TO: WECC REMWG

FROM: POUYAN POURBEIK, PEACE®

SUBJECT: PROPOSAL FOR NEW PLANT CONTROLLER AND ELECTRICAL CONTROLLER

DATE: 5/13/22 (REVISED 9/19/22; 2/6/23; 7/5/23; 8/18/23; 9/7/23; 9/15/23; 12/14/23; 1/9/24;1/10/24;1/15/24)

CC: D. RAMASUBRAMANIAN, J. BOEMER, AND A. GAIKWAD, EPRI

Revision Note:

This is the fourth revision of this memo capturing comments and edits from several webcast meetings of the core team, plus items discussed at the last several WECC MVS meetings. The changes are mainly editorial changes/fixes and clarifications of the model elements. These will need to be discussed and refined, and then implemented for benchmark testing and final approval.

1.0 Electrical Control Model:

Presently, there are three (4) existing REEC_* models in the WECC RES generic models:

1. **REEC_A:** the most commonly used model for both wind and PV plants.
2. **REEC_B:** a simplified version of the electrical controls, which was previously used for PV plants but is no longer accepted and has been disbanded. **This model should not be used moving forward.**
3. **REEC_C:** a model intended primarily for use in modeling battery energy storage systems (BESS).
4. **REEC_D:** the most recent electrical controller model for use with wind, PV and BESS.

As indicated above, the REEC_B model is no longer accepted for use by WECC (and other regions) since it is devoid of the ability to represent the voltage-dependent limits (VDL) on the inverter current, and cannot emulate inverter blocking. The REEC_A and REEC_C models, although quite comprehensive, and though both do include the VDL tables, there are some improvements that where pointed out in recent past by several vendors, and thus REEC_D incorporates these improvements (see [1]).

Here in this memo we present yet another proposed new version of the electrical controls model, which is called REEC_E, to provide even more updates. This is not intended to replace the earlier versions (REEC_A, REEC_C and REEC_D) but simply to give even more features for modeling newer plants.

1.1 REEC_E (new model)

The REEC_E (Figure 1) model is identical to the REEC_D model [1], with the following additions/modifications:

1. Allow for local PI Q or pf control: Add another path forward through *QFlag* to allow for local PI control of constant-Q or constant-pf.
2. Active power PI control: Add a new flag *PEFlag* and two additional PI gains *Kpp*/*Kpi* to allow for local PI control of active power.
3. P/Q-priority during Faults: Add a new flag, *PqflagFRT*, to allow for a different P/Q priority during fault conditions.

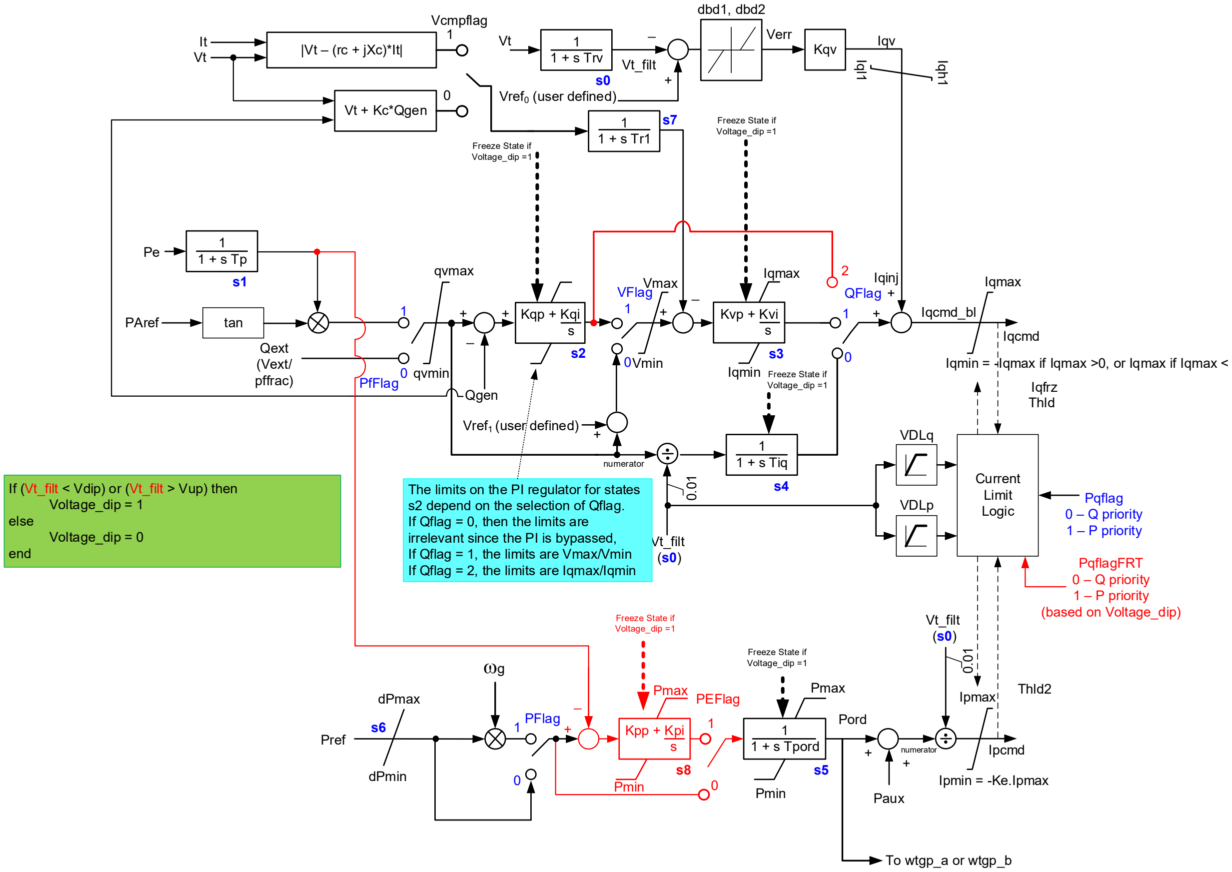


Figure 1: REEC_E model.

Table 1: Additional Parameters REEC_E – only new additional parameters are listed here; all other parameters are identical to REEC_D.

Parameter	Description	Typical Range/Value
Qflag	2 – new option to allow for PI control of constant-Q or constant-pf	N/A
PEFlag	0 – no PI control of active power 1 – allow for local PI control of active power	N/A
PqflagFRT	0 – Q priority and 1 – P priority Gets invoked under fault conditions (when Voltage_dip = 1) and can be set opposite to the *Pqflag*, which applies to no-fault (Voltage_dip = 0) conditions	N/A
Kpp	Proportional control gain for active power [pu/pu]	N/A
Kpi	Integral control gain for active power [pu/pu/s] **Important Note:** if *PEFlag* = 1 then Kpi MUST be greater than 0	N/A

2.0 Plant Controller Models

4.1 REPC_D (new model)

This proposed new plant-controller model is based off of the existing REPC_C model, which interfaces to a single aggregated WTG model. This model then builds on REPC_C to make it like REPC_B for controlling multiple aggregated renewable system models downstream, but without some of the limitations of REPC_B. Thus, REPC_D will have the exact same structure as REPC_C with the only change being the addition that which is shown in RED in Figure 2, to allow the plant controller to control multiple aggregated units.

Figure 2 shows the proposed REPC_D model. In essence the only difference between REPC_C and REPC_D is the end interface. The concept behind the interface may be explained as follows:

1. The model has its own MVA base, as with REPC_C (*MVAplt*). The user written parameter behaves in the following way:
   a. If the user inputs a value for *MVAplt*, then this is treated as a "fixed" MVA base and all the model parameters are on this fixed base.
   b. If the user inputs a value of "0" for MVAplt, then *MVAplt* is calculated by the model (both at initialization and thereafter at every time-step) as = {the sum of the MVA base of all down-stream units that are on-line, i.e. status of the models = 1}.
2. All the down-stream devices must receive a Q-command. Mixing down-stream devices with Q-command and voltage-reference control is more complex and outside of the present context. Likewise, for down-stream devices where all receive a voltage-reference from the plant controller is to be considered in the future.
3. Here we will assume that there can be up to fifty (50) downstream inverter-based resources (IBR).
4. If the initial power flow condition is such that:
   a. IBR 1 has an output of $Q1_o$ (MVar) and $P1_o$ (MW), and
   b. IBR 2 has an output of $Q1_2$ (MVar) and $P1_2$ (MW), etc.

   then upon initialization

   $$Qref_o = \frac{\sum_{i=1}^{n} Qi_o}{MVAplt} \text{ and } Pref_o = \frac{\sum_{i=1}^{n} Pi_o}{MVAplt}$$

   $$qref_i = \frac{Qi_0}{MVA_i} \text{ and } pref_i = \frac{Pi_0}{MVA_i}$$

   By initializing the model in this way (as opposed to the current REPC_B model), the limitations of the REPC_B model are removed. This is because, the initial Q-command and P-command will initialize to the total sum of the Q-output and P-output of the down-stream connected devices, in per-unit on the total plant MVA base. Moreover, the model will always flat-start for a no-disturbance simulation. **Important Note:** this does, however, put burden on the user to ensure that the initial power flow solution is meaningful.

   As for the gains (Kw_i and Kz_i), these are entered by the user, such that for the vast majority of cases they are all to be set to 1. When all the K's are set to 1, this is interpreted as meaning that the scaling is based on MVA ratios. That is, each downstream plant picks-up power real/reactive in proportion to its MVA base. So for example, if we have two (2) downstream plants each of rating 100 MVA, then the total hybrid-plant rating is 200 MVA. So if the hybrid-PPC asks for 0.1 pu increase in MW (i.e. 0.1 × 200 = 20 MW), then 10 MW (0.1 pu on 100 MVA base) will be provided by the first 100 MVA plant, and 10 MW (0.1 pu on 100 MVA). So it is easy to see that the $\overline{K}$'s in the block diagram will all be 1. For the rare case where some of the K's are not 1, then the K's should be normalized by the user such that the largest K = 1, and the remaining K's are interpreted as the fraction of response relative to the largest K . That is, for example, if there are two (2) downstream units and K1 = 1 and K2 =0.5, this means that unit 2 will respond with only 50% of the response of unit 1. For all cases the relationship $\left\{\sum_{i=1}^{n} \frac{\bar{K}_i.MVA_i}{MVAplt}\right\} = 1$ must be satisfied.

It is thus easy to show that in general:

$$\bar{K}_{Z/Wi} = \frac{MVAplt}{MVA_i} \times K_{Z/Wi} \times \frac{MVA_i}{MVAplt} \times \frac{MVAplt}{\sum K_{Z/Wi}.MVA_i} = \frac{MVAplt}{\sum K_{Z/Wi}.MVA_i} \quad (1)$$

Clearly, for the most common case when all the user inputted K's are equal to one, then all the $\bar{K}$'s will also be 1, since $\sum K_{Z/Wi}.MVA_i = MVAplt$.

5. This model will be connected to its own bus. Multiple instances of this model should not be made at a single bus. Each hybrid-plant controller should be connected to its own unique bus. The model is bus connected to avoid the model being disconnected or tripped during a simulation, when an individual aggregated downstream generator connected to this hybrid-plant controller is tripped. Typically, though not necessary, the models bus will be the plants point of interconnection or point of measurement.

6. The one difference between this model and REPC_C in terms of the main controls is that the switching point of the MSSs is entered by the user as MVAr values instead of pu values, since the switching points will be at a constant MVAr value and since this hybrid-controller may have a varying MVA base (see 1b. above) the switching points cannot be entered in per unit quantities.

7. One other simple change has been made, as compared to REPC_C. To allow for HVRT cases one additional parameter has been added to the model *Vfrzhigh*, such that, If (*Vfltr* < *Vfrz*) or (*Vfltr* > *Vfrzhigh*), then states s2, s3, s5, and s6 are frozen, and remain frozen for the time duration *Tfrz* after the filtered voltage, *Vfltr*, recovers above *Vfrz* and below *Vfrzhigh*. This is illustrated pictorially in Figure 3. Namely, in that example at point (1) the filtered voltage (**BLUE** trace) goes above *Vfrzhigh* so the controls would be frozen. The timer does not start until the voltage goes below *Vfrzhigh* – point (2). Then the freezing stops only when the filtered voltage remains between *Vfrzhigh* and *Vfrz* for at least *Tfrz* – point (3); otherwise a whole new freeze cycle starts. The same is true for the low side as shown, i.e., freeze at point (4) and unfreeze at point (5).

PowerTech Labs has graciously already implemented the interface shown in Figure 2 (i.e. what is marked in RED) as an end-block and it has been used in some user-developed models (UDMs) and shown to be effective. For actual UDMs additional complexities apply in some cases. What is shown is the simplest form.

Limits have been added on each of the outputs in the interface, but this is likely to be superfluous in most cases since the down-stream devices will have their own limits.

Thus, for this model, there will be:

- 10 × *N* new parameters, namely:

  (Bus_i, id_i, Kw_i, Kz_i, $Pmax_i$, $Pmin_i$, $Qmax_i$, $Qmin_i$, Tw_i and Tz_i) × *N* where

  N – is the number of down-stream connected devices (to be decided by WECC MVS)

  Bus_i – is the bus number of the i^{th} device

  id_i – is the id of the i^{th} device

  and the other parameters are self-explanatory from Figure 2.

- Two (2) additional flags will be needed (one for the reactive power path and one for the active power path) to defined the method of scaling the K's in each path, i.e. option 5a., 5b., 5c. or straight pass through.

For this initial proposed implementation, the intent is that the *N* downstream devices can be any combinations of:

1. Any REEC_* model (and note for a type 3 WTG the active power reference needs to go to the WTGQ_A and WTGP_* models), and
2. A REPC_C model, where the model is confirmed with constant-Q control (or straight pass through Q control) and pass-through P control

Other possibilities exist in real plants, but for now this may need to be the first step.

Important Note: With this model the limitations of REPC_B are removed (i.e. the model has its own MVA base and the limits on the controls are now the same as any plant controller model and no longer relative limits). *However, this introduces a significant burden on the user to ensure that the initial power flow solution for the plant is reasonable. For example, the initial Q output of all the downstream aggregated units in general should be in proportion to their respective MVA bases, etc.*

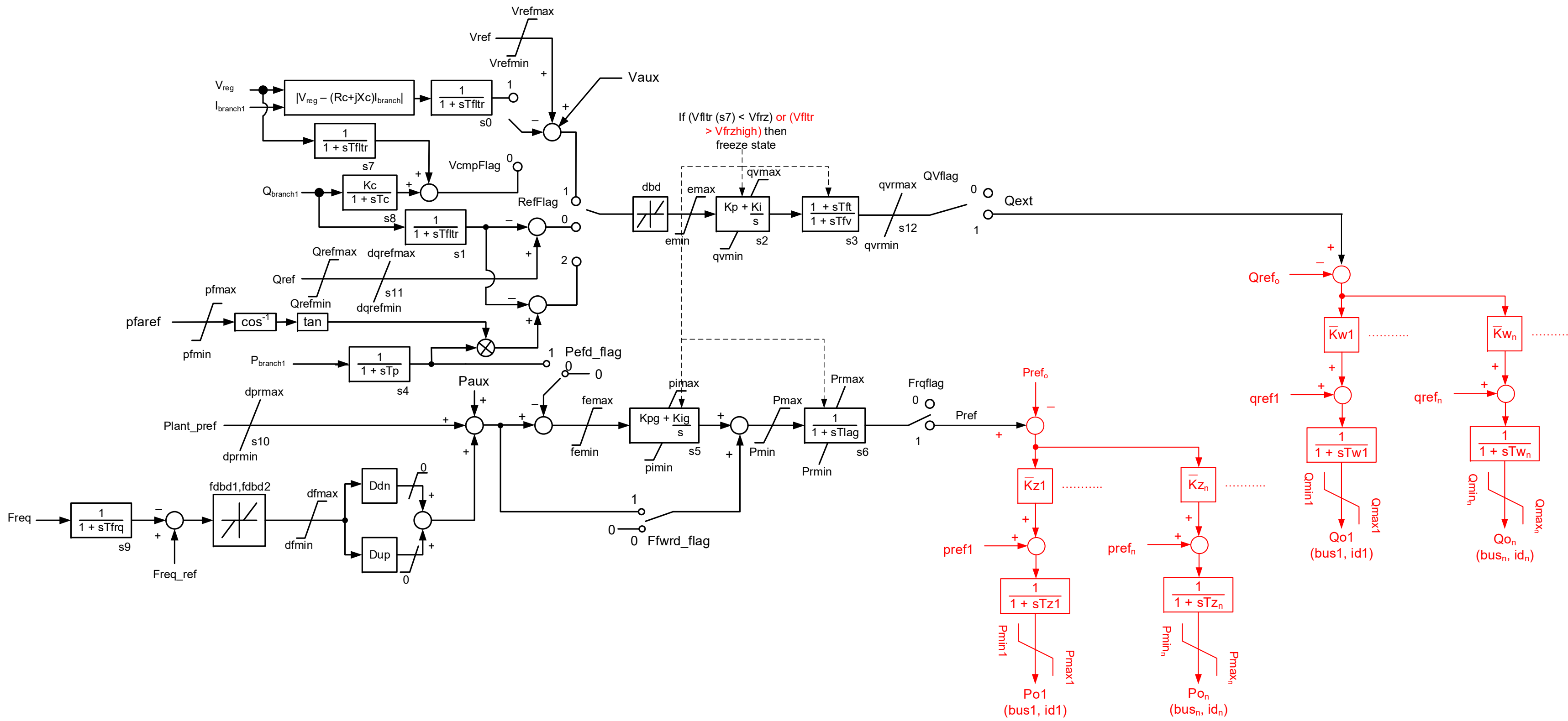
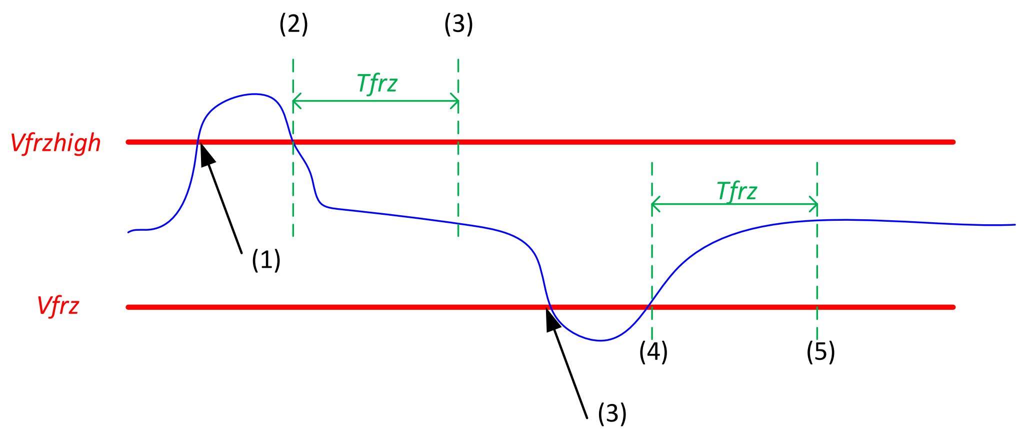


Figure 2: REPC_D model




```
//----------------------------------------------------
// Vfrz     : input parameter
// Vfrzhigh : input parameter
// Tfrz     : input parameter
// Vfltr    : returns the state Vfltr floating point
// TfrzLimitsTimeNeedsReset : internal boolean to indicate the timer needs to be reset (initially FALSE)
// FreezeStates             : internal boolean to indicate the states are frozen (initially FALSE)
// TfrzLimitsTime           : floating point used as a timer
// PresentTime              : floating point with the present time of the simulation
//----------------------------------------------------
// Following code is run each time-step of the simulation
//----------------------------------------------------
    if (Vfltr < Vfrz) OR (Vfltr > Vfrzhigh) then
      TfrzLimitsTimeNeedsReset = TRUE
      FreezeStates = TRUE
    else
      if TfrzLimitsTimeNeedsReset then begin
        TfrzLimitsTimeNeedsReset = FALSE
        TfrzLimitsTime = PresentTime
      endif
      if FreezeStates AND (GetDiffSeconds(PresentTime,TfrzLimitsTime) >= Tfrz) then
        FreezeStates = TRUE
      endif
endif
```

Figure 3: Illustrating the freezing logic.

Acknowledgements:

We gratefully acknowledge all who contributed to this effort, including:

Pouyan Pourbeik (PEACE®), Jamie Weber (PowerWorld), Jay Senthil (Siemens PTI), Juan Sanchez-Gasca (GE), Mengxi Chen (GE), Deepak Ramasubramanian (EPRI) and Jeff Bloemink (PowerTech Labs), as well as many others within WECC MVS who provided comments and feedback.

PEACE® wishes to gratefully acknowledge EPRI research funding for time spent by PEACE® staff to prepare this memo.

PEACE® also very gratefully acknowledges PowerTech Labs, and in particular Pouya Zadkhast, for their implementation of the end-block that already accomplishes the task of the end interface shown in Figure 2 (in RED).

We apologize for any inadvertent omissions.

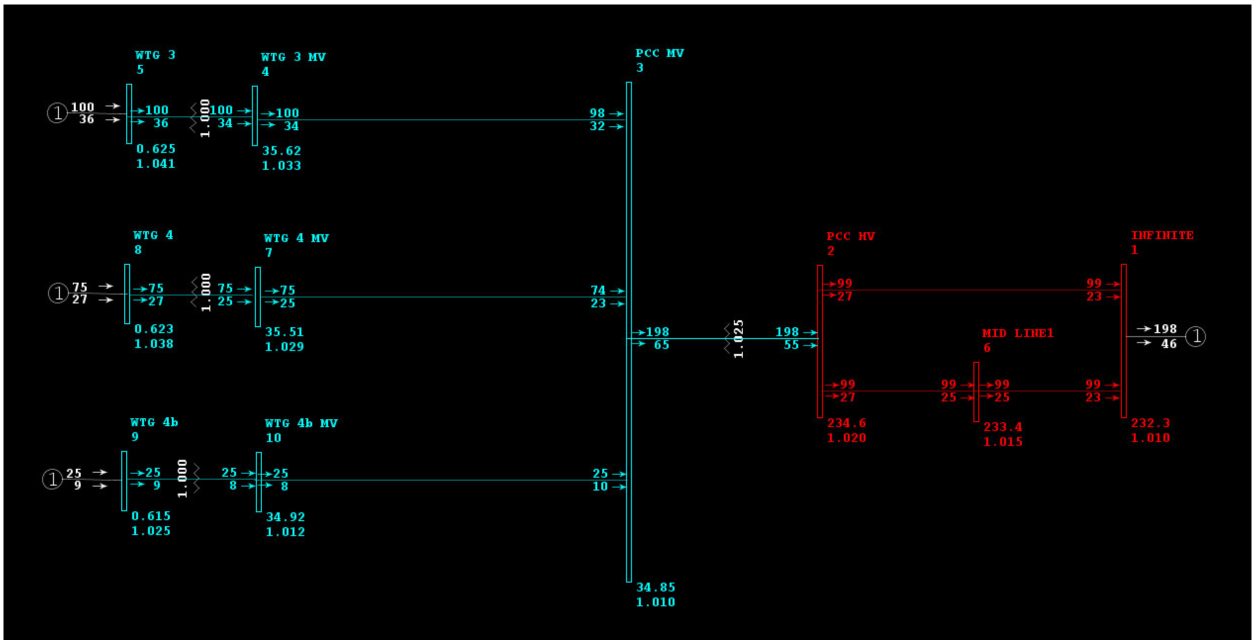
References:

[1] Memo on *Proposal for New Features for the Renewable Energy System Generic Models,* Revisions 27, Dated: 2/16/23

https://www.wecc.org/Reliability/Memo_RES_Modeling_Updates_021623_Rev27_Clean.pdf

<u>Appendix A</u> – Simple Test Case for REPC_D Model

The REPC_C model has been previously in 2022 benchmark tested across the various software platforms and released in the various software, and WECC approved. The REPC_D is identical in its core control structure to the REPC_C model, thus the software vendors will essentially copy the code from REPC_C to the heart of REPC_D and so a comprehensive benchmarking is not needed. The key difference between REPC_D and REPC_C is in its interface to multiple downstream renewable energy system (RES) models. Thus, it is this aspect that requires benchmark testing across the software platforms. To that end, the following simple test case was developed, together with the proposed tests listed below, in early September 2023 to be used for software benchmarking.



The case consists of three (2) aggregated RES models:

A 200 MVA WTG type 3 plant

A 100 MVA WTG type 4 plant, and

A 50 MVA PV plant

All of these are "fictitious" and do not represent any specific plant or equipment. This is all simply for the purpose of software testing.

The REPC_D model is set up to control all three (3) plants in two ways:

<u>Option 1</u> – all Ks = 1

<u>Option 2</u> – the Ks set up as follows: K1s = 1, K2s =0.5 and K3s = 0.25 (where 1, 2 and 3 correspond to the 200 MVA, 100 MVA and 50 MVA plants, respectively)

In the above by Ks is meant both Kwi and Kzi.

Then four (4) test simulations are to be run for both options above and compared across the software tools:

- Test 1 – place a 5% (0.05 pu) *Paux* reference step on the plant (see figure below) at T = 1 second and run to 30 seconds

- Test 2 – place a 0.25% (0.0025 pu) *Vaux* reference step on the plant at T = 1 second and run to 30 seconds
- Test 3 – used the constant-Q control version of the dynamic model and increase the *Qref* by 5% (0.05 pu) at T = 1 second and run to 30 seconds
- Test 4 – place a 0.25% (0.0025 pu) *Vaux* reference step on the plant at T = 1 second and run to 20 seconds, then at T = 20 seconds trip the 50 MVA unit.

The simple test case has been setup and shared amongst the group and can be obtained upon request.

Test 1b, 2b, 3b and 4b, are the repeat of the above tests with the Option 2 Ks.