MatPSST 1.0

Smart Grid Operation & Control center (SGO)

School of Electrical and Electronic Engineering

State Key Laboratory of Advanced Electromagnetic Engineering and Technology

Huazhong University of science and technology (HUST), Wuhan, P. R. China

By Prof. Wei Yao (w.yao@hust.edu.cn)

Mr. Siqi Lin (lin_sq047@hust.edu.cn)

August 2021

CONTENTS

1 Introduction

The MatPSST is a Matlab/Simulink-based power system simulation toolbox for electric power system analysis and simulation. It is a user-friendly, flexible, modularized, and supporting real-time simulation toolbox. The overarching goal of the MatPSST is *to enable users without advanced software and hardware knowledge to simulating power system and combine with real-time simulation platforms.* To achieve this, MatPSST minimizes code and makes full use of the solvers and toolboxes in MATLAB. So that the MatPSST has a smooth learning curve, and is very suitable for power system research and education.

MatPSST can perform the following task:

（1） Power flow calculation.

（2） Power system dynamic simulation.

（3） Small-signal analysis without additional programming.

（4） Various advanced controller designs.

（5） Combining with the real-time platform, such as dSPACE and RTLAB.

MatPSST has a library containing common device models, such as converters and renewable energy generators. Besides, with the GUI provided by Simulink, users can *easily incorporate the user-defined model and control strategy* into the MatPSST.

Since the Simulink is natively supported by some real-time simulation platforms, such as dSPACE and RTLAB, the model run on MatPSST can easily transfer to the dSPACE and RTLAB. It reduces the difficulty of real-time simulation of power system

MatPSST is an open-access toolbox. Which can be download from [1]. We hope the toolbox can help you overcome the problems in power system research and education.

PART I

User's Manual

2 Install

The prerequisites for MatPSST are the following:

- The latest version of MATLAB and Simulink should be installed. The MatPSST has been tested on MATLAB 2016B and is not guaranteed to run on older versions.
- It is recommended to install the control design toolbox. The model uses internal functions and Simulink modules in these two toolboxes.
- It is recommended to install MATPOWER 4.1 [2], it can be download from [3]
- It is recommended to install MATACDC 1.0 [4]. it can be download from [5]
- Set the current path to the location where the model is located.

Then the toolbox can be used.

3 Running a simulation

To start the simulation, take the M10B39 file as an example. This is a benchmark which is the New England system

- First, you should open the *initAll.m* file and run it, it is a file used to carry out the initialization process.
- Then, open the *M10B39.slx* file, it is a Simulink file.
- Finally, click the *run* button, the simulation can be started, and the simulation result can get from the scope.

4 Model and Data

4.1 Model

4.1.1 Synchronous generator

The synchronous generator is the basis of the power system. The parameters of the synchronous generator include motor parameters, governor parameters, PSS parameters, excitation system parameters, etc. The model in MatPSST is shown in Figure 1.

Input and output of generator modular are shown in Table 1. To the trip signal *trip*. When it is vacant, the input is 0, and the generator is connected to the grid. When the cut signal input is 1, this generator is cut off.

Table 1 Input and output of generator modular

Signal name	Meaning
Input	
Vxy	Terminal voltage
Trip	cut signal
Output	
Igxy1	Injection current
GBxy	Equivalent admittance
m	Measurement

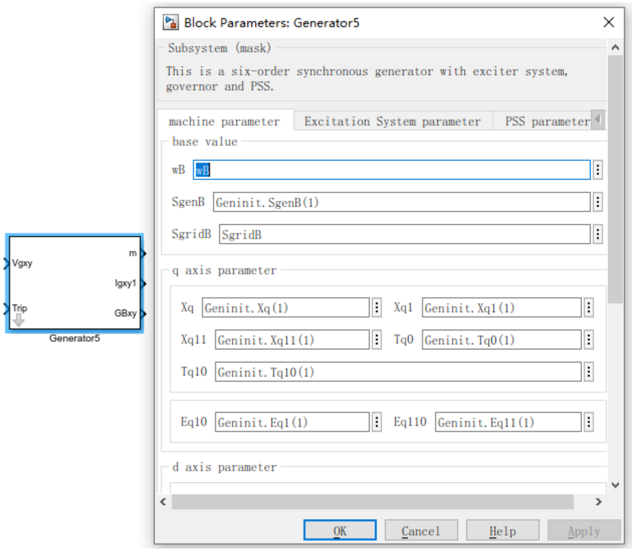


Figure 1 Synchronous Generator Module

To the generator, the basic frequency is used to define the frequency it works on, the SgenB and SgridB are used to define the basic capacity. Table 2 shows the parameter.

Table 2 General parameter

variable name	meaning
wB	Basic frequency value
SgenB	The basic capacity of the model
SgridB	Grid reference capacity

The generator model in the MatPSST takes a six-order model, the function of the machine can be written as (1). To model this function in MatPSST, the parameters in Table 3 needs to be set in the model.

$$
\begin{cases}
\dot{\delta}=\Omega_b(\omega-1) \\
\dot{\omega}=\left(P_m-P_e-D(\omega-1)\right)/T_j \\
T'_{d0}\dot{e}_q=e'_q-e''_q-(X'_d-X''_d)\,i_d+T'_{d0}pe'_q \\
T_{d0}\dot{e}_q{}'=E_f-e'_q-(X_d-X'_d)\,i_d \\
T'_{d0}\dot{e}_d{}''=e'_d-e''_d-(X'_q-X''_q)\,i_q+T'_{q0}pe'_d \\
T_{q0}\dot{e}_d{}'=-e'_d-(X_q-X'_q)\,i_q
\end{cases}
\tag{1}
$$

Table 3 Generator parameters

Variable name	Symbol	Meaning
Xd	X_d	d-axis reactance
Xq	X_q	q-axis reactance
Xd1	X'_d	d-axis transient reactance
Xq1	X'_q	q-axis transient reactance
Xd11	X''_d	d-axis sub-transient reactance
Xq11	X''_q	q-axis sub-transient reactance
Td0	T_{d0}	d-axis transient time constant
Tq0	T_{q0}	q-axis transient time constant
Td10	T'_d	d-axis sub-transient time constant
Tq10	T'_q	q-axis sub-transient time constant
Tj	T_j	Constant of inertia
D	D	Damping constant

Governors define the primary frequency control of synchronous machines. The transfer function of the governor is as (2). To model this function in MatPSST, the parameters in Table 4 need to be set in the model.

$$\Delta P_M = \begin{cases} 0 & Enable = 0 \\ \dfrac{1}{R}\left(\dfrac{1}{T_s+1} * \dfrac{sT_3+1}{sT_c+1} * \dfrac{sT_4+1}{sT_5+1}\right)\Delta\omega & Enable = 1 \end{cases} \tag{2}$$

Table 4 Governor parameters

Variable name	Symbol	Meaning
Enablegov		Whether to enable the governor
R	R	Droop coefficient
Ts	T_s	Transfer function parameters
T3	T_3	Transfer function parameters
Tc	T_c	Transfer function parameters
T4	T_4	Transfer function parameters
T5	T_5	Transfer function parameters

The excitation system defines the primary voltage regulation of synchronous machines. The transfer function of the excitation system is as (3). To model this function in MatPSST, the parameters in Table 5 need to be set in the model.

$$E_f = K_A\left[\left(\frac{1}{1+sT_b}\right)\left(1-\frac{T_c}{T_b}\right)+\frac{T_c}{T_b}\right](V_{ref}-V_g+V_{PSS}) \tag{3}$$

Table 5 Excitation system parameters

Variable name	Symbol	Meaning
Ka	K_a	Excitation multiple
Tb	T_b	Transfer function parameters
Tc	T_c	Transfer function parameters
Efmin	E_{fmin}	Minimum excitation system output
Efmax	E_{fmax}	Maximum excitation system output
Vref	V_{ref}	The terminal voltage reference value
Vf1	V_{f1}	Initial value of excitation system

Power System Stabilizers (PSSs) are typically used for damping power system oscillations. The transfer function of PSS in MatPSST is (4). To model this function in MatPSST, the parameters in Table 6 needs to be set in the model.

$$U_{pss} = \begin{cases} 0 & Enable = 0 \\ K_{pss}\dfrac{T_w s}{T_w s+1}\dfrac{T_1 s+1}{T_2 s+1}\dfrac{T_3 s+1}{T_4 s+1} & Enable = 1 \end{cases} \tag{4}$$

Table 6　PSS parameters

Variable name	Symbol	Meaning
Kpss	K_{pss}	gain
Tw	T_w	Transfer function parameters
T1	T_1	Transfer function parameters
T2	T_2	Transfer function parameters
T3	T_3	Transfer function parameters
T4	T_4	Transfer function parameters
Upmax	U_{pmin}	The maximum output of the PSS
Upmin	U_{pmax}	The minimum output of the PSS

4.1.2　Double fed inductor generator (DFIG)

MatPSST provides a double-fed inductor generator (DFIG) model for the wind generator. The structure of the DFIG model is shown in Figure 2. a DFIG model contains the motor parameters, phase-locked loop parameters, rotor-side control parameters, and stator-side control parameters.

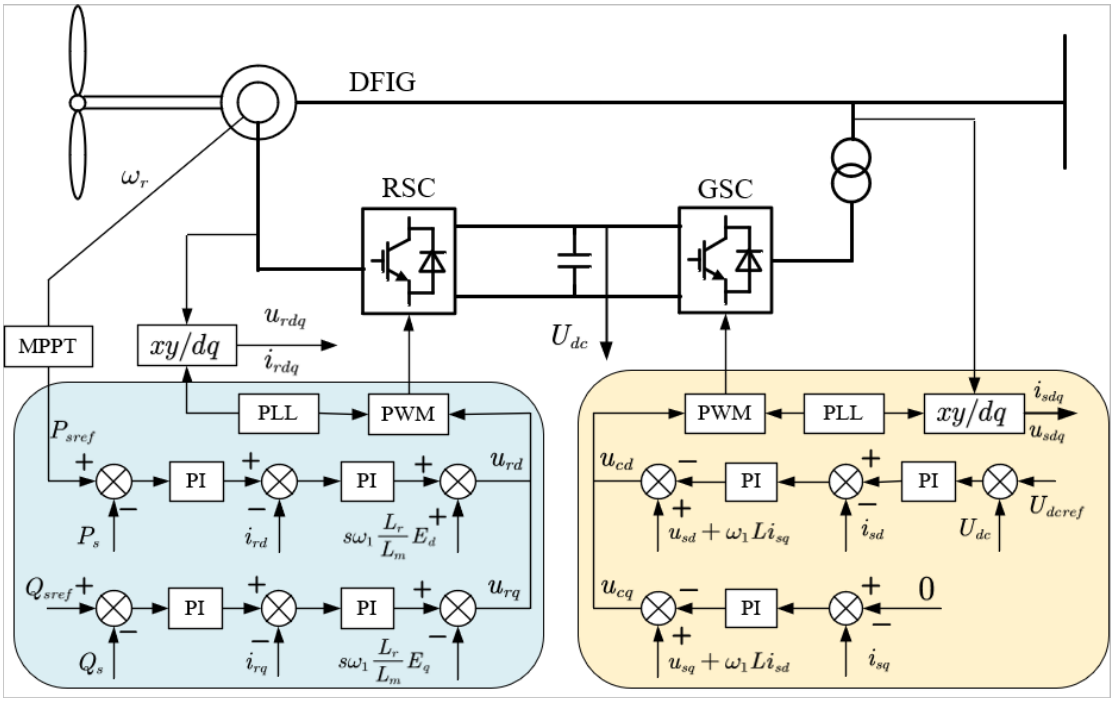


Figure 2　The structure of the DFIG

The model of DFIG in MatPSST is shown in Figure 3.

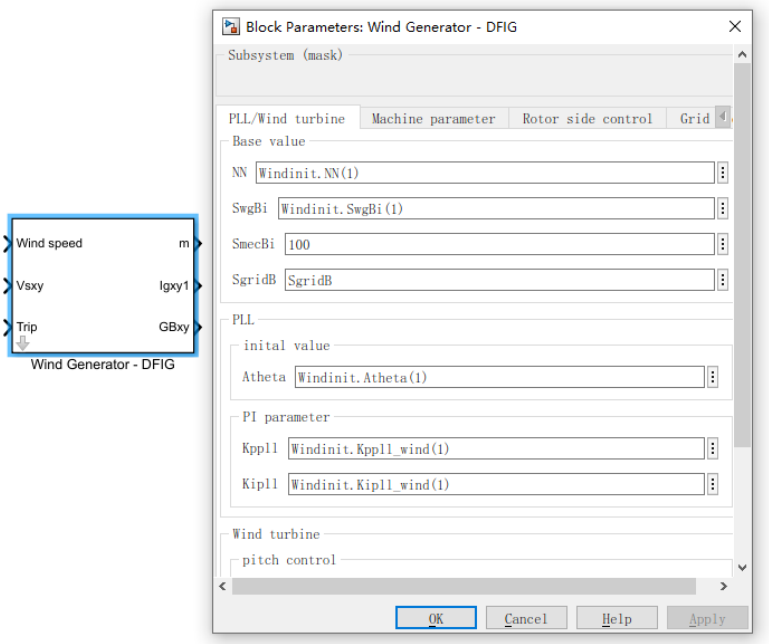


Figure 3　DFIG modular

Input and output of DFIG modular are shown in Table 7. To the trip signal *trip*. When it is vacant, the input is 0, and the DFIG is connected to the grid. When the cut signal input is 1, this DFIG is cut off.

Table 7　Input and output of DFIG

Signal name	Meaning
	Input
Wind speed	The wind speed of DFIG
Vxy	Terminal voltage
Trip	Cut signal
	Output
Igxy1	Injection current
GBxy	Equivalent admittance
m	Measurement

Because of the small capacity of a single wind generator, the wind farm contains multiple wind generators. Usually, researchers aggregate multiple wind generators into one equivalent wind generator. To help users model the wind generator, the basic parameter is shown in Table 8.

Table 8 Basic parameters

Variable name	Meaning
NN	Number of DFIGs
SwgBi	The reference value of single DFIG
SgridB	AC network benchmark value
SmecB	Wind Turbine Reference Value
wB	Frequency reference value

In MatPSST, the d component of the stator voltage was selected as the real part of the busbar voltage and q component was selected as the imaginary part. The q-axis was assumed to be 90° ahead of the d-axis with respect to the direction of rotation. This transformation is realized by the phase-locked loop (PLL). The structure is shown in Figure 4. To model this function, the parameters in Table 9 needs to be set in the model.

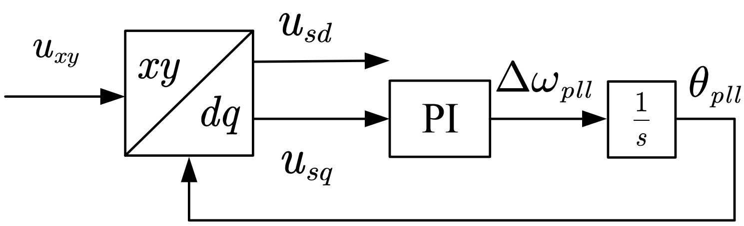


Figure 4 PLL model

Table 9 PLL parameters

Variable name	Meaning
Kppll_wind	PLL PI proportional parameters
Kipll_wind	PLL PI integration parameters

The function of DFIG takes a 3-order model (5), which can refer to [6]. To model this function, the parameters in Table 10 needs to be set in the model.

$$
\begin{cases}
v_{sd}=R_s i_{sd}-X_{a1} i_{sq}+e_d \\
v_{sq}=R_s i_{sq}+X_{a1} i_{sd}+e_q \\
\dot{e}_d=-\dfrac{1}{T_{a0}}\left[e_d+\dfrac{L_m^2}{L_r}\times i_{sq}\right]+s\omega_B e_q-\omega_B\dfrac{L_m}{L_r}v_{rq} \\
\dot{e}_q=-\dfrac{1}{T_{a0}}\left[e_q-\dfrac{L_m^2}{L_r}\times i_{sd}\right]-s\omega_B e_d+\omega_B\dfrac{L_m}{L_r}v_{rq} \\
\dot{\omega}_r=\dfrac{1}{H}(T_e-T_m)
\end{cases}
\tag{5}
$$

The rotor side and the grid side control take the double loop control. The structure of them is shown in Figure 2. To model them, the control parameter of the RSC and the GSC in Table 11 and Table 12 should be set.

Table 10 Machine parameters

Variable name	Symbol	Meaning
H	H	Rotor inertia
Xa1	X_{a1}	Asynchronous motor sub-transient reactance
Lm	L_m	Mutual inductance of the asynchronous motor
Rr	R_r	Asynchronous motor rotor resistance
Lr	L_r	Asynchronous motor rotor self-inductance
Rs	R_s	Asynchronous motor stator resistance
Ls	L_s	Asynchronous motor stator leakage inductance
Ta0	T_{a0}	The asynchronous motor time constant

Table 11 Rotor side parameters

Variable name	Meaning
KpP_rw	Active outer loop proportional coefficient
KiP_rw	Active power outer loop integral coefficient
KpId_rw	d-axis current proportional coefficient
KiId_rw	d-axis current integral coefficient
KpQ_rw	Reactive power outer loop proportional coefficient
KiQ_rw	Reactive power outer loop integral coefficient
KpIq_rw	q-axis current proportional coefficient
KiIq_rw	d-axis current integral coefficient
Qs0	The reactive power reference value

Table 12 Grid side parameters

Variable name	Meaning
KpDC_gw	DC voltage outer loop proportional control coefficient
KiDC_gw	DC voltage outer loop integral control coefficient
KpId_gw	d-axis current proportional control coefficient
KiId_gw	d-axis current integral control coefficient
KpIq_gw	q-axis current proportional control coefficient
KiIq_gw	q-axis current integral control coefficient
Vdc_w	DC voltage reference value

There is a capacitance between the RSC and GSC used to exchange energy, and the value of it is decided by the parameter in Table 13.

Table 13　DC parameters

Variable name	Meaning
C_w	Back-to-back converter capacitors

4.1.3 Voltage source converter (VSC)

The model of voltage source converter (VSC) is provided in the MatPSST, which is shown in Figure 5. The function of the VSC can be referred to [7]. However, the VSC usually adopts different control strategies, such as the master-slave control and the droop control, which is difficult to integrate into one model. So, the MatPSST only provides two different control strategies. The input and output of the VSC modular are shown in Table 14.

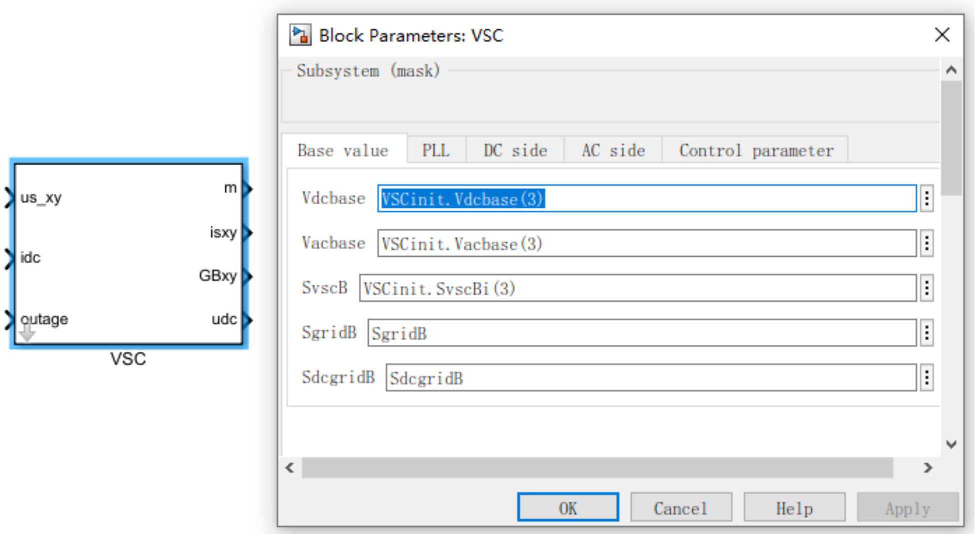


Figure 5　VSC modular

Table 14　Input and output of VSC

Signal name	Meaning
Input	
Vxy	Terminal voltage
idc	The dc current of VSC
outage	Cut signal
Output	
Igxy1	Injection current
GBxy	Equivalent admittance
m	Measurement
Udc	The dc voltage of the VSC

To the trip signal *outage*. When it is vacant, the input is 0, and the VSC is connected to the grid. When the cut signal input is 1, this VSC is cut off.

A series of general values are used for the VSC model, shown in Table 15.

In MatPSST, the d component of the stator voltage was selected as the real part of the busbar voltage and q component was selected as the imaginary part. The q-axis was assumed to be 90° ahead of the d-axis with respect to the direction of rotation. This transformation is realized by the phase-locked loop (PLL). The structure is shown in Figure 4.

To model this function, the parameters in Table 16 needs to be set in the model.

Table 15 General parameters

Variable name	Meaning
Vdcbase	DC voltage reference value
Vacbase	AC voltage reference value
SvscBi	converter reference capacity
SgridB	AC grid base capacity
SdcgridB	DC grid base capacity
wB	Frequency reference value

Table 16 PLL parameters

Variable name	Meaning
Kppll	PLL proportional coefficient
Kipll	PLL integral coefficient

The model of the VSC is shown in Figure 6, which can be written as (6). The parameters are shown in Table 17.

$$\begin{aligned}
&\frac{L_{pr}}{\omega_B}\dot{i}_{sd}=u_{sd}-u_{cd}-R_{pr}i_{sd}+\omega_1 L_{pr}i_{sq}\\
&\frac{L_{pr}}{\omega_B}\dot{i}_{sq}=u_{sq}-u_{cq}-R_{pr}i_{sq}-\omega_1 L_{pr}i_{sd}\\
&C_{dc}\dot{U}_{dc}=\frac{1}{U_{dc}}(P_{conv}-P_{line})
\end{aligned} \tag{6}$$

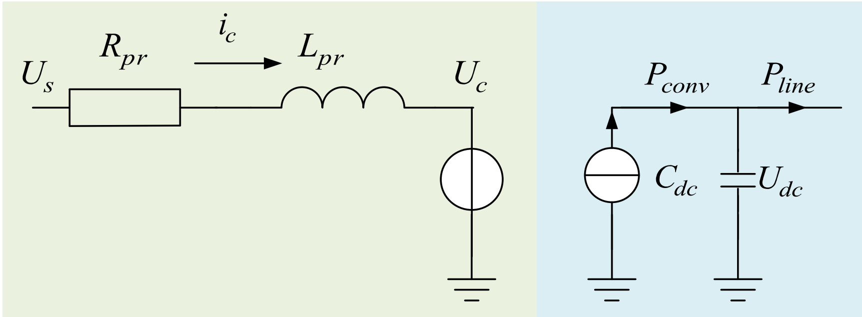


Figure 6 The structure of VSC

Table 17 VSC parameters

Variable name	Symbol	Meaning
Lpr	L_{pr}	converter coupling inductance
Rpr	R_{pr}	converter coupling resistance
Cdc	C_{dc}	Converter capacity

The outer loop controller of the converter contains different control methods, which are determined by dq_axis_control variable and specified in the *converter_init.m*. Among them, d-axis 1 mode represents active power control and 2 represents DC voltage control. q-axis 1 mode represents reactive power control and 2 represents AC voltage control. The control parameter is shown in Table 18.

Table 18 Control parameter

Variable name	Meaning
d_axis_control	d-axis control parameters
KpP	Active outer loop proportional coefficient
KiP	Active power outer loop integral coefficient
Kpdc	DC voltage outer loop proportional coefficient
Kidc	DC voltage outer loop integral coefficient
KpId	d-axis current inner loop proportional coefficient
KiId	d-axis current inner loop integral coefficient
q_axis_control	q-axis control coefficient
KpQ	Reactive power outer loop proportional coefficient
KiQ	Reactive power outer loop integral coefficient
Kpac	AC voltage outer loop proportional coefficient
Kiac	AC voltage outer loop integral coefficient
KpIq	q-axis current inner loop proportional coefficient
KiIq	q-axis current inner loop integral coefficient

4.1.4 AC grid

AC grid model is used to generate and store the admittance of the network, the model is shown in Figure 7. Input and output of AC grid modular are shown in Table 19.

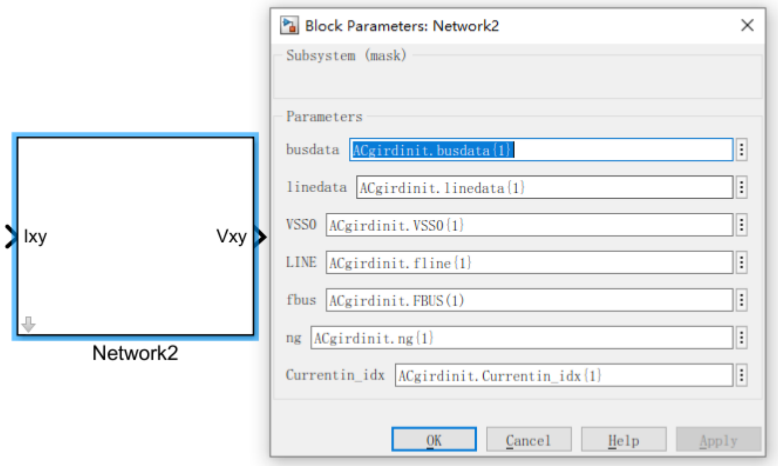


Figure 7 AC grid module

The input of the AC grid is a vector containing all the injection current and the equivalent admittance from the model, the structure of the input is like Figure 8. The n means the index in *Currentin_idx*. The components are aggregated through the mux module, then they are sent into the AC grid model.

Table 19 Input and output of AC

Signal name	Meaning
	Input
Ixy	Injection current and equivalent admittance
	Output
Vxy	Terminal voltage

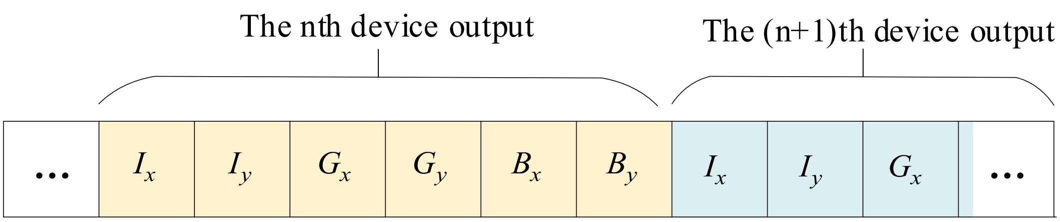


Figure 8 The structure of AC grid input

The output is the terminal voltage of each component V_{xy}. The order of the output element is the same as the input element.

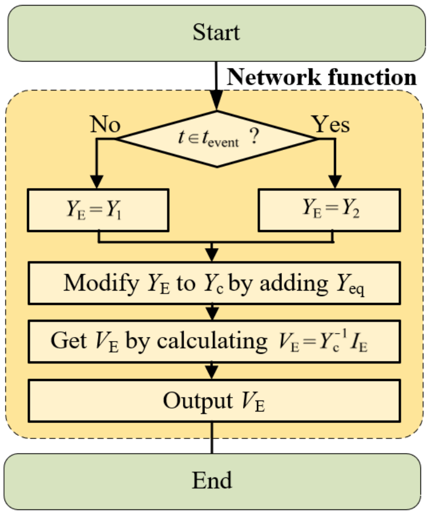


Figure 9 The flow chart of the function

The admittance of the AC power grid is automatically generated by extracting the power flow calculation data from the initial function and generally does not need to be modified. The AC model is modeled by S-function blocks, the flow chart of the function is shown in Figure 9.

To form the matrix and set fault time, the parameters are listed in Table 20.

Table 20 AC grid parameters

Variable name	Meaning
busdata	Node data
linedata	Line data
VSS0	Load data
fline	Fault removal circuit
FBUS	short-circuit node
ng	Number of dynamic models connected to the network
Curriencidx_in	The order of the device connected to the network model

The fault line and short-circuit node need to be manually specified in the initial function of the AC network. The time of fault removal needs to be specified in the AC network module.

4.1.5 DC grid

DC grid model is used to connect different VSCs to model the DC grid. The model is shown in Figure 10.

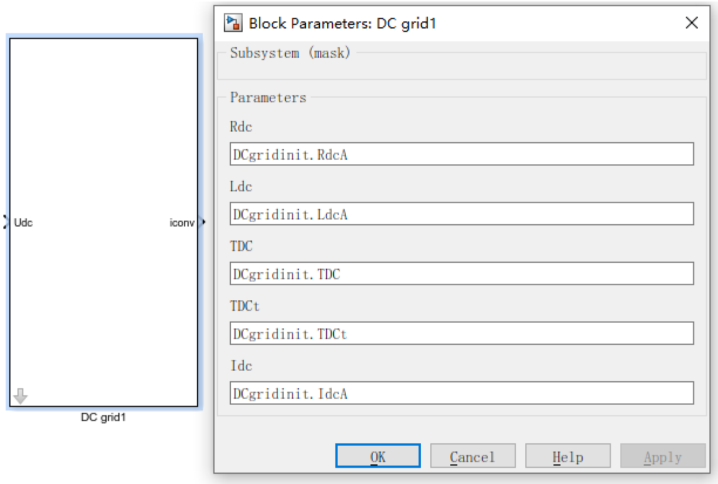


Figure 10　DC grid module

A DC grid containing n nodes and b branches can be described by an incidence matrix T, with element T_{ik}=1 if branch k leaves node *i*, T_{ik}=−1 if branch k enters node *i*, and T_{ik}=0 if there is no direct connection between branch k and node i. So, to a DC grid, the function can be written as (7). The input and output of the DC grid are shown in Table 21.

$$\begin{cases} \frac{1}{L}\dot{i}_{line} = T_{dc}^{T} u_{dc} - R i_{line} \\ i_{conv} = T_{dc} i_{line} \end{cases} \tag{7}$$

Where the i_{line} is the vector of all line current, *R* and *L* are the vector of all line resistance and inductance, the i_{conv} is the vector of VSC DC current, and the u_{dc} is the vector of VSC terminal voltage. All these parameters in the function are listed in Table 21 and Table 22.

Table 21　The input and output of DC grid

Signal name	Meaning
	Input
Vdc	The terminal voltage of all the VSC.
	Output
Iconv	DC current of all the VSC.

Take the system in Figure 11 as an example, the direction of VSC DC current enters the DC grid and the direction of the DC current in DC grid is the same as the direction in power flow calculation. The incidence matrix T of this DC grid is the (8).

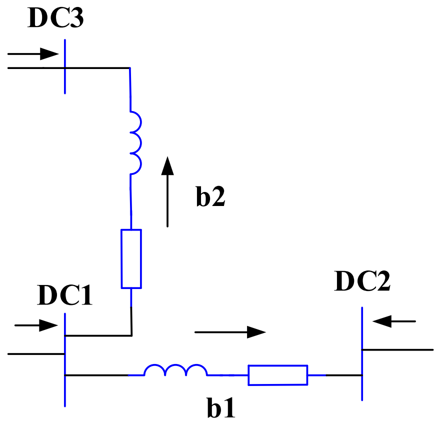


Figure 11 Structure of DC grid

$$T = \begin{array}{c} \\ n_1 \\ n_2 \\ n_3 \end{array} \begin{array}{c} \begin{array}{cc} b_1 & b_2 \end{array} \\ \begin{bmatrix} 1 & 1 \\ -1 & 0 \\ 0 & -1 \end{bmatrix} \end{array} \tag{8}$$

The parameter of the DC grid model is shown in Table 22.

Table 22 DC line parameters

Variable name	Symbol	Meaning
R	R	The resistance of all DC line
L	L	The induction of all DC line
TDC	T_{dc}	The incidence matrix of DC grid
TDCt	T_{dc}^{T}	The transposition of the incidence matrix

4.2 The output of simulation results

The output of the simulation results can use the blocks in the sinks library like Figure 12. You can view the output curve, or input the data into the work area with the blocks.

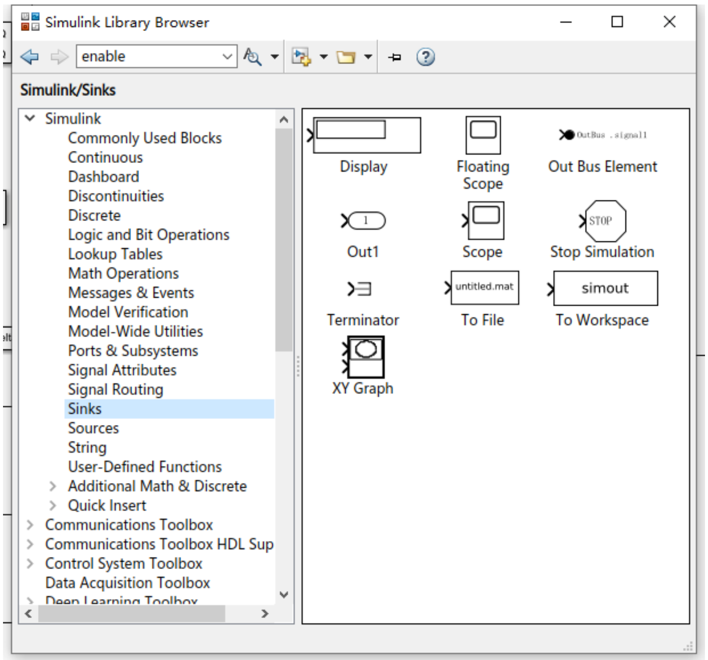


Figure 12　Sink library

4.3　User-defined model

4.3.1　　Use existing models to build a new system

1、Draw the system diagram and number the nodes and generators correctly.

This step helps you understand the structure of the system.

2、Input the power flow data into the power flow program provided in the MatPSST or the MATPOWER and MATACDC to adjust the power flow to convergence.

Please refer to the manual for the use of MATPOWER and MATACDC. However, the power flow calculation software here is not limited to MATPOWER and MATACDC, other power flow calculation software can be used when necessary. But it must be able to output network data and power flow calculation results to carry out the initialization process.

3、Add the initialization function corresponding to the equipment model included in the system, modify the number and value of the parameters in the initial function

The vector calculation is adopted in the initialization function, and the dimensions of the variables need to be consistent. For example, If the power flow result contains three devices, but the parameters of four devices are given, the dimensions will be different and calculation errors may happen.

4、Drag in the required modules to Simulink, modify the parameters, and connect them as required.

4.3.2 Customize the new model

1、List the dynamic equations of the components, including differential equations and algebraic equations, and implement them in Simulink.

Since the network is modeled by an admittance matrix, the input of the device is voltage and the output of the device is current.

2、Connect the model to the network.

3、Set the model parameters and calculate the initial conditions of the model.

Because the startup process of the device often involves changes in equations and control logic, which does not model in the MatPSST. The steady state of the model needs to be obtained to avoid the start-up process of the device. The initial condition of the model is determined by the initial value of each integral module.

Generally speaking, the equation of the model can be summarized as (9):

$$\begin{aligned} \dot{\vec{x}} &= f\left(\vec{x}, \vec{y}, \vec{\lambda}\right) \\ y &= g\left(\vec{x}, \vec{y}, \vec{\lambda}\right) \end{aligned} \tag{9}$$

The meaning of initialization is to calculate by solving the equation. Because the component parameters also need to be considered in the calculation process, the parameters of the equipment model need to be given in the initial files. When solving (10), the power flow calculation results generally need to be used. Based on the power flow calculation results, write your program to calculate the initial value of each state variable (integral module) in the component, and complete the initialization of the model.

$$\begin{aligned} 0 &= f\left(\vec{x}, \vec{y}, \vec{\lambda}\right) \\ y &= g\left(\vec{x}, \vec{y}, \vec{\lambda}\right) \end{aligned} \tag{10}$$

5 Numerical integration algorithm

MatPSST uses the numerical solvers in Simulink to solve dynamic functions. Currently, Simulink includes eight fixed-step solvers, ode1, ode2, ode3, ode4, ode5, ode8, ode14x and ode1be, as well as eight variable-step solvers, ode45, ode23, ode113, ode15s, ode23t, ode23t, ode23tb and odeN [8]. They can solve traditional power system dynamic problems easily and cope with the stiff problems caused by the device with fast dynamics. Also, the power system model usually contains discontinuities components, such as switch and saturation, discrete components such as digital controller, and logic operations. The Simulink solvers can easily handle these hybrid models, which overcome the shortcomings of most free toolboxes with few solving algorithms. Users can directly select the appropriate solver and set step size and tolerances in the Simulink setting page as needed, without modifying the model. The setting page is shown in Figure 13.

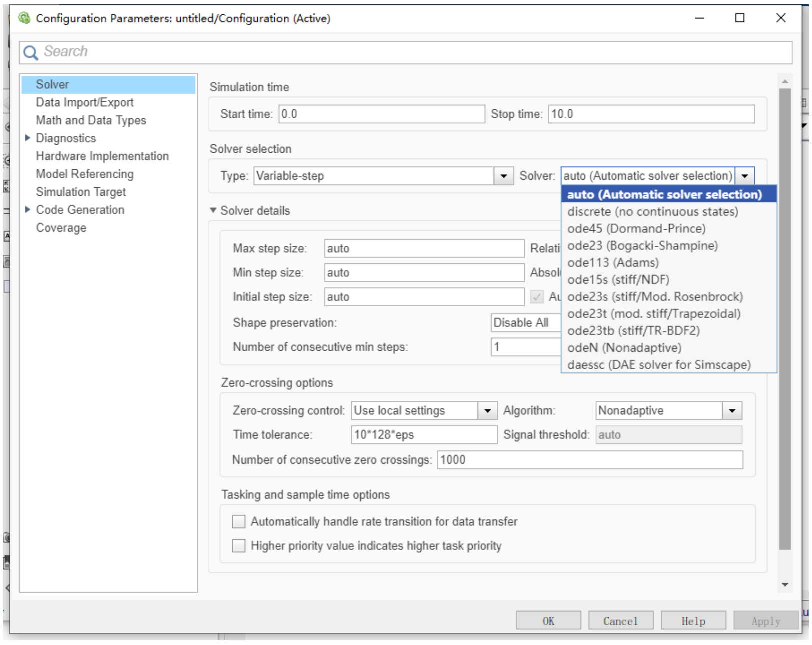


Figure 13 Solver setting page

6 Model small signal analysis

Small-signal analysis of the power system is an important tool to analyze the character of power systems. This requires the use of the Model Linearizer in Simulink [9]. The steps of linearization are:

1. Specify the linearization analysis point: right-click on a signal line and select the menu-Linear Analysis point, you can select the open-loop analysis point and the closed-loop analysis point. As shown in Figure 14.

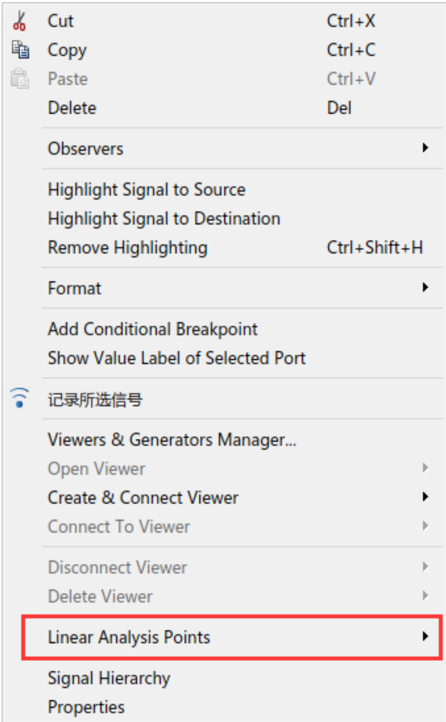


Figure 14　Linear Analysis Points

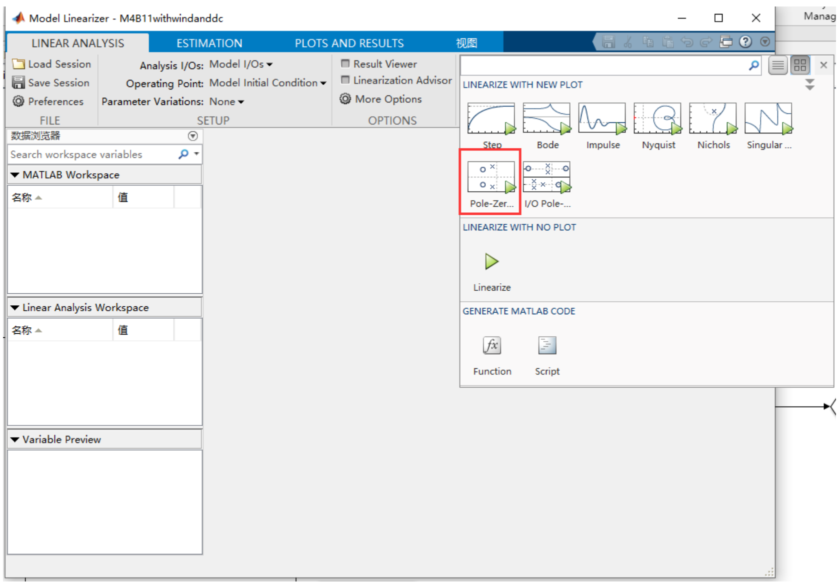


Figure 15　Plot pole-zero diagram.

2. Open the linearization toolbox, as shown in Figure 15. In linear analysis, click linearize to select the content of small-signal analysis. The Pole-zero diagram is usually used.

For more information, you can read the related help files in MATLAB.

7 Combining with RTLAB

For the MatPSST is based on Simulink, it can generate the C code and support RTLAB natively. Users can use MatPSST to realize the power system real-time simulation in a short time, at a low cost. It means users can easily carry out distributed processing, parallel computing, and interacting with the actual device, which facilitates the design and validation process of newly developed controllers and components in complex embedded systems.

The steps are shown in Figure 16. First, a model of the system should be built in MatPSST. Because the real-time simulation only supports a fixed-step solver, the model needs to be carried out the off-line simulation with a fixed-step solver normally. Then, the AC grid has been divided into different parts according to the area. Users can divide the whole system into different parts and place them into three different submodules. RTLAB software in the upper computer will translate the model into code, assign and download each submodule to a CPU core in the lower computer automatically. Finally, the model can be run on lower computer and connected with the actual controller.

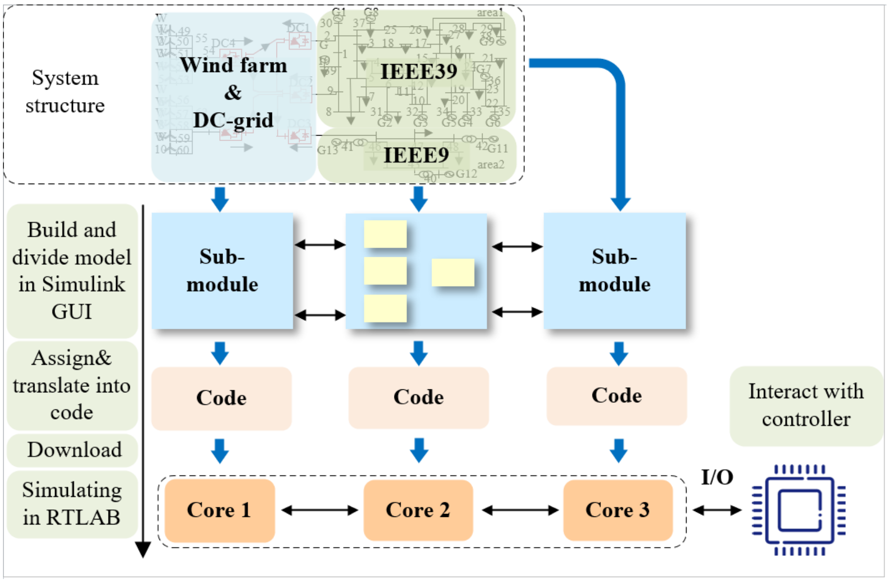


Figure 16 The step to combine MatPSST with RTLAB

PART II

Examples

8 Validation

MatPSST is validated by comparing its results with those obtained by using the open-source software package PSAT [10], which can be download from [11], and Simscape [12].

8.1 Verification with Electromagnetic Transient Model

To verify the accuracy of the MatPSST, the dynamic model and the electromagnetic transient model of the system in Figure 17 are built in the MatPSST and Simscape respectively.

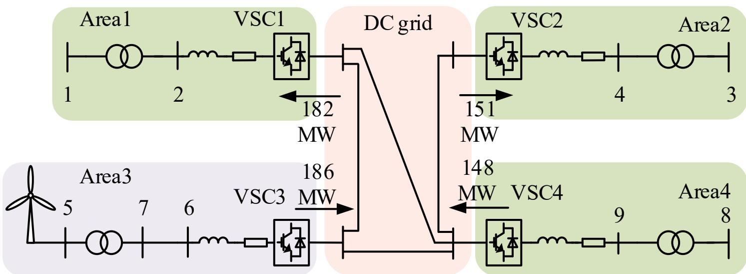
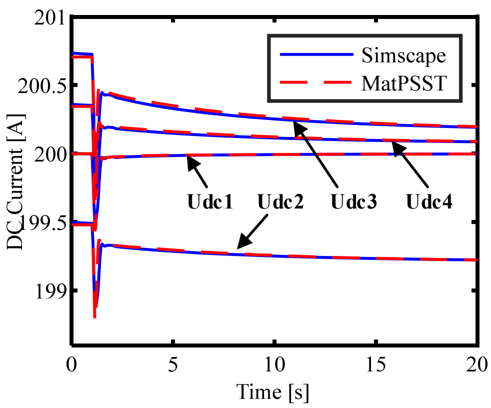
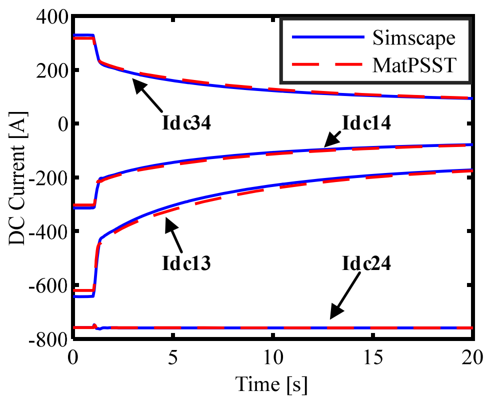


Figure 17 System for electromagnetic validation

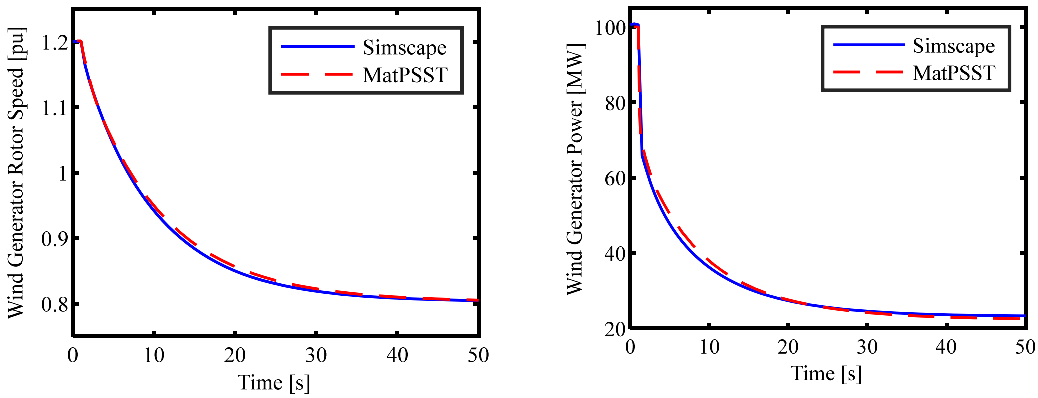
Scenario one is the wind speed changes. In the beginning, the wind speed is 15 m/s, the rotor speed is 1.2 pu. At 1 s, the wind speed drops by 7.5 m/s. The result is shown in Figure 18.



(a)



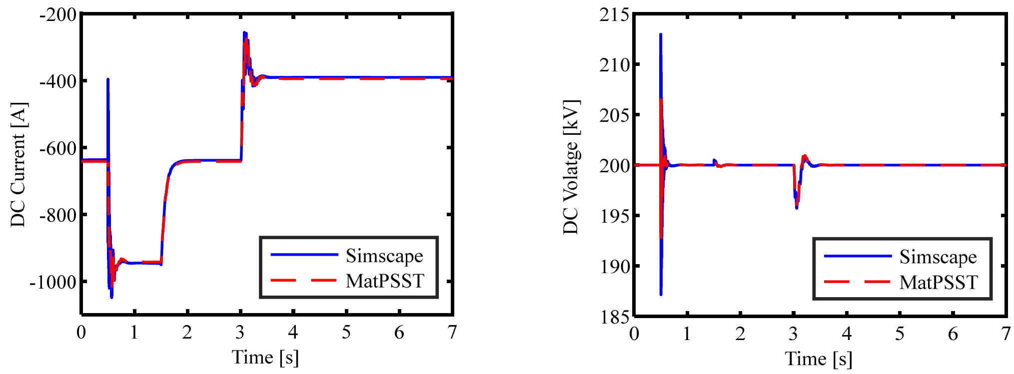
(b)



(c) (d)

Figure 18 Dynamics of the system during the wind speed change: (a) The DC voltage; (n) The DC current. (c) The rotor speed of wind generator; (d) The output power of wind generator.

Scenario two is the DC fault, at 1.5 s, the DC line 14 was cut off and reconnected at 2.5 s. At 4 s, an outage happens in VSC4. The dynamics of the DC system during the DC faults are depicted in



(a) (b)

Figure 19 Dynamics of DC grid during the DC fault: (a) The DC voltage of DC 1; (b) The DC current of line 14.

It can be seen that the results of MatPSST closely matches the results of Simscape except for some high-frequency components, which verifies the accuracy of MatPSST.

8.2 Verification with PSAT

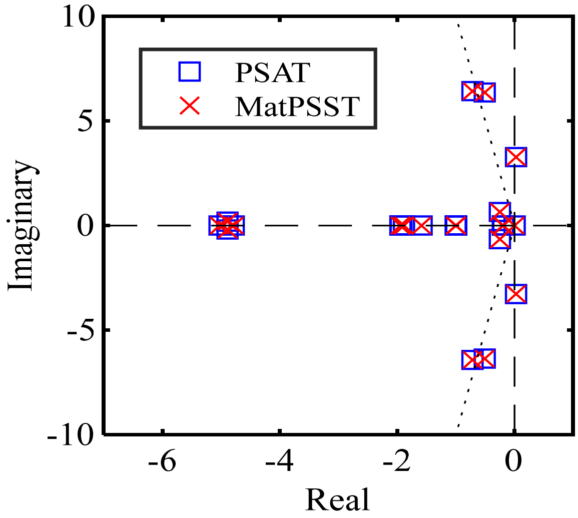


Figure 20 Relevant pole for Kundur's system obtained by PSAT and MatPSST

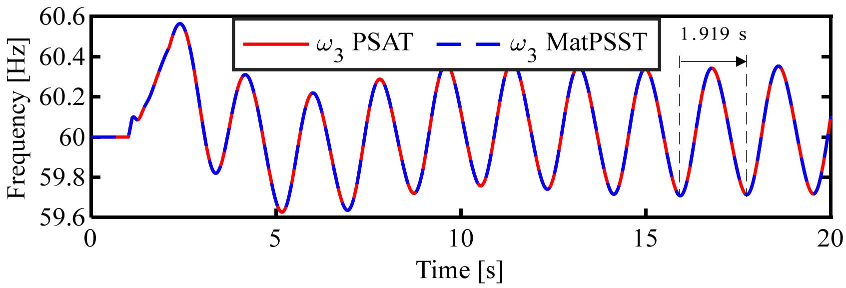


Figure 21 Time-domain results in PSAT and MatPSST

The small-signal analysis results obtained in PSAT and MatPSST are shown in Figure 20. A negative damping mode exists in the system, the oscillation frequency of the mode is 0.521 Hz. Moreover, at 1 s, a three-phase fault is applied in a line near bus seven, the faulty line is tripped out in 100 ms and reclosed successfully in another 1 s. The dynamic simulation result shown in Figure 21 verifies the correctness of the small-signal analysis results.

The results show that MatPSST acquires almost the same result as PSAT.

9 Model introduction

Three cases are shown here to help users to simulate the system in MatPSST.

9.1 New England system

A benchmark system called the New England system is shown in MatPSST in Figure 22. The dynamic model is built in *M10B39.slx* and the initialization process is set in *initAll.m*. Users should carry out the *initAll.m* file first to get the power flow result and initial value of the model. Then start the simulation in *M10B39.slx*.

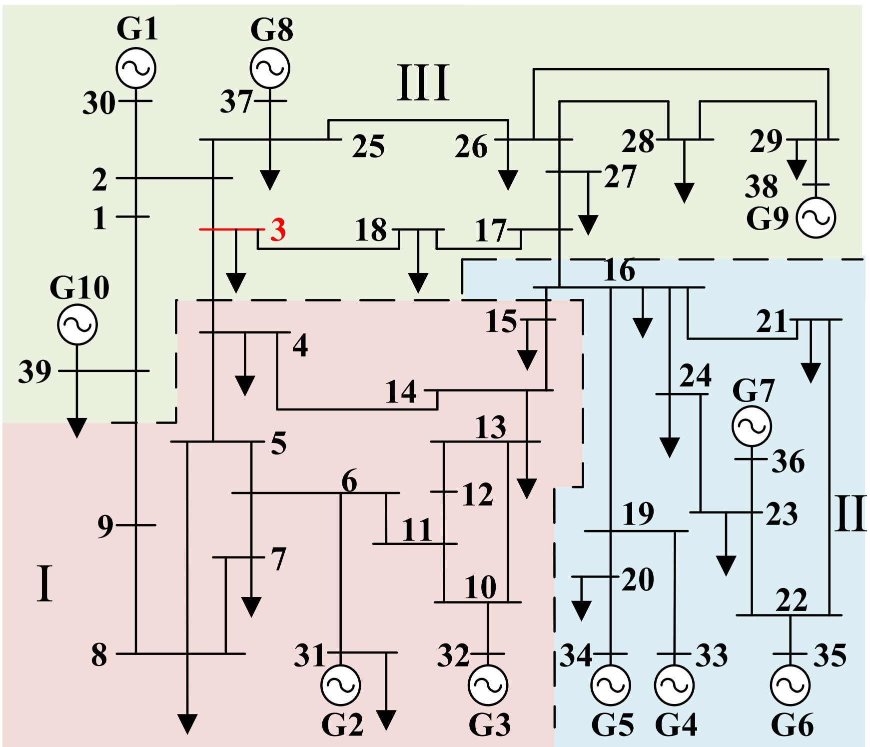


Figure 22 The single line diagram of the New England system

The files in this case are introduced here:

initAll.m: It is the MATLAB script file to carry out the initialization process, other files and functions are called in this file. So, the users can open this file to understand the process.

PowerFlowData.m: It is the MATLAB function to store the power flow data. This case uses the power flow program provided by MatPSST. This function is called in *initAll.m*

Powerflowcaculate.m: It is the MATLAB function to get the power flow result, the power flow program is in this file. This function is called in *initAll.m*.

Generator_init.m: It is the MATLAB function to set up the generator parameter and calculate the initial value of the generator. This function is called in *initAll.m*.

M10B39.slx: It is the Simulink model. All the dynamic function is modeled and calculated in this file. In this file, users can simulate the power system dynamic and carry out the small-signal analysis.

TS_net_1_new.m: It is the Simulink S-function, it is called in Simulink to model the network function. In this file, the network matrix is augmented by adding the model equivalent admittance.

9.2 Kundur two area system

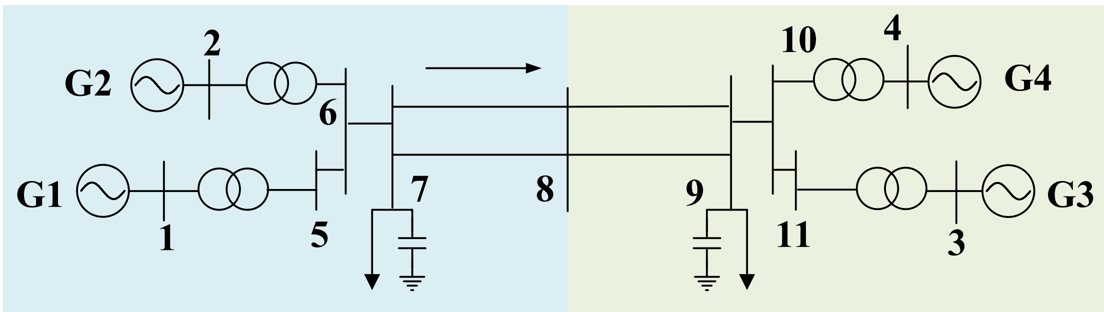


Figure 23 The single line diagram of Kundur two area system

A classic system is Kundur two-area system [13]. It is shown in Figure 23 and the model is built in MatPSST. This model uses MATPOWER to get the power flow result. So, to run this model, users should install the MATPOWER and MATACDC.

initAll.m: It is the MATLAB script file to carry out the initialization process, other files and functions are called in this file. So, the users can open this file to understand the process.

Generator_init.m: It is the MATLAB function to set up the generator parameter and calculate the initial value of the generator. This function is called in *initAll.m.*

M4B11.slx: It is the Simulink model. All the dynamic function is modeled and calculated in this file. In this file, users can simulate the power system dynamic and carry out the small-signal analysis.

ACgrid_init.m: This function is used to calculate the AC grid matrix. This function is called in *initAll.m.*

TS_net_1_new.m: It is the Simulink S-function, it is called in Simulink to model the network function. In this file, the network matrix is augmented by adding the model equivalent admittance.

9.3 Two area system with wind farm

A two area system with wind farm based on the Kundur two-area system is shown in MatPSST. This model uses the MATPOWER and the MATACDC to get the power flow result. So, to run this model, users should install the MATPOWER and MATACDC.

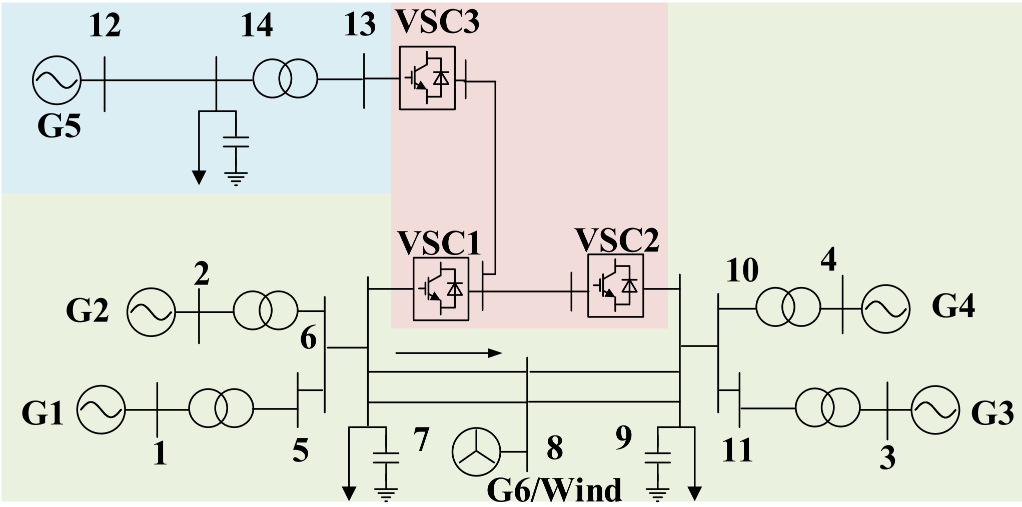


Figure 24 The single line diagram of two area system with wind farm

As shown in Figure 24, a three-terminal DC grid is connected to node seven, node nine and node thirteen. And a wind generator (DFIG) is connected to node eight.

The files in this case are introduced here:

initAll.m: it is the MATLAB script file to carry out the initialization process, other files and functions are called in this file. So, the users can open this file to understand the process.

P_ACdata.m: This file contains the AC power flow data used in MATPOWER.

P_DCdata.m: This file contains the DC power flow data used in MATACDC.

Powerflowcaculate.m: It is the MATLAB function to get the power flow result, the matpower and matacdc are called in this file. This function is called in *initAll.m.*

mpcTVSA.m: This model uses the *table* to store the network information and the initial value since this data structure in the Matlab can store more various data compare with the matrix. However, the table is not compatible with the MATPOWER and

MATACDC, so this function is used to convert the MATPOWER matrix to table and convert table to the matrix.

ACgrid_init.m: This function is used to calculate the AC grid matrix. This function is called in *initAll.m.*

nDCgrid_init.m: this function is used to set up the DC grid parameter and calculate the initial value. This function is called in *initAll.m.*

converter_init.m: It is the MATLAB function to set up the converter parameter and calculate the initial value of the converter. This function is called in *initAll.m.*

nWind_machine_init.m: It is the MATLAB function to set up the DFIG parameter and calculate the initial value of DFIG. This function is called in *initAll.m.*

Generator_init.m: It is the MATLAB function to set up the generator parameter and calculate the initial value of the generator. This function is called in *initAll.m.*

Two_area_system_with_wind_farm.slx: It is the Simulink model. All the dynamic function is modeled and calculated in this file. In this file, users can simulate the power system dynamic and carry out the small-signal analysis.

TS_net_1_new.m: It is the Simulink S-function, it is called in Simulink to model the network function. In this file, the network matrix is augmented by adding the model equivalent admittance.

TS_net_2_new.m: It is the Simulink S-function, it is called in Simulink to model the network function. In this file, the network matrix is augmented by adding the model equivalent admittance.

Reference

[1] *MatPSST*. [Online]. Available: http://www.xihe-energy.com/matpsst

[2] R. D. Zimmerman, C. E. Murillo-Sánchez, and R. J. Thomas, “MATPOWER: steady-state operations, planning, and analysis tools for power systems research and education,” *IEEE Trans. Power Syst.*, vol. 26, no. 1, pp. 12–19, Feb. 2011, doi: 10.1109/TPWRS.2010.2051168.

[3] *MATPOWER*. [Online]. Available: https://matpower.org/

[4] J. Beerten and R. Belmans, “Development of an open source power flow software for high voltage direct current grids and hybrid AC/DC systems: MATACDC,” *IET Gener. Transm. Distrib.*, vol. 9, no. 10, pp. 966–974, Jul. 2015, doi: 10.1049/iet-gtd.2014.0545.

[5] *MATACDC*. [Online]. Available: https://www.esat.kuleuven.be/electa/teaching/matacdc/

[6] J. B. Ekanayake, L. Holdsworth, and N. Jenkins, “Comparison of 5th order and 3rd order machine models for doubly fed induction generator (DFIG) wind turbines,” *Electric Power Systems Research*, vol. 67, no. 3, pp. 207–215, 2003.

[7] S. Cole, J. Beerten, and R. Belmans, “Generalized dynamic VSC MTDC model for power system stability studies,” *IEEE Trans. Power Syst.*, vol. 25, no. 3, pp. 1655–1662, Aug. 2010, doi: 10.1109/TPWRS.2010.2040846.

[8] Mathworks Inc, “Solver,” 2021. https://ww2.mathworks.cn/help/simulink/gui/solver.html?lang=en

[9] Mathworks Inc, “Linearization,” 2021. https://ww2.mathworks.cn/help/slcontrol/linearization.html

[10] F. Milano, “An open source power system analysis toolbox,” *IEEE Trans. Power Syst.*, vol. 20, no. 3, pp. 1199–1206, Aug. 2005, doi: 10.1109/TPWRS.2005.851911.

[11] *Power System Analysis Toolbox*. [Online]. Available: http://faraday1.ucd.ie/psat.html

[12] *Specialized Power Systems*. Mathworks Inc. [Online]. Available: https://www.mathworks.com/help/physmod/sps/specialized-power-systems.html

[13] Prabha kundur, *Power system stability and control*. McGraw-hill New York, 1994.