
R_c	DayAheadBid	Float (1x1)	Rated charging power in kW for the EV battery
R_d	DayAheadBid	Float (1x1)	Rated discharging power in kW for the EV battery
L_{in}	DayAheadBid	Float (1x1)	EV Battery charging loss in %
L_{out}	DayAheadBid	Float (1x1)	EV Battery discharging loss in %
C_{min}	DayAheadBid	Float (1x1)	Minimum allowable stored energy in kWh (state of charge lower limit)
C_{max}	DayAheadBid	Float (1x1)	Maximum allowable stored energy in kWh (state of charge upper limit)
C_{init}	DayAheadBid	Float (1x1)	Initial stored energy in the EV battery in kWh
ProfitMargin	DayAheadBid	Float (1x1)	Specified in % and used to modify slope of bid curve. Set to 0 to disable
BindingObjFunc	DayAheadBid	Boolean (1x1)	If True, then optimization considers cleared price, quantities from previous iteration in the objective function
f_{DA}	DayAheadBid	List of Floats (1 x *windowLength*)	Forecasted prices in $/kWh for all the hours in the duration of *windowLength*
prev_clr_Quantity	DayAheadBid	List of Floats (1 x *windowLength*)	Cleared quantities (kWh) from previous market iteration for all hours
prev_clr_Price	DayAheadBid	List of Floats (1 x *windowLength*)	Cleared prices ($/kWh) from previous market iteration
batteryLifeDegFactor	DayAheadBid	Float (1x1)	Constant to model EV battery degradation
bidSpread	DayAheadBid	Integer (1x1)	This can be used to spread out bids in multiple hours. When set to 1 hour (recommended), it's effect is none
windowLength	DayAheadBid	Integer (1x1)	Length of day ahead optimization period in hours (e.g. 48 hours)

dayAheadCapacity	DayAheadBid	Float (1x1)	% of EV battery capacity reserved for day ahead bidding
EV_Avail	DayAheadBid	List of Booleans (1 x *windowLength*)	Vehicle availability forecast for each hour of the window. Specifies if the vehicle is available (to grid) or not (being driven) =1 Vehicle available to grid =0 Vehicle not available to grid (being driven)
V2G	DayAheadBid	Boolean (1x1)	When True, models V2G EV. Set to False for V1G EV variant
EV_hourly_consumption	DayAheadBid	Float (1x1)	Energy (kWh) consumed by EV when it is driven
From Day Ahead Bid EV module to Master Script			
Parameter Name	**Function Involved**	**Parameter Type and size**	**Description**
CurveSlope	DayAheadBid	List of Floats (1 x *windowLength*)	Slopes of bid curves for all hours of the window specified by *windowLength*
yIntercept	DayAheadBid	List of Floats (1 x *windowLength*)	y-intercept of bid curves for all hours of the window specified by *windowLength*
bidTrack	DayAheadBid	List of Integers (1 x *windowLength*)	Specified for all hours of the window = -1 in case of no bid = 0 in case of charging bid = 1 in case if discharging bid
bidMade	DayAheadBid	List of Floats (1 x *windowLength*)	Optimal quantity from optimization for all hours of the window specified by *windowLength*

Table 4 Data Exchange between Master script and Market module

From Master Script to Market module			
Parameter Name	**Function Involved**	**Parameter Type and size**	**Parameter Description**
max_price	demand_aggregator	Float (1x1)	Maximum price ($/kWh) for aggregate demand curve
m_Bat	demand_aggregator	List of Floats (1 x NumberOfBatteries)	Slope of bid curves for all batteries
c_Bat	demand_aggregator	List of Floats (1 x NumberOfBatteries)	y-intercept of bid curves for all batteries
baseload	demand_aggregator, marketClearing	Float (1x1)	Current base load
bt_Bat	demand_aggregator	List of Floats (1 x NumberOfBatteries)	Specifies bid type for all Batteries = -1 in case of no bid = 0 in case of charging bid = 1 in case if discharging bid
dataPoints	demand_aggregator	Integer (1x1)	Number of data points in aggregate demand curve
batMaxChargeRate	demand_aggregator	List of Floats (1 x NumberOfBatteries)	Rated maximum charging power in kW for each battery
batMaxDischargeRate	demand_aggregator	List of Floats (1 x NumberOfBatteries)	Rated maximum discharging power in kW for each battery
energyConstraintCharge	demand_aggregator	List of Floats (1 x NumberOfBatteries)	Max kWh – currently stored kWh for each battery
energyConstraintDischarge	demand_aggregator	List of Floats (1 x NumberOfBatteries)	currently stored kWh – Min kWh for each battery
m_hvac	demand_aggregator, clearQuantitiesHVAC, clearTemperatureHVAC,	List of Floats (1 x NumberOfHVACs)	Slope of bid curves for all HVACs
c_hvac	demand_aggregator, clearQuantitiesHVAC, clearTemperatureHVAC,	List of Floats (1 x NumberOfHVACs)	y-intercept of bid curves for all HVACs
max_quan_hvac	demand_aggregator, clearQuantitiesHVAC	Float (1x1)	Maximum bid quantity for an HVAC (assumed same for all)
bt_hvac	demand_aggregator	List of Floats (1 x NumberOfHVACs)	Specifies bid type for all HVACs = -1 in case of no bid = 0 in case of consumption bid

m_EV	demand_aggregator	List of Floats (1 x NumberOfEVs)	Slope of bid curves for all EVs
c_EV	demand_aggregator	List of Floats (1 x NumberOfEVs)	y-intercept of bid curves for all EVs
bt_EV	demand_aggregator	List of Floats (1 x NumberOfEVs)	Specifies bid type for all EV Batteries = -1 in case of no bid = 0 in case of charging bid = 1 in case if discharging bid
EvbatMaxChargeRate	demand_aggregator	List of Floats (1 x NumberOfEVs)	Rated maximum charging power in kW for each EV battery
EvbatMaxDischargeRate	demand_aggregator	List of Floats (1 x NumberOfEVs)	Rated maximum discharging power in kW for each EV battery
EvenergyConstraintCharge	demand_aggregator	List of Floats (1 x NumberOfEVs)	Max kWh – currently stored kWh for each EV battery
EvenergyConstraintDischarge	demand_aggregator	List of Floats (1 x NumberOfEVs)	currently stored kWh – Min kWh for each EV battery
dp	marketClearing	List of Floats (1 x *dataPoints*)	Demand prices forming the aggregate demand curve
dq	marketClearing	List of Floats (1 x *dataPoints*)	Demand quantities forming the aggregate demand curve
sp	marketClearing	Float (1x1)	Supply price ($/kWh)
ssl	marketClearing	Float (1x1)	Substation limit (kW)
clrPrice	clearQuantitiesHVAC	Float (1x1)	Cleared price ($/kWh)
clrquan	clearTemperatureHVAC	Float (1x1)	Cleared quantity (kW) for the hour
temp_min	clearTemperatureHVAC	Float (1x1)	Minimum temperature output
temp_max	clearTemperatureHVAC	Float(1x1)	Maximum temperature output
From Market module to Master Script			
Parameter Name	**Function Involved**	**Parameter Type and size**	**Description**
prices	demand_aggregator	List of Floats (1 x *dataPoints*)	Prices forming the aggregate demand curve
quantity	demand_aggregator	List of Floats (1 x *dataPoints*)	Quantities forming the aggregate demand curve
dp[index]	marketClearing	Float (1x1)	Cleared price ($/kWh)
dq[index]	marketClearing	Float (1x1)	Cleared quantity (kW) for the hour
Cleared HVAC quantity	clearQuantitiesHVAC	Float (1x1)	Cleared quantity for HVAC
Cleared HVAC temperature	clearTemperatureHVAC	Float (1x1)	Cleared temperature for HVAC

Cleared quantity battery	clearQuantities	Float(1x1)	Cleared quantity for Battery (kWh)