Sensitivity Calculation of Different Parameters for Power Plant Collaboration

Ziqi Yang, E.I.T., University of Wisconsin Madison Electrical and Computer Engineering Department

***Abstract*—This paper introduces a dynamic simulation method to simulate a WECC 4 bus power plant model and discusses various optimization approaches to better simulate the model by varying some parameter values.**

***Index Terms*— Block Diagram Conversion, Control Modeling, Dynamic Simulation, Perturbation Analysis, Power System Modeling, Sensitivity Analysis.**

I. Introduction

THIS paper focuses on implementing a power system dynamic simulation method in Matlab using Newton-Trapezoidal Rule integration. A specific application of this approach is on the simulation of a WECC power plant model.

In order to simulate the model properly, this paper introduces necessary guidelines of converting model block diagrams to differential and algebraic equations. It also provides the basic method of converting a 4 bus power system to a two buses system through Thevenin equivalents.

II. Background Theory

In this section, the background of the Newton-Raphson Method and Trapezoidal Rule will be introduced.

A. Trapezoidal Rule

The trapezoidal Rule, which is similar to Forward Euler and Backward Euler methods, is a common technique for simulating nonlinear systems [1].

Assume a system has time variables x, y and u, and constant parameter p, then for differential equations, we have

$$\frac{dx}{dt} = f(x, y, u; p)$$

For algebraic equations, we have

$$0 = g(x, y, u; p)$$

By applying the discrete derivative with a small enough time step, the differentiation equation then can be re-written as

$$\frac{dx}{dt} = \lim_{h\to 0}\frac{x(t+h) - x(t)}{h} = \lim_{h\to 0}\frac{x_{n+1} - x_n}{h}$$

Where h is the time step. Still considering a small enough time step, we can then use the numerical method to approximate the differential equation

$$\frac{x_{n+1} - x_n}{h} = \frac{1}{2}f(x_{n+1}, y_{n+1}, u_{n+1}; p) + \frac{1}{2}f(x_n, y_n, u_n; p)$$
$$0 = g(x_{n+1}, y_{n+1}, u_{n+1}; p)$$

The approach above is called the Trapezoidal Rule Approximation, where $x_n = x(t)$ and $x_{n+1} = x(t+h)$. Although the Trapezoidal Rule appears as a simple average of the Forward Euler and Backward Euler schemes, the local truncation error scales with h^3. The Trapezoidal Rule is more accurate than the Euler Approximations.

B. Newton-Raphson Method

The Newton-Raphson method is a powerful technique for solving equations numerically [2].

Assume two equations $f_1(x, y) = 0$ and $f_2(x, y) = 0$, then

$$\begin{bmatrix} x(i+1) \\ y(i+1) \end{bmatrix} = \begin{bmatrix} x(i) \\ y(i) \end{bmatrix} + \begin{bmatrix} \frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} \\ \frac{\partial f_2}{\partial x} & \frac{\partial f_2}{\partial y} \end{bmatrix}^{-1} \times \begin{bmatrix} 0 - f_1(x(i), y(i)) \\ 0 - f_2(x(i), y(i)) \end{bmatrix}$$

The inverse matrix is called the Jacobian matrix. The Newton-Raphson method is one of the fastest convergences methods. However, the Newton-Raphson method depends on the initial values, in the example above, the initial values are $x(1)$ and $y(1)$

III. Network Thevenin Equivalent

The power system network used for this project is a four buses system as shown in Figure 1.

In Table I, values of parameters in Figure 1 are listed. The generator has a specific power base, and the rest of the network has another specific power base. For the transformer, ‘a’ is the primary to the secondary tap ratio.

Since the model analyses in the following sections are based on a two bus system, it is required to convert the system

---

This work was initially submitted to Dr. Bernard Lesieutre for review on May 2nd, 2014. This work was supported in part by the Western Electricity Coordinating Council (WECC).

Ziqi Yang is with the University of Wisconsin Madison Electrical and Computer Engineering Department, Madison, WI 53706 USA (email: zyang79@wisc.edu)

in Figure 1 to its Thevenin equivalent circuit as shown in Figure 2.

TABLE I
NETWORK VALUES

Symbol	Value
Z_1	0.0032+j0.01895 pu
Z_2	0.0002+j0.00382 pu
Z_3	0.00030 pu
Y	j0.35034 pu
a	1:0.92911
S_G	599.8+j5.74 MVA
S_L	27.51+j14.39 MVA
S_{Base2}	676 MVA
S_{Base1}	100 MVA
V_1	1.09661 pu
Z_L	$(\frac{1.014^2}{S_L/S_{Base2}})^*$ pu

For the value of Z_L, 1.014 represents the magnitude of generator voltage.

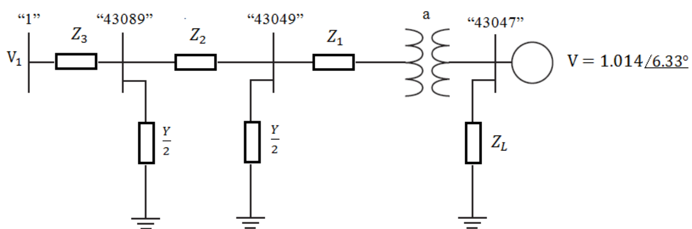

Fig.1 Power System Network

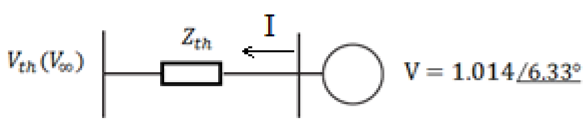

Fig.2 Thevenin equivalent system for the network

For Thevenin equivalent circuit transformation, the basic idea is to short out all the voltage sources and open all the current sources to calculate the Thevenin equivalent impedance, and apply the node or mesh circuit analysis for the Thevenin equivalent voltage calculation [3].

Based on the given values of the system in Figure 1, we can calculate the Thevenin equivalent voltage is $V_{th} = 1.016676 - j0.005374$ pu and the Thevenin equivalent impedance is $Z_{th} = 0.003732 + j0.134233$ pu. The base power is also unified to be 676MVA, which is same as the base power of the generator in Figure 1.

For the following model analysis and calculation, the network model will be based on the Thevenin equivalent network model in Figure 2.

IV. MODEL DESCRIPTIONS

In this part, the synchronous machine and the network model with saturation will be given. Also, the Stabilizer Model and Voltage Regulator/Exciter Model which are applied in this project will be described. Necessary approaches of converting block diagrams to mathematical equations will be introduced as well.

A. Synchronous Machine and Network Model

The reduced Synchronous Machine Modeling is represented by the following equations [4]. In this model, all the variables are in per-unit values except ω_r, which is in real value.

$$T'_{do}\frac{dE'_q}{dt} = -E'_q - (X_d - X'_d)[\frac{X''_d - X_{ls}}{X'_d - X_{ls}}I_d - \frac{X'_d - X''_d}{(X'_d - X_{ls})^2}\psi_{1d} + \frac{X'_d - X''_d}{(X'_d - X_{ls})^2}E'_q] + E_{fd} \quad (1)$$

$$T'_{qo}\frac{dE'_d}{dt} = -E'_d + (X_q - X'_q)[\frac{X''_q - X_{ls}}{X'_q - X_{ls}}I_q - \frac{X'_q - X''_q}{(X'_q - X_{ls})^2}\psi_{2q} - \frac{X'_q - X''_q}{(X'_q - X_{ls})^2}E'_d] \quad (2)$$

$$T''_{do}\frac{d\psi_{1d}}{dt} = -\psi_{1d} + E'_q - (X'_d - X_{ls})I_d \quad (3)$$

$$T''_{qo}\frac{d\psi_{2q}}{dt} = -\psi_{2q} - E'_d - (X'_q - X_{ls})I_q \quad (4)$$

$$\frac{d\delta}{dt} = \omega_r - \omega_e \quad (5)$$

$$\frac{2H}{w_s}\frac{d\omega_r}{dt} = T_M - (\psi_d I_q - \psi_q I_d) - T_{FW} = T_M + (X''_d - X''_q)I_d I_q - \frac{X''_d - X_{ls}}{X'_d - X_{ls}}E'_q I_q - \frac{X''_q - X_{ls}}{X'_q - X_{ls}}E'_d I_d - \frac{X'_d - X''_d}{X'_d - X_{ls}}\psi_{1d} I_q + \frac{X'_q - X''_q}{X'_q - X_{ls}}\psi_{2q} I_d - T_{FW} \quad (6)$$

$$0 = (R_s + R_e)I_d - \frac{\omega_r}{\omega_s}(X''_q + X_e)I_q - \frac{\omega_r}{\omega_s}\frac{X''_q - X_{ls}}{X'_q - X_{ls}}E'_d + \frac{\omega_r}{\omega_s}\frac{X'_q - X''_q}{X'_q - X_{ls}}\psi_{2q} + V_\infty sin(\delta) \quad (7)$$

$$0 = (R_s + R_e)I_q + \frac{\omega_r}{\omega_s}(X''_d + X_e)I_d - \frac{\omega_r}{\omega_s}\frac{X''_d - X_{ls}}{X'_d - X_{ls}}E'_q - \frac{\omega_r}{\omega_s}\frac{X'_d - X''_d}{X'_d - X_{ls}}\psi_{1d} + V_\infty cos(\delta) \quad (8)$$

$$0 = R_e I_d - \frac{\omega_r}{\omega_s}X_e I_q + V_\infty sin(\delta) - V_d \quad (9)$$

$$0 = R_e I_q + \frac{\omega_r}{\omega_s}X_e I_d + V_\infty cos(\delta) - V_q \quad (10)$$

$$0 = (V_d^2 + V_q^2)^{\frac{1}{2}} - V_t \quad (11)$$

Where $T_{FW} = D(\omega_r - \omega_e)$.

B. Saturation Model

In addition to the machine and network model, the next step is to combine the Saturation model with the system. The following equations represent the saturation model of this project [5].

$$0 = \frac{\psi''_d}{\psi''}S_G(|\psi''|) - S_{fd} \quad (12)$$

$$0 = \frac{\psi''_q}{\psi''}\frac{X_q - X_{ls}}{X_d - X_{ls}}S_G(|\psi''|) - S_{1q} \quad (13)$$

$$0 = [(\psi''_d)^2 + (\psi''_q)^2]^{\frac{1}{2}} - |\psi''| \quad (14)$$

$$0 = \frac{X''_d - X_{ls}}{X'_d - X_{ls}}E'_q + \frac{X'_d - X''_d}{X'_d - X_{ls}}\psi_{1d} - \psi''_d \quad (15)$$

$$0 = -\frac{X_q''-X_{ls}}{X_q'-X_{ls}}E_d' + \frac{X_q'-X_q''}{X_q'-X_{ls}}\psi_{2q} - \psi_q'' \quad (16)$$

Where $S_G(|\psi''|) = Ae^{B|\psi''|}$. And $S_G(1.0) = 0.124$, $S_G(1.2) = 0.504$, then A and B are calculated to be A = 0.00011178, B = 7.0114735.

Also, Eqn. (1) and Eqn. (2) should have saturation terms, and values should be slightly changed.

$$T'_{do}\frac{dE'_q}{dt} = -E'_q - (X_d - X'_d)[\frac{X''_d-X_{ls}}{X'_d-X_{ls}}I_d - \frac{X'_d-X''_d}{(X'_d-X_{ls})^2}\psi_{1d} + \frac{X'_d-X''_d}{(X'_d-X_{ls})^2}E'_q] + E_{fd} - S_{fd} \quad (1)$$

$$T'_{qo}\frac{dE'_d}{dt} = -E'_d + (X_q - X'_q)\left[\frac{X''_q-X_{ls}}{X'_q-X_{ls}}I_q - \frac{X'_q-X''_q}{(X'_q-X_{ls})^2}\psi_{2q} - \frac{X'_q-X''_q}{(X'_q-X_{ls})^2}E'_d\right] + S_{1q} \quad (2)$$

C. Voltage Regulator/Exciter Model

Voltage regulator is designed to maintain a constant voltage level [6], while the generator exciter is used to supply the magnetizing current to generate the working flux [7].

The voltage regulator/exciter model used for this simulation is the IEEE (2005) type ST4B excitation system[1].

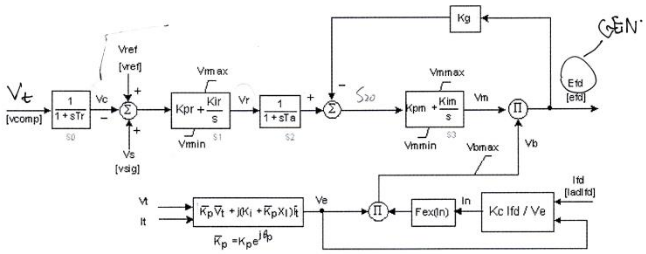

Fig.3 Block diagram for ST4B excitation System [8]

As shown in Figure 3, for the block diagrams with upper and lower limit, the implementations in Figure 4 and Figure 5 will be applied.

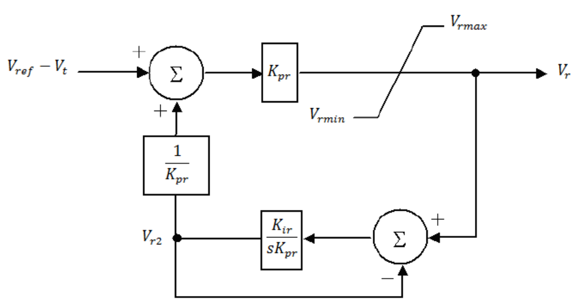

Fig.4 Implementation for Block s1

If $K_{pr}\left[(V_{ref} - V_t) + \frac{V_{r2}}{K_{pr}}\right] > V_{rmax}, V_r = V_{rmax}$

If $K_{pr}\left[(V_{ref} - V_t) + \frac{V_{r2}}{K_{pr}}\right] < V_{rmin}, V_r = V_{rmin}$

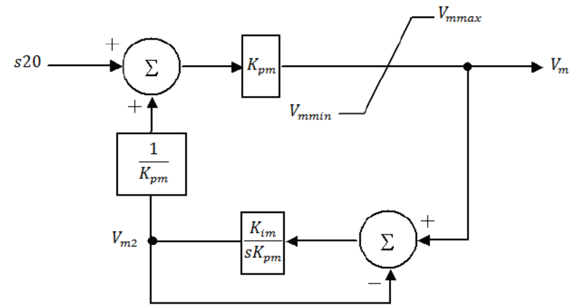

Fig.5 Implementation for Block s3

If $K_{pm}\left[s20 + \frac{V_{m2}}{K_{pm}}\right] > V_{mmax}, V_m = V_{mmax}$

If $K_{pm}\left[s20 + \frac{V_{m2}}{K_{pm}}\right] > V_{mmax}, V_m = V_{mmax}$

Due to parameters T_r and T_a are equal to zero, the blocks s0 and s2 can be eliminated. Also, all variables are in per-unit values. Therefore, based on Figure 1, the block diagram will be converted to equations as shown below[2].

$$\frac{dV_{r2}}{d_t} = \frac{K_{ir}}{K_{pr}}(V_r - V_{r2}) \quad (17)$$

$$\frac{dV_{m2}}{d_t} = \frac{K_{im}}{K_{pm}}(V_m - V_{m2}) \quad (18)$$

$$0 = K_{pr}(V_{ref} - V_t) + V_{r2} - V_r \quad (19)$$

$$0 = K_{pm}s20 + V_{m2} - V_m \quad (20)$$

$$0 = V_r - K_g E_{fd} - s20 \quad (21)$$

$$0 = \{[K_pV_d\cos(\theta_p) - K_pV_q\sin(\theta_p) - K_pI_dX_l\sin(\theta_p) - K_pI_qX_l\cos(\theta_p) - K_iI_q]^2 + [K_pV_q\cos(\theta_p) + K_pV_d\sin(\theta_p) + K_pI_dX_l\cos(\theta_p) - K_pI_qX_l\sin(\theta_p) + K_iI_d]^2\}^{\frac{1}{2}} - V_e \quad (22)$$

$$0 = \frac{1}{X_d - X_{ls}}\left[E_q' + (X_d - X_d')(I_d - I_{1d})\right] - I_{fd} \quad (23)$$

$$0 = \frac{X_d'-X_d''}{(X_d'-X_{ls})^2}\left[\psi_{1d} + (X_d' - X_{ls})I_d - E_q'\right] - I_{1d} \quad (24)$$

$$0 = \frac{K_cI_{fd}}{V_e} - I_n \quad (25)$$

For Eqn. (26)we have

If $I_n \le 0.433, 0 = 1 - 0.577I_n - Fex$

If $0.433 < I_n \le 0.75, 0 = (0.75 - I_n^2)^{\frac{1}{2}} - Fex$

If $0.75 < I_n \le 1, 0 = 1.732(1 - I_n) - Fex$

If $1 < I_n, 0 = -Fex$

$$0 = V_eFex - V_b \quad (27)$$

$$0 = V_bV_m - E_{fd} \quad (28)$$

D. Power System Stabilizer Model

Power system stabilizer (PSS) control provides a positive contribution by damping generator rotor angle swings, which are in a broad range of frequencies in the power system [9].

The power system stabilizer model used for this simulation is the Dual input Power System Stabilizer as shown in Figure 6[3].

[1] For this voltage regulator/exciter model ST4B, the block diagram is considered without OEL & UEL inputs & V_{gmax}.

[2] If only consider the voltage regulator/exciter model, simply set V_{st} to be 0 initially.

[3] Power System Stabilizer IEEE PSS2A type.

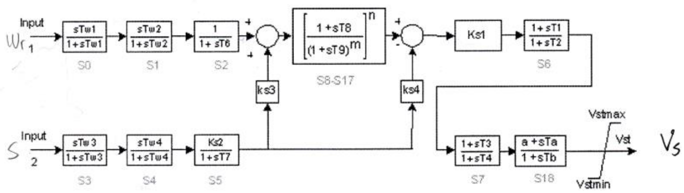

Fig.6 Block diagram for PSS2A Power System Stabilizer [8]

Based on Figure 6, implementation for blocks as s0 will be applied in Figure 7 using the equivalent feedback control block.

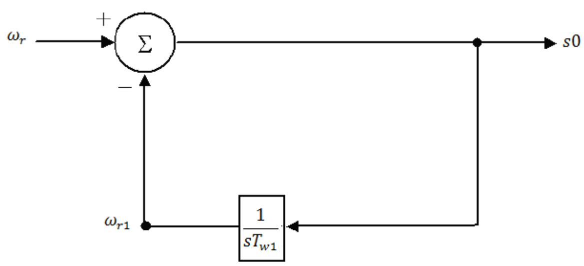

Fig.7 Implementation for Block s0

Also, implementation for blocks as s5 will be given in Figure 8.

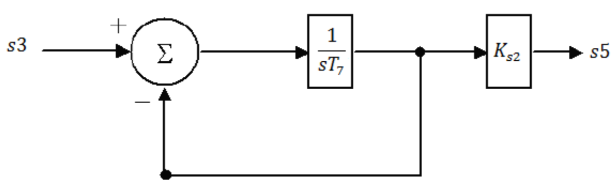

Fig.8 Implementation for Block s5

In addition, Figure 9 illustrates the implementation for blocks as s7.

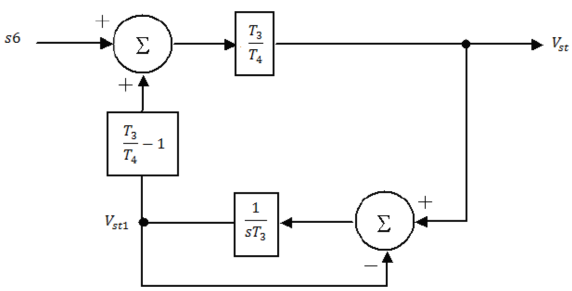

Fig.9 Implementation for Block s7

Because m = 4 and n = 2, the compounded blocks in s8-s17 shown in Figure 6 can be converted into several blocks in series connections.

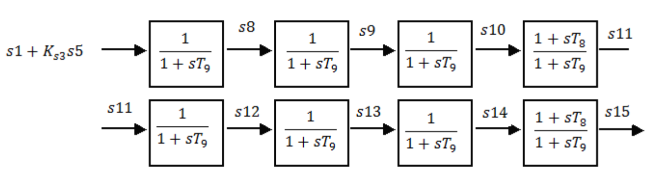

Fig.10 Conversion for compounded blocks

Due to T_6, T_{w4}, T_a and T_b are equal to zero, and a = 1, blocks s2, s4 and s18 can be eliminated. Also, Input1 ω_r (rotor speed) and Input2 S (power) are in per-unit values in this PSS2A model. Based on Figure 6, the block diagram then will be converted to equations as shown below.

$$\frac{d\omega_{r1}}{dt} = \frac{s0}{T_{w1}} \tag{29}$$

$$\frac{ds0_1}{dt} = \frac{s1}{T_{w2}} \tag{30}$$

$$\frac{ds3_1}{dt} = \frac{s3}{T_{w3}} \tag{31}$$

$$\frac{ds5}{dt} = \frac{K_{s2}s3 - s5}{T_7} \tag{32}$$

$$\frac{ds8}{dt} = \frac{s1 + K_{s3}s5 - s8}{T_9} \tag{33}$$

$$\frac{ds9}{dt} = \frac{s8 - s9}{T_9} \tag{34}$$

$$\frac{ds10}{dt} = \frac{s9 - s10}{T_9} \tag{35}$$

$$\frac{ds10_1}{dt} = \frac{s11 - s10_1}{T_8} \tag{36}$$

$$\frac{ds12}{dt} = \frac{s11 - s12}{T_9} \tag{37}$$

$$\frac{ds13}{dt} = \frac{s12 - s13}{T_9} \tag{38}$$

$$\frac{ds14}{dt} = \frac{s13 - s14}{T_9} \tag{39}$$

$$\frac{ds14_1}{dt} = \frac{s15 - s14_1}{T_8} \tag{40}$$

$$\frac{ds6_1}{dt} = \frac{s6 - s16_1}{T_1} \tag{41}$$

$$\frac{dV_{st1}}{dt} = \frac{V_{st} - V_{st1}}{T_3} \tag{42}$$

$$\text{s0} = \omega_r - \omega_{r1} \tag{43}$$

$$\text{s1} = \text{s0} - s0_1 \tag{44}$$

$$\text{s3} = \text{S} - s3_1 \tag{45}$$

$$\text{s11} = \frac{T_8}{T_9}s10 + (1 - \frac{T_8}{T_9})s10_1 \tag{46}$$

$$\text{s15} = \frac{T_8}{T_9}s14 + (1 - \frac{T_8}{T_9})s14_1 \tag{47}$$

$$\text{s6} = \frac{T_1}{T_2}K_{s1}s15 - \frac{T_1}{T_2}K_{s1}K_{s4}s5 + (1 - \frac{T_1}{T_2})s6_1 \tag{48}$$

$$V_{st} = \frac{T_3}{T_4}s6 + (1 - \frac{T_3}{T_4})V_{st1} \tag{49}$$

E. Model combination

Once combined the saturated machine model with the stabilizer model and the voltage regulator/exciter model, it is required to change Eqn. (43) so that all ω_r will appear as real values. Also, after the combination, Eqn. (19) in voltage regulator/exciter model is required to consider V_{st}. In addition, a new S equation needs to be added[4]. Therefore, the following equations are the changed/new equations.

$$0 = K_{pr}(V_{ref} - V_t + V_{st}) + V_{r2} - V_r \tag{19}$$

[4] This new S equation represents the power equation. In the power system stabilizer model, S was set to be a constant with the value of 0.8873.

$$s0 = \frac{\omega_r}{\omega_s} - \omega_{r1} \quad (43)$$

$$S = [(V_d I_d + V_q I_q)^2 + (V_q I_d - V_d I_q)^2]^{\frac{1}{2}} \quad (50)$$

The 50 equations listed above will then be used for the following project simulation. However, before starting the dynamic simulations, there still lefts an important step.

V. Dynamic Parameters Initialization

Based on the model descriptions in the previous sections, the next step is to calculate the initial value.

With the constant parameters listed in Table II, the following equations represent the initial value calculation for this system [4], and the network system is based on the circuit in Figure 2.

$$\begin{bmatrix} I_{th} \\ I \end{bmatrix} = \begin{bmatrix} Y_{th} & -Y_{th} \\ -Y_{th} & Y_{th} \end{bmatrix} \begin{bmatrix} V_{th} \\ V \end{bmatrix},\ Y_{th} = \frac{1}{Z_{th}}$$

$$I = I_s e^{j\varphi_s}, V = V_s e^{j\theta_s} = 1.014 e^{j6.33°}$$

$$\delta = \text{angle}\left[V + (R_s + jX_q)I\right] \quad (51)$$

$$I_d = I_s \cos(\varphi_s - \delta + 90°) \quad (52)$$

$$I_q = I_s \sin(\varphi_s - \delta + 90°) \quad (53)$$

$$V_d = I_s \cos(\theta_s - \delta + 90°) \quad (54)$$

$$V_q = I_s \cos(\theta_s - \delta + 90°) \quad (55)$$

$$V_t = (V_d^2 + V_q^2)^{\frac{1}{2}} \quad (56)$$

$$E'_d = (X_q - X'_q) I_q \quad (57)$$

$$E'_q = V_q + X'_d I_d \quad (58)$$

$$E_{fd} = E'_q + (X_d - X'_d)\, I_d \quad (59)$$

$$\psi_{2q} = -E'_d - (X'_q - X_{ls}) I_q \quad (60)$$

$$\psi_{1d} = E'_q - (X'_d - X_{ls}) I_d \quad (61)$$

$$T_M = E'_d I_d + E'_q I_q + (X'_q - X'_d) I_d I_q \quad (62)$$

$$S = [(V_d I_d + V_q I_q)^2 + (V_q I_d - V_d I_q)^2]^{\frac{1}{2}} \quad (63)$$

$$\psi''_d = \frac{X''_d - X_{ls}}{X'_d - X_{ls}} E'_q + \frac{X'_d - X''_d}{X'_d - X_{ls}} \psi_{1d} \quad (64)$$

$$\psi''_q = -\frac{X''_q - X_{ls}}{X'_q - X_{ls}} E'_d + \frac{X'_q - X''_q}{X'_q - X_{ls}} \psi_{2q} \quad (65)$$

$$|\psi''| = [(\psi''_d)^2 + (\psi''_q)^2]^{\frac{1}{2}} \quad (66)$$

$$S_{fd} = \frac{\psi''_d}{\psi''} S_G(|\psi''|) \quad (67)$$

$$S_{1q} = \frac{\psi''_q}{\psi''} \frac{X_q - X_{ls}}{X_d - X_{ls}} S_G(|\psi''|) \quad (68)$$

$$I_{1d} = \frac{X'_d - X''_d}{(X'_d - X_{ls})^2} [\psi_{1d} + (X'_d - X_{ls}) I_d - E'_q] \quad (69)$$

$$I_{fd} = \frac{1}{X_d - X_{ls}} [E'_q + (X_d - X'_d)(I_d - I_{1d})] \quad (70)$$

$$V_e = \{[K_p V_d \cos(\theta_p) - K_p V_q \sin(\theta_p) - K_p I_d X_l \sin(\theta_p) - K_p I_q X_l \cos(\theta_p) - K_i I_q]^2 + [K_p V_q \cos(\theta_p) + K_p V_d \sin(\theta_p) + K_p I_d X_l \cos(\theta_p) - K_p I_q X_l \sin(\theta_p) + K_i I_d]^2\}^{\frac{1}{2}} \quad (71)$$

$$I_n = \frac{K_c I_{fd}}{V_e} \quad (72)$$

TABLE II
CONSTANT PARAMETERS OF EACH MODEL

Model	Parameter	Value
Machine and network (saturation included)	T'_{do}	4.893
	T'_{qo}	0.54
	T''_{do}	0.0416
	T''_{qo}	0.067
	X_d	2.238
	X_q	2.001
	X'_d	0.357
	X'_q	0.573
	X''_d	0.293
	X''_q	0.293
	X_{ls}	0.249
	R_e	0.003732
	X_e	0.134233
	H	2.25
	D	0
	$S_G(1.0)$	0.124
	$S_G(1.2)$	0.504
	R_a	0
	R_{comp}	0
	X_{comp}	0
	T_M	0.8873
	ω_s	2π*60
Voltage Regulator/Exciter	T_r	0
	K_{pr}	40
	K_{ir}	4
	T_a	0
	V_{rmax}	4
	V_{rmin}	-2.67
	K_{pm}	20
	K_{im}	2
	V_{mmax}	4
	V_{mmin}	-2.67
	K_g	1
	K_p	1
	θ_p	0
	K_i	0
	K_c	0.062
	X_l	0
	V_{bmax}	1.2
	V_{ref}	1.014
Power System Stabilizer	T_{w1}	15
	T_{w2}	15
	T_{w3}	15
	T_{w4}	0
	T_1	0.11
	T_2	0.02
	T_3	0.11
	T_4	0.03
	T_6	0
	T_7	15
	T_8	0.3
	T_9	0.15
	T_a	0
	T_b	0
	K_{s1}	10
	K_{s2}	10/3
	K_{s3}	1
	K_{s4}	1
	n	2
	m	4
	a	1
	V_{stmax}	0.1
	V_{stmin}	-0.05

$R_e + jX_e = Z_{th}$, and T_M and V_{ref} are calculated to be constants.

If $I_n \leq 0.433, Fex = 1 - 0.577I_n$ (73)

If $0.433 < I_n \leq 0.75, Fex = (0.75 - I_n^2)^{\frac{1}{2}}$

If $0.75 < I_n \leq 1, Fex = 1.732(1 - I_n)$

If $1 < I_n, Fex = 0$

$$V_b = V_e Fex \quad (74)$$

$$V_m = \frac{E_{fd}}{V_b} \quad (75)$$

$$V_{m2} = V_m \quad (76)$$

$$s20 = 0 \quad (77)$$

$$V_r = K_g E_{fd} - s20 \quad (78)$$

$$V_{r2} = V_r \quad (79)$$

$$V_{ref} = V_t \quad (80)$$

$$\omega_r = 2\pi \times 60 \quad (81)$$

$$\omega_{r1} = \frac{\omega_r}{\omega_s} \quad (82)$$

$$s3_1 = S \quad (83)$$

For the rest variables, based on equations of the power system stabilizer model, the initial values are all equal to zero.

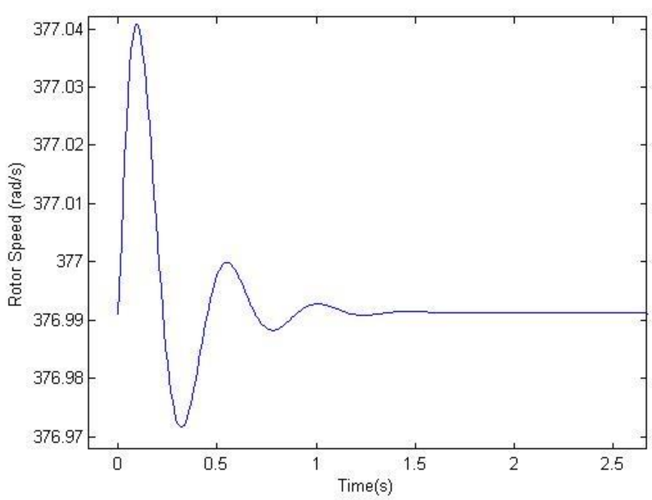

Fig.11 plot of ω_r to reach new steady state value

Figure 11 illustrates the process of ω_r converging to the new steady state value, which is the same as the calculated value $2\pi \times 60$rad/s.

Next, without changing V_{inf} and ω_e($V_{inf} = 1.0167 = |V_{th}|$, calculated from the Thevenin equivalent circuit in Figure 2, and $\omega_e = 2\pi \times 60$), applying these calculated initial values to the system with all models implemented[5], the system will then converge to a new steady state with these time variables converging to new values.

Table III records the differences between the calculated initial values and the new steady state value, it only shows small differences between these two groups of numbers.

We will take the new steady state values listed in Column 3 of Table III to be the new initial values for the rest simulation of this project.

[5] With the consideration of the new S equation, the total number of equations of this integrated model simulation is 50.

VI. Sensitivity Analysis and Perturbation Elimination

In this section, by analyzing selected parameters in Table II, some most sensible parameters will be changed to new values in order to better match the simulation with the WECC model.

A. WECC model descriptions

The WECC model records the instant V_1 (in Figure 1) and f_e readings at every $\frac{1}{30}$ second for 150 seconds. It also provides real power and reactive power readings at bus "1" with the same time points.

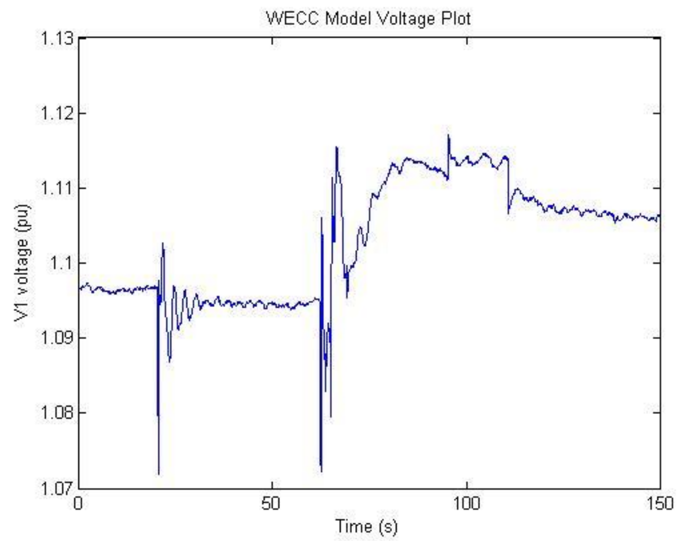

Fig.12 Plot of V_1 of WECC model, the unit of V_1 is kV

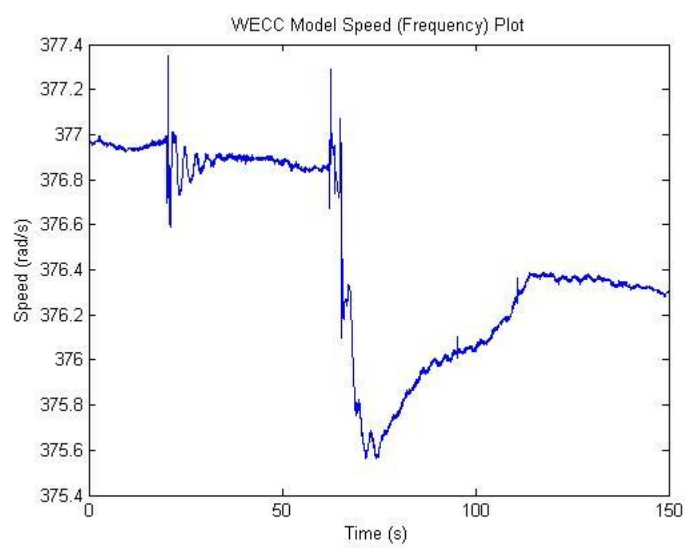

Fig.13 Plot of f_e of WECC model, the unit of f_e Hz

According to Figure 12 and Figure 13, the voltage varies from 535.96 kV to 558.56 kV, while the frequency ranges from 59.773 Hz to 60.057 Hz. In order to better apply these varied voltages and frequencies to the model described in previous sections, the initial step is to convert these V_1's to the corresponding Thevenin equivalents V_∞'s as shown in Section

III, and we should also multiply each f_e with 2π to have corresponding ω_e values.

B. Test run and sensitivity analysis

Similar to the new initial value identifications in Section IV, without changing values of any given parameters, the first step is to have a test run by applying these time-variant V_∞'s and ω_e's to those 50 simulation model equations.

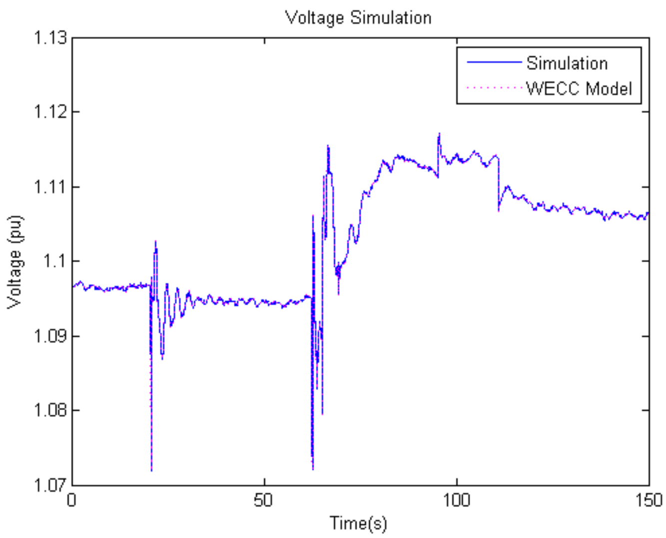

Fig.14 Voltage comparison

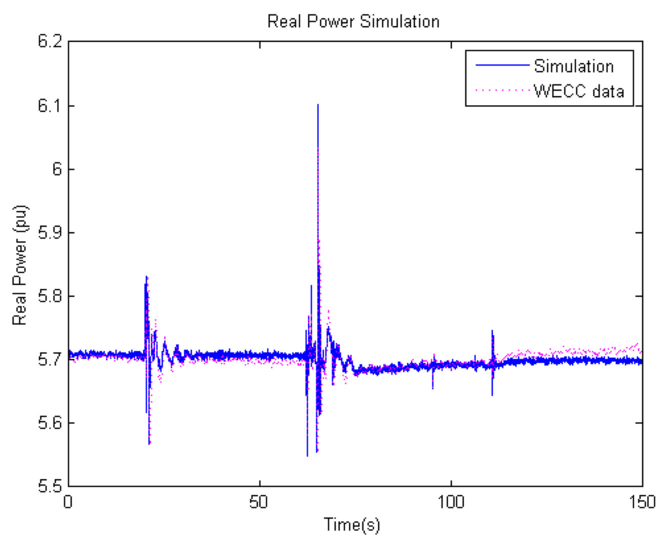

Fig.15 Real power comparison

By choosing time step h to be $\frac{0.05}{6}$, we can have the simulation results in the figures above. Although Figure 15 indicates the simulated voltage matches the WECC model, without changing any parameters in the model, the simulated real power (Figure 15) and reactive power (Figure 16) are still slightly different from the WECC model.

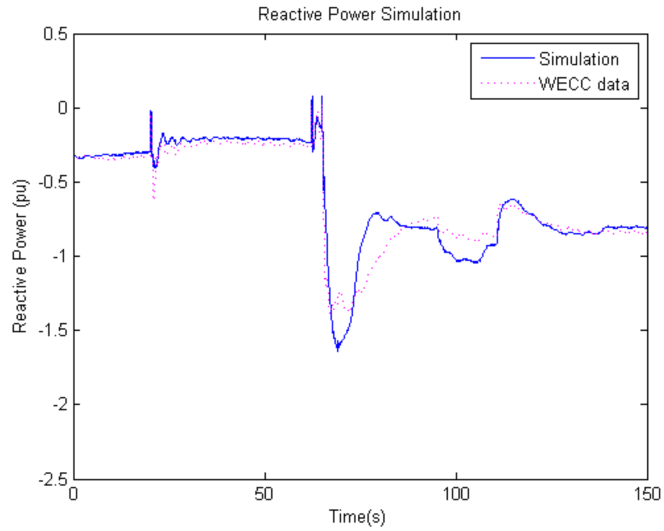

Fig.16 Reactive power comparison

In order to correct the simulation and make it become closer to the WECC model, the next step is to find the most sensible parameter/parameters in the model. These parameters include (H), ($T_{w1}, T_{w2}, T_{w3}, T_7$), ($K_{s1}$), ($K_{s2}$), ($T_1, T_2$), ($T_3, T_4$), and($T_8, T_9$)[6]. In addition, still maintain the time step h to be $\frac{0.05}{6}$.

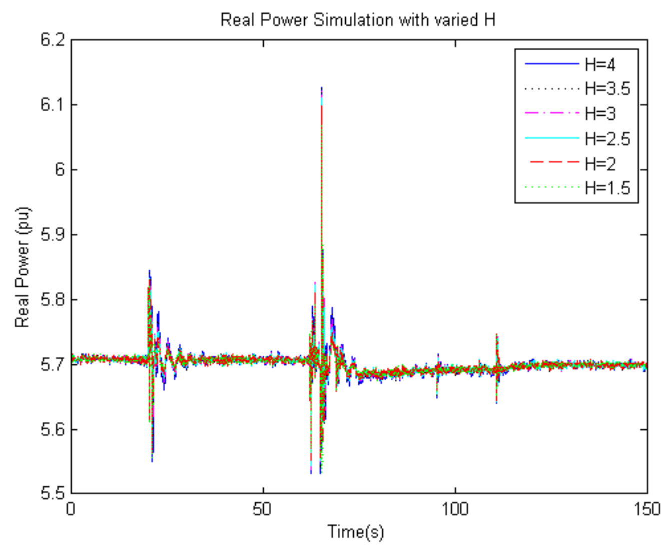

Fig.17 Real power without changing other parameters, vary H

In Figure 17 and Figure 18, with H varies from 1.5 to 4, the plots of P and Q change obviously. Since in the real power plot of the test run, some places of the WECC model have higher magnitudes than the simulation model, therefore, the comparatively higher H is desired. However, with the higher value of H, the larger distinctions appear in the reactive power plot in compare with the WECC model.

[6] Parameters inside each parenthesis are logically related, parameters in the same parenthesis will be analyzed simultaneously for the sensitivity analysis.

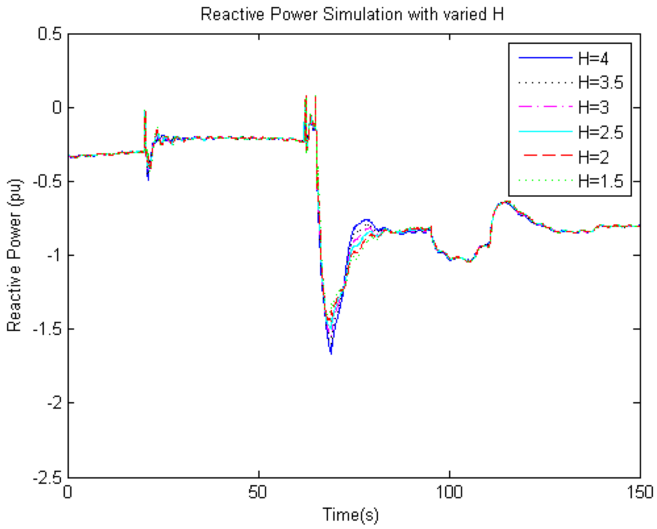

Fig.18 Reactive power without changing other parameters, vary H

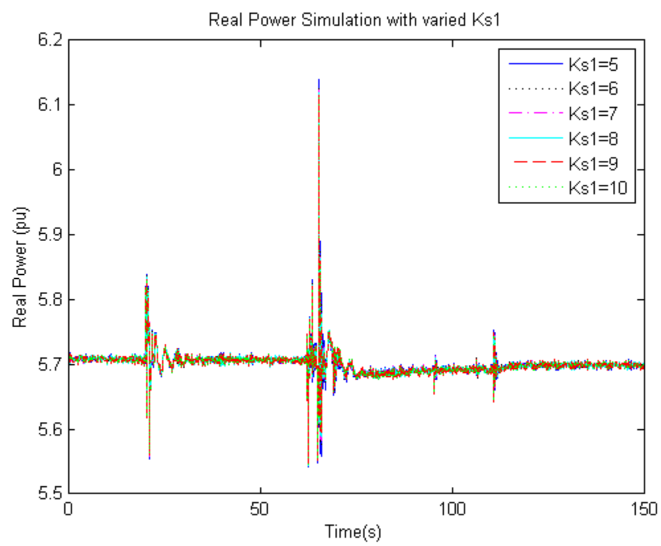

Fig.19 Real power without changing other parameters, vary K_{s1}

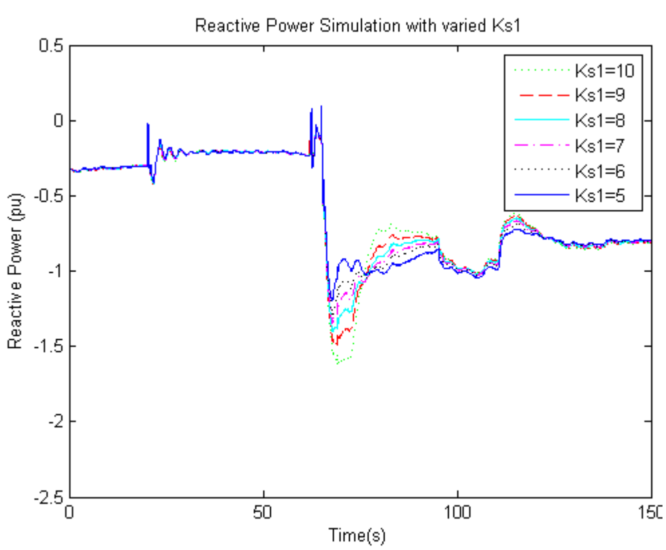

Fig.20 Reactive power without changing other parameters, vary K_{s1}

Based on Figure 19 and Figure 20, distinct differences exist in both real power plot and reactive power plot with changing the value of K_{s1} from 5 to 10. As Ks1 becoming smaller, for the reactive plot, the distinctions to the reactive power plot of the WECC model become smaller, and for the real power plot, the smaller disturbances will be reached, but undesired oscillations at some places in the real power plot turn to become more and more obvious.

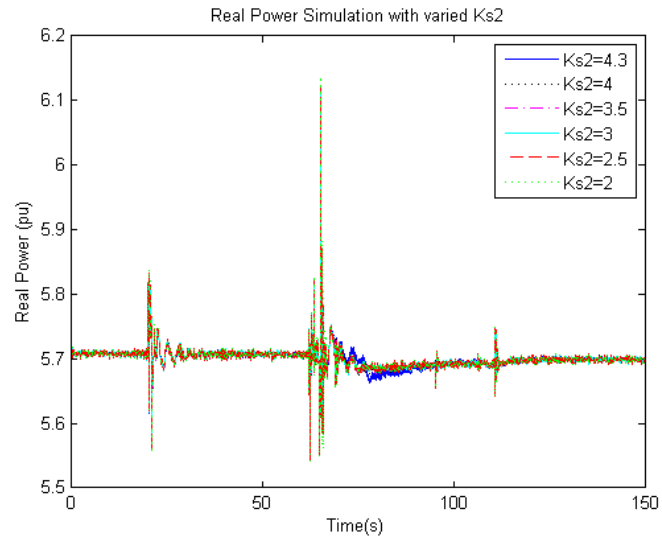

Fig.21 Real power without changing other parameters, vary K_{s2}

According to Figure 21 and Figure 22, similar to the effect of changing K_{s1}, with the value of K_{s2} becoming smaller, the better reactive power plot will be gained, but simultaneously, larger disturbance will also appear in the real power plot.

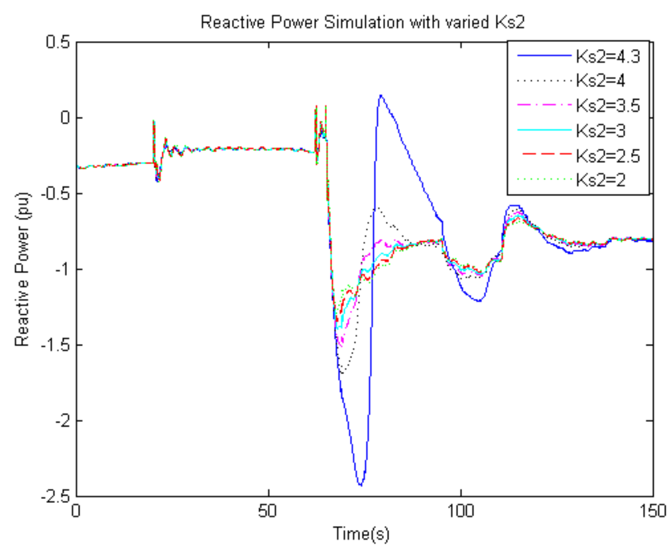

Fig.22 Reactive power without changing other parameters, vary K_{s2}

Although varying T_1 and T_2 largely, based on Figure 23 and Figure 24, both real power plot and reactive power plot do not have obvious changes. Same situation also occurs when changing other T values[7].

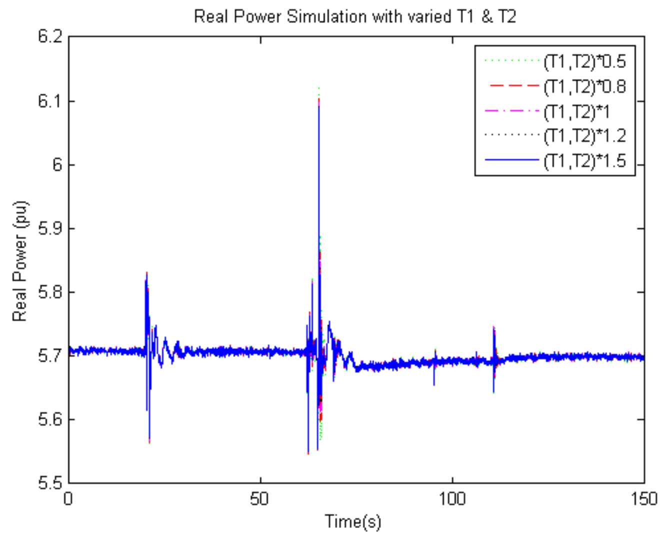

Fig.23 Real power without changing other parameters, vary T_1 and T_2

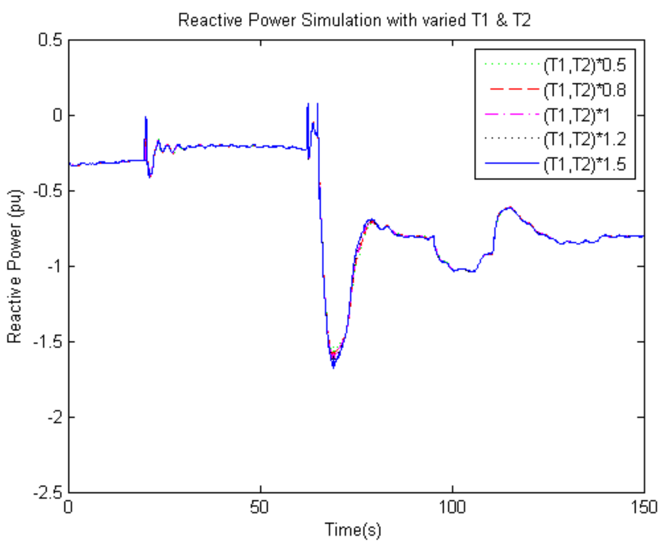

Fig.24 Reactive power without changing other parameters, vary T_1 and T_2

In general, the most sensible parameters are H, K_{s1} and K_{s2}, and based on the sensitivity analysis results, the trends to change these parameters are to increase the value of H and decrease the values of K_{s1} and K_{s2}.

C. Optimization result

In accordance with the sensitivity analysis results in the previous part, the optimal solutions of these three parameters are chosen to be $H = 3.6, K_{s1} = 7$ and $K_{s2} = 2.2$.

Next, apply these changes to re-simulate the model, and similar to the test run part, we also compare the simulation results with the WECC model data.

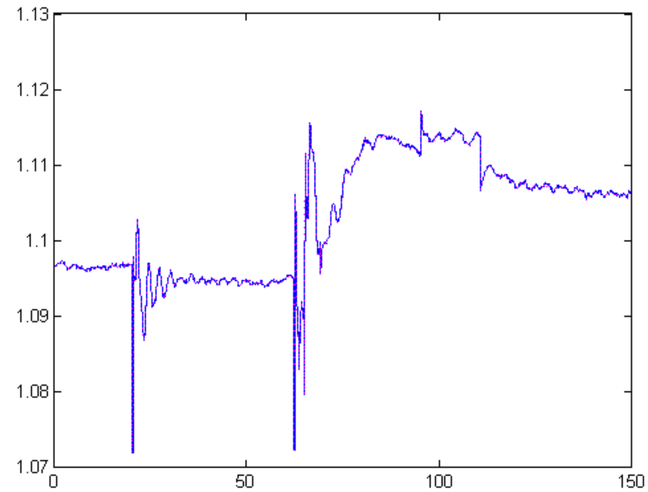

Fig.25 Voltage comparison

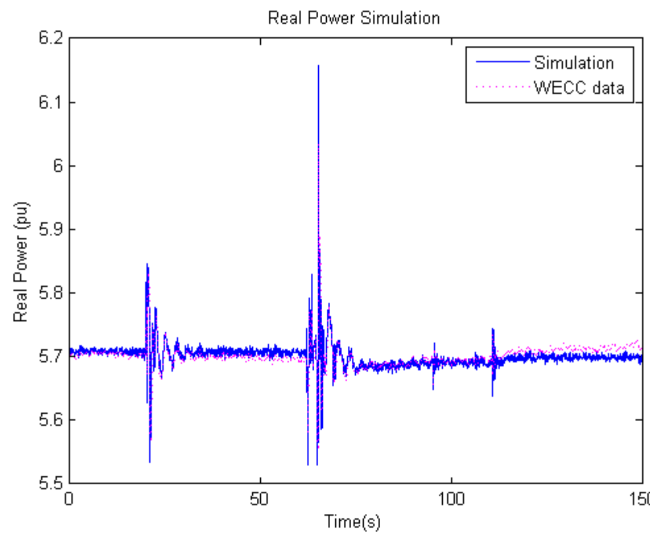

Fig.26 Real Power comparison

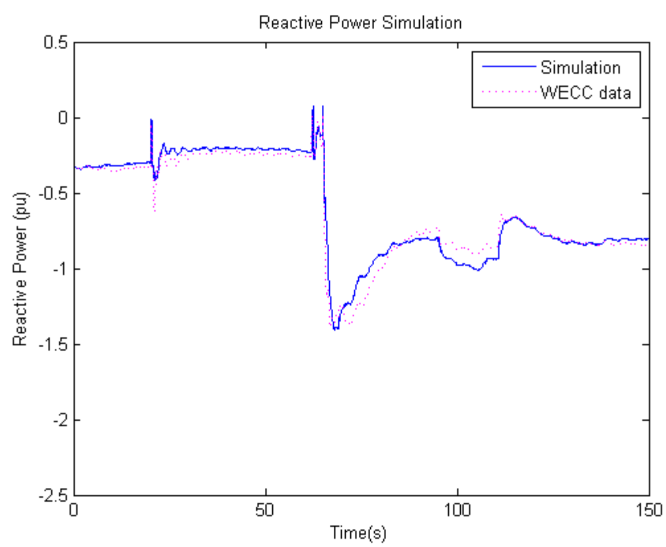

Fig.27 Reactive Power comparison

[7] Based on the sensitivity analysis, these T values are $T_{w1}, T_{w2}, T_{w3}, T_7, T_3, T_4, T_8, T_9$.

Based on the simulation with the renewed parameters in Figures 25-27, it is not hard to tell that the distinction has been improved.

VII. Conclusion

In conclusion, this project introduces to simulate dynamic models through Newton-Raphson Method and Trapezoidal Rule approximation. In addition, necessary sensitivity analyses are important to modify model parameters and to receive optimized simulations.

In this specific project, manually make $H = 3.6, K_{s1} = 7$ and $K_{s2} = 2.2$ will help to improve the simulation. For the future work, a programming approach for the sensitivity analysis can be applied by detecting and minimizing the real and reactive power differences between simulation and WECC data, using the feedback of the error to modify the model parameters.

VIII. Appendix

The appendix provides Matlab codes of WECC dynamic model simulation with modified parameters.

```
clear
clc

Z1 = 0.00032+0.01895i;
Z2 = 0.0002+0.00382i;
Z3 = 0.0003i;
Y = 0.35034i/2;
a = 1/0.92911;
P = 599.8/676;
Q = 5.74/676;
PL = 27.51/676;
QL = 14.39/676;
SLoad = (27.51+14.39i)/676;
ZLoad = conj(1.014^2/(SLoad)); % Network parameters
Ptable = xlsread('WECCdata.xlsx','D2:D4501')/100;
Qtable = xlsread('WECCdata.xlsx','E2:E4501')/100; %
WECC real power and reactive power data reading

ws = 2*pi*60;
Tdo1 = 4.893;
Tqo1 = 0.54;
Tdo2 = 0.0416;
Tqo2 = 0.067;
Xd = 2.2380;
Xq = 2.001;
Xd1 = 0.357;
Xq1 = 0.573;
Xd2 = 0.2930;
Xq2 = 0.2930;
H = 3.6;
M = H*2/ws;
D = 0;
Rs = 0;
Xls = 0.2490;
Kpr = 40;
Kir = 4;
Vrmax = 4;
Vrmin = -2.67;
Kpm = 20;
Kim = 2;
Vmmax = 4;
Vmmin = -2.67;
Kg = 1;
Kp = 1;
thetap = 0;
Ki = 0;
Kc = 0.062;
Xl = 0;
Vbmax = 1.2;
Tw1 = 15;
Tw2 = 15;
Tw3 = 15;
T7 = 15;
Ks2 = 2.2;
Ks3 = 1;
Ks4 = 1;
T8 = 0.3;
T9 = 0.15;
Ks1 = 7;
T1 = 0.11;
T2 = 0.02;
T3 = 0.11;
T4 = 0.03;
Vstmax = 0.1;
Vstmin = -0.05;
Re = 0.003732;
Xe = 0.134233;
TM = 0.8873;
A = 0.00011178;
B = 7.0114735;
Vref = 1.014; % Constant parameters of each model

h = 0.01*5/6;
Eq(1) = 0.7835;
Ed(1) = 0.6243;
psi1d(1) = 0.7016;
psi2q(1) = -0.7659;
delta(1) = 1.1563;
wr(1) = 2*pi*60;
Id(1) = 0.7581;
Iq(1) = 0.4372;
Vd(1) = 0.8748;
Vq(1) = 0.5128;
Vt(1) = 1.014;
Efd(1) = 2.2093;
Vr(1) = 2.2093;
Vr2(1) = 2.2093;
Vm(1) = 2.2677;
Vm2(1) = 2.2677;
s20(1) = 0;
Ifd(1) = 1.1108;
I1d(1) = 0;
In(1) = 0.0679;
Fex(1) = 0.9608;
Ve(1) = 1.014;
Vb(1) = 0.9743;
wr1(1) = 1;
s0(1) = 0;
s0_1(1) = 0;
s1(1) = 0;
s3(1) = 0;
s3_1(1) = 0.8873;
s5(1) = 0;
s6(1) = 0;
s6_1(1) = 0;
s8(1) = 0;
s9(1) = 0;
s10(1) = 0;
s10_1(1) = 0;
s11(1) = 0;
s12(1) = 0;
s13(1) = 0;
s14(1) = 0;
s14_1(1) = 0;
s15(1) = 0;
Vst(1) = 0;
Vst1(1) = 0;
S(1) = 0.8873;
psid2(1) = 0.7349;
psiq2(1) = -0.7467;
```

```
psi2(1) = 1.0477;
Sfd(1) = 0.1215;
S1q(1) = -0.1087; % Variables initializations

Veq_1 = xlsread('WECCdata.xlsx','B2:B4501')/500;
Ieq1 = Veq_1./Z3;
Zeq1 = 1/(1/Z3+Y);
Veq2 = Ieq1.*Zeq1;
Zeq2 = Zeq1+Z2;
Ieq2 = Veq2./Zeq2;
Zeq3 = 1/(1/Zeq2+Y);
Veq3 = Ieq2.*Zeq3;
Zeq4 = Zeq3+Z1;
Zeq5 = Zeq4.*6.76/a^2;
Veq4 = Veq3./a;
Ieq3 = Veq4./Zeq5;
Zth = 1/(1/Zeq5+1/ZLoad);
Vth = Ieq3.*Zth; % Network Thevenin equivalent
circuit conversion

Vs = abs(Vth);
we = xlsread('WECCdata.xlsx','C2:C4501')*2*pi;

V(1) = Vd(1)+1i*Vq(1);
I(1) = Id(1)+1i*Iq(1);
Vbus1(1) = V(1);
Ibus1(1) = V(1)/ZLoad-I(1);
Vbus2(1) = a*Vbus1(1)+Ibus1(1)*6.76*Z1/a;
Ibus2(1) = Ibus1(1)*6.76/a;
Vbus3(1) = (1+Y*Z2)*Vbus2(1)+Z2*Ibus2(1);
Ibus3(1) = (2*Y+Y^2*Z2)*Vbus2(1)+(1+Y*Z2)*Ibus2(1);
Vbus4(1) = Vbus3(1)+Z3*Ibus3(1);
Ibus4(1) = Ibus3(1);
Sinf(1) = Vbus4(1)*conj(Ibus4(1));
Pinf(1) = -real(Sinf(1));
Qinf(1) = -imag(Sinf(1)); % Calculating bus "1"
voltage, current and power

for i = 1:1:18000
    deltaV = ones(50,1);
    r = fix((i-1)/4+1);
    k = 1;
    CEq(1) = 0;
    CEd(1) = 0;
    Cpsi1d(1) = 0;
    Cpsi2q(1) = 0;
    Cdelta(1) = 0;
    Cwr(1) = 0;
    CId(1) = 0.1;
    CIq(1) = 0;
    CVd(1) = 0.1;
    CVq(1) = 0.1;
    CVt(1) = 0;
    CEfd(1) = 0;
    CVr(1) = 0;
    CVr2(1) = 0;
    CVm(1) = 0;
    CVm2(1) = 0;
    Cs20(1) = 0;
    CIfd(1) = 0;
    CI1d(1) = 0;
    CIn(1) = 0;
    CFex(1) = 0;
    CVe(1) = 0.1;
    CVb(1) = 0;

    Cwr1(1) = 0;
    Cs0(1) = 0;
    Cs0_1(1) = 0;
    Cs1(1) = 0;
    Cs3(1) = 0;
    Cs3_1(1) = 0;
    Cs5(1) = 0;
    Cs6(1) = 0;
    Cs6_1(1) = 0;
    Cs8(1) = 0;
    Cs9(1) = 0;
    Cs10(1) = 0;
    Cs10_1(1) = 0;
    Cs11(1) = 0;
    Cs12(1) = 0;
    Cs13(1) = 0;
    Cs14(1) = 0;
    Cs14_1(1) = 0;
    Cs15(1) = 0;
    CVst(1) = 0;
    CVst1(1) = 0;
    CS(1) = 0;

    Cpsid2(1) = 0;
    Cpsiq2(1) = 0.1;
    Cpsi2(1) = 0.1;
    CSfd(1) = 0;
    CS1q(1) = 0;

    while
sqrt(deltaV(1,:)^2+deltaV(2,:)^2+deltaV(3,:)^2+delta
V(4,:)^2+deltaV(5,:)^2+deltaV(6,:)^2+deltaV(7,:)^2+d
eltaV(8,:)^2+deltaV(9,:)^2+deltaV(10,:)^2+deltaV(11,
:)^2+deltaV(12,:)^2+deltaV(13,:)^2+deltaV(14,:)^2+de
ltaV(15,:)^2+deltaV(16,:)^2+deltaV(17,:)^2+deltaV(18
,:)^2+deltaV(19,:)^2+deltaV(20,:)^2+deltaV(21,:)^2+d
eltaV(22,:)^2+deltaV(23,:)^2+deltaV(24,:)^2+deltaV(2
5,:)^2+deltaV(26,:)^2+deltaV(27,:)^2+deltaV(28,:)^2+
deltaV(29,:)^2+deltaV(30,:)^2+deltaV(31,:)^2+deltaV(
32,:)^2+deltaV(33,:)^2+deltaV(34,:)^2+deltaV(35,:)^2
+deltaV(36,:)^2+deltaV(37,:)^2+deltaV(38,:)^2+deltaV
(39,:)^2+deltaV(40,:)^2+deltaV(41,:)^2+deltaV(42,:)^
2+deltaV(43,:)^2+deltaV(44,:)^2+deltaV(45,:)^2+delta
V(46,:)^2+deltaV(47,:)^2+deltaV(48,:)^2+deltaV(49,:)
^2+deltaV(50,:)^2)>0.001^2
        dF(1,1) = 1+h/2/Tdo1*(1+(Xd-Xd1)*(Xd1-
Xd2)/((Xd1-Xls)^2));
        dF(1,3) = -h/2/Tdo1*(Xd-Xd1)*(Xd1-
Xd2)/((Xd1-Xls)^2);
        dF(1,7) = h/2/Tdo1*(Xd-Xd1)*(Xd2-Xls)/(Xd1-
Xls);
        dF(1,12) = -h/(2*Tdo1);
        dF(1,46) = h/2/Tdo1;
        dF(2,2) = 1+h/2/Tqo1*(1+(Xq-Xq1)*(Xq1-
Xq2)/((Xq1-Xls)^2));
        dF(2,4) = h/2/Tqo1*(Xq-Xq1)*(Xq1-Xq2)/((Xq1-
Xls)^2);
        dF(2,8) = -h/2/Tqo1*(Xq-Xq1)*(Xq2-Xls)/(Xq1-
Xls);
        dF(2,47) = -h/2/Tqo1;
        dF(3,1) = -h/2/Tdo2;
        dF(3,3) = 1+h/2/Tdo2;
        dF(3,7) = h*(Xd1-Xls)/2/Tdo2;
        dF(4,2) = h/2/Tqo2;
        dF(4,4) = 1+h/2/Tqo2;
        dF(4,8) = h*(Xq1-Xls)/2/Tqo2;
        dF(5,5) = 1;
        dF(5,6) = -h/2;
        dF(6,1) = h*(Xd2-Xls)*CIq(k)/2/M/(Xd1-Xls);
        dF(6,2) = h*(Xq2-Xls)*CId(k)/2/M/(Xq1-Xls);
        dF(6,3) = h*(Xd1-Xd2)*CIq(k)/2/M/(Xd1-Xls);
        dF(6,4) = -h*(Xq1-Xq2)*CId(k)/2/M/(Xq1-Xls);
        dF(6,6) = 1+h*D/2/M;
        dF(6,7) = -h/2/M*((Xd2-Xq2)*CIq(k)-(Xq2-
Xls)/(Xq1-Xls)*CEd(k)+(Xq1-Xq2)/(Xq1-Xls)*Cpsi2q(k));
        dF(6,8) = -h/2/M*((Xd2-Xq2)*CId(k)-(Xd2-
Xls)/(Xd1-Xls)*CEq(k)-(Xd1-Xd2)/(Xd1-Xls)*Cpsi1d(k));
        dF(7,2) = -Cwr(k)/ws*(Xq2-Xls)/(Xq1-Xls);
        dF(7,4) = Cwr(k)/ws*(Xq1-Xq2)/(Xq1-Xls);
        dF(7,5) = Vs(r)*cos(Cdelta(k));
        dF(7,6) = -1/ws*(Xq2+Xe)*CIq(k)-(Xq2-
Xls)/ws/(Xq1-Xls)*CEd(k)+(Xq1-Xq2)/ws/(Xq1-
Xls)*Cpsi2q(k);
        dF(7,7) = 3*(Rs+Re);
        dF(7,8) = -Cwr(k)/ws*(Xq2+Xe);
        dF(8,1) = -Cwr(k)/ws*(Xd2-Xls)/(Xd1-Xls);
```

```
        dF(8,3) = -Cwr(k)/ws*(Xd1-Xd2)/(Xd1-Xls);
        dF(8,5) = -Vs(r)*sin(Cdelta(k));
        dF(8,6) = 1/ws*(Xd2+Xe)*CId(k)-(Xd2-
Xls)/ws/(Xd1-Xls)*CEq(k)-(Xd1-Xd2)/ws/(Xd1-
Xls)*Cpsi1d(k);
        dF(8,7) = Cwr(k)/ws*(Xd2+Xe);
        dF(8,8) = 3*(Rs+Re);
        dF(9,5) = Vs(r)*cos(Cdelta(k));
        dF(9,6) = -Xe*CIq(k)/ws;
        dF(9,7) = Re;
        dF(9,8) = -Cwr(k)/ws*Xe;
        dF(9,9) = -1;
        dF(10,5) = -Vs(r)*sin(Cdelta(k));
        dF(10,6) = Xe*CId(k)/ws;
        dF(10,7) = Cwr(k)/ws*Xe;
        dF(10,8) = Re;
        dF(10,10) = -1;
        dF(11,9) = CVd(k)*((CVd(k))^2+(CVq(k))^2)^(-
1/2);
        dF(11,10) =
CVq(k)*((CVd(k))^2+(CVq(k))^2)^(-1/2);
        dF(11,11) = -1;
        dF(12,14) = 1+h*Kir/2/Kpr;
        dF(12,13) = -h*Kir/2/Kpr;
        dF(13,16) = 1+h*Kim/2/Kpm;
        dF(13,15) = -h*Kim/2/Kpm;
        dF(14,11) = -Kpr;
        dF(14,13) = -1;
        dF(14,14) = 1;
        dF(14,43) = Kpr;
        dF(15,15) = -1;
        dF(15,16) = 1;
        dF(15,17) = Kpm;
        dF(16,12) = -Kg;
        dF(16,13) = 1;
        dF(16,17) = -1;
        dF(17,7) = ((Kp*CVd(k)*cos(thetap)-
Kp*CVq(k)*sin(thetap)-Kp*CId(k)*Xl*sin(thetap)-
Kp*CIq(k)*Xl*cos(thetap)-
Ki*CIq(k))^2+(Kp*CVq(k)*cos(thetap)+Kp*CVd(k)*sin(th
etap)+Ki*CId(k)+Kp*CId(k)*Xl*cos(thetap)-
Kp*CIq(k)*Xl*sin(thetap))^2)^(-1/2)*(-
(Kp*CVd(k)*cos(thetap)-Kp*CVq(k)*sin(thetap)-
Kp*CId(k)*Xl*sin(thetap)-Kp*CIq(k)*Xl*cos(thetap)-
Ki*CIq(k))*Kp*Xl*sin(thetap)+(Kp*CVq(k)*cos(thetap)+
Kp*CVd(k)*sin(thetap)+Ki*CId(k)+Kp*CId(k)*Xl*cos(the
tap)-
Kp*CIq(k)*Xl*sin(thetap))*(Ki+Kp*Xl*cos(thetap)));
        dF(17,8) = ((Kp*CVd(k)*cos(thetap)-
Kp*CVq(k)*sin(thetap)-Kp*CId(k)*Xl*sin(thetap)-
Kp*CIq(k)*Xl*cos(thetap)-
Ki*CIq(k))^2+(Kp*CVq(k)*cos(thetap)+Kp*CVd(k)*sin(th
etap)+Ki*CId(k)+Kp*CId(k)*Xl*cos(thetap)-
Kp*CIq(k)*Xl*sin(thetap))^2)^(-1/2)*(-
(Kp*CVd(k)*cos(thetap)-Kp*CVq(k)*sin(thetap)-
Kp*CId(k)*Xl*sin(thetap)-Kp*CIq(k)*Xl*cos(thetap)-
Ki*CIq(k))*(Ki+Kp*Xl*cos(thetap))-
(Kp*CVq(k)*cos(thetap)+Kp*CVd(k)*sin(thetap)+Ki*CId(
k)+Kp*CId(k)*Xl*cos(thetap)-
Kp*CIq(k)*Xl*sin(thetap))*Kp*Xl*sin(thetap));
        dF(17,9) = ((Kp*CVd(k)*cos(thetap)-
Kp*CVq(k)*sin(thetap)-Kp*CId(k)*Xl*sin(thetap)-
Kp*CIq(k)*Xl*cos(thetap)-
Ki*CIq(k))^2+(Kp*CVq(k)*cos(thetap)+Kp*CVd(k)*sin(th
etap)+Ki*CId(k)+Kp*CId(k)*Xl*cos(thetap)-
Kp*CIq(k)*Xl*sin(thetap))^2)^(-
1/2)*((Kp*CVd(k)*cos(thetap)-Kp*CVq(k)*sin(thetap)-
Kp*CId(k)*Xl*sin(thetap)-Kp*CIq(k)*Xl*cos(thetap)-
Ki*CIq(k))*Kp*cos(thetap)+(Kp*CVq(k)*cos(thetap)+Kp*
CVd(k)*sin(thetap)+Ki*CId(k)+Kp*CId(k)*Xl*cos(thetap
)-Kp*CIq(k)*Xl*sin(thetap))*Kp*sin(thetap));
        dF(17,10) = ((Kp*CVd(k)*cos(thetap)-
Kp*CVq(k)*sin(thetap)-Kp*CId(k)*Xl*sin(thetap)-
Kp*CIq(k)*Xl*cos(thetap)-
Ki*CIq(k))^2+(Kp*CVq(k)*cos(thetap)+Kp*CVd(k)*sin(th
etap)+Ki*CId(k)+Kp*CId(k)*Xl*cos(thetap)-
Kp*CIq(k)*Xl*sin(thetap))^2)^(-1/2)*(-
(Kp*CVd(k)*cos(thetap)-Kp*CVq(k)*sin(thetap)-
Kp*CId(k)*Xl*sin(thetap)-Kp*CIq(k)*Xl*cos(thetap)-
Ki*CIq(k))*Kp*sin(thetap)+(Kp*CVq(k)*cos(thetap)+Kp*
CVd(k)*sin(thetap)+Ki*CId(k)+Kp*CId(k)*Xl*cos(thetap
)-Kp*CIq(k)*Xl*sin(thetap))*Kp*cos(thetap));
        dF(17,22) = -1;
        dF(18,1) = 1/(Xd-Xls);
        dF(18,18) = -1;
        dF(18,19) = -(Xd-Xd1)/(Xd-Xls);
        dF(18,7) = (Xd-Xd1)/(Xd-Xls);
        dF(19,1) = -(Xd1-Xd2)/(Xd1-Xls)^2;
        dF(19,3) = (Xd1-Xd2)/(Xd1-Xls)^2;
        dF(19,19) = -1;
        dF(19,7) = (Xd1-Xd2)/(Xd1-Xls);
        dF(20,22) = -Kc*CIfd(k)/(CVe(k))^2;
        dF(20,18) = Kc/(CVe(k));
        dF(20,20) = -1;
        if CIn(k)<=1
            if CIn(k)<=0.75
                if CIn(k)<=0.433
                    dF(21,20) = -0.577;
                    dF(21,21) = -1;
                else
                dF(21,20) = -(0.75-(CIn(k))^2)^(-
1/2)*CIn(k);
                dF(21,21) = -1;
                end
            else
            dF(21,20) = -1.732;
            dF(21,21) = -1;
            end
        else
            dF(21,20) = 0;
            dF(21,21) = -1;
        end
        dF(22,22) = CFex(k);
        dF(22,21) = CVe(k);
        dF(22,23) = -1;
        dF(23,12) = -1;
        dF(23,15) = CVb(k);
        dF(23,23) = CVm(k);
        dF(24,24) = 1;
        dF(24,25) = -h/2/Tw1;
        dF(25,26) = 1;
        dF(25,27) = -h/2/Tw2;
        dF(26,28) = -h/2/Tw3;
        dF(26,29) = 1;
        dF(27,28) = -h*Ks2/2/T7;
        dF(27,30) = 1+h/2/T7;
        dF(28,27) = -h/2/T9;
        dF(28,30) = -h*Ks3/2/T9;
        dF(28,33) = 1+h/2/T9;
        dF(29,33) = -h/2/T9;
        dF(29,34) = 1+h/2/T9;
        dF(30,34) = -h/2/T9;
        dF(30,35) = 1+h/2/T9;
        dF(31,37) = -h/2/T8;
        dF(31,36) = 1+h/2/T8;
        dF(32,37) = -h/2/T9;
        dF(32,38) = 1+h/2/T9;
        dF(33,38) = -h/2/T9;
        dF(33,39) = 1+h/2/T9;
        dF(34,39) = -h/2/T9;
        dF(34,40) = 1+h/2/T9;
        dF(35,42) = -h/2/T8;
        dF(35,41) = 1+h/2/T8;
        dF(36,31) = -h/2/T1;
        dF(36,32) = 1+h/2/T1;
        dF(37,43) = -h/2/T3;
        dF(37,44) = 1+h/2/T3;
        dF(38,6) = 1/ws;
        dF(38,24) = -1;
        dF(38,25) = -1;
        dF(39,25) = 1;
        dF(39,26) = -1;
```

```
        dF(39,27) = -1;
        dF(40,28) = -1;
        dF(40,29) = -1;
        dF(40,45) = 1;
        dF(41,35) = T8/T9;
        dF(41,36) = 1-T8/T9;
        dF(41,37) = -1;
        dF(42,40) = T8/T9;
        dF(42,41) = 1-T8/T9;
        dF(42,42) = -1;
        dF(43,30) = -T1/T2*Ks1*Ks4;
        dF(43,31) = -1;
        dF(43,32) = 1-T1/T2;
        dF(43,42) = T1/T2*Ks1;
        dF(44,31) = T3/T4;
        dF(44,43) = -1;
        dF(44,44) = 1-T3/T4;
        dF(45,45) = -1;
        dF(45,9) =
(((CVd(k))^2+(CVq(k))^2)*((CId(k))^2+(CIq(k))^2))^(-
1/2)*((CId(k))^2+(CIq(k))^2)*CVd(k);
        dF(45,10) =
(((CVd(k))^2+(CVq(k))^2)*((CId(k))^2+(CIq(k))^2))^(-
1/2)*((CId(k))^2+(CIq(k))^2)*CVq(k);
        dF(45,7) =
(((CVd(k))^2+(CVq(k))^2)*((CId(k))^2+(CIq(k))^2))^(-
1/2)*((CVd(k))^2+(CVq(k))^2)*CId(k);
        dF(45,8) =
(((CVd(k))^2+(CVq(k))^2)*((CId(k))^2+(CIq(k))^2))^(-
1/2)*((CVd(k))^2+(CVq(k))^2)*CIq(k);
        dF(46,1) = (Xd2-Xls)/(Xd1-Xls);
        dF(46,3) = (Xd1-Xd2)/(Xd1-Xls);
        dF(46,46) = -1;
        dF(47,2) = -(Xq2-Xls)/(Xq1-Xls);
        dF(47,4) = (Xq1-Xq2)/(Xq1-Xls);
        dF(47,47) = -1;
        dF(48,46) =
Cpsid2(k)*((Cpsid2(k))^2+(Cpsiq2(k))^2)^(-1/2);
        dF(48,46) =
Cpsiq2(k)*((Cpsid2(k))^2+(Cpsiq2(k))^2)^(-1/2);
        dF(48,48) = -1;
        dF(49,46) = A*exp(B*Cpsi2(k))/(Cpsi2(k));
        dF(49,48) =
A*Cpsid2(k)*exp(B*Cpsi2(k))*(B*Cpsi2(k)-
1)/(Cpsi2(k))^2;
        dF(49,49) = -1;
        dF(50,47) = (Xq-Xls)/(Xd-
Xls)*A*exp(B*Cpsi2(k))/(Cpsi2(k));
        dF(50,48) = (Xq-Xls)/(Xd-
Xls)*Cpsiq2(k)*A*exp(B*Cpsi2(k))*(B*Cpsi2(k)-
1)/(Cpsi2(k))^2;
        dF(50,50) = -1;

        F(1,:) = CEq(k)-Eq(i)-h/(2*Tdo1)*(-CEq(k)-
(Xd-Xd1)*((Xd2-Xls)/(Xd1-Xls)*CId(k)-(Xd1-
Xd2)/((Xd1-Xls)^2)*Cpsi1d(k)+(Xd1-Xd2)/((Xd1-
Xls)^2)*CEq(k))+CEfd(k)-Eq(i)-(Xd-Xd1)*((Xd2-
Xls)/(Xd1-Xls)*Id(i)-(Xd1-Xd2)/((Xd1-
Xls)^2)*psi1d(i)+(Xd1-Xd2)/((Xd1-
Xls)^2)*Eq(i))+Efd(i));
        F(2,:) = CEd(k)-Ed(i)-h/(2*Tqo1)*(-
CEd(k)+(Xq-Xq1)*((Xq2-Xls)/(Xq1-Xls)*CIq(k)-(Xq1-
Xq2)/((Xq1-Xls)^2)*Cpsi2q(k)-(Xq1-Xq2)/((Xq1-
Xls)^2)*CEd(k))-Ed(i)+(Xq-Xq1)*((Xq2-Xls)/(Xq1-
Xls)*Iq(i)-(Xq1-Xq2)/((Xq1-Xls)^2)*psi2q(i)-(Xq1-
Xq2)/((Xq1-Xls)^2)*Ed(i)));
        F(3,:) = Cpsi1d(k)-psi1d(i)-h/2/Tdo2*(-
Cpsi1d(k)+CEq(k)-(Xd1-Xls)*CId(k)-psi1d(i)+Eq(i)-
(Xd1-Xls)*Id(i));
        F(4,:) = Cpsi2q(k)-psi2q(i)-h/2/Tqo2*(-
Cpsi2q(k)-CEd(k)-(Xq1-Xls)*CIq(k)-psi2q(i)-Ed(i)-
(Xq1-Xls)*Iq(i));
        F(5,:) = Cdelta(k)-delta(i)-h/2*(Cwr(k)-
we(r)+wr(i)-we(r));
        F(6,:) = Cwr(k)-wr(i)-h/2/M*(TM+(Xd2-
Xq2)*CId(k)*CIq(k)-(Xd2-Xls)/(Xd1-
Xls)*CEq(k)*CIq(k)-(Xq2-Xls)/(Xq1-
Xls)*CEd(k)*CId(k)-(Xd1-Xd2)/(Xd1-
Xls)*Cpsi1d(k)*CIq(k)+(Xq1-Xq2)/(Xq1-
Xls)*Cpsi2q(k)*CId(k)-D*(Cwr(k)-we(r))+TM+(Xd2-
Xq2)*Id(i)*Iq(i)-(Xd2-Xls)/(Xd1-Xls)*Eq(i)*Iq(i)-
(Xq2-Xls)/(Xq1-Xls)*Ed(i)*Id(i)-(Xd1-Xd2)/(Xd1-
Xls)*psi1d(i)*Iq(i)+(Xq1-Xq2)/(Xq1-
Xls)*psi2q(i)*Id(i)-D*(wr(i)-we(r)));
        F(7,:) = (Rs+Re)*CId(k)-
Cwr(k)/ws*(Xq2+Xe)*CIq(k)-Cwr(k)/ws*(Xq2-Xls)/(Xq1-
Xls)*CEd(k)+Cwr(k)/ws*(Xq1-Xq2)/(Xq1-
Xls)*Cpsi2q(k)+Vs(r)*sin(Cdelta(k));
        F(8,:) =
(Rs+Re)*CIq(k)+Cwr(k)/ws*(Xd2+Xe)*CId(k)-
Cwr(k)/ws*(Xd2-Xls)/(Xd1-Xls)*CEq(k)-Cwr(k)/ws*(Xd1-
Xd2)/(Xd1-Xls)*Cpsi1d(k)+Vs(r)*cos(Cdelta(k));
        F(9,:) = Re*CId(k)-
Cwr(k)/ws*Xe*CIq(k)+Vs(r)*sin(Cdelta(k))-CVd(k);
        F(10,:) =
Re*CIq(k)+Cwr(k)/ws*Xe*CId(k)+Vs(r)*cos(Cdelta(k))-
CVq(k);
        F(11,:) = ((CVd(k))^2+(CVq(k))^2)^(1/2)-
CVt(k);
        F(12,:) = CVr2(k)-Vr2(i)-
h*Kir/2/Kpr*(CVr(k)-CVr2(k)+Vr(i)-Vr2(i));
        F(13,:) = CVm2(k)-Vm2(i)-
h*Kim/2/Kpm*(CVm(k)-CVm2(k)+Vm(i)-Vm2(i));
        F(14,:) = Kpr*(Vref-CVt(k)+CVst(k))+CVr2(k)-
CVr(k);
        F(15,:) = Kpm*Cs20(k)+CVm2(k)-CVm(k);
        F(16,:) = CVr(k)-Kg*CEfd(k)-Cs20(k);
        F(17,:) = ((Kp*CVd(k)*cos(thetap)-
Kp*CVq(k)*sin(thetap)-Kp*CId(k)*Xl*sin(thetap)-
Kp*CIq(k)*Xl*cos(thetap)-
Ki*CIq(k))^2+(Kp*CVq(k)*cos(thetap)+Kp*CVd(k)*sin(th
etap)+Ki*CId(k)+Kp*CId(k)*Xl*cos(thetap)-
Kp*CIq(k)*Xl*sin(thetap))^2)^(1/2)-CVe(k);
        F(18,:) = (1/(Xd-Xls))*(CEq(k)+(Xd-
Xd1)*(CId(k)-CI1d(k)))-CIfd(k);
        F(19,:) = (Xd1-Xd2)/(Xd1-
Xls)^2*(Cpsi1d(k)+(Xd1-Xls)*CId(k)-CEq(k))-CI1d(k);
        F(20,:) = Kc*CIfd(k)/CVe(k)-CIn(k);
        if CIn(k)<=1
            if CIn(k)<=0.75
                if CIn(k)<=0.433
                    F(21,:) = 1-0.577*CIn(k)-CFex(k);
                else
                F(21,:) = (0.75-(CIn(k))^2)^(1/2)-
CFex(k);
                end
            else
            F(21,:) = 1.732*(1-CIn(k))-CFex(k);
            end
        else
            F(21,:) = -CFex(k);
        end
        F(22,:) = CVe(k)*CFex(k)-CVb(k);
        F(23,:) = CVb(k)*CVm(k)-CEfd(k);
        F(24,:) = Cwr1(k)-wr1(i)-
h/2/Tw1*(Cs0(k)+s0(i));
        F(25,:) = Cs0_1(k)-s0_1(i)-
h/2/Tw2*(Cs1(k)+s1(i));
        F(26,:) = Cs3_1(k)-s3_1(i)-
h/2/Tw3*(Cs3(k)+s3(i));
        F(27,:) = Cs5(k)-s5(i)-h/2/T7*(Ks2*Cs3(k)-
Cs5(k)+Ks2*s3(i)-s5(i));
        F(28,:) = Cs8(k)-s8(i)-
h/2/T9*(Cs1(k)+Ks3*Cs5(k)-Cs8(k)+s1(i)+Ks3*s5(i)-
s8(i));
        F(29,:) = Cs9(k)-s9(i)-h/2/T9*(Cs8(k)-
Cs9(k)+s8(i)-s9(i));
        F(30,:) = Cs10(k)-s10(i)-h/2/T9*(Cs9(k)-
Cs10(k)+s9(i)-s10(i));
        F(31,:) = Cs10_1(k)-s10_1(i)-
h/2/T8*(Cs11(k)-Cs10_1(k)+s11(i)-s10_1(i));
```

```
        F(32,:) = Cs12(k)-s12(i)-h/2/T9*(Cs11(k)-
Cs12(k)+s11(i)-s12(i));
        F(33,:) = Cs13(k)-s13(i)-h/2/T9*(Cs12(k)-
Cs13(k)+s12(i)-s13(i));
        F(34,:) = Cs14(k)-s14(i)-h/2/T9*(Cs13(k)-
Cs14(k)+s13(i)-s14(i));
        F(35,:) = Cs14_1(k)-s14_1(i)-
h/2/T8*(Cs15(k)-Cs14_1(k)+s15(i)-s14_1(i));
        F(36,:) = Cs6_1(k)-s6_1(i)-h/2/T1*(Cs6(k)-
Cs6_1(k)+s6(i)-s6_1(i));
        F(37,:) = CVst1(k)-Vst1(i)-h/2/T3*(CVst(k)-
CVst1(k)+Vst(i)-Vst1(i));
        F(38,:) = Cwr(k)/ws-Cwr1(k)-Cs0(k);
        F(39,:) = Cs0(k)-Cs0_1(k)-Cs1(k);
        F(40,:) = CS(k)-Cs3_1(k)-Cs3(k);
        F(41,:) = T8/T9*Cs10(k)+(1-T8/T9)*Cs10_1(k)-
Cs11(k);
        F(42,:) = T8/T9*Cs14(k)+(1-T8/T9)*Cs14_1(k)-
Cs15(k);
        F(43,:) = T1/T2*Ks1*Cs15(k)-
T1/T2*Ks1*Ks4*Cs5(k)+(1-T1/T2)*Cs6_1(k)-Cs6(k);
        F(44,:) = T3/T4*Cs6(k)+(1-T3/T4)*CVst1(k)-
CVst(k);
        F(45,:) =
((CVd(k)*CId(k)+CVq(k)*CIq(k))^2+(CVq(k)*CId(k)-
CVd(k)*CIq(k))^2)^(1/2)-CS(k);
        F(46,:) = (Xd2-Xls)/(Xd1-Xls)*CEq(k)+(Xd1-
Xd2)/(Xd1-Xls)*Cpsi1d(k)-Cpsid2(k);
        F(47,:) = -(Xq2-Xls)/(Xq1-Xls)*CEd(k)+(Xq1-
Xq2)/(Xq1-Xls)*Cpsi2q(k)-Cpsiq2(k);
        F(48,:) =
((Cpsid2(k))^2+(Cpsiq2(k))^2)^(1/2)-Cpsi2(k);
        F(49,:) =
Cpsid2(k)/(Cpsi2(k))*(A*exp(B*Cpsi2(k)))-CSfd(k);
        F(50,:) = Cpsiq2(k)/(Cpsi2(k))*(Xq-Xls)/(Xd-
Xls)*(A*exp(B*Cpsi2(k)))-CS1q(k);

        deltaV = dF\(-F);

        CEq(k+1) = deltaV(1,:)+CEq(k);
        CEd(k+1) = deltaV(2,:)+CEd(k);
        Cpsi1d(k+1) = deltaV(3,:)+Cpsi1d(k);
        Cpsi2q(k+1) = deltaV(4,:)+Cpsi2q(k);
        Cdelta(k+1) = deltaV(5,:)+Cdelta(k);
        Cwr(k+1) = deltaV(6,:)+Cwr(k);
        CId(k+1) = deltaV(7,:)+CId(k);
        CIq(k+1) = deltaV(8,:)+CIq(k);
        CVd(k+1) = deltaV(9,:)+CVd(k);
        CVq(k+1) = deltaV(10,:)+CVq(k);
        CVt(k+1) = deltaV(11,:)+CVt(k);
        CEfd(k+1) = deltaV(12,:)+CEfd(k);
        CVr(k+1) = deltaV(13,:)+CVr(k);
        CVr2(k+1) = deltaV(14,:)+CVr2(k);
        CVm(k+1) = deltaV(15,:)+CVm(k);
        CVm2(k+1) = deltaV(16,:)+CVm2(k);
        Cs20(k+1) = deltaV(17,:)+Cs20(k);
        CIfd(k+1) = deltaV(18,:)+CIfd(k);
        CI1d(k+1) = deltaV(19,:)+CI1d(k);
        CIn(k+1) = deltaV(20,:)+CIn(k);
        CFex(k+1) = deltaV(21,:)+CFex(k);
        CVe(k+1) = deltaV(22,:)+CVe(k);
        CVb(k+1) = deltaV(23,:)+CVb(k);
        Cwr1(k+1) = deltaV(24,:)+Cwr1(k);
        Cs0(k+1) = deltaV(25,:)+Cs0(k);
        Cs0_1(k+1) = deltaV(26,:)+Cs0_1(k);
        Cs1(k+1) = deltaV(27,:)+Cs1(k);
        Cs3(k+1) = deltaV(28,:)+Cs3(k);
        Cs3_1(k+1) = deltaV(29,:)+Cs3_1(k);
        Cs5(k+1) = deltaV(30,:)+Cs5(k);
        Cs6(k+1) = deltaV(31,:)+Cs6(k);
        Cs6_1(k+1) = deltaV(32,:)+Cs6_1(k);
        Cs8(k+1) = deltaV(33,:)+Cs8(k);
        Cs9(k+1) = deltaV(34,:)+Cs9(k);
        Cs10(k+1) = deltaV(35,:)+Cs10(k);
        Cs10_1(k+1) = deltaV(36,:)+ Cs10_1(k);
        Cs11(k+1) = deltaV(37,:)+Cs11(k);
        Cs12(k+1) = deltaV(38,:)+Cs12(k);
        Cs13(k+1) = deltaV(39,:)+Cs13(k);
        Cs14(k+1) = deltaV(40,:)+Cs14(k);
        Cs14_1(k+1) = deltaV(41,:)+Cs14_1(k);
        Cs15(k+1) = deltaV(42,:)+Cs15(k);
        CVst(k+1) = deltaV(43,:)+CVst(k);
        CVst1(k+1) = deltaV(44,:)+CVst1(k);
        CS(k+1) = deltaV(45,:)+CS(k);
        CSfd(k+1) = deltaV(49,:)+CSfd(k);
        CS1q(k+1) = deltaV(50,:)+CS1q(k);
        Cpsi2(k+1) = deltaV(48,:)+Cpsi2(k);
        Cpsid2(k+1) = deltaV(46,:)+Cpsid2(k);
        Cpsiq2(k+1) = deltaV(47,:)+Cpsiq2(k);
        k = k+1;
    end
    Eq(i+1) = CEq(k);
    Ed(i+1) = CEd(k);
    psi1d(i+1) = Cpsi1d(k);
    psi2q(i+1) = Cpsi2q(k);
    delta(i+1) = Cdelta(k);
    wr(i+1) = Cwr(k);
    Id(i+1) = CId(k);
    Iq(i+1) = CIq(k);
    Vd(i+1) = CVd(k);
    Vq(i+1) = CVq(k);
    Vt(i+1) = CVt(k);
    Vr2(i+1) = CVr2(k);
    Vm2(i+1) = CVm2(k);
    Vr(i+1) = CVr(k);
    Efd(i+1) = CEfd(k);
    s20(i+1) = Cs20(k);
    Vm(i+1) = CVm(k);
    Ve(i+1) = CVe(k);
    Ifd(i+1) = CIfd(k);
    I1d(i+1) = CI1d(k);
    In(i+1) = CIn(k);
    Fex(i+1) = CFex(k);
    Vb(i+1) = CVb(k);
    wr1(i+1) = Cwr1(k);
    s0(i+1) = Cs0(k);
    s0_1(i+1) = Cs0_1(k);
    s1(i+1) = Cs1(k);
    s3(i+1) = Cs3(k);
    s3_1(i+1) = Cs3_1(k);
    s5(i+1) = Cs5(k);
    s6(i+1) = Cs6(k);
    s6_1(i+1) = Cs6_1(k);
    s8(i+1) = Cs8(k);
    s9(i+1) = Cs9(k);
    s10(i+1) = Cs10(k);
    s10_1(i+1) = Cs10_1(k);
    s11(i+1) = Cs11(k);
    s12(i+1) = Cs12(k);
    s13(i+1) = Cs13(k);
    s14(i+1) = Cs14(k);
    s14_1(i+1) = Cs14_1(k);
    s15(i+1) = Cs15(k);
    Vst(i+1) = CVst(k);
    Vst1(i+1) = CVst1(k);
    S(i+1) = CS(k);
    Sfd(i+1) = CSfd(k);
    S1q(i+1) = CS1q(k);
    psi2(i+1) = Cpsi2(k);
    psid2(i+1) = Cpsid2(k);
    psiq2(i+1) = Cpsiq2(k);

    if Vb(i+1)>=Vbmax
        Vb(i+1) = Vbmax;
    end
    if Vm(i+1)<=Vmmin
        Vm(i+1)=Vmmin;
    elseif Vm(i+1)>=Vmmax
        Vm(i+1)=Vmmax;
    end
    if Vr(i+1)<=Vrmin
        Vr(i+1)=Vrmin;
```

```
    elseif Vr(i+1)>=Vrmax
        Vr(i+1)=Vrmax;
    end

    if Vst(i+1)<=Vstmin
        Vst(i+1)=Vstmin;
    elseif Vst(i+1)>=Vstmax
        Vst(i+1)=Vstmax;
    end

    V(i+1) = Vd(i+1)+1i*Vq(i+1);
    I(i+1) = Id(i+1)+1i*Iq(i+1);
    Vbus1(i+1) = V(i+1);
    Ibus1(i+1) = V(i+1)/ZLoad-I(i+1);
    Vbus2(i+1) = a*Vbus1(i+1)+Ibus1(i+1)*6.76*Z1/a;
    Ibus2(i+1) = Ibus1(i+1)*6.76/a;
    Vbus3(i+1) = (1+Y*Z2)*Vbus2(i+1)+Z2*Ibus2(i+1);
    Ibus3(i+1) =
(2*Y+Y^2*Z2)*Vbus2(i+1)+(1+Y*Z2)*Ibus2(i+1);
    Vbus4(i+1) = Vbus3(i+1)+Z3*Ibus3(i+1);
    Ibus4(i+1) = Ibus3(i+1);
    Sinf(i+1) = Vbus4(i+1)*conj(Ibus4(i+1));
    Pinf(i+1) = -real(Sinf(i+1));
    Qinf(i+1) = -imag(Sinf(i+1));
end
t = 0:h:h*18000;
m = 0:0.04*5/6:0.04*5/6*4499;
figure(1)
plot(t,Pinf)
hold on
plot(m,Ptable,'m:')
legend('Simulation','WECC data')
axis([0 150 5.5 6.2])
hold off
figure(2)
plot(t,Qinf)
hold on
plot(m,Qtable,'m:')
legend('Simulation','WECC data')
axis([0 150 -2.5 0.5])
hold off
figure(3)
plot(t,abs(Vbus4))
hold on
plot(m,Veq_1,'m:')
hold off
```

IX. Acknowledgement

This paper is made possible through the help and support from Dr. Bernard Lesieutre. The author would like to thank Dr. Bernard Lesieutre for his support and valuable advices. He kindly prepared me well with necessary knowledge before conducting this project, he also carefully read my paper and offered invaluable detailed advices on grammar, organization, and the theme of the paper.

X. References

[1] K. E. Atkinson, "Numerical integration," in *An Introduction to Numerical Analysis*, 2nd ed. Canada: John Wiley & Sons, 1989, pp. 251-263.

[2] J. D. Glover, M. S. Sarma, and T. J. Overbye, *Power System Analysis and Design*, 5th ed. Stamford, CT: Thomson Learning, 2012, pp. 311-342.

[3] C. K. Alexander, and M. N. Sadiku, "Circuit Theorems," in *Fundamentals of Electric Circuit*, 4th ed. New York: McGraw-Hill, 2009, pp. 127-175.

[4] P. W. Sauer, and M. A. Pai, "Synchronous machine dynamic models," in *Power System Dynamics and Stability*, Upper Saddle River, NJ: Prentice Hall, 1998, pp. 89-123.

[5] P. W. Sauer, "Constraints on saturation modeling in AC machines," IEEE Transactions on Energy Conversion, vol. 7, no. 1, pp. 161-167, Mar. 1992.

[6] *The Voltage Regulator Handbook*, Texas Instruments Inc., Dallas, TX, 1977.

[7] P. C. Krause, *Analysis of Electric Machinery*, New York: McGraw-Hill, 1986, 211-296.

[8] IEEE Recommended Practice for Excitation System Models for Power System Stability Studies, IEEE Standard 421.5, 2005.

[9] P. Kunder, *Power System Stability and Control*, New York: McGraw-Hill, 1994, pp. 699-827.

Ziqi Yang received the B.S. degree in Electrical Engineering from Ohio University, Athens, Ohio, in 2012 and the M.S. degree in Electrical Engineering Power Systems from University of Wisconsin Madison, Madison, Wisconsin, in 2014.

From 2013 to 2014, he was a Research Assistant with Dr. Bernard Lesieutre of University of Wisconsin Madison Electrical and Computer Engineering Department. His research interests include power system trajectory sensitivity analysis, and power system modeling and simulation. He is student members of Power System Engineering Research Center and Wisconsin Electric Machine and Power Electronics Consortium.

Mr. Yang's was a winner of the Global Stewards Sustainability Prize in 2013, the Outstanding Senior Award in 2012, and the Outstanding Junior Award in 2011.