TO: WECC REMWG

FROM: POUYAN POURBEIK, PEACE®; PPOURBEIK@PEACE-PLLC.COM

SUBJECT: PROPOSAL FOR NEW FEATURES FOR THE RENEWABLE ENERGY SYSTEM GENERIC MODELS

DATE: 07/23/18 (REVISED 8/3/18; 11/14/18; 11/18/18; 12/13/18; 1/20/19; 2/7/19; 3/5/19; 6/24/19; 6/28/19; 7/3/19; 7/17/19; 8/19/19; 12/17/19; 8/14/20; 8/24/20;8/31/20; 11/12/20; 4/6/21; 4/15/21; 9/22/21; 11/11/21; 9/19/22;12/12/22; 1/5/23)

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

Recent efforts of the WECC REMWG[1] have resulted in development of the now so-called second-generation generic renewable energy system (RES) models [1] and [2]. These models were first released in the major commercial software platforms in 2014. Since that time adoption of these models has been gradually increasing. WECC started adopting the models in late 2014, early 2015, and by this time the vast majority of the first-generation generic models have been replaced in the WECC database. Adoption of these models in the Eastern Interconnection has been slower.

As with any model development process there needs always to be a balance between perfection and timely development for application to yield value. Thus, the initial development of the second-generation RES models started wholesale in late 2010/early 2011, within the WECC REMWG, and culminated in January 2014. It is now time to revisit the models and propose modifications to further improve their applicability. At the last two WECC MVS[2] meetings, with input from several of the major equipment vendors and other stake-holders, discussions have ensued to propose new features and additions to the models. Given the modular nature of the second-generation RES models, these proposed changes can be readily adopted through the implementation of a new set of modules to be added to the RES model library.

In parallel to the efforts of WECC REMWG, and in collaboration in the early stages of both efforts, the IEC TC88 WG27 has been developing international standard specifications for generic wind turbine generator models since 2010. The IEC work came to fruition in 2020, however, it only pertains to wind turbine generators and their controls.

With the above background, this document presents the collection of proposed additions/changes to the 2nd generation RES models developed in WECC, based on input from several vendors and discussions at the most recent WECC REMWG meetings. With respect to the proposed new inertia-based fast-frequency-response (IBFFR) model, the presentation here is our initial proposal based on an understanding of the general nature of these controls, it is not by any means an exact representation of any vendors equipment. In general, this is true of all the model modifications presented. These so-called generic models are able to reasonably "emulate" the behavior of equipment, if parameterized appropriately.

Revision Note:

The original version of this memo was circulated publicly to the forums listed above, and discussed at length on a WECC REMWG webcast/call on 9/6/18. Furthermore, a subsequent revision was discussed at the WECC MVS meeting in Salt Lake City, UT on November 28th, 2018. Thus, this revision includes comments received from many of the various stake holders during both the webcast/call on 9/6/18 and the MVS meeting

---

[1] Throughout this document references are made to the new name of the Renewable Energy Modeling Task Force (REMTF), which changed in early 2020 to the Renewable Energy Modeling Working Group (REMWG).

[2] Throughout this document references are made to the new name of the Modeling Validation Working Group (MVWG), which changed in early 2020 to the Modeling and Validation Subcommittee (MVS).

on 11/28/18. This includes comments from several major wind and solar PV vendors (First Solar, GE, Senvion, Siemens-Gamesa, and Vestas), as well as the major commercial software vendors for power system simulation tools (GE, PowerWorld, PowerTech Labs, Siemens PTI). The input from all these entities, WECC REMWG members, and EPRI are most gratefully acknowledged.

One of the key outcomes of the 11/28/18 MVS meeting was a prioritization of the implementation of the proposed new modules in this memo. The agreed to prioritization was as follows:

1. REGC_B

2. REEC_D

As of August 14, 2020, the above two models were fully tested and approved in the August, 2020 MVS meeting.

3. REPC_C & WTGWGO_A

4. WTGP_B, WTGT_B and WTGIBFFR_A

As of the December, 2021 MVS meeting the above five models were approved after implementation, benchmark testing etc. (see and https://www.wecc.org/Reliability/Memo_IBFFR_071719.pdf for example for some verification cases for the IBFFR model).

5. REGC_C – already implemented by all the major software platforms, as of September 2022, and benchmark tested.

As of this revision 25 of the memo, dated 1/5/23 all of the models discussed herein have been approved by WECC and benchmarked and tested in the major software tools used in WECC and should be available in the respective tools in the latest versions of the tools. The only model not yet formally approved and pending approval is the REGC_C model. It was discussed at the September 2022 MVS meeting and will go up for final approval at the MVS February, 2023 meeting. Thus, this should hopefully be the final version of the memo.

1. 0 Overview:

The RES models are composed of four (4) categories of modules [1] and [2]:

1. The renewable energy generator-converter models (REGC_*)
2. The renewable energy electrical-control models (REEC_*)
3. The renewable energy plant-controller models (REPC_*)
4. The mechanical and aero-dynamic models associated with wind turbine generators

New features or modifications have been proposed in all the above categories. Here the most significant of these, which in principle have been agreed to in the recent WECC REMWG meetings, will be described. It should be noted that work continues with respect to some other modification that are only mentioned here, but not described in detail, since they will be forthcoming in other presentations once finalized.

In addition to the above, in this document, we are proposing the introduction of a new class of modules which we will call "**auxiliary controls**". In this category will fit at present two (2) new models being proposed, that have been under discussion in the WECC MVS for some time

2.0 Generator/Converter Models:

Presently, the only available REGC model is the REGC_A model (see [1] or [2]) which is a simple current-source model. The main limitation of this model is numerical stability when applied for modeling RESs connected to a relatively weak grid node. There are numerical tricks that can be played to extend its use (e.g. shortening integration time steps, etc.), however, none of this is truly conducive. Thus, several proposals have been put forth for new REGC models to extend the range of applicability of the REGC models. These proposals are as follows:

2.1 REGC_B (new model)

This is an extension of the REGC_A model by changing the network interface between the generator/converter model from a current-source to a voltage-source, based on [3].

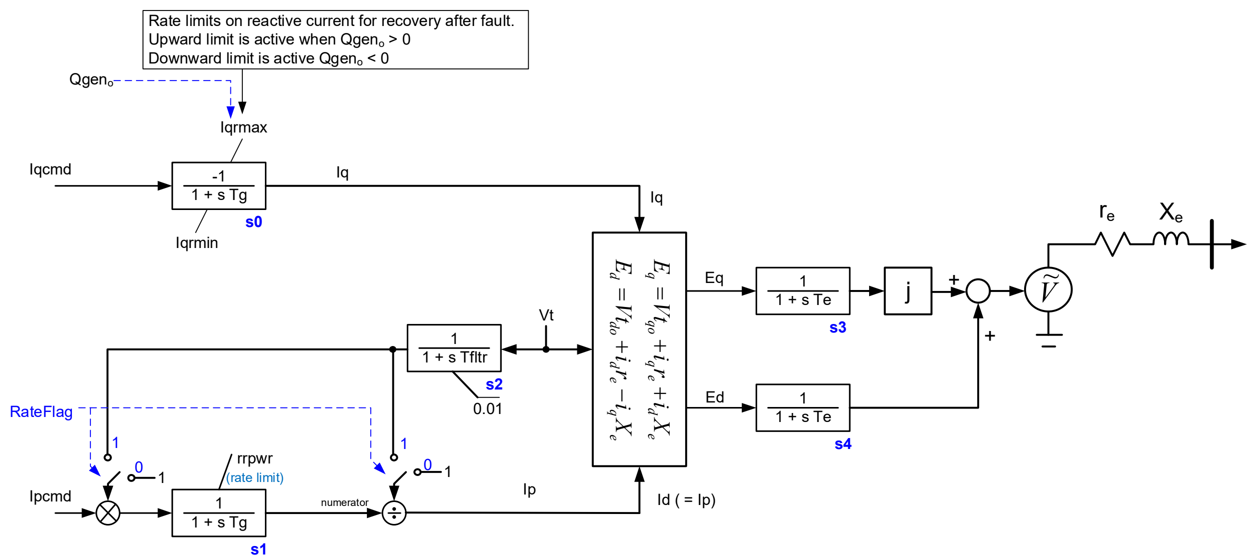

Figure 1: REGC_B model

The above figure defines the REGC_B model. The reactive-current arm is identical to REGC_A. The interface to the grid is a simple voltage-source, based on [3]. The active-current arm, as presented here, is significantly different to REGC_A and hitherto not presented in the form shown in Figure 1. The changes, and the reasoning behind the changes are as follows:

- The new flag *RateFlag* allows the user to make the effect of the rate limit for the increase in active-current (*rrpwr*) to be either a rate-limit in active-current (by setting *RateFlag* = 0) or active-power (by setting *RateFlag* = 1). This is because some vendors have indicated that the rate limit is implemented in their controls as an active-power rate limit.

- The lag block (*Tfltr*) allows for modeling of a small delay in the measurement of terminal voltage (*Vt*). *Tfltr* can be set to zero.

- The hardcoded lower limit on the voltage measurement lag block (0.01) is implemented to prevent a divide by zero. The limit must be effected even if *Tfltr* = 0. Also, the limit should be on Vt (as shown) and not on only the division arm, so that the same value of *Vt* is first multiplied by *Ipcmd* and then later *Ipcmd* is divided by the same number.

- The LVPL block and associated parameters (used in REGC_A – see [1]) has been completely removed. The reasoning is as follows:

- This block is typically only used by a few vendors
- With the extended VDL blocks (see below section 3), we believe that the function of the LVPL block can be easily emulated with the VDL2 block and thus it is not necessary and would add confusion (and based on many questions from users, the LVPL function already does cause confusion).
- However, once REGC_B is implemented and tested, if some vendors find that it is absolutely necessary to re-introduce the block, that will be considered. This remains to be seen.

- **Current Limits:** during the network solution, at the interface of the model, based on discussions within the group, it was concluded that some sort of current limits might be useful in order to minimize the potential for current spikes due to numerical issues. It should be understood, that there are no "current limits" applied in reality at physical interface of the inverter to the grid. All such limits are applied in the controls (REEC_* models). The reason for introducing this algebraic manipulation at the network interface is to help to mitigate some of the un-realistic current spikes often seen in the RMS models. For a detailed explanation of the implementation that was developed by EPRI see [9]. EPRI [4] has demonstrated the potential benefit of this algebraic manipulation at the network interface. This will need further discussion and investigation once a beta version of the model is available for testing in the commercial tools to assess the efficacy of these limits. The *Imax* and *pqflag* parameters below are the maximum current limit and P/Q priority settings on the converter (these values should be consistent with the associated REEC_* model), which is to be used for applying this limit in the network solution iterations.
- The ramp-rate of recovery after a fault on active-current (*rrpwr*) should also be imposed (in the opposite direction) when the model is being used to "emulate" charging of an energy storage device. That is, when *Pgen* is negative, then *rrpwr* should have its sign changed and it becomes the ramp-rate at which charging power (power being absorbed by the model) increases after a fault[3]. (*It is recommended that this also be applied to REGC_A to ensure it too behaves in this way when used with a battery model*)
- **Note:** a final feature to be introduced in REGC_B is to implement blocking such that an "link" is established inside the software platforms that comes from the REEC_D model and directly tells the machine to block, i.e. set active and reactive current coming out of the machine to "zero" the instant blocking is invoked in REEC_D.

Table 1: Parameter List for REGC_B

Parameter	Description	Typical Range
Tg	Emulated delay in converter controls [s] (Cannot be zero; at minimum will be set to 4 × integration time step)	0.02 – 0.05
Iqrmax	Rate at which reactive current recovers after a fault when the initial reactive power output ($Qgen_o$) of the unit is greater than zero (typically disabled by setting to 999) [pu/s]	1 – 999
Iqrmin	Rate at which reactive current recovers after a fault when the initial reactive power output ($Qgen_o$) of the unit is less than zero (typically disabled by setting to -999) [pu/s]	-1 – -999
Tfltr	Filter time constant for voltage measurement. Can be set to "zero". [s]	0.02 – 0.05

[3] A simple way to model this is:
```
If ds1/dt (derivative of state 1) > +rrpwr then
  If s1 >= 0 then ds1/dt = +rrpwr
Elseif ds1/dt < -rrpwr then
  If s1 <= 0 then  ds1/dt = -rrpwr
EndIf
```
This is checked at every-time step.

rrpwr	Rate at which active current (power) recovers after a fault [pu/s]	1 – 20
RateFlag	0 – *rrpwr* represents active-current ramp rate; 1 – *rrpwr* represents active-power ramp rate	N/A (if in doubt set to 0)
Te	Emulated delay in converter controls [s] (Can be zero)	0.0 – 0.02
Imax	Maximum current rating of the converter [pu]	1.1 – 1.4
pqflag	1 – P priority and 0 – Q priority on the current limits	N/A
re	Source resistance [pu]; typically set to 0	0 – 0.01
Xe	Source reactance [pu]	0.05 – 0.2

2.2 REGC_C (new model)

Another more complicated proposal for a generator/converter model is an alternative to REGC_B which EPRI has been working on and presented at the last WECC MVS meeting [4]. The core difference between what is presented in [4] and the REGC_B model shown above is the following:

1. The addition of a simplified phase-lock loop (PLL) model (blocks associated with *Kppll, Kipll*), based on the attempt that was made in the 1st generation generic models for wind turbine generators (see https://www.esig.energy/wiki-main-page/wt3-generic-wind-model/), and
2. The addition of a simplified representation of the inner-current control loops (PI controllers with gains *Kip, Kii*) – this has not been attempted previously in the generic models.

Based on the results shown in [4] we believe it is clear that the combination of the PLL, inner-current control loops and the voltage-source representation in REGC_B does collectively present some potential additional fidelity and usefulness for extending the range of applicability of the generic models. Here the block diagram presented below combines the concepts presented in [4], together with the changes presented above in REGC_B to propose a so-called REGC_C model.

Note that the purpose of the *Imax* and *pqflag* parameters are introduced here for the same reason as in REGC_B, to be used for imposing the maximum current limit during network solution iterations (see [9]).

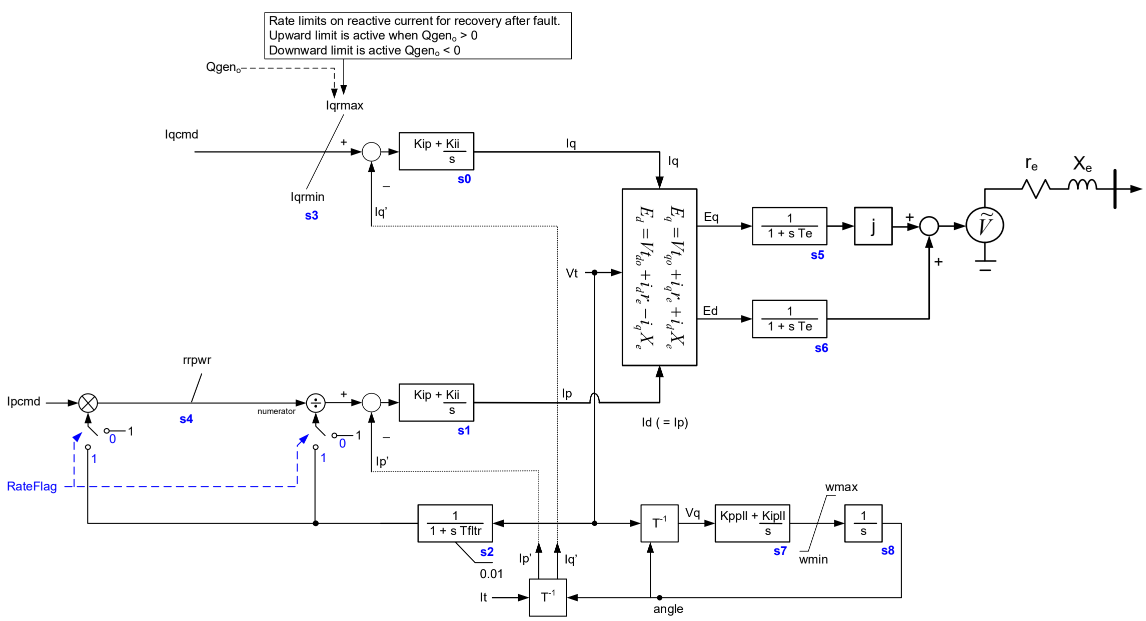

Figure 2: REGC_C model[4]. The block T-1 represent the reference frame transformation between the network and controls. Details of the network solution iterations are not shown that are discussed in [4].

[4] This is a very simplified diagram; the details can be obtained in EPRI's presentation reference [4].

This model still requires some further investigation and fine tuning. What is presented here in Figure 2 is slightly different from that in [4], in that it introduces some of the new features of REGC_B presented above.

Table 3: Parameter List for REGC_C

Parameter	Description	Typical Range[5]
Iqrmax	Rate at which reactive current recovers after a fault when the initial reactive power output ($Qgen_o$) of the unit is greater than zero (typically disabled by setting to 999) [pu/s]	1 – 999
Iqrmin	Rate at which reactive current recovers after a fault when the initial reactive power output ($Qgen_o$) of the unit is less than zero (typically disabled by setting to -999) [pu/s]	-1 – -999
Tfltr	Filter time constant for voltage measurement. Can be set to "zero". [s]	0.02 – 0.05
rrpwr	Rate at which active current (power) recovers after a fault [pu/s][6]	1 – 20
RateFlag	0 – *rrpwr* represents active-current ramp rate; 1 – *rrpwr* represents active-power ramp rate	N/A (if in doubt set to 0)
Imax	Maximum current rating of the converter [pu]	1.1 – 1.4
pqflag	1 – P priority and 0 – Q priority on the current limits	N/A
Te	Emulated delay in converter controls [s] (Can be zero)	0.0 – 0.02
re	Source resistance [pu]; typically set to 0	0 – 0.01
Xe	Source reactance [pu]	0.05 – 0.2
Kip	Proportional-gain of the inner-current control loop [pu/pu]	1 – 10 (suggested range)
Kii	Integral-gain of the inner-current control loop [pu/pu.s^{-1}]	20 – 100 (suggested range)
Kppll	Proportional-gain of the PLL [rad.s^{-1}/pu]	1 – 10 (suggested range)
Kipll	Integral-gain of the PLL [rad.s^{-1}/pu.s^{-1}]	500 – 3000 (suggested range)
wmax	Upper limit on the PLL [rad.s^{-1}]	1 – 10
wmin	Lower limit on the PLL [rad.s^{-1}]	-10 – -1
Vpllfrz	If measured terminal voltage is below *Vpllfrz*, then PLL state is frozen	0.1 – 0.5

Note: a final feature to be introduced in REGC_C is to implement blocking such that an "link" is established inside the software platforms that comes from the REEC_D model and directly tells the machine to block, i.e. set active and reactive current coming out of the machine to "zero" the instant blocking is invoked in REEC_D.

Important Notes on REGC_C:

The REGC_C model has been tested and verified against some vendor data (see [10] as an example) and also shown to yield expected results in terms of both small-signal and frequency response (see presentation at WECC MVS [11]). Moreover, this model is intended primarily for use with renewable energy systems (RES) that are connected to the grid through a full-converter (e.g. type 4 WTG, PV and BESS). Although not intended for use for type 3 WTGs, if a type 3 WTG vendor is able to parameterize the model to reasonably represent their equipment and demonstrate this (e.g. comparison of results with a more detailed model or factory type tests) then that is acceptable. Also, this model IS NOT intended to replace *REGC_A* or *REGC_B*, but rather to offer a more detailed model for circumstances where an RES plant is connected to an extremely low short-circuit ratio node (e.g. SCR < 2), where the other models may result in numerical problems.

[5] The typical range of parameters is for guidance only and not to be interpreted as a hard limit/range on values.

[6] See implementation of rrpwr for REGC_B on page 4.

2.3 IEC TC88 WG27 Modules

There are two other models specifically for type 3 WTG generator/converters, that would fall under this same category of the REGC models which have been proposed by the IEC TC88 WG27 group in the recently published IEC 61400-27-1standard Wind energy generation systems – Part 27-1: Electrical simulation models – Generic models, Edition 2.0 2020-07. They are called the "type3A" and "type3B" models in the IEC Standard document. Here we could call them for example, *REGC_IEC3A* and *REGC_IEC3B*. Consideration should be given to adopting those models in due course. They include items such as the active-crow bar emulation[7], which was previously investigated also by the WECC MVS but not adopted.

In addition, there is a second more complex aero-dynamics model proposed in the IEC standard, which may be useful to have implemented in order to have it available in the mix of models.

---

[7] An early presentation of the active-crow bar simulation was presented in section 5 of https://www.wecc.biz/Reliability/WECC-Type-3-Wind-Turbine-Generator-Model-Phase-II-012314.pdf but at the time not adopted by WECC due to its complexity.

3.0 Electrical Control Models:

Presently, there are three (3) existing REEC_* models (see [2]):

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 recommended[8] (originally proposed in [5]).
3. **REEC_C:** a model intended primarily for use in modeling energy storage systems.

As indicated above, the REEC_B model is no longer recommended for use in most cases since it is devoid of the ability to represent the voltage-dependent limits (VDL) on the inverter current. The REEC_A and REEC_C models, although quite comprehensive, and though both do include the VDL tables, there are some improvements that have been pointed out by several vendors recently, including Siemens-Gamesa, Vestas and Senvion. Here those suggested improvements have been collected into a new proposed model – REEC_D.

3.1 REEC_D (new model)

The REEC_D (Figure 3) model is identical to the REEC_A model[9], with the following additions/modifications:

1. The VDL tables in REEC_D should have ten (10) pairs of points[10]: That is,
   a. VDLq reactive-current limits = {*(iq1, vq1), (iq2, vq2), (iq3, vq3), (iq4, vq4), (iq5, vq5), (iq6, vq6), (iq7, vq7), (iq8, vq8), (iq9, vq9), (iq9, vq9), (iq10, vq10)*}[11]
   b. VDLp active-current limits = {*(ip1, vp1), (ip2, vp2), (ip3, vp3), (ip4, vp4), (ip5, vp5), (ip6, vp6), (ip7, vp7), (ip8, vp8), (ip9, vp9), (ip9, vp9), (ip10, vp10)*}

   The *iq's* can be positive or negative

   The *ip's* must all be greater than or equal to zero

   Furthermore,

   If (*Pqflag* = 0) (i.e. Q – priority)

   Iqmax = min {VDLq, *Imax*}

   If *Iqmax* < 0

   Iqmin = *Iqmax* (important new logic)

   else

   Iqmin = -*1* × *Iqmax*

   end

   Ipmax = min {VDLp, $\sqrt{Imax^2 - Iqcmd^2}$}

   Ipmin = -*Ke* × *Ipmax* (important new logic)

   else (i.e. P – priority)

[8] https://www.nerc.com/comm/PC/NERCModelingNotifications/Modeling_Notification_-_Modeling_Momentary_Cessation_-_2018-02-27.pdf

[9] It is important to note that all other aspects of REEC_A should be copied into REEC_D. What is presented here are only the suggested changes and additions.

[10] Also, based on recommendations by some users, the names of the tables are to be changed from VDL1 and VDL2 to VDLq and VDLp, so that it is easier to identify which table is associated with reactive and active current, respectively.

$Iqmax$ = min {VDLq, $\sqrt{Imax^2 - Ipcmd^2}$}

If $Iqmax < 0$

$Iqmin = Iqmax$ (important new logic)

else

$Iqmin = -1 \times Iqmax$

end

$Ipmax$ = min {VDLp, $Imax$}

$Ipmin = -Ke \times Ipmax$ (important new logic)

end

The explanation for the new logic (marked in blue comments) above is as follows:

- If $Ke = 0$, then the model mimics a generator, that is $Ipmin = 0$ and the unit cannot absorb active power from the grid (see Figure 4).
- If ($1 \geq Ke > 0$), then the model mimics a storage device, that is capable of also absorbing active current. A value of $Ke < 1$ implies that the device has a lower capacity for absorbing instantaneous power, as compared to its capacity of generating instantaneous power (see Figure 5).
- Ke cannot be negative or greater than 1.
- In the case that *Iqmax* is negative, then *Iqmin* must also be negative and the same value in order to force *Iqcmd* to this limit. This is used in some cases during extreme high voltages to make the inverter absorb reactive power (see Figure 6).

2. The addition of two new blocks:

   a. A local current-compensation block ($|Vt - (rc + j\ Xc).It|$) with a lag block to emulate measurement delays (*Tr1*). The lag block time-constant (*Tr1*) can be set to zero. Likewise, *rc*/*Xc* can both be set to zero to eliminate modeling of current-compensation. The inputs to this block are the terminal-voltage (*Vt*) of the generator/converter model (REGC_*) which is downstream of this model, and the terminal-current (It) of the same. Both these values (*Vt* and *It*) are the complex (*real* + *j.imaginary*) values of voltage and current.

   b. A local reactive-droop compensation block (*Kc*) with a lag block to emulate measurement delays (*Tr1*). The lag block time-constant (*Tr1*) can be set to zero. Likewise, *Kc* can be set to zero to eliminate modeling of reactive-current compensation. The input to this block is the terminal generated reactive-power (*Qgen*) voltage of the generator/converter model (REGC_*) which is downstream of this model.

   c. The reactive-current injection arm (which has the output *Iqinj*), is slightly different in this model, as compared to REEC_A. First, the logic around the switch at the output of this arm has been completely removed. This arm is always active (as long as *Kqv* is non-zero). To completely disable this arm, *Kqv* can be set to zero and *Vdip*/*Vup* set to e.g. -1 and 2 to completely turn-off the voltage-dip logic. The two parameters *Iqfrz* and *Thld* remain, but now have a slightly different function. The logic is as follows:

      i. If *Thld* = 0 – no other action is taken.

      ii. If *Thld* > 0, then for *Thld* seconds following a voltage dip (i.e. *voltage_dip* goes from 1 back to 0) *Iqcmd_bl* is held at its current value (i.e. value just prior to the end of the *voltage_dip*) for *Thld* seconds and is then released.

iii. If *Thld* < 0, then for *Thld* seconds following a voltage dip (i.e. *voltage_dip* goes from 1 back to 0) *Iqcmd_bl* is held equal to *Iqfrz* for *Thld* seconds and is then released.

Note: The value of *Iqcmd_bl* that is held/frozen is the value that is after the summing junction and just before the *Iqmax*/*Iqmin* limits as shown in Figure 3. The purpose of this feature is primarily to help with better modeling of momentary cessation.

The REEC_D has logic that holds/freezes active current command (and active current *Ipmax*) at the previous value (i.e. at the value that they both were at during the voltahe-dip and just prior to the release of the voltage dip) for *Thld2* seconds following a voltage dip, i.e. *voltage_dip* goes from 1 back to 0.

3. Blocking Logic: At very low voltages at the terminals of the converter the converter power electronic will block. In recent work within NERC and WECC this has been referred to as momentary cessation". A detailed discussion of this subject is outside of the scope of this document. Although it may be possible to model inverter blocking by properly parameterizing the VDL tables and the *Thld* and *Thld2* parameters, this may not be entirely desirable since those parameters are more typically used for modeling the voltage dependence of the inverter current limits and the voltage-dip logic, which can be independent of blocking. Thus, the following three new parameters are proposed to be completely independent of all the other parameters and to be used for modeling converter blocking:

   a. *vblkl* – this is the voltage below which the converter will block, that is if the measured terminal voltage of the generating device (*Vt_filt* in Figure 3, i.e. state S0) is less than or equal to *vblkl* then *Iqmax* and *Ipmax* are forced to 0 (i.e. *Iqmax = Ipmax = Iqmin = Ipmin = 0*, and thus both the *Ipcmd* and *Iqcmd* = 0).

   b. *vblkh* – this is the voltage above which the converter will block, that is if the measured terminal voltage of the generating device (Vt_filt in Figure 3, i.e. state S0) is greater than or equal to *vblkh* then *Iqmax* and *Ipmax* are forced to 0 (i.e. *Iqmax = Ipmax = Iqmin = Ipmin = 0*, and thus both the *Ipcmd* and *Iqcmd* = 0).

   c. *Tblk_delay* – once the converter comes out of the blocking mode (i.e. voltage recovers after a blocking incident back within the range of *vblkl* < *Vt_filt* < *vblkh*) the current limits are released only after Tblk_delay seconds (i.e. *Iqmax = Ipmax = Iqmin = Ipmin = 0* for another *Tblk_delay* seconds after the voltage recovers outside of the blocking range).

4. The non-wind up limits: The non-wind up limits shown on the two PI controllers in the REEC_* models can be implemented several different ways, all of which are legitimate non-winding limit representations, but which will yield subtly different results for extreme cases that force the controllers into their limits. Based on a discussion as to how to attempt to harmonize the performance of these models across the various commercial software platforms, here we present one way of implementing the non-windup limits, which all software vendors can adopt, if they wish, to make the implementation uniform. This can be adopted for REEC_D. However, this discussion among the software vendors, rightfully so, identified one key challenge with trying to do this: if all software vendors implement the exact same form of the non-windup limit (as shown below) in REEC_D and yet this implementation is not the same as was used by the software vendors in previous models (e.g. REEC_C, REEC_A, etc.) then users may complain about inconsistency of results, even if subtle, when going from one model to the other with all the same parameters.

   Some of the vendors are actually already using the approach shown below, or something quite similar. Thus, in the end we have no choice but to leave the exact implementation of the non-windup limits to the judgement and discretion of each software vendor.

   Here an example way of implementing a non-windup limit is given. Consider the PI controller associated with state 3, it could be implemented as follows:

   $ds3 = Kvi \times error$

```
if (s3 ≥ Iqmax)
   s3 = Iqmax
elseif (s3 ≤ Iqmin)
   s3 = Iqmin
end
if ((s3 ≥ Iqmax) and (ds3 > 0))
   ds3 = 0
elseif ((s3 ≤ Iqmin) and (ds3 < 0))
   ds3 = 0
end
PI = s3 + Kvp × error
if (PI ≥ Iqmax)
   PI = Iqmax
elseif (PI ≤ Iqmin)
   PI = Iqmin
end
```

where *error* is the input to the PI block (i.e. the output of the summing junction which is going into the PI block), *PI* is the total output of the PI block, *ds3* is the derivative of the integrator, and *s3* is the state (output) of the integrator.

Further to the above, for these two PI blocks a special additional requirement is (see footnote on page 3-6 of [1]), that the non-windup integrators for *s3* and *s2* are linked. This link is as follows: if *s3* hits its maximum limit and *ds3* is positive, then *ds3* is set to 0 (as shown in the pseudo code above), and further more if *ds2* is also positive, then it is also set to 0 to prevent windup, but, if *ds2* is negative, then *ds2* is not set to 0. A similar rule is applied for *s3* hitting the lower limit, but the check is whether *ds3* and *ds2* are negative.

5. <u>The addition of *Paux*:</u> The new input *Paux* should be accessible both by the user for manipulation by an external user-written model, or by the auxiliary control models discussed below in section 6.0.
6. <u>Filter Time Constant:</u> Note that now *voltage_dip* is determined from the filtered (*Vt_filt*) voltage rather than *Vt*.
7. <u>This model should also obey the so-</u>called baseload flag that is used in the major North American software tools. Namely,
   a. If baseload flag = 0, then the model behaves normally,
   b. If baseload flag = 1, then *Pmax* = initial power flow MW output of the plant ($Pgen_o$) and thus the power order can only go down and not up, and
   c. If baseload flag = 2, then *Pmax*=*Pmin* = $Pgen_o$ and thus the power order is fixed.

Apart from the above additions/changes all other aspects of this model are identical to REEC_A. Thus, the software vendors may start with the code for REEC_A and make the above changes to get to REEC_D.

<u>**Important Note:**</u> *all input references (i.e.* $Vref_0$*, pfaref,* $Vref_1$*, Pref, Qext and Paux) should be accessible to the user after model initialization such that they can be either step-changed or controlled by an external user-written model. Clearly, if this model is connected to one of the standard plant controller models (repc_*) then Qext and Pref will be controlled by that model and cannot be also controlled by another user-written model.*

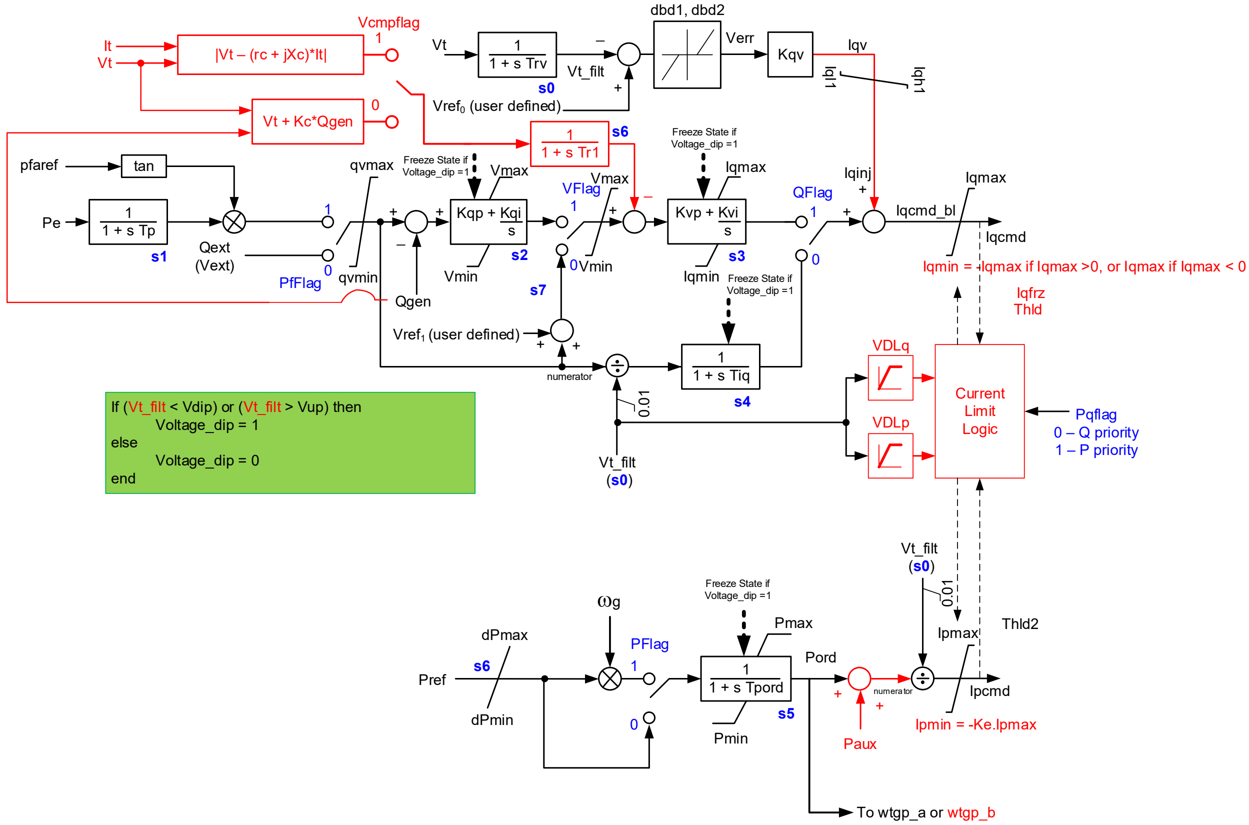

Figure 3: REEC_D model.

Table 3: Parameter List for REEC_D – only new additional parameters are listed here; all other parameters are identical to REEC_A.

Parameter	Description	Typical Range/Value[12]
rc	Current-compensation resistance [pu]	0 – 0.02
Xc	Current-compensation reactance [pu]	0.01 – 0.12
Tr1	Filter time constant for voltage measurement. Can be set to "zero". [s]	0.02 – 0.05
Kc	Reactive-current compensation gain	0.01 – 0.05 (to be discussed)
Vcmpflag	1 – use current compensation, 0 – use reactive droop	N/A
Ke	Scaling on Ipmin; set to 0 for a generator, set to a value between 0 and 1 for a storage device, as appropriate	0 – 1
Iqfrz	Value to which reactive-current command is frozen after a voltage-dip [pu]	0
Thld	Time for which reactive-current command is frozen after a voltage-dip [s]; if positive then *Iqcmd* is frozen to its final value during the voltage-dip; if negative then *Iqcmd* is frozen to *Iqfrz*	0
VDLq	10 pairs of values defining the voltage dependent reactive-current limits [pu]	N/A
VDLp	10 pairs of values defining the voltage dependent active-current limits [pu]	N/A
qvmax	The maximum value of the incoming Qext or Vext [pu]	N/A
qvmin	The minimum value of the incoming Qext or Vext [pu]	N/A
vblkl	Voltage below which the converter is blocked (i.e. Iq = Ip = 0)	N/A
vblkh	Voltage above which the converter is blocked (i.e. Iq = Ip = 0)	N/A
Tblk_delay	The time delay following blocking of the converter after which the converter is released from being blocked	0.04 – 0.1

Important Note: note that the position of *qvmax*/*qvmin* is different to previous versions of the REEC_* models. It is now being applied on any incoming signal from the plant controller, or power factor control, regardless of the chosen flags within REEC_D.

[12] The typical range of values here are a preliminary set of values discussed briefly with several vendors. They are to be further discussed and verified with the vendors once the model is implemented.

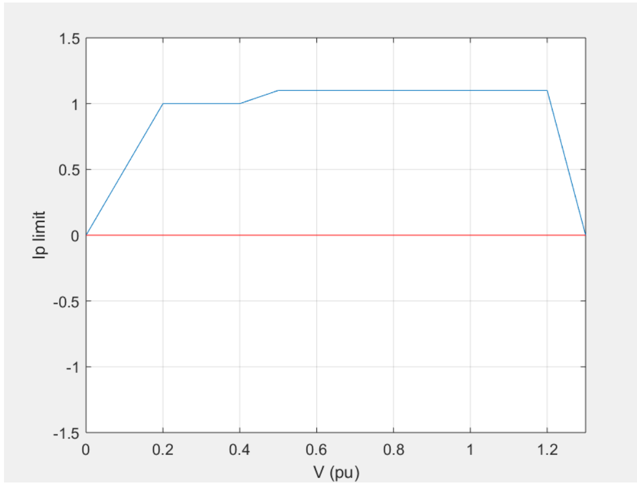

Figure 4: Example of VDLp table with $Ke = 0$.

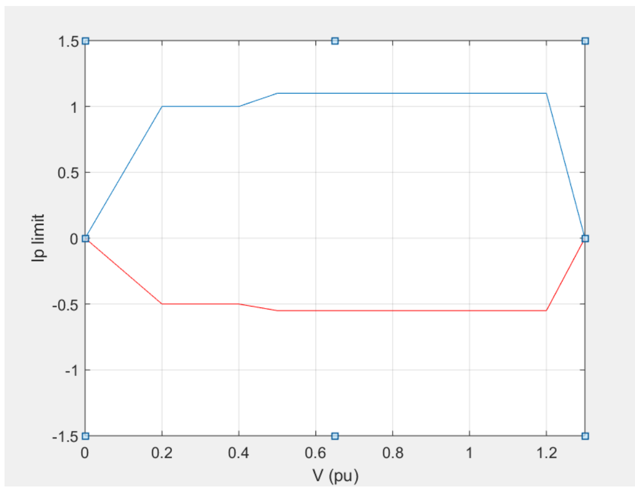

Figure 5: Example of VDLp table with $Ke = 0.5$ to emulate a storage system which can charge at a maximum charging current of half as much as the maximum discharging current.

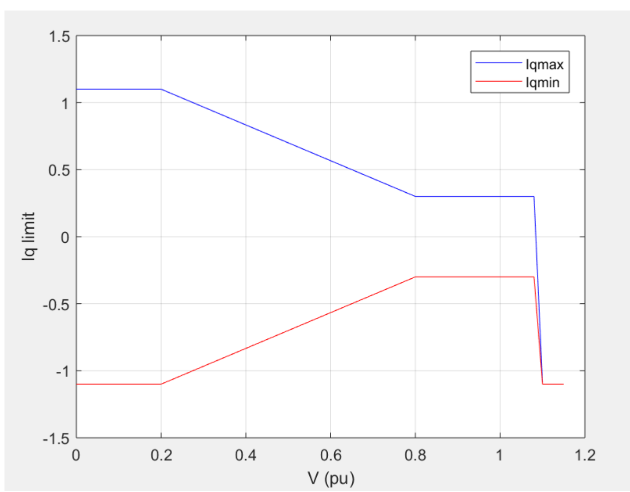

Figure 6: Example of VDLq table.

4.0 Plant Controller Models

4.1 REPC_C (new model)

This proposed new plant-controller model is based off of the existing REPC_A model, which interfaces to a single aggregated WTG model. These same changes may also be applied to the REPC_B model to create a second complex plant controller – whether that is needed should be further discussed at the next WECC MVS meeting.

The REPC_C model is identical to the REPC_A model, with the following additions/modifications (see Figure 7):

1. Limits on Vref and Qref: Adding max/min limits on the Vref and Qref input signals.
2. Auxiliary Inputs: Adding a *Vaux* and *Paux* signal to the voltage and power reference summing junctions, which can be accessed through programing by the user to implemented user-written auxiliary controls.
3. Measurement Time Constants: The measurement time-constant (*Tfltr*) is added also to the V_{reg} signal, and it has been moved for the current-compensation block. Also, a separate measurement delay time-constant (*Tc*) has been added to the reactive-current compensation block and frequency measurement (*Tfrq*).
4. Rate Limits: rate limits have been added to *Qref* (*dqrefmax/dqrefmin*), *Plant Pref* (*dprmax/dprmin*), the power factor reference (*dpfmax/dpfmin*) and the output of the reactive-power command signal (*qvrmax/qvrmin).* The rate limits on the reactive-power command (*qvrmax/qvrmin*) should be disabled (e.g. set to 9999/-9999) when the plant controller is producing a voltage signal (*Vext*). The model should determine automatically, upon initialization, whether the output (*Qext/Vext*) is a Q-command or V-command, depending on the downstream *reec_** model.
5. Power Factor Control: Adding the functionality of power factor control at the plant level. The constant pfaref is not a user entered value. Upon initialization, *pf* is internally calculated by the program to be the power factor based on the power flow solution from the Q_{branch} and P_{branch}), where here the values of Q_{branch} and P_{branch} are the initial values of these variables at time = 0 in the simulation. Moreover,
    a. the calculate Q = abs(*Pbranch*)×pf because we want to cater for a battery case where *Pbranch* may be negative for charging (otherwise the sign convention of the power factor limits will flip between charging and discharging and cause confusion specially if inside a simulation the battery goes from charging to dischraging), and
    b. If *Pbranch* < 1% of the plant rating and the plant is in *PF* control (*RefFlag* = 2), then upon initialization the software will force it to constant-Q control (i.e. *RefFlag* is forced from 2 to 0), and a warning message is issued to the user that this has been done since constant power factor control at very low loads does not make sense (e.g. *P* = 0 how does one even calculate power factor?).
6. A feedforward path in the frequency control: A new feedforward path can be introduced by setting the flag *Ffwrd_flag = 1*. See the special instructions (yellow high-lighted box in Figure 7) on how this flag is to be utilized. This is used by some vendors and should be used with specific information from the vendor. When the *Ffwrd_flag* is used, the values of *pimax/pimin* may be set differently to *Pmax/Pmin*.
7. Coordinated Switching of Shunt Capacitors and Reactors: The logic for coordinated switching of mechanically switched shunts (MSS) has been added. This logic is copied from the generic SVS models [6]. For completeness, the switching logic is depicted in Figure 8. There are eight (8) parameters associated with the switching logic. The four reactive thresholds at which switching occurs (*Qdn1, Qdn2, Qup1, Qup2*), the two-time delays for switching (*Tdelay1* and *Tdelay2*), the time delays associated with the opening/closing of the MSS breaker (*Tmssbrk*), and the discharging time of the shunt capacitors (Tout). For more details see [6] or the logic of any of the SVS models (SVSMO1, SVSMO2,

and SVSMO3). In addition, each shunt capacitor/reactor that is to be controlled by this plant controller in the power flow case, during time-domain stability simulations, must also be associated with generator on which the REPC_C model is instantiated. As an example, for the SVSMO* models this is done in GE PSLF™ by specifying the bus number and ID of the SVS device that is controlling the shunt in the shunt data record. The same approach could be used here. For now, we will assume that no more than ten (10) MSSs are to be controlled by a plant controller.

8. <u>Freezing of both Active and Reactive Power Output:</u> the states s2, s3, s5 and s6 are all frozen if the filtered voltage (s7) falls below *Vfrz*. Furthermore, once the filtered voltage recovers above Vfrz, these states remain frozen for another *Tfrz* seconds (which can be 0). This delay has been introduced to allow coordination with the downstream inverter/turbine models such that the freeze remains until the inverter level LVRT actions have been completed. Furthermore, the delay can be used to ensure that the plant level controls do not react to spikes in frequency or voltage caused by simulating a fault at the point-of-measurement (see https://www.wecc.biz/Reliability/WECC_White_Paper_Frequency_062618_Clean_Final.pdf). If Tfrz is none-zero the user must ensure that there is proper LVRT and voltage control actions at the inverter/turbine level controls and it is properly coordinated.

9. If the monitoring branch is no defined by the user, or ill-defined, then the model will default to assuming that the *Pbranch* and *Qbranch* get set to *Pgen* and *Qgen* for the aggregated generating unit being controlled by *REPC_C* and not set to 0. Also, a warning message should be issued to the user that the branch has not been correctly defined and so the software is assuming *Pgen*/*Qgen* which is not entirely correct, but a necessary simplification to allow the model to still function. **<u>Note:</u>** for the MSC switching logic there is a second branch defined to monitor *Qbranch2*. If that branch is not correctly defined then the MSC control logic should be disabled and a warning message issued to the user to this effect. (*It is recommended that this logic around the missing branch data be made retroactively to the REPC_A model as well.)*

10. <u>This model should also obey the so-</u>called baseload flag that is used in the major North American software tools. Namely,
    a. If baseload flag = 0, then the model behaves normally,
    b. If baseload flag = 1, then *Pmax* = initial power flow MW output of the plant ($Pgen_o$) and thus the power order can only go down and not up, and
    c. If baseload flag = 2, then *Pmax*=*Pmin* = $Pgen_o$ and thus the power order is fixed.

<u>Note:</u> the rate-limits shown as state s10, 11 and 12 are implemented by using a very small time-constant (T = 2×delt) to effect the rate limit.

<u>MSS Switching Logic:</u> it is important to note that when switching the MSSs first any in-service MSS that can be switched out to meet the need must be switched out, prior to switching in another MSS. That is, circulating Vars must be avoided. The pseudo code below demonstrates the intent.

(let Q = Qbranch2)

If (Qdn1 < Q < Qup1)

Do nothing

Else

If (Qdn2 < Q < Qup2)

If (Q < Qdn1) {that is Q is too inductive}

After Tdelay1 seconds (Q must remain in this range for that duration before even initiating switching) initiate the switching out of any in-service shunt capacitor first or if all MSCs are out, then switch in the first available shunt reactor, if nothing is available then there is nothing you can do; remember the breaker time after MSS switching engaged; the discharge time of the capacitor must always be obeyed once it is switched out

Else {must be too capacitive, since landing here we are already outside of Qdn1 < Q < Qup1}

After Tdelay1 seconds (Q must remain in this range for that duration before even initiating switching) initiate the switching out of any in-service shunt reactor first or if all MSRs are out, then switch in the first available shunt capacitor, if nothing is available then there is nothing you can do; remember the breaker time after MSS switching engaged; the discharge time of the capacitor must always be obeyed if previously switched out in the run

End

Else

If (Q < Qdn2) {that is Q is too inductive}

After Tdelay2 seconds (Q must remain in this range for that duration before even initiating switching) initiate the switching out of any in-service shunt capacitor first or if all MSCs are out, then switch in the first available shunt reactor, if nothing is available then there is nothing you can do; remember the breaker time after MSS switching engaged; the discharge time of the capacitor must always be obeyed once it is switched out

Else {must be too capacitive, since landing here we are already outside of Qdn2 < Q < Qup2}

After Tdelay2 seconds (Q must remain in this range for that duration before even initiating switching) initiate the switching out of any in-service shunt reactor first or if all MSRs are out, then switch in the first available shunt capacitor, if nothing is available then there is nothing you can do; remember the breaker time after MSS switching engaged; the discharge time of the capacitor must always be obeyed if previously switched out in the run

End

End

End

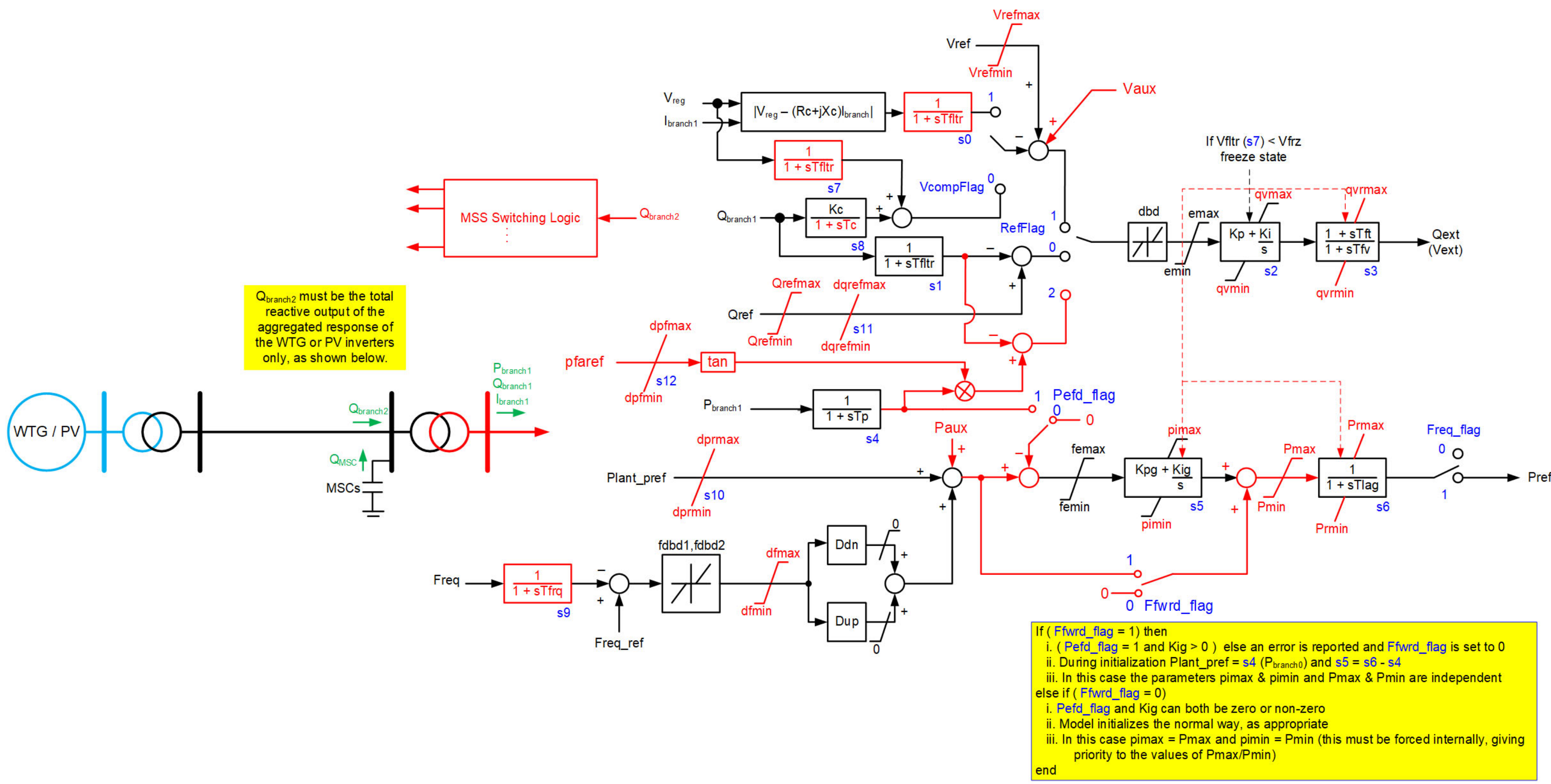

Figure 7: REPC_C model

Table 4: Parameter List for REPC_C – only new additional parameters are listed here; all other parameters are identical to REPC_A.

Parameter	Description	Typical Range/Value
Fbus	Bus at which frequency is measured for the primary frequency response [13]	**N/A**
Vrefmax	Maximum voltage reference [pu]	1.05 – 1.08
Vrefmin	Minimum voltage reference [pu]	0.95 – 1.0
Qrefmax	Maximum Q-refence [pu]	N/A
Qrefmin	Minimum Q-refence [pu]	N/A
dqrefmax	Maximum rate of increase of Q-reference [pu/s]	N/A (set to 9999 to disable)
dqrefmin	Maximum rate of decrease of Q-reference [pu/s]	N/A (set to -9999 to disable)
qvrmax	Maximum rate of increase of Qext (Vext) [pu/s]	N/A (set to 9999 to disable)
qvrmin	Maximum rate of decrease of Qext (Vext) [pu/s]	N/A (set to -9999 to disable)
dprmax	Maximum rate of increase of Plant Pref [pu/s]	N/A (set to 9999 to disable)
dprmin	Maximum rate of decrease of Plant Pref [pu/s]	N/A (set to -9999 to disable)
pfmax	Maximum power factor limit	0.9 to 0.95 (typical)
pfmin	Maximum power factor limit	-0.9 to -0.95 (typical)
Prmax	Maximum rate of increase of Pref [pu/s]	N/A (set to 9999 to disable)
Prmin	Maximum rate of decrease of Pref [pu/s]	N/A (set to -9999 to disable)
pimax	Maximum output of the active power PI controller [pu]	1.0
pimin	Minimum output of the active power PI controller [pu]	0.0
Pefd_flag	Enable (1) or disable (0) electrical power feedback	N/A
Tc	Reactive-current compensation time-constant [s]	0 – 2
Ffwrd_flag	Feedforward flag (1) include feedforward and (0) disable	N/A
Qdn1	First stage of capacitor (reactor) switching out (in) [pu] – *Qdn1 < 0*	N/A
Qdn2	Second stage of capacitor (reactor) switching out (in) [pu] – *Qdn2 < Qdn1*	N/A
Qup1	First stage of capacitor (reactor) switching in (out) [pu] – *Qup1 > 0*	N/A
Qup2	First stage of capacitor (reactor) switching in (out) [pu] – *Qup2 > Qup1*	N/A
Tdelay1	Time delay after which if Q < Qdn1 (or Q > Qup1) a capacitor (reactor) is switched [s]	N/A
Tdelay2	Time delay after which if Q < Qdn2 (or Q > Qup2) a capacitor (reactor) is switched [s] – typically *Tdelay2 < Tdelay1*	N/A
Tmssbrk	Time it takes to switch in (out) a mechanically switched shunt [s]	0.05 – 0.1 (set to zero to disable)
Tout	Time for discharging of a capacitor that has just been switched out; the same capacitor cannot be switched back in until Tout [s] has elapsed	typically, 120 – 300 seconds

[13] **Note:** if *Fbus* = 0 (i.e. no bus number is entered) then the same bus is used for frequency measurement as the bus defined for the POI that is used for the voltage measurement for the volt-var control-loop, which is the from bus of the defined monitored branch.

Tfrz	A time delay during which the states (s2, s3, s5 and s6) are kept frozen even after the filtered voltage recovers above Vfrz. This can be used to ensure the plant controller does not interact with the inverter-level LVRT.	0 – 2 seconds
dfmax	Maximum frequency error [pu]	0.01 to 999
dfmin	Minimum frequency error [pu]	-0.01 to -999

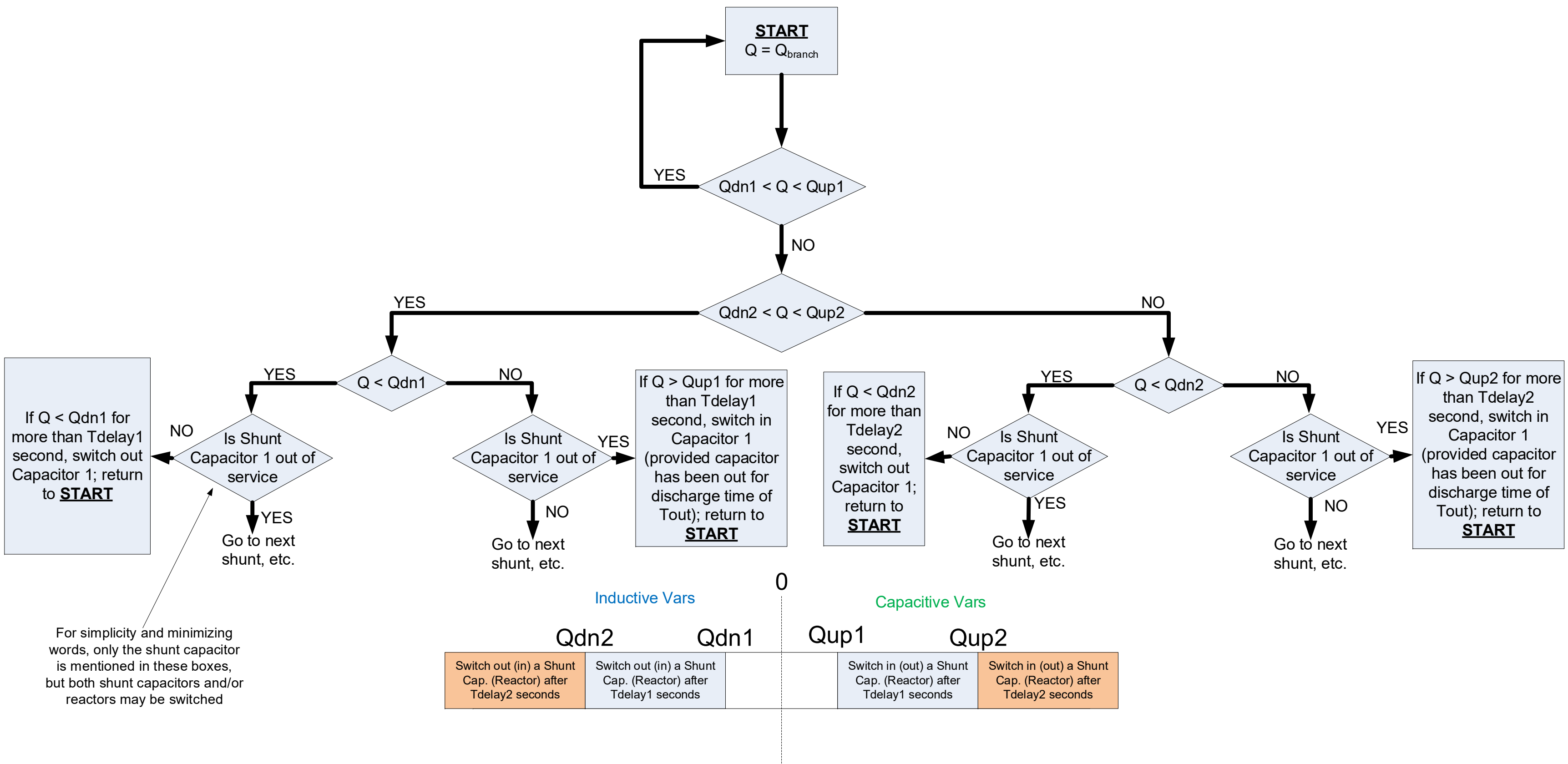

Figure 8: MSS switching logic.

5.0 Changes to Mechanical Side Models for WTGs

5.1 WTGP_B (new model)

Based on experience it has become clear that the pitch-controller model would benefit from the following modification, which is shown here as a proposed new model call WTGP_B (Figure 9). This is because in the actual controls the limits are often independent and thus in some simulation cases this can make a difference. The difference between WTGP_B and the existing WTGP_A model is as follows:

1. The limits on the two integrators are now independent of the limits on the lag block (state s2). This provides for greater flexibility, which is often needed to emulate the behavior of the actual equipment.
2. The non-windup limits on *s0, s1* an *s2* are to be implemented in the following way:
   a. Each of the integrator below has its own non-windup limit, for example, for *s1*

```
ds1 = Kiw × (Pord – Pref)
  if (s1 ≥ θcmax)
    s1 = θcmax
  elseif (s1 ≤ θcmin)
    s1 = θcmin
  end
  if ((s1 ≥ θcmax) and (ds1 > 0))
    ds1 = 0
  elseif ((s1 ≤ θcmin) and (ds1 < 0))
    ds1 = 0
  end
```

   Then s0 and s2 would also have similar non-windup limits of their own.

   b. Then in addition to the above independent non-windup limits on each integrator, the following four statements are implemented which pertain only to the RED arrows below in the diagram:

```
if (s2 ≥ θcmax and ds1 > 0)
        ds1 = 0
end
if (s2 ≤ θcmin and ds1 <0)
        ds1 = 0
end
if (s2 ≥ θcmax and ds0 >0)
        ds0 = 0
end
if (s2 ≤ θcmin and ds0 <0)
        ds0 = 0
end
```

   The point here is that the derivative of the states are set to zero when the pitch-angle hits its limits and any of the integrators are trying to push it further in the said direction.

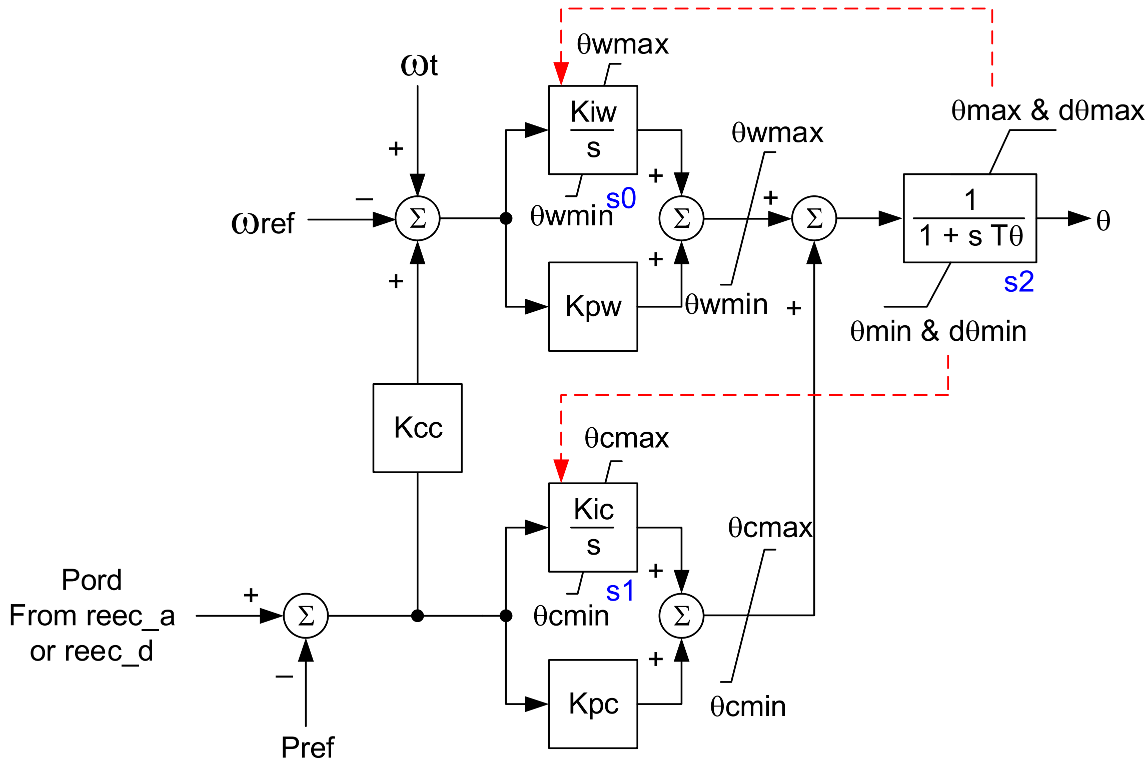

Figure 9: The WTGP_B model

Table 6: Parameter List for WTGP_B – only new additional parameters are listed here; all other parameters are identical to WTGP_A.

Parameter	Description	Typical Range/Value
$\theta cmax$	Maximum output of the pitch compensation controller [degrees]	20 – 30
$\theta cmin$	Minimum output of the pitch compensation controller [degrees]	-5 – 0
$\theta wmax$	Maximum output of the speed error controller [degrees]	20 – 30
$\theta wmin$	Minimum output of the speed error controller [degrees]	-5 – 0

5.2 WTGT_B (new model)

This is a proposed new drive-train "emulation" model where the only input is the electrical power (Pe) of the generator and mechanical power (*Pm*) is simply taken to be a filtered value of Pe. This should **ONLY** be used with the type 4A (i.e. type 4 WTG with torsional emulation – see Figure 3-9 in the model specification for WTGs here: https://www.wecc.biz/Reliability/WECC-Second-Generation-Wind-Turbine-Models-012314.pdf). The idea is simple. When a type 4A WTG is modeled all the mechanical side models are neglected[14] (i.e. pitch control, torque control, etc.). As such, in the *wtgt_a* model mechanical power (*Pm*) remains constant. This results in a net increase in speed after the simulation of a fault or other disturbance that

[14] **Note:** work done years ago showed that for a type 4 WTG, since the generator is fully decoupled from the grid, modeling all the mechanical elements provides little added fidelity for grid electrical response modeling – see https://www.wecc.biz/Reliability/WECC-Type-4-Wind-Turbine-Generator-Model-Phase-II-012313.pdf

momentarily decreased *Pe*, and thus a slight error in Pe in steady-state once the fault clears. In real-life, following a disturbance, the change in turbine speed initiates the pitch control to adjust *Pm* to bring the turbine back to its original steady-state speed and thus power. By doing what is shown in Figure 10, *Pm* changes in a way that is similar (certainly not exactly the same) to the action of the pitch control following a disturbance in *Pe*. Thus, the steady-state error in *Pe* does not occur. This is illustrated by a simple example simulation below.

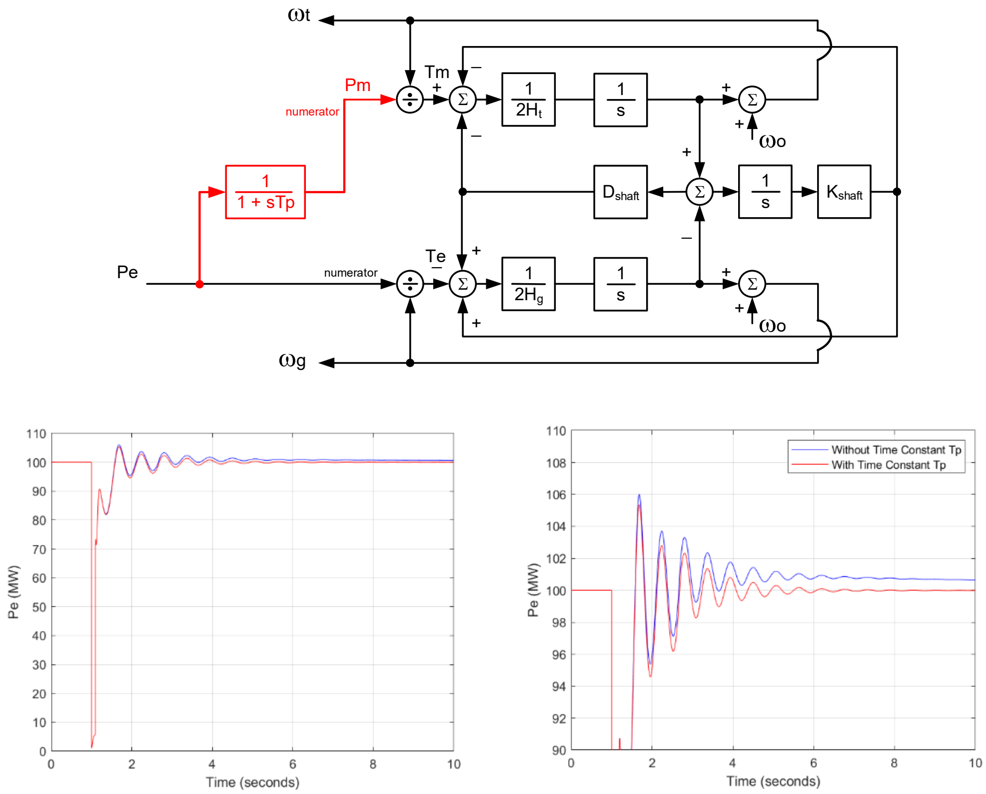

Figure 10: The WTGT_B model

6.0 Proposed New Auxiliary Models

There has been discussion at several previous MVS meetings about two additional modules, which we will call "auxiliary controls". These modules apply only to wind turbine generators presently.

1. Weak-grid option controls
2. Inertia based Fast Frequency response

Below we describe these new proposed options, and we have also implemented them as simple user-written models and thus demonstrate their efficacy below in simple simulations.

6.1 WTGWGO_A (new model)

Some vendors (Siemens-Gamesa in particular) have indicated that in some cases they offer a supplemental control called a weak grid option (WGO). Here we present a simple generic model, as an auxiliary control which can be used to "emulate" this supplemental control. The model is shown in Figure 11. This auxiliary control would fit between the plant controller (REPC_*) and the electrical controls model (REEC_*). That is the Pref coming from the plant controller would go into *Pref_in*, and the output of this model, *Pref_out*, would go into the electrical controller for the WTG. For now, the intent is to have this model used only for type 4 WTGs – i.e. between REPC_* and REEC_*, as shown below. Once further testing is done with the model, it may be extended in the future to also interface between the WTGQ_* and REEC_* model for type 3 WTGs.

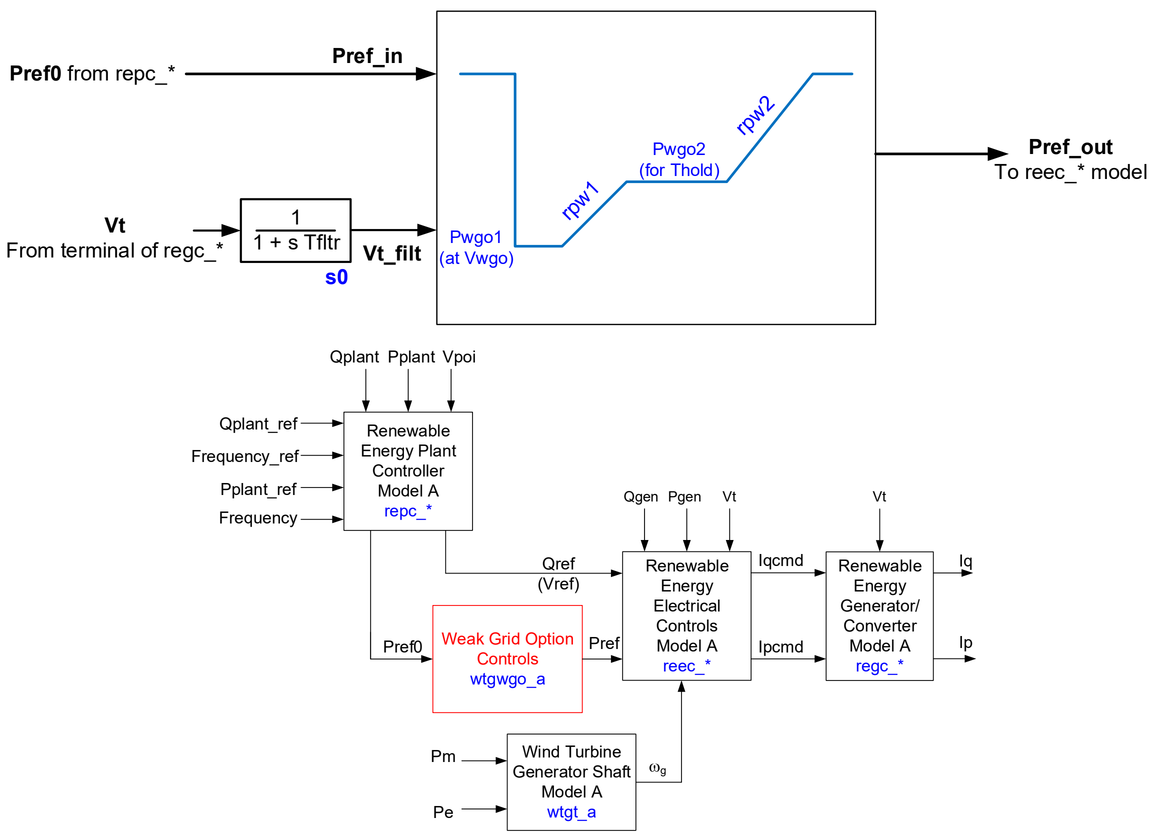

Figure 11: The WTGWGO_A model, and how it fits into the sequence of RES models.

The model can be explained as follows:

1. If the filtered terminal voltage (*Vt_filt*) of the WTG falls below *Vwgo*, then the WGO function is initiated.
2. The power reference is then held at *Pwgo1* until the voltage recovers to above (*Vwgo* + *eps*).

3. After this point the power reference is ramped at a rate of rpw1 to the next level *Pwgo2* and is held there (at *Pwgo2*) for *Thold* seconds.
4. Then, finally, the power reference is ramped back to its initial value at a rate of *rpw2*.
5. Presently, the intent is to implement this model to work as shown in Figure 11 with either the REPC_A or REPC_C models. It will not, for the initial implementation, be configured to function with the REPC_B (multiple device controller) model. Furthermore, at the discretion of the software vendors they may for now configure the initial implementation to work only with type 4, PV and BESS models, to be possibly extended in the future to function with type 3 WTGs models. In the case of type 3 WTG, the *Pref* output from *WGO* will need to be fed to both the *wtgp_a* and *wtgq_** models.

Once initiated, the WGO follows through the whole process.

To illustrate its effectiveness, we developed a user-written model of this proposed auxiliary control in GE PSLF™ and used it on the existing RES generic models to illustrate its behavior for a simple test case. The results are shown in Figures 12 to 16. As can be seen the user-written model performs as expected. Figure 17 shows the actual performance of the type 4 WTG, with the WGO supplemental controls, as presented by Siemens Wind in March 2017 to the WECC MVS [7]. As can be seen the performance of the model is very similar to the actual WTG performance.

Table 7: Parameter List for WTGWGO_A model.

Parameter	Description	Typical Range/Value
Vwgo	Voltage threshold below with the WGO function is initiated [pu]	0.5 – 0.6
Pwgo1	Power reference held during a fault when WGO is initiated [pu]	0.4 – 0.6
rpw1	Ramp rate at which power is increased from *Pwgo1* to *Pwgo2* [pu/s]	0.2 – 2
Pwgo2	Power reference held for Thold seconds after the fault [pu]	0.5 – 0.7
rpw2	Ramp rate at which power is increased from *Pwgo2* back to normal [pu/s]	0.2 – 2
Thold	Time for which the power reference is held at *Pwgo2* [s]	1 – 2
eps	Small hysteresis on voltage recovery to start the first ramp [pu]	0.01 – 0.02

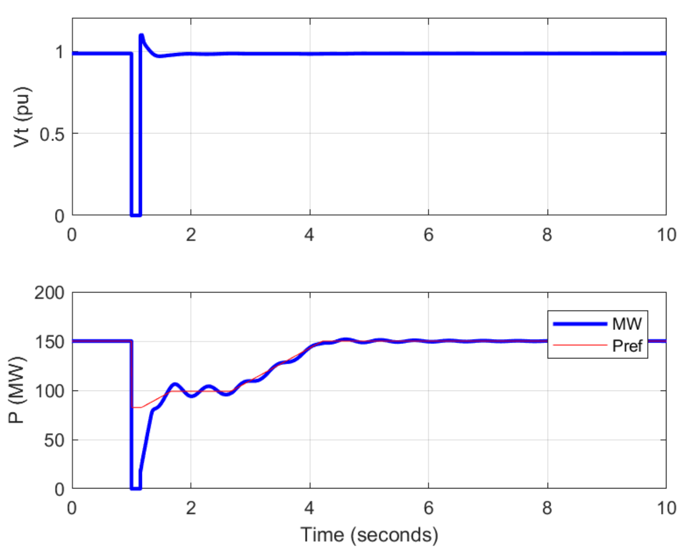

Figure 12: Type 4 WTG generic model with the user-written *WTGWGO_A* model. Fault at WTG terminals resulting in 0% retained voltage.

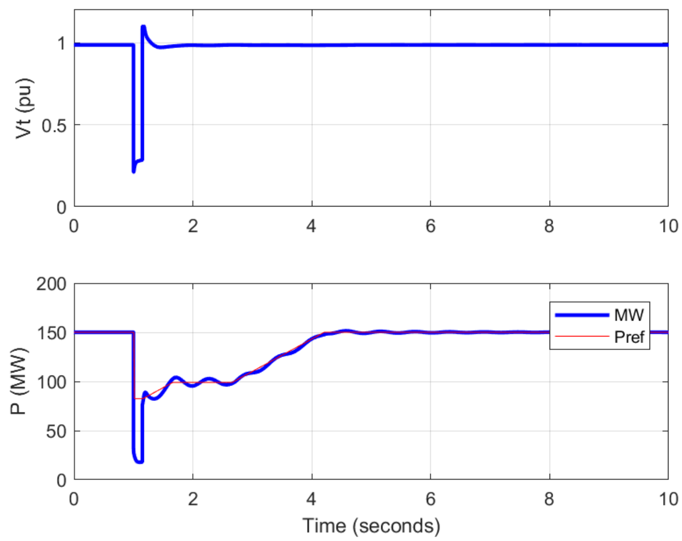

Figure 13: Type 4 WTG generic model with the user-written *WTGWGO_A* model. Fault at WTG terminals resulting in ~30% retained voltage.

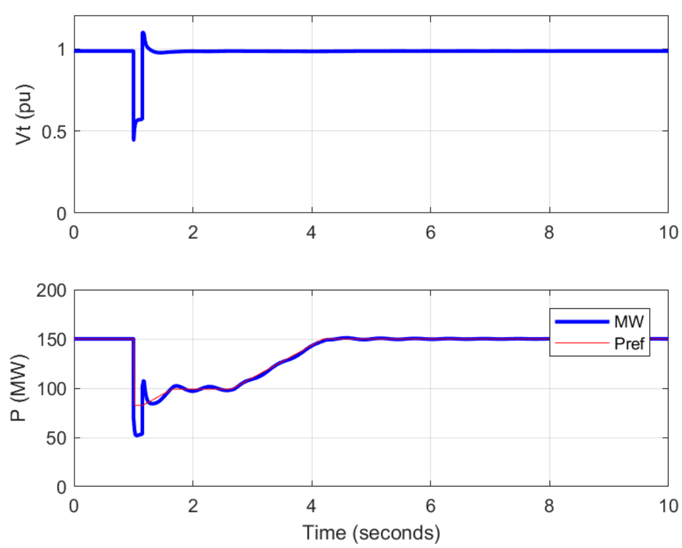

Figure 14: Type 4 WTG generic model with the user-written *WTGWGO_A* model. Fault at WTG terminals resulting in ~60% retained voltage.

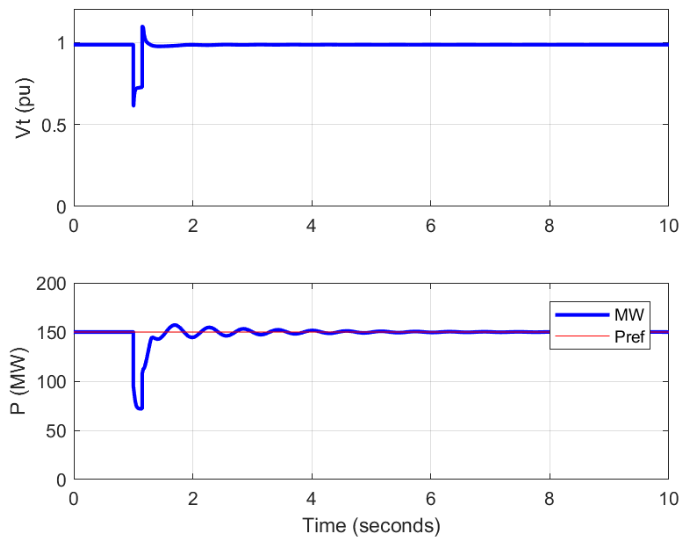

Figure 15: Type 4 WTG generic model with the user-written *WTGWGO_A* model. Fault at WTG terminals resulting in ~75% retained voltage – WGO is not initiated since voltage does not go below the threshold (*Vwgo*).

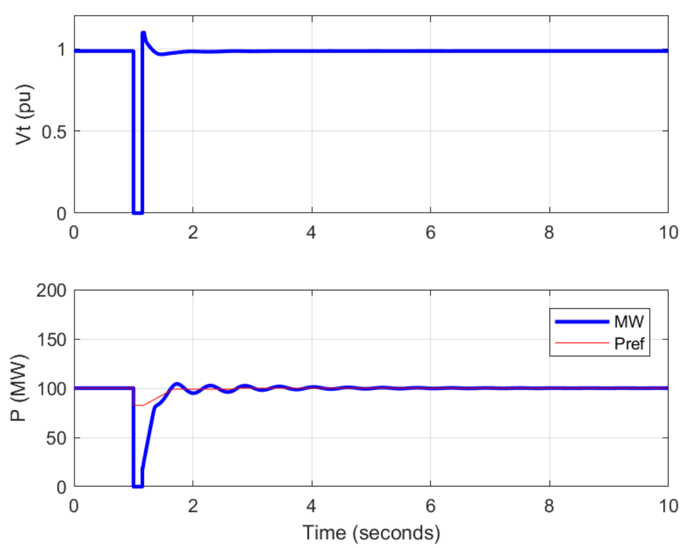

Figure 16: Type 4 WTG generic model with the user-written *WTGWGO_A* model. Fault at WTG terminals resulting in 0% retained voltage. The WGO has little impact, as would be expected, since the initial output of the WTG is below right at the hold threshold *Pwgo2* of the WGO settings.

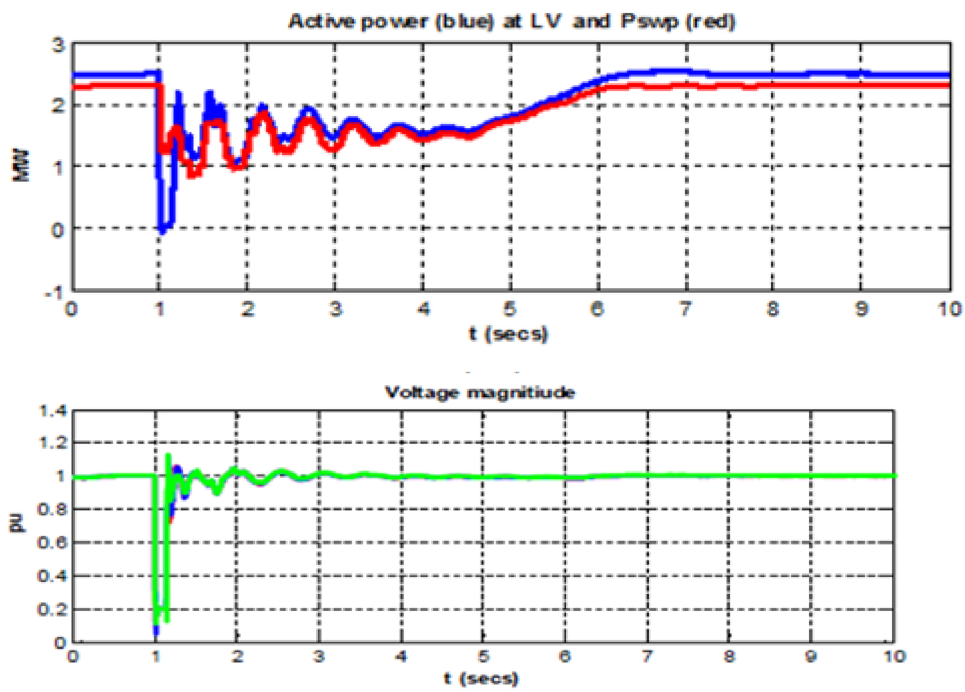

Figure 17: Actual performance of a WTG using the WGO for a Siemens WTG as presented to WECC MVS [7].

6.2 WTGIBFFR_A (new model)

Another auxiliary control feature that is available from many wind turbine manufacturers is the so-called inertial-based fast-frequency response (IBFFR)[15] controls. It is outside of the scope of this document to describe this functionality and its many aspects and nuances. A proposal has been presented at a few previous WECC MVS meetings [8]. Here, based on [8], a simple user-written model was developed in GE PSLF™ to demonstrate the proposed model for IBFFR. We call it the WTGIBFFR_A model. It is shown in Figure 18. The model, the underlying assumptions behind it, and its functionality may be explained as follows:

- Although it is understood that the actual implementation of the so-called inertial-based fast-frequency response[16] is perhaps quite different among the various wind turbine generator manufacturers, none-the-less, for the purposes of a generic model one can define four (4) regions in the response, as shown in Figure 18, namely:
  - *Trise* – which is the time it takes for the power of the unit to rise from its initial power to the peak value during the IBFFR.
  - *Tpeak* – which is the time (duration) that the WTG remains at the peak value of the IBFFR, with the peak of response being a percentage of the initial power, i.e. $Peak = Po \times (1 + dP)$
  - *Tfall* – which is the time it takes for the power to fall back down, and typically (when the incident wind energy is below rated wind power) the turbine will fall below its initial power output. It will fall to a level that is a percentage of the initial power, i.e. $Pmin = Po \times (1 - dPmin)$.

[15] The name IBFFR for this function/model is tentative, and the final naming convention will be discussed in future WECC meetings once this model starts to be implemented by the software vendors.
[16] This is sometime referred to in the literature as “synthetic inertia” or “emulated inertia” but we prefer the terminology of inertia-based fast-frequency response.

- *Trec* – which is the time it takes for the power to recover back to its initial value; this is the time during which energy is given back to the rotor to bring it back to its initial speed.

These parameters thus define the dynamic behavior of the IBFFR. There is, however, a few other caveats that are discussed in the points that follow. Also, there is one other parameter that is needed, namely the deadband (*dbd*) in frequency. When the frequency, as measured at the WTG or wind power plant level, falls below (1 – *dbd*) [pu] then the IBFFR function is initiated. Also, ***note that this supplemental control is only initiated for under-frequency events and for a decrease in system frequency.***

- With the above in mind the proposed model is shown in Figure 19, and Figure 20 shows how it fits into the mix of the RES models. It was implemented, as a user-written model, and tested on a small test system. Some example simulations are shown in Figures 21, 22 and 23. The model may be explained as follows:
  - The error in frequency is calculated (err) where frequency is measured at the point-of-interconnection of the wind power plant[17]. The function F1 represents the following simple logic: *if err ≤ dbd then out1 = 0, else out1 = 1*. Thus, when frequency falls by more than *dbd* [pu] the IBFFR control is initiated.
  - The function F2 represents the following simple logic: *if s0 ≥ dP.Po then out = 1, else out = 0*. Then *out2 = out* (the output of F2) after a delay of *Tpeak* seconds. Where *Po* here (and in all cases below) denotes the initial turbine electrical power output as determined from the initial power flow solution.
  - The function F3 represents the following simple logic: *if s1 ≤ -(dP + dPmin).Po then out3 = 1, else out3 = 0*.
  - The rise for recover (*Trec*) can be calculated from the other parameters (and is not a user-input) in order to ensure that the energy taken out of the shaft (Area A in Figure 18) is equal to the energy returned to the shaft (Area B in Figure 18)[18]. It is can easily be shown that:

$$Trec = \frac{2dP}{dPmin}\left[\frac{Trise}{2} + Tpeak + \left(\frac{dP}{dP + dPmin}\right)\frac{Tfall}{2}\right] - Tfall\left[1 - \frac{dP}{dP + dPmin}\right] \quad (1)$$

  - Finally, a perusal of the table of parameters (Table 8) will show that we have assigned six (6) sets of values of *dP, dPmin, Trise, Tpeak* and *Tfall*, associated with six different power levels of the turbine (*p1* to *p6*). The actual amount of IBFFR available from a WTG is dependent of the incident wind energy (wind speed) and the rotation speed of the shaft. However, for the generic RES models wind speed is not an available input and the shaft speed is not available for some of the type 4 WTGs. Thus, we have made the assumption that the initial power output of the WTG (in per unit of the rated output) is a reasonable indicator of both these variables (i.e. incident wind speed and rotor speed). Thus, this matrix of 6 × 6 values works in the following way:

    if (Po ≥ p6)

[17] Measuring frequency at the POI bus is perhaps preferred in the software programs to minimize the issues related to frequency calculation in positive-sequence programs to the extent possible (see https://www.wecc.biz/Reliability/WECC_White_Paper_Frequency_062618_Clean_Final.pdf)

[18] The energy taken back from the system to speed up the turbine (Area B in Figure 18), in a real system, is likely slightly greater than the energy injected into the grid (Area A in Figure 18) at the onset of IBFFR. Thus, the two are not necessarily equal, since not only does the turbine need to be returned to its original speed, but also there are losses in the process that will need to be covered (e.g. such as loss of lift). However, in the generic stability level model presented here, such losses are not modeled and so it is assumed that Area A = Area B. An important note is that if the incident wind speed is above rated wind speed (i.e. significantly greater than the wind speed at which the WTG produces it rated MW output) then Area B will be essentially zero, since the additional energy can be extracted out of the surplus wind energy.

```
        dP = dP6
        dPmin = dPmin6
        Trise = Trise6
        Tpeak = Tpeak6
        Tfall = Tfall6
elseif (Po ≥ p5)
        dP = dP5
        dPmin = dPmin5
        Trise = Trise5
        Tpeak = Tpeak5
        Tfall = Tfall5
elseif (Po ≥ p4)
        dP = dP4
        dPmin = dPmin4
        Trise = Trise4
        Tpeak = Tpeak4
        Tfall = Tfall4
elseif (Po ≥ p3)
        dP = dP3
        dPmin = dPmin3
        Trise = Trise3
        Tpeak = Tpeak3
        Tfall = Tfall3
elseif (Po ≥ p2)
        dP = dP2
        dPmin = dPmin2
        Trise = Trise2
        Tpeak = Tpeak2
        Tfall = Tfall2
elseif (Po ≥ p1)
        dP = dP1
        dPmin = dPmin1
        Trise = Trise1
        Tpeak = Tpeak1
        Tfall = Tfall1
else
        Model is inactive – that is there is no IBFFR
end
```

Note that the *Po* (initial power output of the "aggregated" wind turbine) that is used as the base from which to calculate the change in power etc. should be the value of the initial steady-state power output of the "aggregated" wind turbine generator just prior to the initiation of the IBFFR event. Thereafter, once an IBFFR event starts, the value of *Po* is unchanged.

Now it should be noted that the actual IBFFR that is supplied by each individual turbine in a wind power plant is dependent on many factors, and most importantly on (i) the incident wind energy (wind speed) on a turbine, and (ii) the initial speed of the rotor of the turbine. When performing large scale stability studies, whether using a generic model such as those discussed here, or detailed user-written vendor specific models, one thing is for certain and that is we cannot predict with much accuracy what the wind-speed and rotor-speed of each wind turbine in a wind power plant (WPP) is going to be for a future scenario. Furthermore, the accepted practice for modeling WPPs in large scale stability studies is by using an aggregated WTG model with a simple feeder model. Thus, it is not feasible to model such details even if such data were available. In short, IBFFR cannot be made to emulate exactly what actual field response will be due to the stochastic nature of the resource. As such, we are in need of a simplifying assumption to make the model usable. Although clearly not representative of what would happen in the field, the most conducive assumption is to assume that all the WTGs in the WPP are at the same power level and experiencing the same wind speed.

With all of the above in mind, Figures 19 to 23 show example simulations with the model and we see a reasonable performance, and the fact that the IBFFR changes with the initial WTG power level as expected. Also, we see that the speed transients in the WTG are reasonable – i.e. speed initial declines as energy is extracted from the shaft and then speeds up again once energy is slowly put back into the shaft after the IBFFR is completed and we are in the recovery phase of IBFFR. **Note:** the IBFFR model should be used only in frequency stability studies. This simple model will not behave well if used when performing simulations of transmission faults close to the terminals of the WTGs (see https://www.wecc.org/Reliability/WECC_White_Paper_Frequency_062618_Clean_Final.pdf which is the WECC white paper on frequency calculation at or near faulted buses; such issues could falsely initiate an IBFFR event), thus it is not for studying a combination of faults and frequency events.

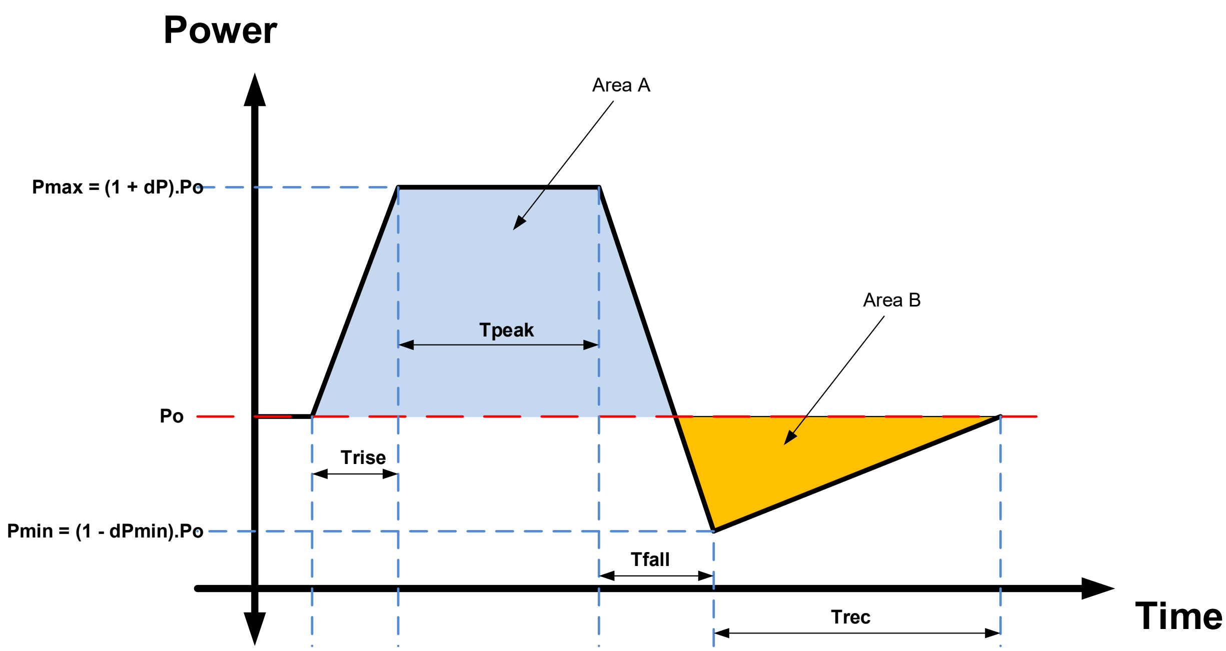

Figure 18: Characteristic of the IBFFR (from [8]).

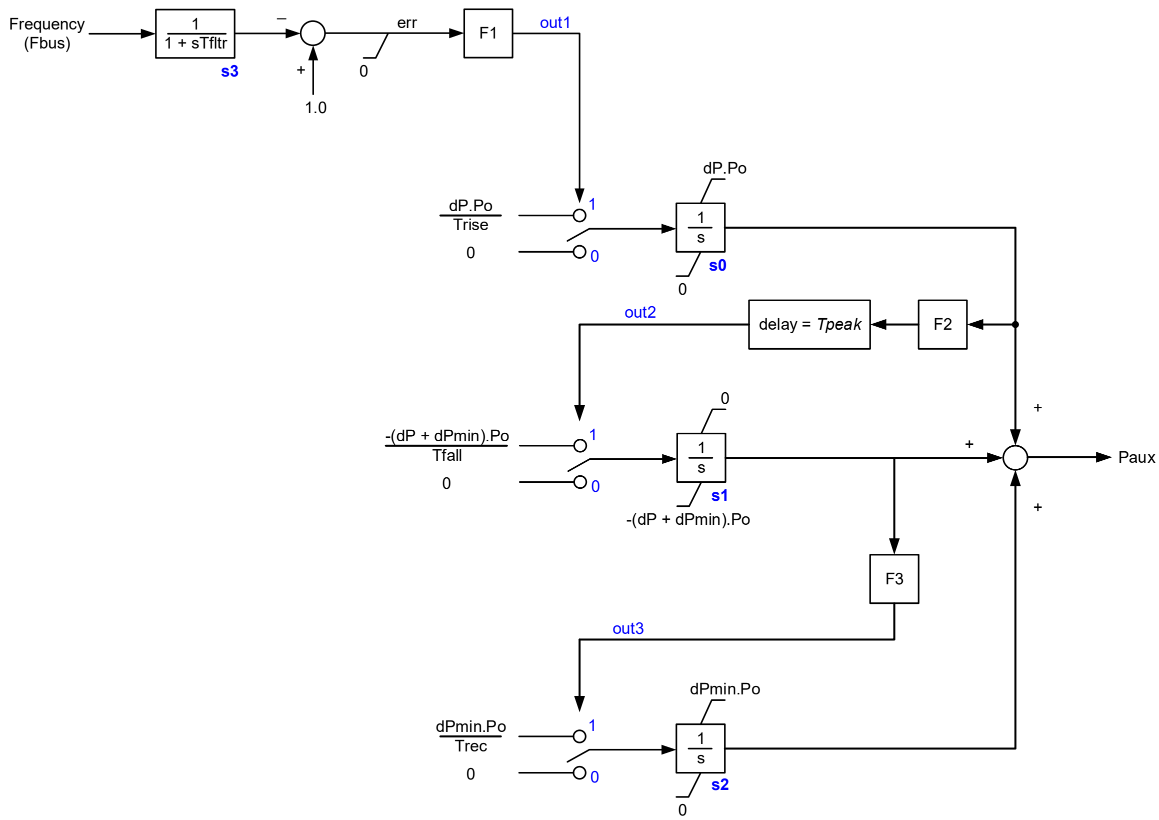

Figure 19: Block-diagram of the implemented WTGIBFFR_A model.

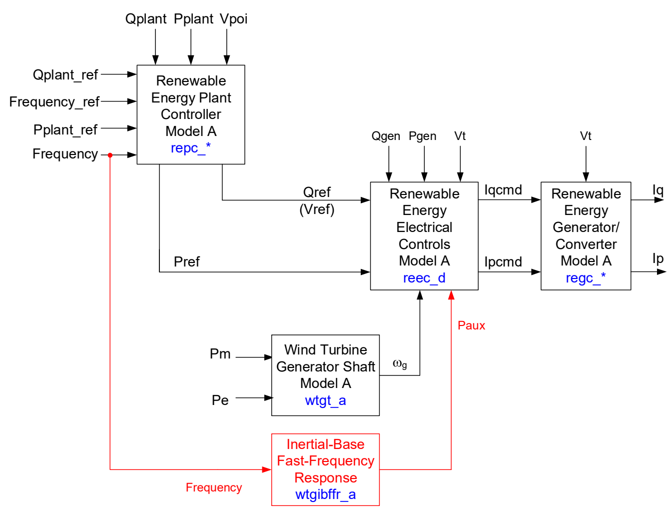

Figure 20: High-level block-diagram showing, as an example for a type 4 WTG, how WTGIBFFR_A fits into the RES models. For a type 3 WTG, it would work the same.

A final note is that once the IBFFR function is initiated at the beginning of an under-frequency event (under-frequency only), it must run its full course and the recovery period be completed. Depending on the value of the *Tlapse* parameter, the next time that the IBFFR function is made available will be *Tlapse* seconds after the full completion of the first instance of IBFFR. Typically, *Tlapse* is of the order of many minutes and much care should be exercised by the user not to enter an unrealistic value here. If a number is entered by the user for *Tlapse* that is ≤ 0, then the software will internally set *Tlapse* = *infinity*, that is IBFFR is exercised once and once only during the entire simulation. **The software should put out a warning message to the user during initialization if a value of *Tlapse* is entered by the user that is less than 60 seconds, indicating that this appears to be an unreasonably short value for *Tlapse* and that *Tlapse* is more typically 120 seconds or more.**

__Note:__ this model has also been tested/verified against field data for at least one vendor, see [12].

Table 8: Parameter List for WTGIBFFR_A model.

Parameter	Description	Typical Range
Tfltr	Filter time-constant for frequency measurement	0.02 – 0.05
Fbus	Bus number at which frequency is measured (generator bus if = 0)	N/A
Tlapse	Time lapse from the end of an IBFFR event, until IBFFR is available again [s]	120 – 1800[19]
dbd	Deadband below which IBFFR is initiated, that is when (1 – frequency) ≥ *dbd,* then the IBFFR is initiated [pu]	0.0008 – 0.0017
[p1, p2, p3, p4, p5, p6]	Six (6) power points corresponding to the six sets of parameters [pu]	N/A
[dP1 to dP6]	Six *dP* values (see Figure 18) [pu]	0.05 – 0.1
[dPmin1 to dPmin6]	Six *dPmin* values (see Figure 18) [pu]	0 – 0.08
[Trise1 to Trise6]	Six *Trise* values (see Figure 18) [s]	0.1 – 0.2
[Tpeak1 to Tpeak6]	Six *Tpeak* values (see Figure 18) [s]	1 – 2
[Tfall1 to Tfall6]	Six *Tfall* values (see Figure 18) [s]	0.2 – 0.5
[Trec1 to Trec6]	Six Trec values (see Figure 18) [s]	0

__**Very Important Note:**__

The six (6) *Trec* values in the model should typically be set to zero (0). By doing so the model internally calculates the value of *Trec* for each operating point using equation (1) in the above section. If, however, the user wishes to defined a *Trec* that is greater than the calculated value using equation (1), for each of the designated six operation points, then the user may populate the *Trec* parameters. This may be done in cases where the user may wish to represent the fact that Area B is actually larger than Area A in some cases due to losses during the period of power injection as the turbine speed significant declines from its optimal point of efficiency. In doing so, it must be noted that since positive sequence models do not represent such losses, by making Area B larger, the speed of the machine in simulation may end up artificially higher at the end of the simulation. Furthermore, if the user defined a value for *Trec* that is less than the value calculated by equation (1), then the model will ignore the user defined value and use that calculated by equation (1). This is because, except under operation when the incident wind speed is greater than rated wind speed, in all other cases Area B must always be either equal to or greater than Area A.

[19] In most cases it will be of the order of many minutes.

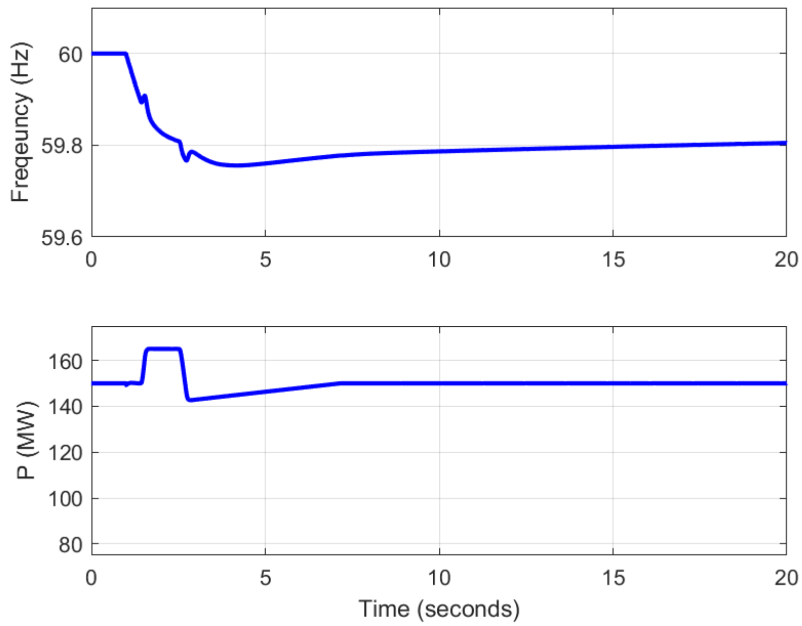

Figure 21: Simulation of the performance of the WTGIBFFR_A on a test case for a type 4 WPP using the generic RES models. Plant is initially at a high output.

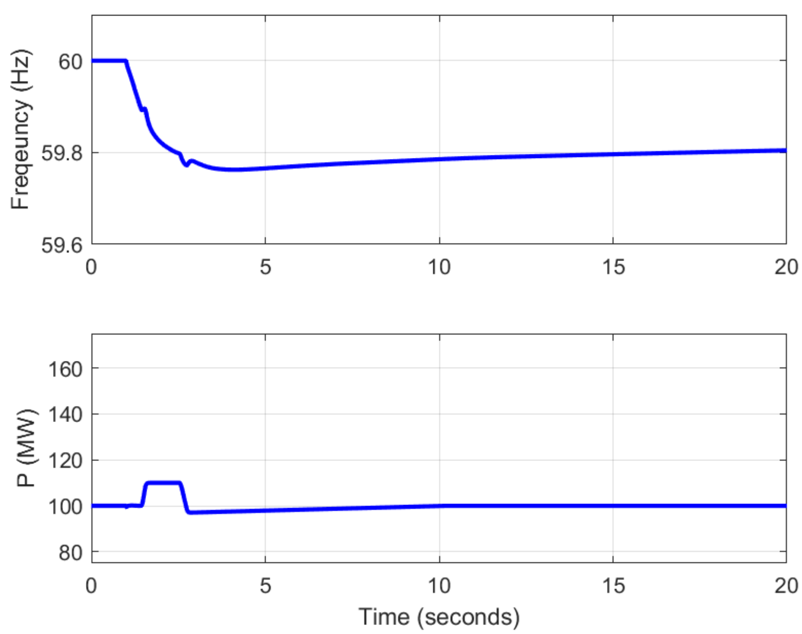

Figure 22: Simulation of the performance of the WTGIBFFR_A on a test case for a type 4 WPP using the generic RES models. Plant is initially at a lower output than the previous case. We can see the difference in the IBFFR response for the same event.

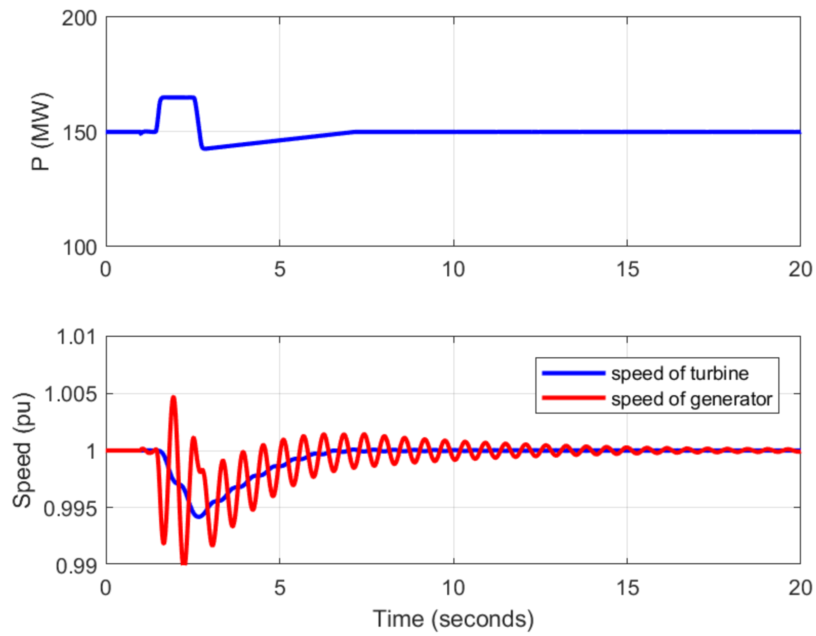

Figure 23: Same cases as Figure 21 – illustrating the fact that the energy for the IBFFR response is being taken out of the rotor-shaft and then put back in – i.e. shaft speed goes down initially and then slowly speeds back up to its initial value.

Acknowledgements:

PEACE® wishes to gratefully acknowledge EPRI research funding for time spent by PEACE® staff to prepare this memo and develop and test the user-written models presented in section 6.0. Furthermore, EPRI's funding and development of the user-written versions of REGC_B and REGC_C in [4] are gratefully acknowledged. Moreover, the continued efforts and contributions of GE, Siemens PTI, PowerWorld and PowerTech Labs are gratefully acknowledged. In addition, we acknowledge the input, comments and participation of many of the equipment vendors in this effort, including FirstSolar, Senvion and Siemens-Gamesa, as well as GE and Vestas who have been present at various meetings where the work here has been discussed.

Needless-to-say, a debt of gratitude is due to the WECC Modeling and Validation Working Group (S. Wang chairperson) and Renewable Energy Modeling Task Force (S. Zhu chairperson) for hosting the open forum for these discussions and developments. There are numerous members of the WECC MVS and REMWG who have provided comments, input and suggestions during the course of these efforts, and it would be difficult at best to attempt to properly capture all names.

Finally, needless-to-say, all entities involved in this activity self-fund within their own organizations their travel expenses to come to the WECC MVS meetings where much of this is discussed. This is gratefully acknowledged.

We apologize for any inadvertent omissions.

References:

[1] WECC Second Generation Wind Turbine Models, 2014 https://www.wecc.biz/Reliability/WECC-Second-Generation-Wind-Turbine-Models-012314.pdf

[2] EPRI Report, *Model User Guide for Generic Renewable Energy System Models*, Product Id: 3002014083, Date Published: July 13, 2018 https://www.epri.com/#/pages/product/3002014083/

[3] D. Ramasubramanian, Z. Yu, R. Ayyanar, V. Vittal and J. M. Undrill, "Converter Model for Representing Converter Interfaced Generation in Large Scale Grid Simulations", *IEEE Trans. PWRS*, April 2016.

[4] D. Ramasubramanian, E. Farantatos, A. Gaikwad and W. Wang, "Proposal for improved voltage source converter model – an update", Presented at the WECC MVWG Meeting, May 10, 2018; https://www.wecc.biz/Administrative/08%20Proposed%20Improved%20Voltaged%20Converter%20Model-%20Ramasubramanian.pdf

[5] WECC Generic Solar Photovoltaic System Dynamic Simulation Model Specification, September 2012, https://www.wecc.biz/Reliability/WECC-Solar-PV-Dynamic-Model-Specification-September-2012.pdf

[6] P. Pourbeik, D. J. Sullivan, A. Boström, J. Sanchez-Gasca, Y. Kazachkov, J. Kowalski, A. Salazar, A. Meyer, R. Lau, D. Davies and E. Allen, "Generic Model Structures for Simulating Static Var Systems in Power System Studies—A WECC Task Force Effort", *IEEE Transactions on PWRS*, August 2012.

[7] H. Ma and J. Nielsen, "*Siemens Wind Turbine Weak Grid Option Modeling in PSSE WECC library*", presented to the WECC MVWG in March 2017.

[8] P. Pourbeik, "*Emulating Synthetic Inertia Response for Wind Generation*", presented to the WECC MVWG in March 2017.

[9] D. Ramasubramanian, "A method of implementation of current limits in regc_b and regc_c", EPRI memo, dated 3/4/19. Sent to WECC REMTF.

[10] D. Ramasubramanian, W. Wang, P. Pourbeik, E. Farantatos, A. Gaikwad, S. Soni and V. Chadliev, "A Positive Sequence Voltage Source Converter Mathematical Model for Use in Low Short Circuit Systems", IET Generation, Transmission & Distribution, January 2020. https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/iet-gtd.2019.0346

[11] J. J. Sanchez-Gasca, S. D. Rao and D. Ramasubramanian, "Model regc_c preliminary evaluation", Presented at WECC MVS, May 18, 2022

https://www.wecc.org/_layouts/15/WopiFrame.aspx?sourcedoc=/Administrative/2022-05-18%20MVS%20Mtg%20regc_c%20tests_%20Sanchez-Gasca.pdf&action=default&DefaultItemOpen=1

[12] P. Pourbeik, "Example Verification of the Proposed IBFFR Model", Memo issued to WECC REMTF and EPRI, 7/17/19. https://www.wecc.org/Reliability/Memo_IBFFR_071719.pdf