1. **Introduction:** The present work describes the different sensitivity scenarios and simulated conditions under which we conduct the capacity expansion planning studies using GenX, a highly reconfigurable Mixed-Integer Linear Programming (MILP) based capacity-expansion planning model. In this set of simulations, we consider
2. **Data Processing:** In the present work, we have run the GenX simulations and analyzed the cases pertaining to the year, 2030. According to the EIA reports, there’s a 7.5% (approx.) demand growth from the year 2020 t0 2030. For the purposes of running the GenX model, we need the following input data:
   * Generators\_data.csv: This consists of data pertaining to existing capacity (projected and scaled back to 2020 values), new-built units, fuel types, costs etc. organized into several clusters
   * Generators\_variability.csv: This consists of the values of capacity factors of each of the generator clusters listed in “Generators\_data.csv” for an entire year, with an hourly resolution variation
   * Load\_data: This lists the hourly resolution based MW load consumption values for an entire year.
   * Fuels\_data: This input table lists the fuel cost in $/MMBtu and CO2 content in tons/MMBtu for the following fuel types: coal, natural gas, uranium, and natural gas\_ccs100

All the above input files for GenX are automatically produced by the Python based open-source software, “PowerGenome” (<https://github.com/gschivley/PowerGenome>), which stitches together data from Annual Energy Outlook (AEO), NREL Annual Technology Baseline (ATB) etc. in order to produce the above data suitable for GenX model, to operate on. The particular branch that has been used in PowerGenome for this work is “Sams\_USA\_Single.” Here, we have run the powergenome for a single aggregated IPM region (which we have named “USA\_Single”) for the entire USA.

Data Processing for “Load\_data.csv”: This file lists the hourly values of demand consumption for the year, 2030, for which we are running the GenX model. however, we have normalized the load values against an assumed peak of 100000 MW, for the sake of ease of comparison when we look at the scenarios where we consider an equivalent load for entire USA corresponding to Texas, California, and New York state-wide consumption values. The scaling is done according to:

Data Processing for Generators\_data: The Existing\_Cap\_MW field in the file gives the currently installed capacities of the different generator clusters. We have scaled these values taking into consideration that there is a 7.5 % demand growth from 2020 to 2030 (according to EIA) and so, projecting back to 2020 installed capacities, now that we have normalized the demand values against a peak of 100000 MW, therefore, if the 2030 demand peak is 100000 MW (instead of the actual, 802010 MW), the scaled existing capacities would be given by:

The above equation essentially tells that, if we were to assume that 100000 MW is the peak demand in 2030 and scale back the existing MW generation capacities to 2020 values (taking into account the 7.5% demand growth) then what those values would have been. We have applied the same scaling to “Max\_Cap\_MW” field for the new-built resource clusters for “landbasedwind” and “utilitypv\_losangeles” as well.

Additionally, in order to reduce the computational burden, without compromising on the quality of the solution, we have only considered those generator clusters for which, the “Heat\_rate\_MMBTU\_per\_MWh” has values lying between 3 and 15. We also ignored those clusters for which the “Fixed\_OM\_cost\_per\_MWyr” is zero. All of these cluster pruning and data preprocessing before feeding the input data to GenX is automatically performed by the Python script, “genClusterRem.py.” This does the cluster pruning, scaling of “Existing\_Cap\_MW” as well as the “Max\_Cap\_MW” values and rounding them off to the nearest integers, rounds off the “Inv\_cost\_per\_MWyr” and “Fixed\_OM\_cost\_per\_MWyr” to the nearest integers, sets the “Cap\_size” for all the clusters to 1, if the “UCommit” option is “GenX\_settings.yml” is set to 2 (in which case, the unit commitment problem is also solved). It also renumbers the “R\_ID” of the generator clusters after pruning, so that they are ion sequence. Finally, it also prunes the corresponding columns in “Generators\_variability.csv” file, corresponding to which the clusters in “Generators\_data.csv” has been deleted. With all these changes, we expect to see a lot more new units to be built for the 2030 year, when we run GenX.

1. **Results:** Scenarios considered:

* **Base-case/No Policy (without CO2 Tax):** The base case scenario is the status-quo case without any policy
* **High Wind case (without CO2 Tax):** Using the ATB2020 low costs (FOM, capex etc.) only for the new-build wind clusters (projected for 2030)

* **High Solar case (without CO2 Tax):** Using the ATB2020 low costs (FOM, capex etc.) only for the new-build solar clusters (projected for 2030)
* **NY Winter Peaking (without CO2 Tax):** Use the New York load profile for 2030, where the peak load is normalized to 100 GW
* **Base-case/No Policy (with CO2 Tax):** Base case with $60 CO2 tax to breakeven the new-built NGCC plants with CCS versus the new-built NGCC plants without CCS
* **High Wind case (with CO2 Tax):** Using the ATB2020 low costs (FOM, capex etc.) only for the new-build wind clusters (projected for 2030) with $60 CO2 tax
* **High Solar case (with CO2 Tax):** Using the ATB2020 low costs (FOM, capex etc.) only for the new-build solar clusters (projected for 2030) with $60 CO2 tax
* **NY Winter Peaking (with CO2 Tax):** Use the New York load profile for 2030, where the peak load is normalized to 100 GW with $60 CO2 tax
* **Repeat all the scenarios above with Capacity Reserve Margin imposed:** We repeat all the cases above with 0.8 of the maximum capacities of the new-built wind and solar clusters as the Capacity Reserve Margins

Shown below are the price-duration curves (for $9000 and $2000 price cap for the scenarios without and with Capacity reserve margin enforced, respectively). Figure 1 shows the price-duration curve of the four cases above, without any CO2 tax and without capacity margin reserve constraints enforced. Figure 2 shows the price-duration curve without the CO2 tax, but with the Capacity Reserve Margin constraints enforced. Figure 3 and 4 are respectively, the $60/ton CO2 tax imposed-counterparts of the plots shown before. Please refer to the “Duration curves” tab of the “Price\_Series\_Summary\_Comprehensive.xlsx” for the details, which shows all the price-duration curves for all the scenarios mentioned here. In the same spreadsheet, the “Case Descriptions”, “Prices”, and “Week\_Viewer” tabs detail, respectively, the descriptions of the different scenarios (described here in section 3), the extended list of the prices for all the 8760 hours for each scenario, and the weekly view of the prices.

Figure 1: No CO2 Tax, No Capacity Reserve Margin, $9000 Price cap Price-Duration Curves

Figure 2: No CO2 Tax, with Capacity Reserve Margin, $2000 Price cap Price-Duration Curves

Figure 3: $60/ton CO2 Tax, No Capacity Reserve Margin, $9000 Price cap Price-Duration Curves

Figure 4: $60/ton CO2 Tax, with Capacity Reserve Margin, $2000 Price cap Price-Duration Curves