

CHAPTER 5

STATE SPACE MODEL GENERATION ALGORITHM

5.1 Problem Formulation

Many references [5, 6, 9] use a simple AC-DC power system to calculate the state space representation and design the controller in the HVDC link. They use one generator and one HVDC link which connect to infinite bus. The development of state space representation is simple. The following figure shows configurations of the system.

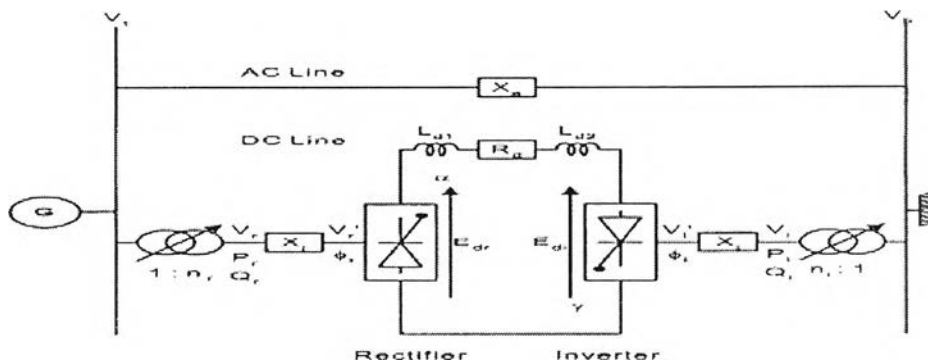


Figure 5.1 A simplified single machine and infinite bus power system with HVDC link [5].

This kind of configuration did not show the actual system. In the real system we have many generators, HVDC links, buses and branches. With system depicted in Fig. 5.1 the investigation that we can do is only local oscillations. We cannot see any inter-area/ global oscillations in the system because the representation of system is simple.

In this thesis the power systems (see Fig. 3.1 in Chapter 3) can be consist of :

- Multiple number of generators
- Multiple number of HVDC links
- AC network which consist of multiple number of buses and branches depend on the system

This kind of system will be useful for complete analysis of small signal stability in the system. We can study both local and inter-area/ global oscillations. Thus, we can see the interaction between one generator to other generators or interaction between group of generators to other groups. In this system also account the presence of HVDC, so that we can study also the interaction between generators and HVDCs in the system.

5.2 Input Data Preparation

We use the actual system data to get the state space representation of the system. No need to reduce the size of the system. In this program we need three kind of data : network data, generator and exciter data, HVDC data.

The developed program can handle power system with many numbers of generators, HVDCs and buses. The input which is needed by the program are data of

generators (including exciter), data of HVDC's, and data of network. The same network data for load flow studies is used for the calculation.

The initial conditions of the state variables for the model in Chapter 2 are computed by systematically solving the load-flow of the network first, and then computing the other algebraic and state variable.

5.2.1 Network Data

The network data is useful to calculate the operating condition of the system. In this thesis we refer to *MATPOWER* for running the load flow. The format of the input network data are shown in Table 5.1-5.4. However, the original data for *MATPOWER* is not account for the presence of HVDC. Because the configuration in our system is including HVDC, we have to add to the network data with HVDC information.

The network data is consist of : bus data, generator data, HVDC data and branch data. Here are the detailed data needed for each subdata.

Table 5.1 Bus data format

Bus	Type	Pd	Qd	Gs	Bs	Area	Vm	Va	baseKV	Zone	Vmax	Vmin	HVDC

Table 5.2 Generator data format

Bus	Pg	Qg	Qmax	Qmin	Vg	MBase	Sta	Pmax	Pmin

Table 5.3 HVDC data format

Bus	Interface	Idc	Alfa	Gamma	Tap_min	Tap_max	Edi

Table 5.4 Branch data format

fbus	tbus	R	X	B	rateA	rateB	rateC	Ratio	Angle	Status

The data files used by *MATPOWER* are simply MATLAB M-files or MAT-files which define and return the variables baseMVA, bus, branch, gen, areas, and gencost. The baseMVA variable is a scalar and the rest are matrices. Each row in the matrix corresponds to a single bus, branch, or generator. The columns are similar to the columns in the standard IEEE and PTI formats [20]. The details of the specification of the *MATPOWER* case file can be found in the help for *caseformat.m*.

Load flow has been the traditional mechanism for computing a proposed steady-state operating point. Loads are of the constant power type. The standard requirements of load flow analysis are as follows :

- Specify bus voltage magnitudes numbered 1 to m .
- Specify bus voltage angle at bus number 1 (slack bus).

- Specify net injected real power P_i at buses numbered 2 to m .
- Specify load powers P_{Li} and Q_{Li} at buses numbered $m+1$ to n .

In this case, we have to add HVDC information. Because of this reason, we modify format data in bus data and adding one more data which is HVDC data. In bus data we add one more column, such that we have 14 columns now (see Table 5.1). In the 14th column of Table 5.1, we fill 0 (zero) if the bus is not HVDC bus and 1(one) if the bus is HVDC bus.

For HVDC data (Table 5.3), in column 1 we fill the number of HVDC bus. In column 2 we fill 1 when it exports power and fill 2 when it imports power. Alfa and Gamma are in radian. I_{dc} and E_{di} are in pu. We have to choose the difference of tap_{max} and tap_{min} to be as small as possible to make the calculation more accurate.

In this developed program, we order and arrange the number of the bus to be the first ' m ' buses for generator buses, the next ' d ' buses for HVDC buses and the last ' p ' buses are load buses.

5.2.2 Generator and Exciter Data

In this developed program, we use two axis model for generator and exciter IEEE Type 1. The input parameter needed for each generator and exciter are shown in Table 5.5 and 5.6.

Table 5.5 Generator data

Parameters	Unit	Explanation
H	secs	the inertia
X_d	pu	the d axis synchronous reactance
X'_d	pu	the d axis transient synchronous reactance
X_q	pu	the q axis synchronous reactance
X'_q	pu	the q axis transient synchronous reactance
τ'_{d0}	secs	the d axis transient open circuit time constant
τ'_{q0}	secs	the q axis transient open circuit time constant
D	pu	the damping factor

Table 5.6 Exciter data

Parameters	unit	Explanation
K_A	pu	the regulator gain
τ_A	secs	the regulator time constant
K_E	pu	the exciter gain
τ_E	secs	the exciter time constant
K_F	pu	the regulator stabilizing circuit gain
τ_F	secs	the regulator stabilizing circuit time constant
A	pu	the derived saturation constant for rotating exciters
B	pu	the derived saturation constant for rotating exciters

5.2.3 HVDC Data

The developed program needs data of parameter in HVDC. Table 5.7 shows the input parameter needed for each HVDC with control scheme such as : constant current (CC) control at the rectifier side and constant-extinction angle (CEA) control at the inverter side.

Table 5.7 Parameter of HVDC link

Parameter	Unit	Explanation
K_r	pu	the gain for inverter
T_r	secs	the time constant for inverter
K_i	pu	the gain for inverter
T_i	secs	the time constant for inverter
R_{dc}	pu	the resistance in HVDC link
X_r	pu	the leakage reactance in rectifier
X_i	pu	the leakage reactance in inverter
L_{dr}	pu	the smoothing reactance in rectifier
L_{di}	pu	the smoothing reactance in inverter

5.3 Programming Flows

In this part we will explain how the program works. We will give the main program and some detailed program.

5.3.1 Main Program

In this flow chart, its give the main programming flow. The first step is read input data from all components. After that the program will run load flow to calculate the operating point of the network. From operating point, program will continue to calculate initial condition all variables in generator and HVDC. Then program will calculate the matrix for each equation and do many calculations based on the algorithm to develop state model. Lastly we will get the matrices A , B , C , and D of the system. The flow chart of this part is in Fig. 5.3 below.

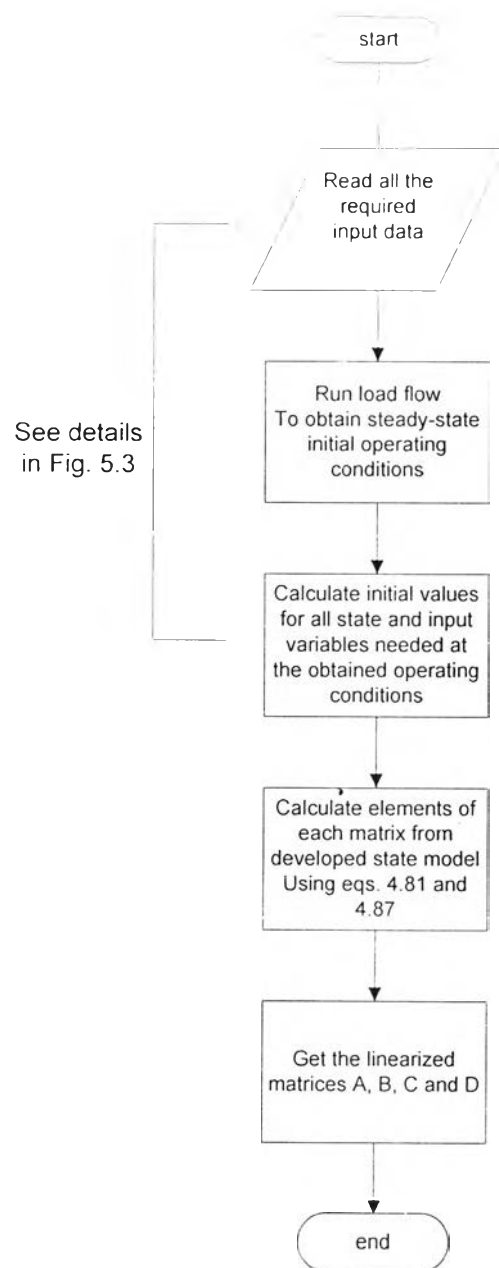


Figure 5.2 Main program to generate a linearized state space model.

5.3.2 Subroutine to Incorporate HVDC Links

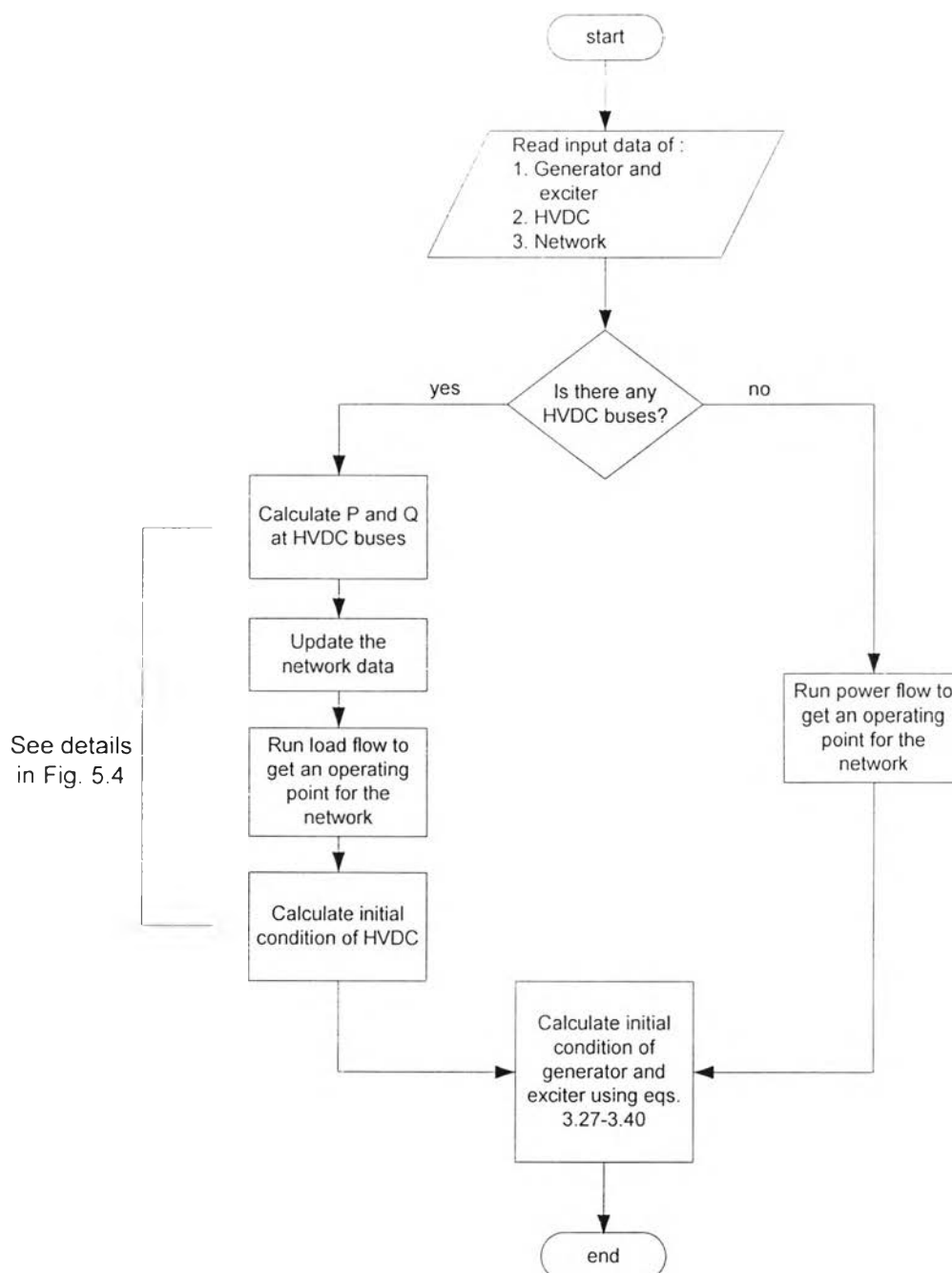


Figure 5.3 Subroutine to incorporate HVDC links.

In this algorithm, we saw that there is a possibility of the program to calculate state space representation of power systems with or without HVDC. The program will check whether there is HVDC in the system or not. If there is HVDC so we have to update the network data by calculating P and Q at the HVDC bus. Thus operating point of all variables will be change. Number of state also increases by the presence of HVDC in the system.

5.3.3 Subroutine to Compute Initial Conditions of HVDC

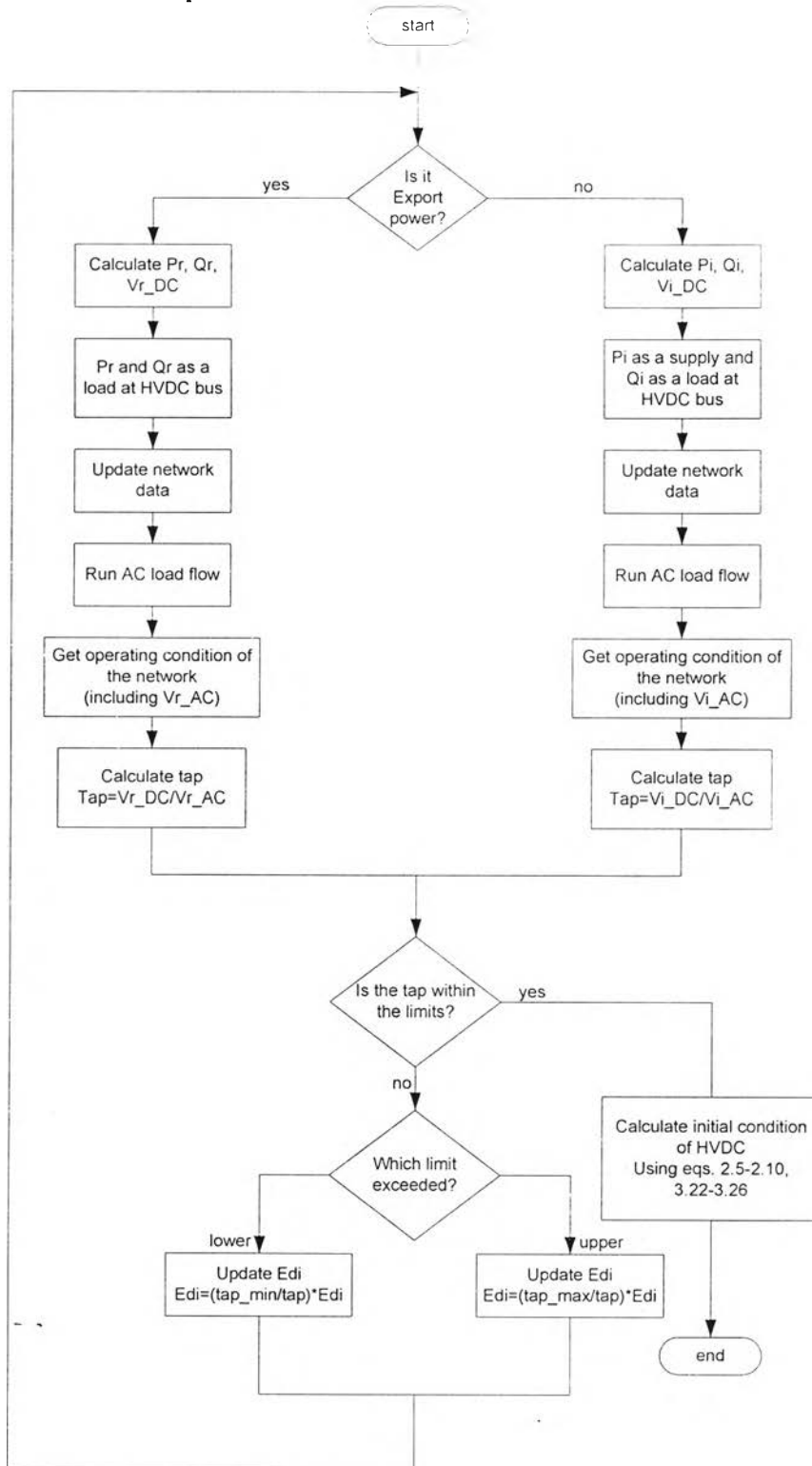


Figure 5.4 Subroutine to compute initial conditions of HVDC.

The previous flow chart in Fig. 5.5 shows algorithm to get initial condition of HVDC. This flow chart originally from literature [18] but we give modification on it. Firstly we will check whether there is HVDC in the system or not. If there is HVDC in the system we have to check first whether its export or import power.

When HVDC exports power to outside of the system, firstly, we calculate P and Q from the data available. After that we will update network data. We put P and Q to be a load at HVDC bus. Then we will run load flow and calculate tap (in this case tap in rectifier). If tap within limits so program will stop, thus we will get operating condition of network and initial condition of all HVDC variables. If tap outside from limits, we have to check which limit exceed, upper or lower limit. After that we will update value of E_{di} and calculate P and Q again until the tap within limits.

When HVDC imports power from outside of the system, same as when it exports power, we also calculate P and Q from the data available. After that we will update network data. We put P as a supply and Q to be a load at HVDC bus. The sign of P is minus (as a supply) and Q is positive (as a load). Then we will run load flow and calculate tap (in this case tap in inverter). If tap within limits so program will stop, thus we will get operating condition of network and initial condition of all HVDC variables. If tap outside from limits, we have to check which limit exceed, upper or lower limit. After that we will update value of E_{di} and calculate P and Q again until the tap within limits.

5.4 Outputs of the Program

The developed program will generate state space representation of the power systems with HVDC link. It produces matrices A , B , C , and D of the systems. We can use this program to calculate those matrices for two cases : when there is HVDC and when there is no HVDC in the systems. Also we have choices whether we import or export power through HVDC link. After having these matrices, the program will calculate eigenvalues, frequency of oscillation, damping ratio, participation factor, controllability matrix and observability matrix of the system.

The developed program is useful to analyze the small signal stability of the system. We can get the eigenvalues of the system thus we can check the stability of it. We also calculate the frequency of oscillation and damping ratio, thus we can investigate what kind of problem in the system. Moreover, we can see the participation factor to see the participation of states to each mode. Lastly, we can check the controllability and observability of the system. By this analysis, we can see how many modes are controllable and observable such that we can design corresponding controller to make the system stable.