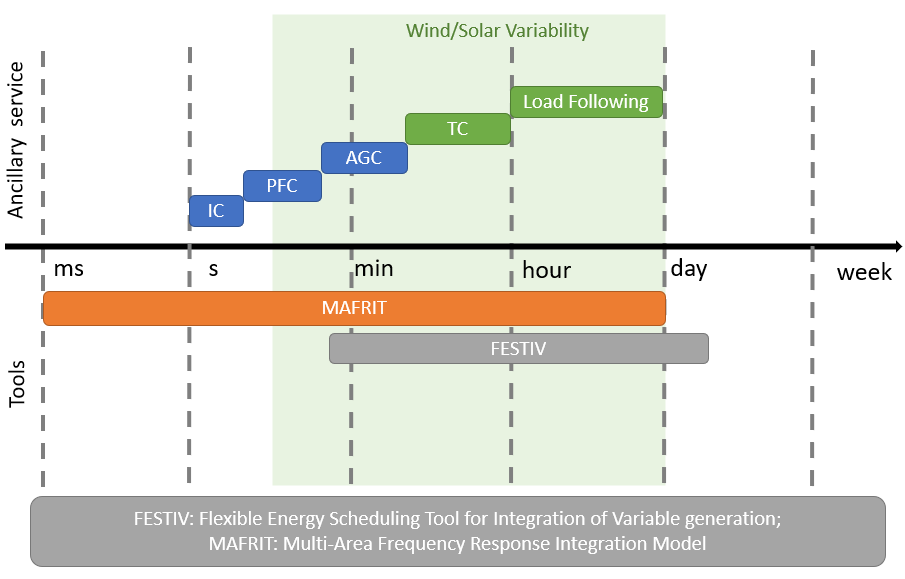
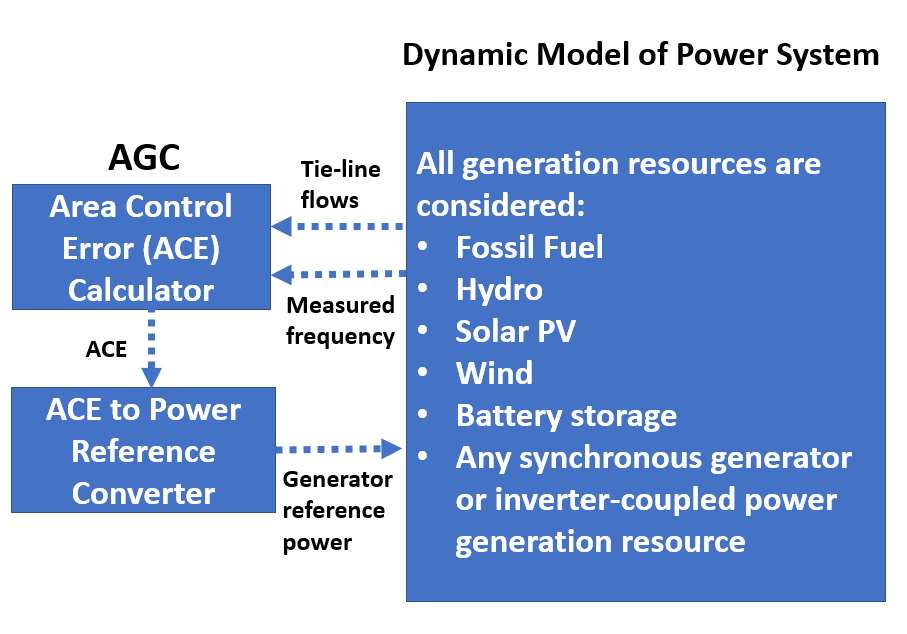
1. **Overview of MAFRIT**

The Multi-Area Frequency Response Integration Tool (MAFRIT) is primarily designed to enable research on high to very high variable renewable energy (VRE) integration. Being an open source tool, researchers in the broader power systems community can use, modify, and expand it to explore the impacts of very high variable renewable energy integration on the reliability of power systems. MAFRIT has been developed at NREL primarily to fill the gap between production cost simulation or scheduling tools and traditional electromechanical transient simulation tools. Production cost simulation tools typically do not consider ac power flow or dynamics of power systems that are needed to model power system frequency response. Similarly, electromechanical dynamics simulations tools are not typically designed to simulate long-term dynamics that are essential to capturing the AGC-timeframe dynamics that have a significant impact on frequency response performance of power systems. The capability offered by MAFRIT to bridge the gap between these two categories of tools has and will become increasingly important as VRE integration increases in the grid and concerns arise regarding the impacts of reduced system inertia and variability and uncertainty of power outputs of VREs on frequency response of power systems. Figure 1 shows how MAFRIT fills the gap between the two categories of tools.

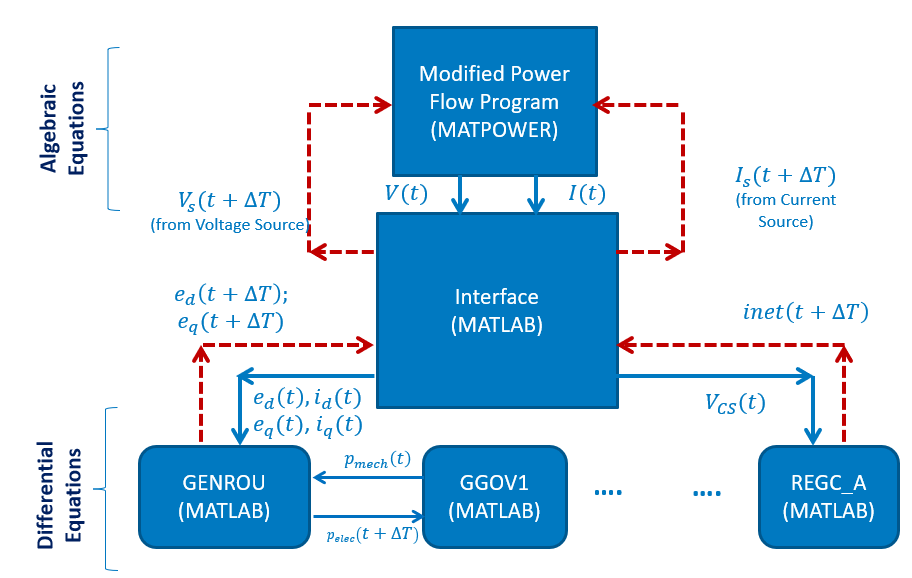


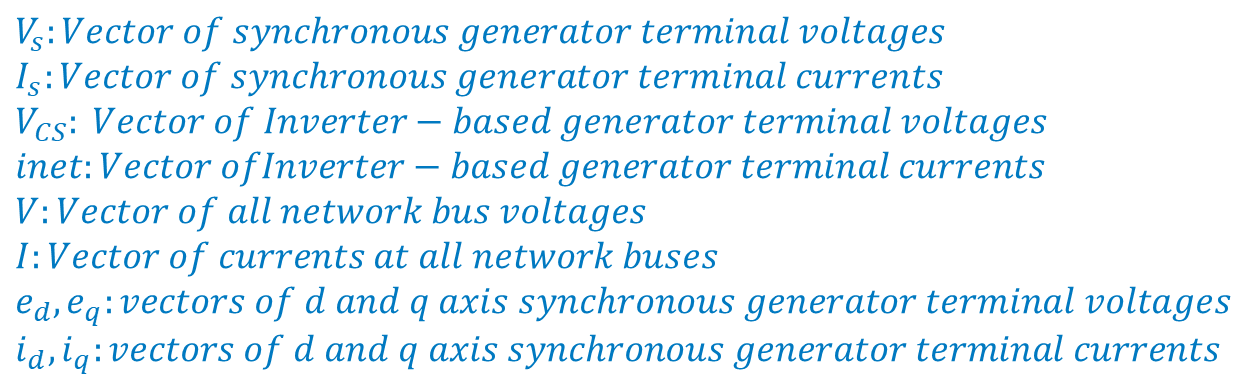
**Figure 1 – MAFRIT integrates Scheduling & Dynamics Simulations**

Figure 2 shows the conceptual diagram of MAFRIT. As shown in the figure, MAFRIT has two distinct modules. The first is the dynamic simulation module that simulates the electromechanical transients of power systems by solving a set of differential algebraic equations (DAEs). The DAEs are solved using sequential-implicit approach meaning that algebraic and differential equations are solved sequentially with implicit integration methods used for solving the differential equations. Figure 3 shows the data flow in the dynamic simulation module. As shown in the figure, MATPOWER, with some modifications, is used for solving the algebraic equations. The network model is also represented in the format used in MATPOWER to model the power flow cases.



**Figure 2 – Key modules of MAFRIT**

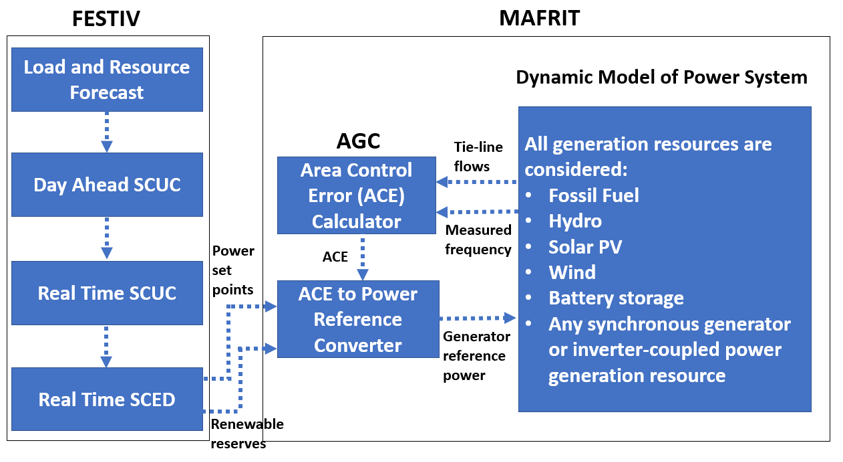




**Figure 3 – Solution of DAEs in MAFRIT**

The second module is the AGC module. Both single area and multi area AGC modules can be easily added in MAFRIT. When AGC is used for a single area it attempts to bring the frequency back to its scheduled value. In the multi-area AGC module, AGC attempts to maintain both the frequency and the tie-line flows at their scheduled values.

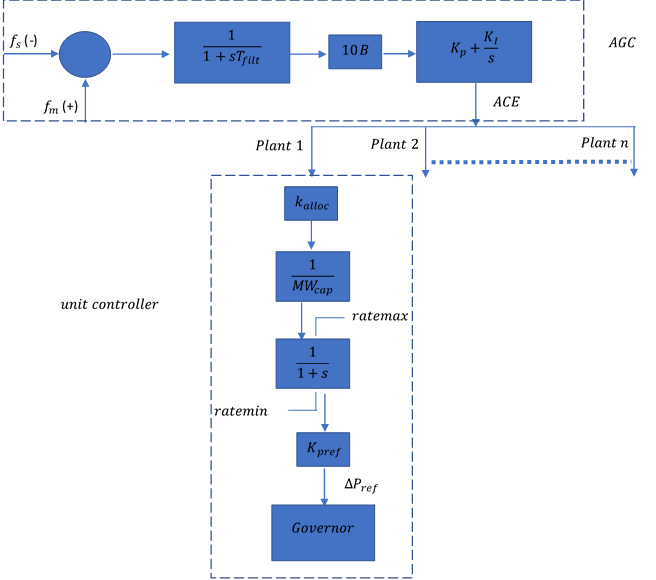
MAFRIT can be integrated with any scheduling tool. At NREL we have primarily integrated it with FESTIV. The integration between FESTIV and MAFRIT is shown in Figure 4. FESTIV provides generation real power dispatch set points and the available real power reserve for renewable generators for a time point of interest, these are translated into an ac power flow case, and the resulting solved power flow case is used to simulate the frequency response under load and generation variations or for the loss of a generator, or for any other contingency of interest.



**Figure 4 – FESTIV and MAFRIT Integration**

1. **Single area AGC and unit controller model for MAFRIT**

The model discussed below is for an isolated system where the entire system is just one area. Therefore, objective of the AGC is to simply restore the frequency to the scheduled value. The AGC model is quite simple but should do the job provided appropriate parameters relevant to the system being studied are used. The model is shown in Figure 5.



**Figure 5 – AGC and Unit Controller Models**

**Equations of AGC**

**Equations of Unit Controller**

If

Else if

Else if

**Initialization**

**AGC:**

Setting (2) and (6) to zero imply that both states 1 and 2, i.e., and are zero.

**Unit Controller:**

Since is zero at initial steady-state, (assuming everything is in the linear region)

1. **Steps to use MAFRIT**

The “mainProgram.m” file is the main file that coordinates the DAE solution between MATPOWER and MATLAB’s ODEs. Several comments have been included in this script that should explain what each part of the code does. Before this file can be used, however, we need the following data:

**1. Power flow case**. This case is expected to be in the case format used in MATPOWER. Following changes ***must*** be made to the case file before it can be used with MAFRIT:

* The user is required to stack all the generator buses modeled as voltage sources (e.g., synchronous generators using the genrou model) on top of the bus table, followed by all the buses that interface with the network as current sources (e.g., synchronous generators modeled using genrouCS model, wind generators, battery storage, and central PV plants), and the rest of the buses should be at the bottom of the table.
* The first row in the bus table should correspond to the swing bus.
* The bus type of the wind generators should be changed to ‘1’ to represent them as load buses in the network.
* All the rows corresponding to wind generators should be removed (or their status made 0) from the generator data table, and the active and reactive powers of wind generators should be assigned with a negative sign to the buses in the bus table.
* A generator bus cannot have load present on it. Including this capability is not difficult, but it is not available yet. An alternative is to add dummy buses connected with the generator bus by low impedance and add generator loads to the dummy buses.
* At present, more than one generator cannot be connected to a generator bus. So, user should consider creating an equivalent generator model if more than one unit is connected at the same bus.
* Voltage and current sources cannot be connected to the same bus.

Besides the above mandatory changes to the power flow case file, no other change is required but following tips will be helpful in the smooth execution of the program:

* MATPOWER cannot represent a constant current load at present, only constant power and constant impedance loads can be modeled.
* Constant P and constant Q values of loads are assigned to the 3rd and 4th columns of the bus table, respectively.
* Constant Impedance loads are represented as equivalent P and Q values at 1.0 p.u. voltage. These are put in the 5th and 6th columns of the bus table, respectively, and Q value should be negative of the load reactive power (e.g., if the load has 30 MVAr of inductive component at 1.0 p.u. voltage, then the 6th column should read -30 MVAr).

Once the power flow case is prepared as above, solve it using the runpf() function and save the workspace (after removing everything else except for the output of runpf()) as a .mat file. Specify the path to this file in the mainProgram.m at the location indicated in the file.

**2. Dynamic Model Parameters:** These parameters should be specified in the same order as in the text files of the existing dynamic models in the “Data file formats” folder. If a user adds a new dynamic model, any desired order of parameters may be used.

3. **Configuration file:** This file, called “power\_plant\_specs.txt”, is used to tell “mainProgram.m” the bus numbers where each dynamic model is connected, the dynamic model of generators used at that bus, their controllers, and whether the generator is on AGC. The parameter files’ names and the names of the text files where dynamic models’ states are saved during the simulation are also specified in this file. The AGC model is added as the last entry of this file and the file should end with “end”. **Each row corresponds to one generator and the order in which the buses appear should be identical to the order in which voltage and current source buses are arranged in the power flow case as discussed earlier**. The program doesn’t read anything after “end”. The exact format of the file is described in the “mainProgram.m” file.

Everything else should be fairly self-explanatory. For any questions, please feel free to create an issue in github.