

2005

# A novel approach for identification and tracing of oscillatory stability and damping ratio margin boundaries

Xiaoyu Wen  
*Iowa State University*

Follow this and additional works at: <https://lib.dr.iastate.edu/rtd>

 Part of the [Electrical and Electronics Commons](#)

## Recommended Citation

Wen, Xiaoyu, "A novel approach for identification and tracing of oscillatory stability and damping ratio margin boundaries " (2005).  
*Retrospective Theses and Dissertations*. 1604.  
<https://lib.dr.iastate.edu/rtd/1604>

This Dissertation is brought to you for free and open access by the Iowa State University Capstones, Theses and Dissertations at Iowa State University Digital Repository. It has been accepted for inclusion in Retrospective Theses and Dissertations by an authorized administrator of Iowa State University Digital Repository. For more information, please contact [digirep@iastate.edu](mailto:digirep@iastate.edu).

A novel approach for identification and tracing of oscillatory stability and damping ratio  
margin boundaries

by

Xiaoyu Wen

A dissertation submitted to the graduate faculty  
in partial fulfillment of the requirements for the degree of  
DOCTOR OF PHILOSOPHY

Major: Electrical Engineering (Electric Power)

Program of Study Committee:  
Venkataramana Ajarapu, Major Professor  
Degang J. Chen  
Wolfgang Kliemann  
James D. McCalley  
Vijay Vittal

Iowa State University

Ames, Iowa

2005

Copyright © Xiaoyu Wen 2005. All rights reserved.

UMI Number: 3184662

Copyright 2005 by  
Wen, Xiaoyu

All rights reserved.

### INFORMATION TO USERS

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleed-through, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

**UMI**<sup>®</sup>

---

UMI Microform 3184662

Copyright 2005 by ProQuest Information and Learning Company.

All rights reserved. This microform edition is protected against unauthorized copying under Title 17, United States Code.

ProQuest Information and Learning Company  
300 North Zeeb Road  
P.O. Box 1346  
Ann Arbor, MI 48106-1346

Graduate College  
Iowa State University

This is to certify that the doctoral dissertation of  
Xiaoyu Wen  
has met the dissertation requirements of Iowa State University

Signature was redacted for privacy.

**Major Professor**

Signature was redacted for privacy.

**For the Major Program**

## Table of content

Chapter 1 Introduction .....	1
Chapter 2 The DAE model of electric power systems.....	6
2.1 Formulation of the Power System DAE model .....	6
2.1.1 Synchronous Generator.....	6
2.1.2 Excitation Control System .....	7
2.1.3 Prime Mover and Speed Governor .....	8
2.1.4 Nonlinear Load Model.....	9
2.1.5 LTC Model.....	10
2.1.6 Network Power Equations .....	10
2.2 The parameterization of DAE model.....	11
Chapter 3 Literature Review .....	15
3.1 Introduction.....	15
3.1.1 The direct method to identify oscillatory stability loading margin.....	16
3.1.2 Eigenvalue based method to identify the oscillatory stability loading margin.....	18
3.1.3 The manifold-based method to identify oscillatory stability margin.....	21
3.2 Damping ratio margin.....	21
3.3 Oscillatory stability and damping margin boundaries tracing .....	23
Chapter 4 Oscillatory stability margin and damping margin identification .....	25
4.1 Oscillatory stability margin identification .....	25
4.2 Simulation result for oscillatory stability margin identification.....	35
4.3 Damping margin identification.....	39
4.4 Simulation result for damping margin identification.....	40
4.5 Discussion of the practical aspects of the proposed algorithm .....	41
4.5.1 Consideration for cases if the eigenvalue index doesn't work.....	41
4.5.2 Consideration of Discrete Events.....	47
4.6 Computational comparison .....	49
4.6.1 Step length selection and accuracy .....	49

4.6.2 Estimation of computational cost.....	53
4.6.3 Robustness of the algorithms .....	55
Chapter 5 Oscillatory stability and damping ratio margin boundary tracing.....	60
5.1 Oscillatory stability margin boundary tracing .....	60
5.1.1 Augmentation for boundary tracing.....	63
5.1.2 Boundary predictor .....	65
5.1.3 Boundary corrector .....	66
5.2 Damping ratio margin boundary tracing.....	69
5.3 Simulation results .....	70
5.3.1 Oscillatory stability margin boundary tracing .....	70
5.3.2 Damping ratio margin boundary tracing .....	74
5.4 Computational aspects .....	78
Chapter 6 Conclusions .....	82
References.....	85
Appendix 1: Data of test case- New England system.....	90
Appendix 2: Application of symbolic computing.....	93
Appendix 3: Some special cases in oscillatory stability margin boundary tracing .....	96
Acknowledgements: .....	98

## Chapter 1 Introduction

The electric power industry is undergoing worldwide deregulation and restructuring. In the past, one company provided all the functions of electric service (generation, transmission, distribution, and retail sales). With competition, these functions are separated into different companies. Generation, or production of electricity, was deregulated in North America around 1995, resulting in an ample supply of new, cleaner and more efficient power plants. With electric competition, retail electric providers sell electricity and provide functions such as customer service and billing. Retail Electric Providers compete for customers business by offering lower prices, renewable energy options, added customer service benefits or other incentives.

On the other hand, deregulation also brought a great challenge and opportunity for the electric power industry. Power producers, distributors and their suppliers are asked to be innovative and satisfy the consumers demand for more choices while operating competitively. The Independent System Operator (ISO) and transmission service provider (TSP) are encountering uncertain power flow patterns in the system due to unpredictable bidding strategies of the power producers. If the ISO or TSP fails to predict or detect congestions due to these patterns, the reliability of the power grid will be significantly reduced. The deterioration of the system operating conditions makes the power system more vulnerable to disturbances. In many cases, power is transferred via a highly stressed network. Large scale system instabilities have been experienced all over the world. On July 2, 1996, the WECC system experienced a major blackout caused by voltage instability. One month

after this disturbance, on August 10, 1996, another major blackout occurred in WECC. This time, as a result of undamped oscillations, the system split into four large islands. Over 7.5 million customers experienced outages ranging from a few minutes to nine hours. On August 4, 2000, poorly damped power oscillations were observed across the WECC system following the Alberta separation [1]. In the New England Electric System, system monitoring devices also recorded oscillatory responses of major event [2].

Currently, in North America, ISO only considers thermal limits in identifying transmission congestion, i.e., for any post-contingency, no transmission line, transformer or breaker's thermal rating is exceeded. Some transmission companies also started to apply voltage stability analysis software to monitor voltage stability in operations. In June, 2005, the ERCOT (Electric Reliability Council of Texas) ISO became the first ISO in North America to implement voltage stability assessment in real time operations. For every hour, VSAT (Voltage Security Assessment Tool by Powertech Labs) is executed with the most updated EMS state estimator results. More than two thousands contingencies and six power transfer scenarios are considered. Any transmission congestion constrained by a voltage stability margin will be identified and considered in market operations.

With VSAT, long execution time is still a problem in power system real time operations. Two thousand contingencies are defined, but only around two hundred contingencies are chosen for the next step screening process according to geographic information. Finally, around twenty of the most critical contingencies are picked for P-V curve tracing to find a voltage collapse point with respect to six different transfer increasing scenarios.



Oscillatory stability also plays a very important role in maintaining power system security as indicated in the disturbances mentioned in the previous pages. Recently (May 27, 2005) an apparent protection system failure in Hydro One's system in Canada produced power system oscillations and line outages that effected frequency in the eastern inter connection.

Oscillatory instability is an inherently nonlinear phenomenon that is related to bifurcation from the viewpoint of nonlinear dynamic systems. Substantial research has been conducted to help understand and analyze the mechanism of this type of instability based on Hopf bifurcation theory [3], [4], [5], [6]. Hopf bifurcation occurs when a pair of complex eigenvalues crosses the imaginary axis when a parameter in the system is varied. Depending on the type of Hopf, this may lead to unstable oscillations in the system. Improper tuning of generation control parameters may lead to Hopf bifurcation [7], [8], [9]. Nonlinear load may also lead to Hopf bifurcation [10]. References [4], [11], [12] present an analysis related to a 1992 disturbance on the midwestern segment of the US interconnected power system and the resulting oscillations caused by line tripping. It confirms that the event was indeed related to a Hopf bifurcation.

The essential information of a Hopf bifurcation can be obtained in terms of eigenproperties of the reduced power system matrix of a structure-preserving power system model described by differential algebraic equations [13]. This matrix is called the dynamic system state matrix.

Damping also plays an important role in power system oscillations. Margin related to damping can be defined as the amount of additional load on a specified pattern of load

increase that would cause the damping ratio to reach its minimum limit. In the oscillatory stability assessment, in order to keep the system far away from the minimum damping limit, the damping ratio margin needs to be checked for each contingency and power transfer scenario.

WECC [14] recommend the following criteria in determining the safe operating limits:

The operating point is acceptable from the damping standpoint if:

With the path flow increased by the larger of 100 MW or 5 percent of the power transfer, any N-1 contingency will not result in undamped oscillations or instability;

With the path flow increased by the larger of 100 MW or 2 1/2 percent of the power transfer, any N-2 contingency will not result in undamped oscillations or instability

The motivation of this research is to establish a framework that can assess the oscillatory security level of the current operating point by quickly identifying and estimating the oscillatory stability margin and damping margin for pre-contingency and post-contingency conditions. This framework provides the relationship between the system parameters (including controllable parameters) and the oscillatory stability and damping margins. This information can be effectively used for optimal control design to avoid undesirable system behavior.

The thesis is organized as follows:

In chapter 2, the Differential-Algebraic Equation (DAE) model of the electric power system will be introduced. .

Chapter 3 provides a critical review of various existing methods that are available for oscillatory stability assessment. Chapter 4 presents, an eigenvalue tracing methodology to identify an oscillatory stability margin, as well as a damping margin. The methodology is demonstrated through the New England test system. Computational issues related to this technique are discussed in detail. Chapter 5 proposes algorithms to trace oscillatory stability and damping margin boundaries. Chapter 6 concludes with the major contributions of this thesis.

## Chapter 2 The DAE model of electric power systems

### 2.1 Formulation of the Power System DAE model

The studied power system is assumed to have  $n$  buses and  $m$  generators. Every generator is assumed to be equipped with the same type of speed governor and excitation control system, but the dynamic model parameters of different generator could be different. The formulation of power system modeling will be presented in this chapter. The most commonly used power system notations are adopted.

#### 2.1.1 Synchronous Generator

Without loss of generality, the  $m^{\text{th}}$  generator's rotor angle can be chosen as the system angle reference. The two-axis model [3][15] describing the synchronous machine dynamics can be given as:

$$\dot{\delta}_i = (\omega_i - \omega_m)\omega_0 \quad i = 1, \dots, m-1 \quad (2.1.1)$$

$$\dot{\omega}_i = M_i^{-1} [P_{mi} - D_i(\omega_i - \omega_m) - (E'_{qi} - X'_{di}I_{di})I_{qi} - (E'_{di} + X'_{qi}I_{qi})I_{di}] \quad i = 1, \dots, m \quad (2.1.2)$$

$$\dot{E}'_{qi} = T_{d0i}^{-1} [E_{fdi} - E'_{qi} - (X_{di} - X'_{di})I_{di}] \quad i = 1, \dots, m \quad (2.1.3)$$

$$\dot{E}'_{di} = T_{q0i}^{-1} [-E'_{di} + (X_{qi} - X'_{qi})I_{qi}] \quad i = 1, \dots, m \quad (2.1.4)$$

Where  $\omega_m$  is the system frequency,  $\omega_i$  is the machine frequency, namely, generator angular speed and  $\omega_0$  is the system rated frequency ( $377.0 \text{ rad/sec} = 60\text{Hz} \times 2\pi \text{ rad}$ ).  $I_{di}$  and  $I_{qi}$

are direct axis and quadrature axis currents respectively;  $E'_{di}$  and  $E'_{qi}$  are transient direct axis and quadrature axis EMF respectively;  $T_{d0i}$  and  $T_{q0i}$  are direct axis and quadrature axis open circuit time constants respectively;  $X'_{di}$  and  $X'_{qi}$  are direct axis and quadrature axis transient reactances and  $R_{si}$  is armature resistance of the machine;  $M_i$  is inertia constant and  $D_i$  is the damping constant of the machine. All the quantities are per unit except  $\omega_0$ .

Interface voltage equations to the network are given as follows:

$$E'_{qi} = V_i \cos(\delta_i - \theta_i) + R_{si} I_{qi} + X'_{di} I_{di} \quad (2.1.5)$$

$$E'_{di} = V_i \sin(\delta_i - \theta_i) + R_{si} I_{di} - X'_{qi} I_{qi} \quad (2.1.6)$$

Where  $V_i$  and  $\theta_i$  are bus voltage and angle respectively.

The machine currents  $I_{di}$  and  $I_{qi}$  can be eliminated by solving the generator interface equations to the network. Hence,

$$I_{di} = [R_{si} E'_{di} + E'_{qi} X'_{qi} - R_{si} V_i \sin(\delta_i - \theta_i) - X'_{qi} V_i \cos(\delta_i - \theta_i)] A_i^{-1} \quad (2.1.7)$$

$$I_{qi} = [R_{si} E'_{qi} - E'_{di} X'_{di} - R_{si} V_i \cos(\delta_i - \theta_i) - X'_{di} V_i \sin(\delta_i - \theta_i)] A_i^{-1} \quad (2.1.8)$$

$$A_i = R_{si}^2 + X'_{di} X'_{qi} \quad (2.1.9)$$

Note that (2.1.1) does not include the differential equation for  $\delta_m$ , and that all the angles here and henceforth are relative angles with respect to the  $m$ th generator's rotor angle.

### 2.1.2 Excitation Control System

The simplified IEEE type DC-1 excitation system [3] as shown in Fig.2.1 is used here. The corresponding mathematical model is

$$E_{fdi} = T_{ei}^{-1} [V_{ri} - [S_{ei}(E_{fdi})] E_{fdi}] \quad i = 1, \dots, m \quad (2.1.10)$$

$$\dot{V}_{ri} = T_{ai}^{-1}[-V_{ri} + K_{ai}(V_{refi} - V_i - R_{fi})] \quad i=1, \dots, m \quad (2.1.11)$$

If

$$V_{ri, \min} \leq V_{ri} \leq V_{ri, \max}, \quad V_{pssi} = 0 \quad (\text{at steady state})$$

$$R_{fi} = T_{fi}^{-1}[-R_{fi} - [K_{ei} + S_{ei}(E_{fdi})]K_{fi}E_{fdi}/T_{ei} + K_{fi}V_{ri}/T_{ei}] \quad i=1, \dots, m \quad (2.1.12)$$

where  $V_{refi}$  is the reference voltage of the automatic voltage regulator (AVR);  $V_{ri}$  and  $R_{fi}$  are the outputs of the AVR and exciter soft feedback;  $E_{fdi}$  is the voltage applied to generator field winding;  $T_{ai}$ ,  $T_{ei}$  and  $T_{fi}$  are AVR, exciter and feedback time constants;  $K_{ai}$ ,  $K_{ei}$  and  $K_{fi}$  are the gains of AVR, exciter and feedback;  $V_{ri, \min}$  and  $V_{ri, \max}$  are the lower and upper limits of  $V_{ri}$ .

### 2.1.3 Prime Mover and Speed Governor

Fig. 2.2 shows the block diagram for a simplified prime mover and speed governor.

Two differential equations are involved to describe the dynamics when no  $\mu_i$  limit is hit.

$$\dot{P}_{mi} = T_{chi}^{-1}(\mu_i - P_{mi}) \quad i=1, \dots, m \quad (2.1.13)$$

$$\dot{\mu}_i = T_{gi}^{-1}[P_{gsi} - (\omega_i - \omega_{ref})/R_i - \mu_i] \quad \text{if } \mu_{i, \min} \leq \mu_i \leq \mu_{i, \max} \quad i=1, \dots, m \quad (2.1.14)$$

where  $P_{gsi} = P_{gsi}^0(1 + K_{gi}\mu)$  is the designated real power generation;  $P_{gsi}^0$  is its setting at base case;  $K_{gi}$  is the generator load pick-up factor that could be determined by AGC, EDC or other system operating practices;  $P_{mi}$  is the mechanical power of prime mover and  $\mu_i$  is

the steam valve or water gate opening;  $R_i$  is the governor regulation constant, representing its inherent speed-droop characteristic;  $\omega_{ref}$  ( $=1.0$ ) is the governor reference speed;  $T_{chi}$  and  $T_{gi}$  are the time constants related to the prime mover and speed governor respectively;  $\mu_{i,min}$  and  $\mu_{i,max}$  are the lower and upper limits of  $\mu$ , where a parameter  $\mu$  is introduced to designate the system load level. At the base case,  $\mu$  equals zero.

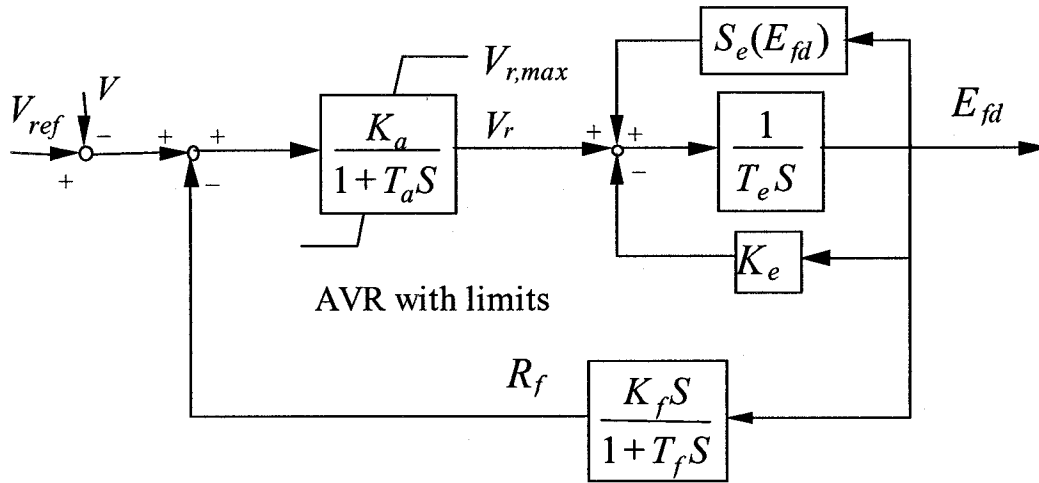


Figure 2.1: The IEEE type DC-1 excitation system

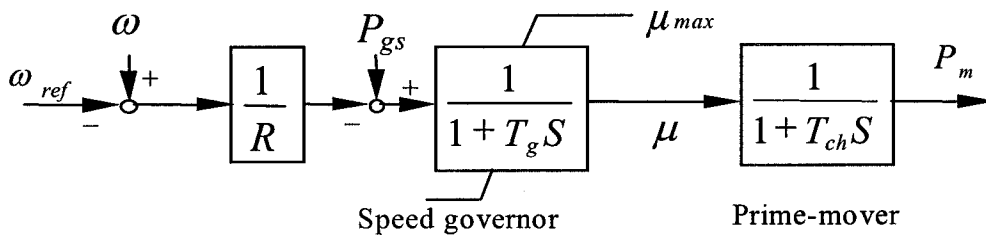


Figure 2.2 The simplified speed governor and prime mover

### 2.1.4 Nonlinear Load Model

The voltage and frequency dependent load is modeled as follows for all the load buses.

$$\begin{cases} P_{li} = P_{li0} (V_i / V_{i0})^{\alpha_i} [1 + K_{lpi} (\omega_m - \omega_r)] \\ Q_{li} = Q_{li0} (V_i / V_{i0})^{\beta_i} [1 + K_{lqi} (\omega_m - \omega_r)] \end{cases} \quad i = 1, \dots, m \quad (2.1.15)$$

Where  $P_{li0}$  and  $Q_{li0}$  are the active and reactive powers consumed by the load at the nominal voltage  $V_i$  and frequency  $\omega_r$  ( $=1.0$ ). The frequency dependent term is included to prevent the equilibrium computation from divergence in case all the generators reach their maximum real power limits due to load increase or generator outages. Here  $K_{lpi}$  and  $K_{lqi}$  are the load changing factors with respect to system frequency.

### 2.1.5 LTC Model

Continuous on Load Tap Changer (LTC) model is taken.

Assume there is an LTC between bus  $i$  and  $j$ ,

$$V_j = rV_i \quad (2.1.16)$$

$$T_i \dot{r} = V_j^{ref} - V_j \quad (2.1.17)$$

Where  $r$  is the ratio position of an LTC;  $t$  is the number of LTC;  $V_j^{ref}$  is the reference voltage at the LTC regulated bus  $j$ ;  $T_i$  is the time constant.

### 2.1.6 Network Power Equations

Corresponding to the above models, the network equations can be written as:

$$\begin{cases} 0 = P_{gi} - (1 + K_{lpi} \mu) P_{li} - P_{ii} \\ 0 = Q_{gi} - (1 + K_{lqi} \mu) Q_{li} - Q_{ii} \end{cases} \quad i = 1, \dots, n \quad (2.1.24)$$

Where



$$\begin{cases} P_{ti} = \sum_{k=1}^n V_i V_k Y_{ik} \cos(\theta_i - \theta_k - \phi_{ik}) \\ Q_{ti} = \sum_{k=1}^n V_i V_k Y_{ik} \sin(\theta_i - \theta_k - \phi_{ik}) \end{cases} \quad i = 1, \dots, n \quad (2.1.25)$$

and

$$\begin{cases} P_{gi} = I_{di} V_i \sin(\delta_i - \theta_i) + I_{qi} V_i \cos(\delta_i - \theta_i) \\ Q_{gi} = I_{di} V_i \cos(\delta_i - \theta_i) - I_{qi} V_i \sin(\delta_i - \theta_i) \end{cases} \quad i = 1, \dots, m \quad (2.1.26)$$

$P_{gi}$  and  $Q_{gi}$  are the generator output powers, which are primarily determined by the inherent characteristics of the speed governor and the AVR regulations. They will change if real power generation rescheduling and secondary voltage control is activated.  $P_{ti}$  and  $Q_{ti}$  are the powers injected into the network at bus  $i$ .  $K_{lpi}$  and  $K_{lqi}$  are the load changing factors specified for bus  $i$ . It should be noted that (2.1.24) is generic in the sense that it is used for all of the buses.

## 2.2 The parameterization of DAE model

The entire electric power system can be represented by differential and algebraic equations (DAE):

$$\begin{cases} \dot{X} = F(X, Y, \alpha, \beta); \\ 0 = G(X, Y, \alpha, \beta); \end{cases} \quad (2.2.1)$$

In the equation (2.2.1), the differential equation describes the dynamics associated with the generators, the excitation systems, the prime movers, and the speed governors. The algebraic equation represents the network power balance equations.  $X$  is the vector of the

state variables for differential equations with length  $n$ ,  $X = (\delta, \omega, E_q', E_d', P_m, \mu, E_{fd}, V_r, R_f)$ .  $Y$  is the vector of algebraic variables with length  $m$ ,  $Y = (V, \theta)$ .  $\alpha$  is the parameter chosen for bifurcation analysis. In the power system,  $\alpha$  represents the load level of the entire system.  $\beta$  represents the control change parameter.

In the load parameter space  $Z_i = (P_{iL}, Q_{iL})$  ( $i = 1, \dots, l$ , where  $l$  denotes the number of load buses in the power system), the load at each bus will be changed with the parameter  $\alpha$ ,

$$\begin{aligned} P_{iL} &= P_{iL}^0 (1 + \alpha K_{iPL}); \\ Q_{iL} &= Q_{iL}^0 (1 + \alpha K_{iQL}); \end{aligned}$$

The constant  $K_{iPL}$  and  $K_{iQL}$  define the scenario for load level change.

In the control parameter space  $U_i = (V_{refi}, K_{ai}, \dots)$  ( $i = 1, \dots, k$ , where  $k$  denotes the total number of possible controls in the system.), all the control parameters are changed with the parameter  $\beta$  change.

$$\begin{aligned} V_{refi} &= V_{refi}^0 (1 + \beta K_{iVref}); \\ K_{ai} &= K_{ai}^0 (1 + \beta K_{iKa}); \\ &\dots \end{aligned} \tag{2.2.2}$$

The constant  $K_{iVref}$ ,  $K_{iKa}$ , etc, define the scenario for control vector change.

For some parameter not controllable in power system, also can be used to find impact of its value variation on oscillatory stability margin, or damping margin. For example, the oscillatory stability margin after transmission line tripping can be obtained if one let susceptances  $B_{ij}$  and Admittance  $Y_{ij}$  of transmission line become

$$\begin{aligned} B_{ij} &= B_{ij}^0(1-\beta); \\ Y_{ij} &= Y_{ij}^0(1-\beta); \end{aligned} \quad \text{Where } \beta \in [0, 1] \quad (2.2.3)$$

When the parameter  $\beta = 0$ ,  $B_{ij} = B_{ij}^0$  and  $Y_{ij} = Y_{ij}^0$ . When the parameter  $\beta = 1$ ,  $B_{ij} = 0$  and  $Y_{ij} = 0$ . This represents the line between bus  $i$  and  $j$  is tripped. When one trace the margin boundary with  $\beta$  value gradually moving from zero to one, the margin value with  $\beta = 1$  is the post contingency oscillatory stability margin.

In the margin identification, the parameter  $\beta$  is fixed as zero. For the margin boundary tracing, both parameters ( $\alpha$  and  $\beta$ ) will be changed. It becomes a two parameters variation problem.

When the parameter in (2.2.1) is varied, the corresponding state vectors  $X$ ,  $Y$  and the eigenvalues of the system matrix are evaluated on this path change accordingly.

Linearization of (2.2.1) at the equilibrium point with parameter  $\alpha$  and  $\beta$  leads to:

$$\begin{bmatrix} \Delta \dot{X} \\ 0 \end{bmatrix} = \begin{bmatrix} F_X & F_Y \\ G_X & G_Y \end{bmatrix} \begin{bmatrix} \Delta X \\ \Delta Y \end{bmatrix} = A_{total} \begin{bmatrix} \Delta X \\ \Delta Y \end{bmatrix} \quad (2.2.4)$$

Matrices  $F_X$ ,  $F_Y$ ,  $G_X$ , and  $G_Y$  contain the first derivatives of  $F$  and  $G$  with respect to  $X$  and  $Y$ , evaluated at the equilibrium point.

Note that the matrix  $G_Y$  is an algebraic Jacobian matrix that contains the power flow Jacobian matrix.

In the above equation, if  $\det(G_Y)$  does not equal zero,

$$\Delta Y = -G_Y^{-1} G_X \Delta X \quad (2.2.5)$$

Substituting in (2.2.4) results in

$$\Delta \dot{X} = A_{sys} \Delta X \quad (2.2.6)$$

$$A_{sys} = F_X - F_Y G_Y^{-1} G_X \quad (2.2.7)$$

The essential small-disturbance stability characteristics of a structure-preserving model are expressed in terms of eigenproperties of the reduced system matrix  $A_{sys}$ . This matrix is called the dynamic system state matrix.

Eigenvalue analysis of  $A_{sys}$  will give small signal stability information of the current equilibrium point under small disturbances. At voltage collapse, the system loses the ability to supply enough power to a heavily-loaded network. At that point, the so-called saddle node bifurcation occurs, which is described by the movement of one eigenvalue of  $A_{sys}$  on the real axis as it crosses the origin from the left half of the complex plane. Eigenvalue computation can detect this movement. When the Hopf bifurcation occurs, the Jacobian matrix  $A_{sys}$  of the system has a simple pair of purely imaginary eigenvalues and there are no other eigenvalues on the imaginary axis. At the point of Hopf bifurcation, the power system can experience undamped oscillations. Similarly to saddle node bifurcation, Hopf bifurcation can also be identified by tracing the critical eigenvalue.

## Chapter 3 Literature Review

### 3.1 Introduction

The loading margin is defined as the amount of additional load on a specified pattern of load increase that would cause power system instability. In Fig 3.1, the load level at A0, after subtracting the base-case load level, is called the voltage stability loading margin. Similarly, the load level at A1, after subtracting the base-case load level, is called the oscillatory stability loading margin. The load level at A2, after subtracting base-case load level, is called the damping ratio loading margin.

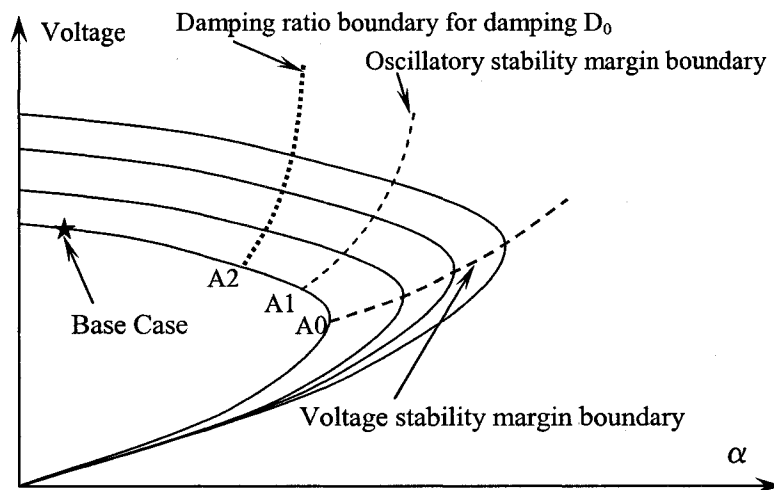


Figure 3.1: Illustration of three types of margin boundaries

The respective margin boundaries (as shown by dashed lines in Fig. 3.1.) can also be obtained by any control parameter change starting with A2, A1 and A0. In the literature various methods are proposed to identify these boundaries. These methods can be broadly classified as direct and indirect methods. The indirect methods can also be divided into

eigenvalue-based methods and the manifold-based method. The following sections will review these methods.

### 3.1.1 The direct method to identify oscillatory stability loading margin

The direct method can identify the oscillatory stability margin directly without computing any intermediate operating points.

Define  $\lambda$  as the eigenvalue of  $A_{sys}$ , and  $v$  represents the eigenvector of  $A_{sys}$ . Then,

$$A_{sys}v = \lambda v \quad (3.1.1)$$

Define the extended eigenvector  $u = -G_Y^{-1}G_X v$ ; the following equation can be obtained.

$$\begin{bmatrix} F_X & F_Y \\ G_X & G_Y \end{bmatrix} \begin{bmatrix} v \\ u \end{bmatrix} = \begin{bmatrix} \lambda v \\ 0 \end{bmatrix} \quad (3.1.2)$$

Reference [16][17] and [18] present the direct method, which rewrites  $u$ ,  $v$  and  $\lambda$  into the form

$$\begin{cases} v = v_R + jv_I; \\ u = u_R + ju_I; \\ \lambda = r + js; \end{cases} \text{ Where } \begin{cases} v_R, v_I \in R^n \\ u_R, u_I \in R^m \\ r, s \in R \end{cases} \quad (3.1.3)$$

At the Hopf bifurcation point, the real part of the dominant (i.e. right-most) eigenvalue is zero, i.e.  $r=0$ .

Then the equation (3.1.2) becomes:

$$\begin{bmatrix} F_X & F_Y \\ G_X & G_Y \end{bmatrix} \begin{bmatrix} v \\ u \end{bmatrix} = \begin{bmatrix} jsv \\ 0 \end{bmatrix} \Rightarrow \begin{cases} F_X v_R + F_Y u_R + sv_I = 0 \\ F_X v_I + F_Y u_I - sv_R = 0 \\ G_X v_R + G_Y u_R = 0 \\ G_X v_I + G_Y u_I = 0 \end{cases} \quad (3.1.4)$$

With

$$\begin{cases} v_{R(i)} = 1 \\ v_{I(i)} = 0 \end{cases} \quad (3.1.5)$$

Where  $v_{R(i)}$  denotes the  $i^{\text{th}}$  element of  $v_R$  vector.

Hopf bifurcation is obtained by solving equations (3.1.4) and (3.1.5) together with system equilibrium equation (3.1.6).

$$\begin{cases} 0 = F(X, Y, \alpha); \\ 0 = G(X, Y, \alpha); \end{cases} \quad (3.1.6)$$

There are  $3(m+n)+2$  independent equations in (3.1.4) (3.1.5) and (3.1.6). There are  $3(m+n)+2$  unknown variables:  $\{X, Y, v_R, v_I, u_R, u_I, \alpha, s\}$ .

References [17] and [18] first applied the direct method in the identification of the power system oscillatory stability margin. Solving of these equations is also complex and sometimes the traditional Newton-based optimization techniques can lead to difficulties or failure. This motivated the effort to find a reliable method to solve these algebraic equations. A genetic algorithm-based solution method was given in reference [19]. All the direct methods require a close initial guess.

### 3.1.2 Eigenvalue based method to identify the oscillatory stability loading margin

Reference [20] presents an eigenvalue-based iterative algorithm that calculates the Hopf bifurcation-related segment of the feasibility boundary for a realistically large power system model. This method needs initial operating points to start the calculation. At the base-case with parameter value  $\alpha^{(0)}$ , by solving the equation (3.1.6), the system state variables  $X$  and  $Y$  are available, represented by  $X^{(0)}$  and  $Y^{(0)}$ . After linearizing the DAE at this operating point, the corresponding Jacobian matrix  $A_{\text{total}}$  is obtained. Then, all the eigenvalues of matrix  $A_{\text{sys}}$  can be calculated. The eigenvalue with maximum real part is selected as the dominant eigenvalue, denoted as  $\lambda^{(0)}$ . If the real part of the dominant eigenvalue is negative, the Hopf bifurcation point is not reached.

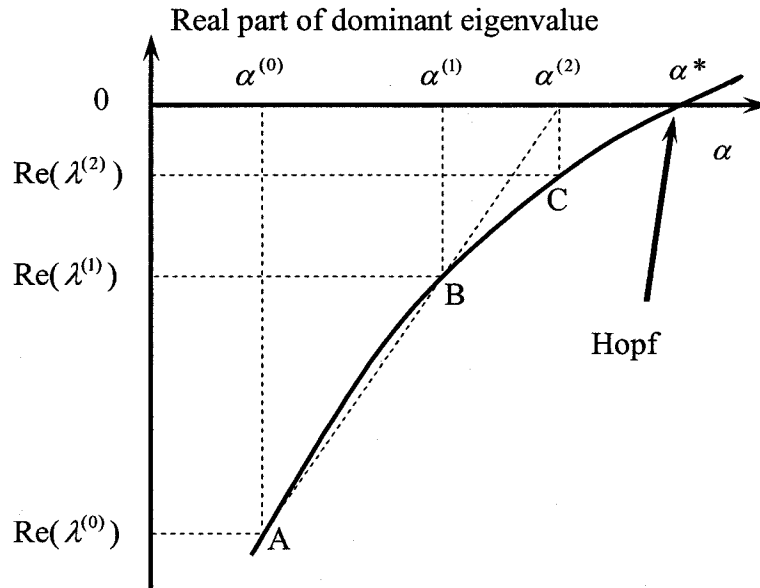


Figure 3.2 The secant method to estimate parameter value

In Fig. 3.2, the vertical axis represents the real part of the dominant eigenvalue; the horizontal axis is the parameter value  $\alpha$ . The smooth curve in Fig. 3.2 is the trajectory of the



real part of the dominant eigenvalue. This trajectory is unknown during the process of Hopf bifurcation identification.  $\alpha^*$  is the parameter value where Hopf bifurcation occurs.

With parameter value  $\alpha^{(0)}$  and the corresponding dominant eigenvalue real part  $\text{Re}(\lambda^{(0)})$ , point A can be located in Fig. 3.2.

Similarly, with another parameter value  $\alpha^{(1)}$  and the corresponding dominant eigenvalue  $\lambda^{(1)}$ , point B can be located in Fig. 3.2. The estimation for  $\alpha^*$  (represented by  $\alpha^{(2)}$ ) can be obtained by using the secant method.

After calculating the eigenvalue at the new estimated operating point  $\alpha^{(2)}$ , point C can be obtained.

Each iteration consists of the parameter estimator and the eigenvalue calculation, also called the eigenvalue corrector. In reference [20], the secant method is used for the parameter estimator. The algorithm for parameter estimator decides the numbers of iterations.

The conventional eigenvalue algorithm calculates all the eigenvalues. Unfortunately, this method is very slow. To speed up the process, references [21] and [22] discuss the robustness and efficiency of existing dominant eigenvalue-computing methods and provide new alternatives. Since only one eigenvalue is calculated each time, the algorithm is very fast.

Reference [23] applies the power method with bilinear transformation to calculate the dominant eigenvalue. In reference [21] the power method is implemented and compared with other dominant eigenvalue-computing algorithms. The author finds that this method belongs to the linear convergence algorithm. Reference [21] also implemented Newton's method, inverse power and Rayleigh quotient iteration, etc. The various algorithms are

compared and evaluated with regards to convergence, performance and applicability. The conclusion of reference [21] points out that linear convergence algorithms, like the power method with bilinear transformation, are more robust than the higher-order methods. However, higher-order methods, such as the quadratic and cubic convergence methods, are much faster. A better result is achieved by combining these two types together. The average iteration number is reduced to approximately 12.

Reference [24] describes new matrix transformations suited to the efficient calculation of the dominant eigenvalue of large-scale power system dynamic models. Since only the most critical eigenvalue is calculated, all the other eigenvalue information is not provided. The critical eigenvalue takes about 8-10 iterations by different transformations.

Reference [25] describes the algorithm which efficiently computes the dominant poles of any specified high-order transfer function. It has the numerical properties of global and ultimately cubic convergence. A numerical example is provided to study low-frequency oscillations in electrical power systems. The transfer function's dominant eigenvalue takes about eleven iterations.

In the large-scale power system simulation, using more than one processor will obviously speed up the calculation. References [23], [26], and [27] present the application of parallel computing in eigenvalue calculation. Parallel processing introduces increased complexity in software and algorithm strategy. Therefore, the task of converting a sequential algorithm into an efficient parallel procedure is always challenging.

Reference [28] presents two sparsity-based eigenvalue techniques for oscillatory stability analysis of large-scale power systems.

### 3.1.3 The manifold-based method to identify oscillatory stability margin

Reference [29] applies  $(A_{total} + A_{total}^T)$  singularity detection to estimate the Hopf bifurcation. This is also called the manifold-based method. It is based on the following proposition:

**Proposition:** Let the maximum eigenvalue of  $(A_{total} + A_{total}^T) = \begin{bmatrix} F_X + F_X^T & F_Y + G_X^T \\ G_X + F_Y^T & G_Y + G_Y^T \end{bmatrix}$  be

$\lambda_1$ , case 1:  $\lambda_1 \geq 0$  is the necessary condition for Hopf bifurcation associated with the power system DAE model  $A_{sys}$ .

Case 2: When  $A_{total}$  is approximate to a normal matrix,  $\lambda_1 = 0$  becomes the necessary condition.

From this proposition, the method provides a conservative estimation for the Hopf bifurcation point. However, it can not provide the damping ratio and other relevant information related to oscillatory stability.

### 3.2 Damping ratio margin

Damping margin can be defined as the amount of additional load on a specified pattern of load increase that would cause the damping ratio reach its minimum limit. In the oscillatory stability assessment, the damping ratio margin needs to be checked for each contingency and scenario in order to keep the system far away from the minimum damping limit.

In the WSCC, reference [14] recommends the following criteria to determine the safe operating limits:

The operating point is acceptable from the damping standpoint if:

With path flow increased by the larger of 100 MW or 5 percent, any N-1 contingency will not result in undamped oscillations or instability;

With the path flow increased by the larger of 100 MW or 2 1/2 percent, any N-2 contingency will not result in undamped oscillations or instability.

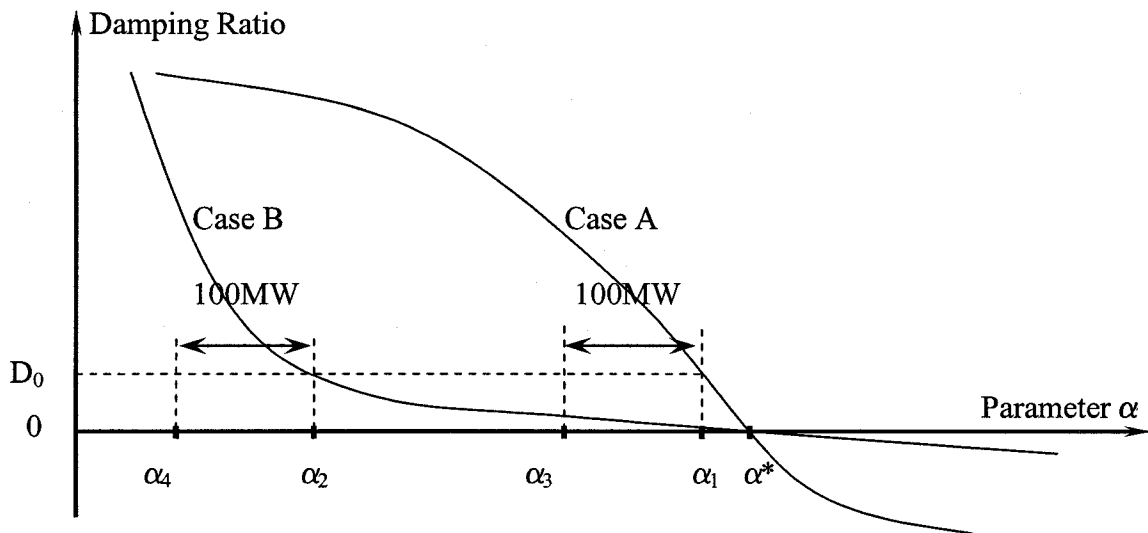


Figure 3.3 Two cases with different damping ratio margins and the same oscillatory stability

Fig. 3.3 shows two cases with same oscillatory stability margin  $\alpha^*$ . If the given minimum damping ratio limit is  $D_0$ , the damping margin will be  $\alpha_1$  and  $\alpha_2$  for case A and case B, respectively. Using this information, in case B the system can only be operated below  $\alpha_4$ . In case A, the system is safe when the parameter is less than  $\alpha_3$ .

Different utilities may have different minimum damping ratio limits. The eigenvalue tracking technique can provide the entire relation curve between the parameter value and the dominant eigenvalue's damping ratio.

Reference [30] mentions that the direct method could be used for computing the damping ratio margin. The numerical difficulties inherent in the direct method for the computation of the stability margin are still present in this case.

In section 4.3, a new approach for calculation of the damping ratio margin is presented with the indirect method. Actually, the new approach is a by-product of eigenvalue tracing. If the eigenvalue tracing method is used in the calculation of oscillatory stability margin, the damping ratio margin can be obtained very quickly.

### **3.3 Oscillatory stability and damping margin boundaries tracing**

In the security assessment, control actions are needed to maintain a given oscillatory stability and also damping ratio margins. How to find the optimal control parameter value or strategy becomes very important. With the margin boundary tracing, one can find the oscillatory stability and damping margins without tracing the entire P-V curve. Reference [31] presents a framework based on a differential manifold approach that combines identification and tracing of both saddle node and Hopf bifurcation margin boundaries. Compared with margin sensitivity, margin boundary tracing provides an actual estimate of the margin by considering all the nonlinear factors. It can provide more accurate information than the margin sensitivity.

Reference [31] doesn't need to trace entire P-V curve, but it still needs to trace P-V curve around the margin point. This will result in some extra power flow solutions. In the proposed method, tracing P-V curve will be totally avoided. Also reference [31] can not trace damping margin boundaries.

Up to this point, we discussed various existing methods (and their different drawbacks) that deal with oscillatory stability margin identification and boundary tracing. Most of the existing literature is concentrated on the identification of oscillatory stability margin. Only reference [31] makes an attempt to trace the oscillatory stability boundary.

## Chapter 4 Oscillatory stability margin and damping margin identification

### 4.1 Oscillatory stability margin identification

To identify the oscillatory stability margin, eigenvalue information of linearized DAE is needed. One of the major contributions of this thesis is to propose an eigenvalue tracking method to obtain eigenvalue related information including sensitivities. A derivative of the eigenvalues which is the by product of this approach can be used to identify the oscillatory stability margin information.

The eigenvalue-tracking method involves a set of differential equations. The derivative in the differential equation denotes the differentiation of the eigenvalue and eigenvector with respect to the system parameter. By integrating in the parameter domain, the curve of the eigenvalue and eigenvector vs. the parameter value can be obtained. Thus, the complete information about how the traced eigenvalue approaches and crosses the imaginary axis is obtained.

References [32], [33] provide an approach for developing the eigenvalue and eigenvector differential equation for the parameterized matrix  $J(\alpha)$ . For any eigenvalue  $\lambda$  of  $J(\alpha)$ , and corresponding eigenvector  $v$ , the following relation is well known.

$$\begin{cases} J(\alpha)v = \lambda v \\ v^T v = 1 \end{cases} \quad (4.1.1)$$

If you differentiate (4.1.1) with respect to  $\alpha$ , then

$$\begin{aligned}
& \begin{cases} \frac{dJ}{d\alpha} v + J \frac{dv}{d\alpha} = \frac{d\lambda}{d\alpha} v + \lambda \frac{dv}{d\alpha} \\ \frac{dv^T}{d\alpha} v + v^T \frac{dv}{d\alpha} = 0 \end{cases} \\
& \Rightarrow \begin{cases} \frac{d\lambda}{d\alpha} v + \lambda \frac{dv}{d\alpha} - J \frac{dv}{d\alpha} = \frac{dJ}{d\alpha} v \\ \frac{dv^T}{d\alpha} v + v^T \frac{dv}{d\alpha} = 0 \end{cases} \\
& \Rightarrow \begin{cases} \frac{d\lambda}{d\alpha} v + \lambda \frac{dv}{d\alpha} - J \frac{dv}{d\alpha} = \frac{dJ}{d\alpha} v \\ \frac{dv^T}{d\alpha} v = 0 \end{cases} \tag{4.1.2} \\
& (\because \frac{dv^T}{d\alpha} v = \sum_{i=1}^n \frac{dv_i}{d\alpha} v_i = v^T \frac{dv}{d\alpha})
\end{aligned}$$

By rearranging the terms in (4.1.2) we get the derivatives of  $v$  and  $\lambda$  with respect to  $\alpha$  as shown in equation (4.1.3)

$$\begin{bmatrix} (\lambda I - J) & v \\ v^T & 0 \end{bmatrix} \begin{bmatrix} \dot{v} \\ \dot{\lambda} \end{bmatrix} = \begin{bmatrix} j_v \\ 0 \end{bmatrix} \tag{4.1.3}$$

where a dot denotes differentiation with respect to  $\alpha$ .

Here  $J$  depends explicitly on  $\alpha$ . In power systems,  $J$  corresponds to the Jacobian matrix. However the Jacobian elements in power systems are not explicitly expressed in terms of  $\alpha$ .

In the differential and algebraic equations (DAE) model of a power system:

$$\begin{cases} \dot{X} = F(X, Y, \alpha) \\ 0 = G(X, Y, \alpha) \end{cases}$$



$X$  and  $Y$  are state and algebraic variable vectors respectively.  $\alpha$  is the parameter to represent load level of the entire system. The Jacobian Matrix of power system DAE model becomes

$$A_{total} = \begin{bmatrix} F_X(\alpha) & F_Y(\alpha) \\ G_X(\alpha) & G_Y(\alpha) \end{bmatrix}$$

$$\text{Thus } A_{sys} = F_X(\alpha) - F_Y(\alpha)G_Y^{-1}(\alpha)G_X(\alpha)$$

When a complex pair of eigenvalues of  $A_{sys}$  crosses the imaginary axis, the system becomes oscillatory unstable.

If  $\lambda$  is an eigenvalue of  $A_{sys}$  and  $v$  denotes the corresponding eigenvector. Then,

$$A_{sys}v = \lambda v$$

Then (4.1.1) will become

$$\begin{cases} A_{sys}(\alpha)v = \lambda v \\ v^T v = 1 \end{cases}$$

$\lambda$  and  $v$  are the eigenvalue and the right eigenvector that we want to trace. To preserve the sparsity, an extended eigenvector is defined as  $u = -G_Y^{-1}(\alpha)G_X(\alpha)v$ . Then,

$$\begin{cases} F_X v + F_Y u = \lambda v & (4.1.4-1) \end{cases}$$

$$\begin{cases} G_X v + G_Y u = 0 & (4.1.4-2) \end{cases}$$

By substituting  $u = -G_Y^{-1}G_X v$  from (4.1.4-2) into (4.1.4-1), we get  $(F_X - F_Y G_Y^{-1} G_X)v = \lambda v$ . If we let  $J(\alpha) = F_X(\alpha) - F_Y(\alpha)G_Y^{-1}(\alpha)G_X(\alpha)$ , and substitute  $J(\alpha)$  into (4.1.3), we obtain

$$\begin{bmatrix} \lambda I - (F_X - F_Y G_Y^{-1} G_X) & \mathbf{v} \\ \mathbf{v}^T & 0 \end{bmatrix} \begin{bmatrix} \dot{\mathbf{v}} \\ \dot{\lambda} \end{bmatrix} = \begin{bmatrix} \frac{d(F_X - F_Y G_Y^{-1} G_X)}{d\alpha} \mathbf{v} \\ 0 \end{bmatrix} \quad (4.1.5)$$

The part of (4.1.5) can be derived as:

$$\begin{aligned} & \frac{d(F_X - F_Y G_Y^{-1} G_X)}{d\alpha} \mathbf{v} \\ &= \dot{F}_X \mathbf{v} + \frac{d(-F_Y G_Y^{-1} G_X)}{d\alpha} \mathbf{v} \\ &= \dot{F}_X \mathbf{v} - \dot{F}_Y G_Y^{-1} G_X \mathbf{v} - F_Y \frac{d(G_Y^{-1} G_X)}{d\alpha} \mathbf{v} \end{aligned} \quad (4.1.6)$$

From  $u = -G_Y^{-1} G_X \mathbf{v}$ , we can get  $\dot{u} = -\frac{d(G_Y^{-1} G_X)}{d\alpha} \mathbf{v} - G_Y^{-1} G_X \dot{\mathbf{v}}$ . So,

$$\frac{d(G_Y^{-1} G_X)}{d\alpha} \mathbf{v} = -\dot{u} - G_Y^{-1} G_X \dot{\mathbf{v}} \quad (4.1.7)$$

Substitute (4.1.7) into (4.1.6),

$$\frac{d(F_X - F_Y G_Y^{-1} G_X)}{d\alpha} \mathbf{v} = \dot{F}_X \mathbf{v} - \dot{F}_Y G_Y^{-1} G_X \mathbf{v} + F_Y (\dot{u} + G_Y^{-1} G_X \dot{\mathbf{v}}) \quad (4.1.8)$$

Substitute (4.1.8) into (4.1.5),

$$\begin{cases} (\lambda I - F_X + F_Y G_Y^{-1} G_X) \dot{\mathbf{v}} + \mathbf{v} \dot{\lambda} = \dot{F}_X \mathbf{v} - \dot{F}_Y G_Y^{-1} G_X \mathbf{v} + F_Y (\dot{u} + G_Y^{-1} G_X \dot{\mathbf{v}}) & (4.1.9-1) \\ \mathbf{v}^T \dot{\mathbf{v}} = 0 & (4.1.9-2) \end{cases}$$

In order to apply a sparse matrix technique, we can't let  $G_Y^{-1}$  show up in explicit form. So, we have to let  $u$  also enter the state variable. Substituting  $u = -G_Y^{-1} G_X \mathbf{v}$  into (4.1.9-1)

$$\begin{cases} (\lambda I - F_X)\dot{v} - F_Y\dot{u} + v\dot{\lambda} = \dot{F}_X v + \dot{F}_Y u \\ -G_X\dot{v} - G_Y\dot{u} = \dot{G}_X v + \dot{G}_Y u \end{cases} \quad (4.1.10)$$

With (4.1.10) and (4.1.9-2), we can derive the following differential equation, similar to (4.1.3)

$$\begin{bmatrix} \lambda I - F_X & -F_Y & v \\ -G_X & -G_Y & 0 \\ v^T & 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{v} \\ \dot{u} \\ \dot{\lambda} \end{bmatrix} = \begin{bmatrix} \dot{F}_X v + \dot{F}_Y u \\ \dot{G}_X v + \dot{G}_Y u \\ 0 \end{bmatrix} \quad (4.1.11)$$

Since  $(\lambda(\alpha), v(\alpha), u(\alpha))^T$  is in  $C \times C^n \times C^m$ , (4.1.11) is a complex differential equation.

Define

$$\begin{cases} v = v_R + jv_I; \\ u = u_R + ju_I; \text{ where} \\ \lambda = \lambda_R + j\lambda_I; \end{cases} \begin{cases} v_R, v_I \in R^n \\ u_R, u_I \in R^m \\ \lambda_R, \lambda_I \in R \end{cases}$$

With the above notation, equation (4.1.11) can be further extended to the following real differential equation form.

$$\begin{bmatrix} \lambda_R I - F_X & -\lambda_I I & -F_Y & 0 & v_R & -v_I \\ \lambda_I I & \lambda_R I - F_X & 0 & -F_Y & v_I & v_R \\ -G_X & 0 & -G_Y & 0 & 0 & 0 \\ 0 & -G_X & 0 & -G_Y & 0 & 0 \\ v_R^T & -v_I^T & 0 & 0 & 0 & 0 \\ v_I^T & v_R^T & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{v}_R \\ \dot{v}_I \\ \dot{u}_R \\ \dot{u}_I \\ \dot{\lambda}_R \\ \dot{\lambda}_I \end{bmatrix} = \begin{bmatrix} \dot{F}_X v_R + \dot{F}_Y u_R \\ \dot{F}_X v_I + \dot{F}_Y u_I \\ \dot{G}_X v_R + \dot{G}_Y u_R \\ \dot{G}_X v_I + \dot{G}_Y u_I \\ 0 \\ 0 \end{bmatrix} \quad (4.1.12)$$

The above formulation leads to an algorithm for tracing any eigenvalue of a parameterized matrix. This formulation also retains the sparsity.

$\dot{F}_{Y(ij)}$  is the element at  $i^{\text{th}}$  row,  $j^{\text{th}}$  column in matrix  $\dot{F}_Y$ . The  $\dot{F}_Y$  can be computed element by element with (4.1.13). Similarly,  $\dot{F}_X$ ,  $\dot{G}_X$  and  $\dot{G}_Y$  can be obtained.

$$\dot{F}_{Y(ij)} = \frac{\partial F_{Y(ij)}}{\partial X} \frac{\partial X}{\partial \alpha} + \frac{\partial F_{Y(ij)}}{\partial Y} \frac{\partial Y}{\partial \alpha} + \frac{\partial F_{Y(ij)}}{\partial \alpha} \quad (4.1.13)$$

The above nonlinear differential equation leads to an algorithm for tracing any eigenvalue and corresponding eigenvector of a parameterized matrix, which retains sparsity. The state variables of this nonlinear differential equation are  $[v_R, v_I, u_R, u_I, \lambda_R, \lambda_I]$ . In the eigenvalue-tracing method, at each operating point the eigenvalue information is obtained through numerical integration. The dimension of the linear equation to be solved is  $2(n+m+1)$ .

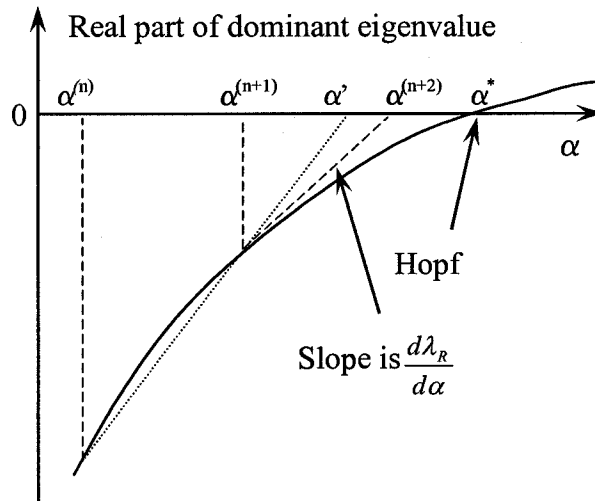


Figure 4.1 The tangent information is used for predicting the next step length

In order to reduce the number of steps to a minimum, derivative  $\frac{d\lambda_R}{d\alpha}$  can be used to calculate the step length for the integration.

In the Newton method, the  $\alpha^{(n+2)}$  is predicted by  $\alpha^{(n+2)} = \alpha^{(n+1)} - \frac{\lambda_R^{(n+1)}}{d\lambda_R^{(n+1)}/d\alpha}$ . (See Fig. 4.1)

The integration step length can be obtained with  $\alpha^{(n+2)} - \alpha^{(n+1)}$ . By the eigenvalue tracking method, the dominant eigenvalue at  $\alpha^{(n+1)}$  can be obtained.

With the same initial operating point  $\alpha^{(n)}$  and  $\alpha^{(n+1)}$ , the secant method [20] will predict the parameter value  $\alpha'$  (in Fig. 4.1) as well as the next step length. In Fig. 4.1, the Newton method estimated parameter value  $\alpha^{(n+2)}$  is closer to  $\alpha^*$  than  $\alpha'$ . Thus, the Newton method can reduce the number of iteration in the oscillatory stability margin identification.

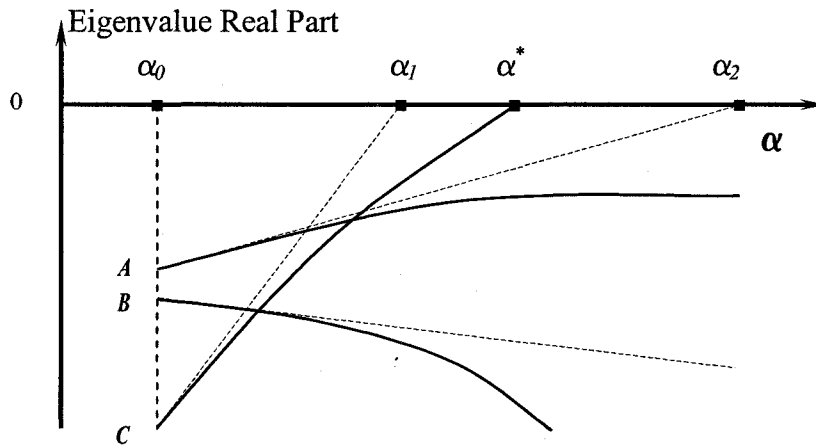


Fig. 4.2 Eigenvalue real part vs. parameter  $\alpha$

From the above formulation (4.1.12), one can trace all the eigenvalues or any specified subset or a single eigenvalue of interest. To detect the Hopf bifurcation, we are

interested in the complex eigenvalue which crosses the imaginary axis first. To identify this eigenvalue an index is derived.

Fig 4.2 shows a conceptual variation of real parts of three eigenvalues with respect to parameter  $\alpha$ . With the equation (4.1.12), any eigenvalue real part derivative can be obtained. The dashed lines in Fig.4.2 indicate the slopes at base case parameter value of  $\alpha_0$ . This information is used to define an index to estimate which eigenvalue will cross the imaginary axis first. This index is given by equation (4.1.13)

$$Index_{Eigen} = -\frac{\lambda_R}{d\lambda_R/d\alpha} = -\frac{\lambda_R}{\dot{\lambda}_R} \quad (4.1.13)$$

If we assume all the eigenvalues at the base case are on the left half of the complex plane, then a positive index means the corresponding eigenvalue moves towards the imaginary axis and negative means it moves away from the imaginary axis. The positive index value indicates, if  $\alpha$  value increases by this index value, it will reach the imaginary axis with the current speed. The negative index value means, if  $\alpha$  value decreases by this index value, it will reach the imaginary axis with the current speed. Thus, a low positive index value relative to others indicates the corresponding eigenvalue is critical and may be the first one to cross the imaginary axis.

In Fig.4.2, the eigenvalues corresponding to points A and C have positive index value, where as point B has a negative index value. Between A and C, C has a less positive index value. In this particular scenario, the Hopf bifurcation is more likely to be identified by tracing the real part of eigenvalue C.

The fourth order integration eigenvalue tracing method to search Hopf bifurcation includes the following steps:

1) Compute the equilibrium point for the DAE model. Linearize the system and calculate  $A_{\text{sys}}$  matrix and its eigenvalue by conventional method. Compute the index for all complex eigenvalues by (4.1.14). Then, rank them by index value. The number one ranked eigenvalue is the one we will trace. Then let the iteration number  $n = 1$  and step length at its maximum limit. Then, enter following loop.

2) Solve linear equation (4.1.12) to compute

$K1 = [\dot{v}_R^{(n)T}, \dot{v}_I^{(n)T}, \dot{u}_R^{(n)T}, \dot{u}_I^{(n)T}, \dot{\lambda}_R^{(n)}, \dot{\lambda}_I^{(n)}]^T$ , where  $n$  denotes the iteration number.

3) Let  $[v_R^{(temp)T}, v_I^{(temp)T}, u_R^{(temp)T}, u_I^{(temp)T}, \lambda_R^{(temp)}, \lambda_I^{(temp)}]^T = [v_R^{(n)T}, v_I^{(n)T}, u_R^{(n)T}, u_I^{(n)T}, \lambda_R^{(n)}, \lambda_I^{(n)}]^T +$

$K1 \cdot \frac{h}{2}$ , where  $h$  is the step length for parameter  $\alpha$ .

4) Increase the load level to  $\alpha^{(n+\frac{1}{2})} = \alpha^{(n)} + \frac{h}{2}$ ; then compute the equilibrium

for  $\begin{bmatrix} X^{(n+\frac{1}{2})} \\ Y^{(n+\frac{1}{2})} \\ \alpha^{(n+\frac{1}{2})} \end{bmatrix}$ .

Solve the linear equation (4.1.12) to compute

$K2 = [\dot{v}_R^{(n+\frac{1}{2})T}, \dot{v}_I^{(n+\frac{1}{2})T}, \dot{u}_R^{(n+\frac{1}{2})T}, \dot{u}_I^{(n+\frac{1}{2})T}, \dot{\lambda}_R^{(n+\frac{1}{2})}, \dot{\lambda}_I^{(n+\frac{1}{2})}]^T$ .

Use  $[\mathbf{v}_R^{(temp1)T}, \mathbf{v}_I^{(temp1)T}, \mathbf{u}_R^{(temp1)T}, \mathbf{u}_I^{(temp1)T}, \lambda_R^{(temp1)}, \lambda_I^{(temp1)}]^T$  as eigen-information for (4.1.12).

5) Let  $[\mathbf{v}_R^{(temp2)T}, \mathbf{v}_I^{(temp2)T}, \mathbf{u}_R^{(temp2)T}, \mathbf{u}_I^{(temp2)T}, \lambda_R^{(temp2)}, \lambda_I^{(temp2)}]^T = [\mathbf{v}_R^{(n)T}, \mathbf{v}_I^{(n)T}, \mathbf{u}_R^{(n)T}, \mathbf{u}_I^{(n)T}, \lambda_R^{(n)}, \lambda_I^{(n)}]^T + \mathbf{K2} \cdot \frac{h}{2}$ , and solve linear equation (4.1.12) again to

compute

$$\mathbf{K3} = [\dot{\mathbf{v}}_R^{(n+\frac{1}{2})T}, \dot{\mathbf{v}}_I^{(n+\frac{1}{2})T}, \dot{\mathbf{u}}_R^{(n+\frac{1}{2})T}, \dot{\mathbf{u}}_I^{(n+\frac{1}{2})T}, \dot{\lambda}_R^{(n+\frac{1}{2})}, \dot{\lambda}_I^{(n+\frac{1}{2})}]^T$$

Use  $[\mathbf{v}_R^{(temp2)T}, \mathbf{v}_I^{(temp2)T}, \mathbf{u}_R^{(temp2)T}, \mathbf{u}_I^{(temp2)T}, \lambda_R^{(temp2)}, \lambda_I^{(temp2)}]^T$  as eigen-information.

6) Let  $[\mathbf{v}_R^{(temp3)T}, \mathbf{v}_I^{(temp3)T}, \mathbf{u}_R^{(temp3)T}, \mathbf{u}_I^{(temp3)T}, \lambda_R^{(temp3)}, \lambda_I^{(temp3)}]^T = [\mathbf{v}_R^{(n)T}, \mathbf{v}_I^{(n)T}, \mathbf{u}_R^{(n)T}, \mathbf{u}_I^{(n)T}, \lambda_R^{(n)}, \lambda_I^{(n)}]^T + \mathbf{K3} \cdot h$ .

Let  $\alpha^{(n+1)} = \alpha^{(n)} + h$ , and compute the equilibrium for load level  $\alpha^{(n+1)}$  to get

$$\begin{bmatrix} X^{(n+1)} \\ Y^{(n+1)} \\ \alpha^{(n+1)} \end{bmatrix}$$

Solve linear equation (4.1.12) again to compute

$$\mathbf{K4} = [\dot{\mathbf{v}}_R^{(n+1)T}, \dot{\mathbf{v}}_I^{(n+1)T}, \dot{\mathbf{u}}_R^{(n+1)T}, \dot{\mathbf{u}}_I^{(n+1)T}, \dot{\lambda}_R^{(n+1)}, \dot{\lambda}_I^{(n+1)}]^T$$

$[\mathbf{v}_R^{(temp3)T}, \mathbf{v}_I^{(temp3)T}, \mathbf{u}_R^{(temp3)T}, \mathbf{u}_I^{(temp3)T}, \lambda_R^{(temp3)}, \lambda_I^{(temp3)}]^T$  is used as eigen-information.

7) Integrate the eigenvalue and eigenvector as

$$[\mathbf{v}_R^{(n+1)T}, \mathbf{v}_I^{(n+1)T}, \mathbf{u}_R^{(n+1)T}, \mathbf{u}_I^{(n+1)T}, \lambda_R^{(n+1)}, \lambda_I^{(n+1)}]^T$$



$$= [v_R^{(n)T}, v_I^{(n)T}, u_R^{(n)T}, u_I^{(n)T}, \lambda_R^{(n)}, \lambda_I^{(n)}]^T + \frac{(K1 + 2 \times K2 + 2 \times K3 + K4)}{6} \cdot h$$

8) If the integrated eigenvalue real part is still negative and the step length is its maximum limit, then go to step 2). Otherwise, go to step 9).

9) If  $|\lambda_{\text{Re}}^{(n+1)}| < 10^{-5}$ , terminate the loop and output.

Otherwise, compute K1 similar in step 2). Then, let the step length

$$h^{(n+1)} = -\frac{\lambda_{\text{Re}}^{(n+1)}}{d\lambda_{\text{Re}}^{(n+1)}/d\alpha} = -\frac{\lambda_{\text{Re}}^{(n+1)}}{\lambda_{\text{Re}}^{(n+1)}} \cdot n = n+1, \text{ go to step 3)}$$

In some complex situations, the eigenvalue index may not always be able to identify which eigenvalue to trace. In Section 4.5, a methodology is proposed to handle this situation.

## 4.2 Simulation result for oscillatory stability margin identification

The algorithm is tested on a New England 39-bus, 10-generator system with two-axis generator model, IEEE DC-I excitation system and governor model, which can be found in reference [3][13]. There are 9 state variables for each generator. The total system Jacobian dimension is 167×167 (89 state variables and 78 algebraic variables). Load consists of 50% constant power, 30% constant current, and 20% constant impedance. Load on all buses will increase with the same percentage. The sum of the initial load on all buses is 6141.3MW. The New England system detailed data can be found in Appendix 1.

The total number of eigenvalues at any given operating equilibrium is 89. Out of which 41 are real and 24 are complex pairs.

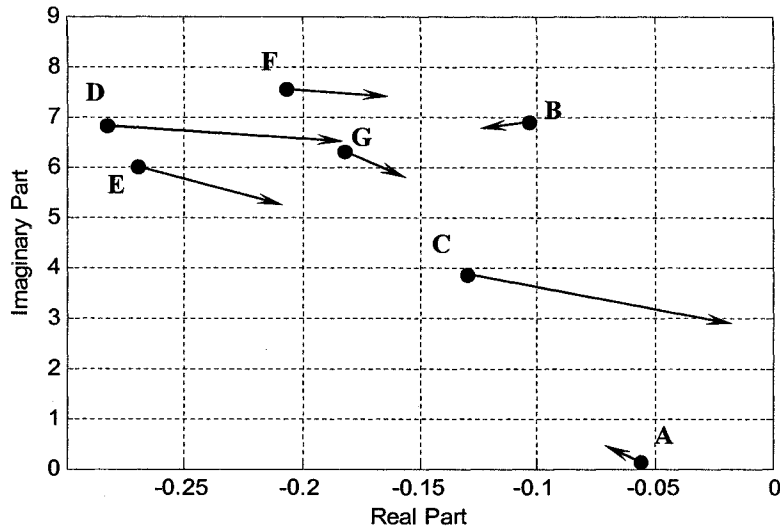


Figure 4.3 Eigenvalues in complex plane at base case

Fig4.3 shows eigenvalues with a real part greater than -0.3 in the upper-half of the complex plane for the base case (total load: 6141.3MW). With (4.1.12), each eigenvalue's  $\dot{\lambda}_R$  and  $\dot{\lambda}_I$  can be calculated. The arrows show the moving direction of each eigenvalue when parameter  $\alpha$  increases. The length of the arrow shows the speed of movement with respect to  $\alpha$ .

Table 4.1 Index of all complex eigenvalues in Fig 4.3

Rank No	Symbol	Index Value	Eigenvalue
1	C	0.49052	$-0.12929 \pm j3.84574$
2	D	0.68478	$-0.28288 \pm j6.80807$
4	E	1.83944	$-0.26939 \pm j5.97408$
6	F	2.26469	$-0.20663 \pm j7.51990$
10	G	9.19457	$-0.18181 \pm j6.28264$
	A	-22.5297	$-0.05609 \pm j0.10762$
	B	-0.89964	$-0.10300 \pm j6.87668$

Table 4.1 shows the index value for all the complex eigenvalues in Fig 4.3. The complex eigenvalues A and B have negative index values. We included these two in the table

to show the importance of the index. These two eigenvalues are very close to the imaginary axis compared to any other ranked eigenvalues. However they are moving away from the imaginary axis for increasing load. By using the index we eliminated these two for further tracing.

The ranked number one eigenvalue was traced through equation (3.1.2) by fourth order Runge-Kutta method. To prevent error caused by too large an integration step length, 0.08 is used as the step length maximum limit for parameter  $\alpha$ . It represents the entire system load increase of 8%. Table 4.2 shows each iteration result during the integration. At each intermediate operating point, a MATLAB provided function is used to obtain the actual eigenvalue for comparison.

**Table 4.2: Hopf bifurcation search for each iteration**

Iter No.	$\alpha$ value	load level (MW)	Integrated $\lambda_R$
0	0	0	-0.12929
1	0.08	491.3	-0.10017
2	0.16	982.61	-0.03869
3	0.19405	1191.72	7.96E-03
4	0.18916	1161.69	1.95E-04
5	0.18903	1160.89	1.27E-07

In Table 4.2, the integrated eigenvalue real part and its actual value are shown. The corresponding error is listed. In the first two steps, the parameter  $\alpha$  is increased by 0.08, which corresponds to  $6141.3\text{MW} \times 8\% = 491.3\text{MW}$ . From Table 4.2, we can see that the accumulated error is well under control. Fig 4.4 shows the eigenvalue's position for every step in the complex plane.

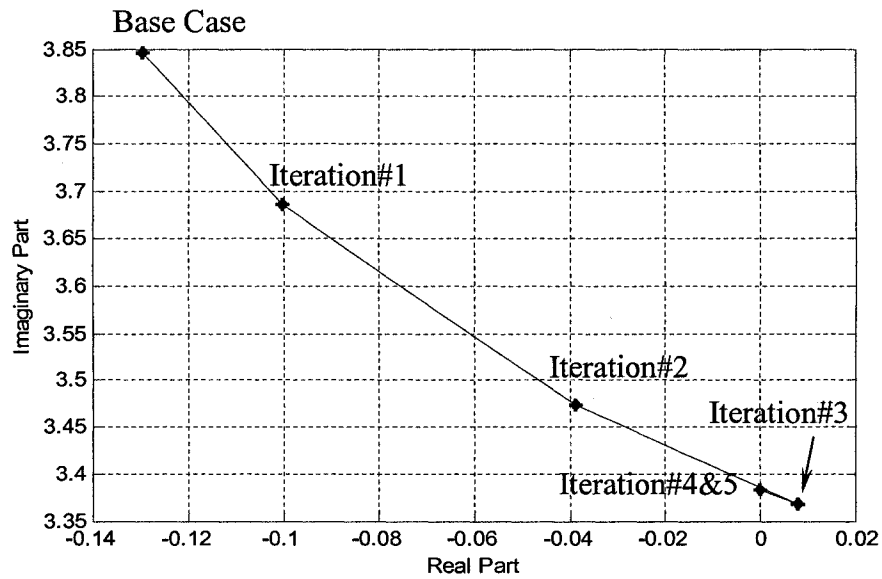


Fig. 4.4 Integrated eigenvalue in complex plane

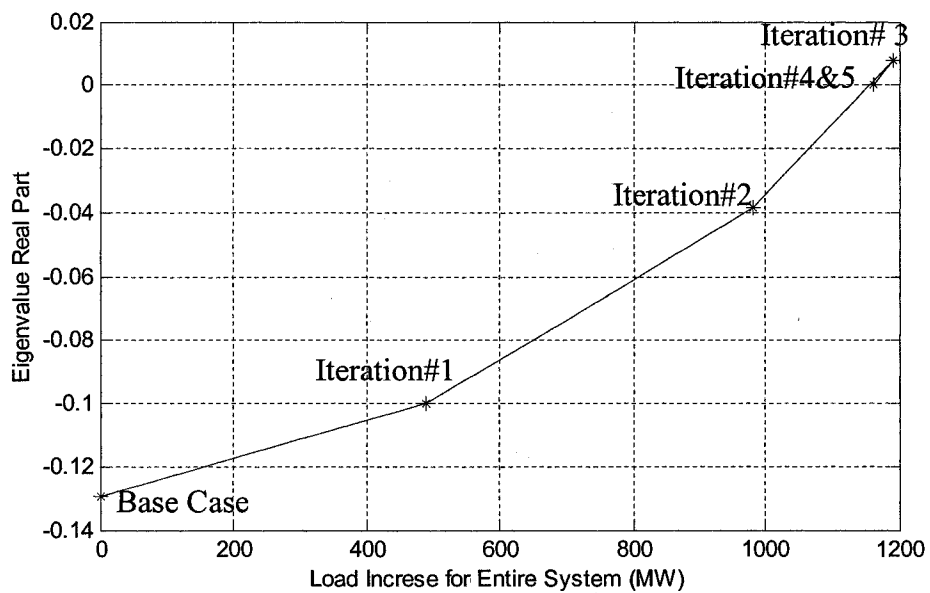


Fig. 4.5 Integrated eigenvalue with total load increase

Fig4.5 shows the eigenvalue's real part with the parameter  $\alpha$ . The eigenvalue position in Steps 4 and 5 are very close.

With the MATLAB provided eigenvalue calculation method, the exact Hopf bifurcation appears at a load increase of 18.90605%. This is equal to  $6141.3\text{MW} \times 18.90605\% = 1161.078\text{MW}$  load increase. This compares well with our value of  $18.90341\% \times 6141.3\text{MW} = 1160.915\text{MW}$ . The error is  $0.163\text{MW}$ .

For each load increase step, we have to solve  $2(m+n)+2$  dimension linear equation four times.

### 4.3 Damping margin identification

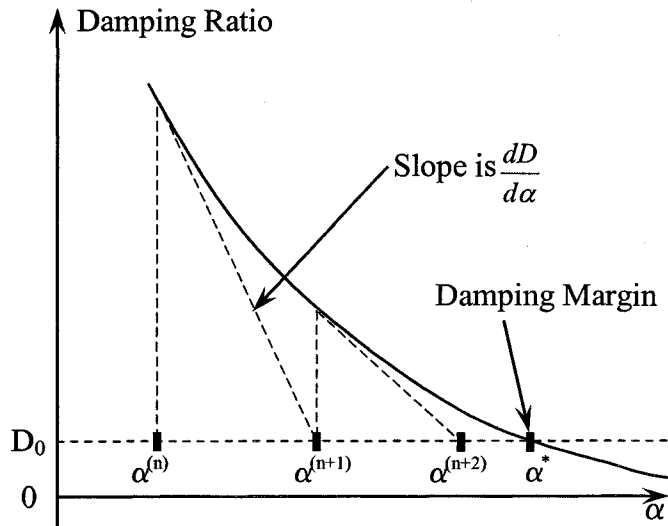


Figure 4.6 The tangent information is used for searching damping margin

The damping ratio is defined as 
$$D = -\cos[\arctan(\frac{\lambda_I}{\lambda_R})] \quad (4.3.1)$$

Since  $\lambda_R$  and  $\lambda_I$  are available, the damping ratio's derivative to parameter  $\alpha$  can be derived and is given by (4.3.2)

$$\frac{dD}{d\alpha} = \sin(\arctan(\frac{\lambda_I}{\lambda_R})) \frac{(\lambda_R \frac{d\lambda_I}{d\alpha} - \lambda_I \frac{d\lambda_R}{d\alpha})}{(\lambda_I^2 + \lambda_R^2)} \quad (4.3.2)$$

The damping index can be obtained by

$$Index_{damping} = -\frac{D - D_0}{dD/d\alpha}$$

Following a similar approach to Hopf identification, the damping ratio of any eigenvalue can be traced with a given damping threshold  $D_0$ . Fig 4.6 shows the process of searching a damping margin.

The slope of the damping ratio curve in Fig 4.6 can be used to predict a parameter value in damping margin identification.

#### 4.4 Simulation result for damping margin identification

Similar to oscillatory stability margin identification, the damping margin identification is also obtained. Table 4.3 shows the simulation results for a damping ratio limit ( $D_0$ ) of 1%. Here the same ranked number one eigenvalue damping ratio is traced.

**Table 4.3: Damping ratio search for each iteration**

Iter No.	$\alpha$ value	Total Load Increase(MW)	Integrated Damping
0	0	0	0.0336
1	0.08	491.30	0.0272
2	0.16	982.61	0.0111
3	0.16358	1004.57	9.97 E-03
4	0.16350	1004.09	9.99999E-03

Figure 4.7 shows the each iteration's load increase and damping ratio.

If we compare the oscillatory stability margin with the damping margin 1004.086MW, there is a 156.992MW (1161.078MW -1004.086MW) difference. When we use  $D_0=0\%$ , the corresponding damping margin becomes the oscillatory stability margin.

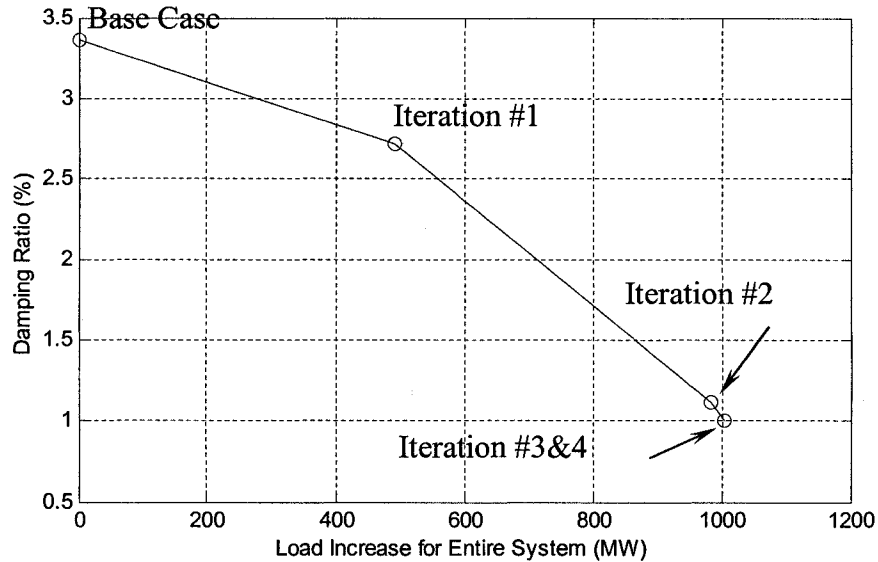


Fig. 4.7 Damping ratio vs. entire system load increase

## 4.5 Discussion of the practical aspects of the proposed algorithm

### 4.5.1 Consideration for cases if the eigenvalue index doesn't work

In the previous section we calculated all the eigenvalues and their indices to trace any specific eigenvalue of interest. In general we do not need to calculate all the eigenvalues. In the previous example the number 1 ranked eigenvalue at the base case is finally crossing the imaginary axis first. However this may not always be true for all operating conditions. Flow chart in Fig. 4.8 describes the steps involved for any general starting point.

First we calculate a subset of eigenvalues with a maximum real part for the base case. As discussed in the introduction, powerful methods are proposed in the literature [21],

[22], [24], [25], [28], [34] to calculate these dominant eigenvalues. Next, the indices for this subset of eigenvalues are calculated. The eigenvalue with minimum positive index in this subset will be traced. During every step of load increase, the eigenvalue moving direction is checked. If  $\frac{d\lambda_R}{d\alpha}$  becomes negative, it indicates that the eigenvalue is moving towards the left hand side in the complex plane. We need to stop tracing this eigenvalue. At this load, calculate another subset of eigenvalues with the maximum real part. Pick an eigenvalue with minimum positive index and trace this new eigenvalue.

When the traced eigenvalue reaches the imaginary axis, it satisfies the minimum tolerance criteria. At this operating point we have to check if some other eigenvalues crossed the imaginary axis already. For this we have to find an eigenvalue with maximum real part. If this eigenvalue is the same as the one we traced before, we find the Hopf and stop. Otherwise, trace the new eigenvalue with the maximum real part backward, until it comes back to the imaginary axis. Our simulation experiments indicated, in a stressed system, the eigenvalue with a minimum positive index will cross the imaginary axis in the first attempt. So, we don't have to waste time calculating an eigenvalue with a maximum real part multiple times. Even if the operating point is far away from the Hopf point, this method took less number of maximum real part calculation updates as described below.

We simulated an operating point which is far away from the Hopf point. In the previous example the total load of the base case is reduced by 50%. At this reduced load, an eigenvalue with the maximum real part is calculated. This eigenvalue turns out to be the same eigenvalue A as in Fig.4.3. However the index for this eigenvalue is negative.



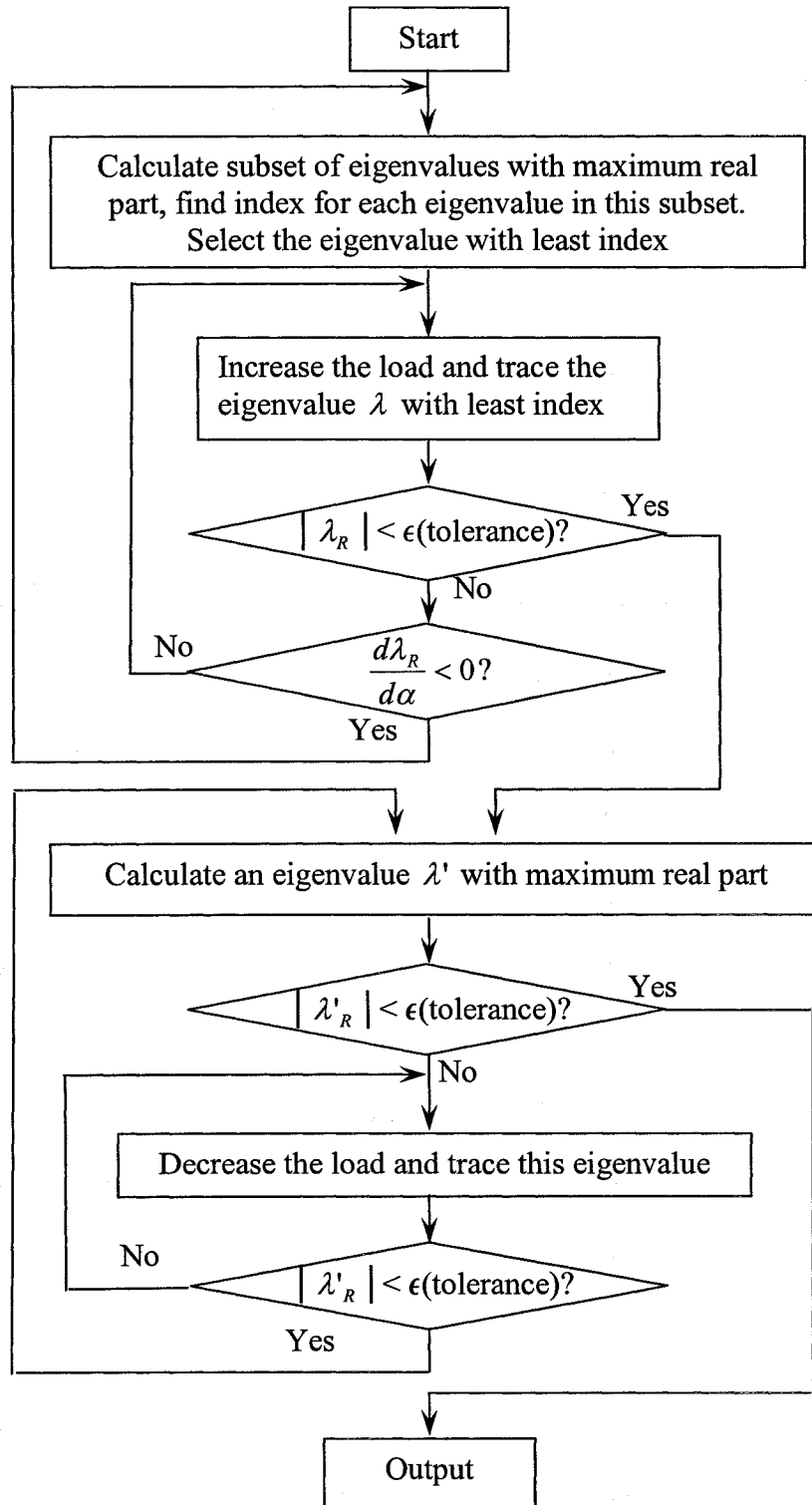


Fig 4.8 Flow chart for oscillatory stability margin estimation

The next eigenvalue with maximum real part is B. At this reduced load, B has a positive index. We traced this B for the Hopf identification. Before it reaches the imaginary axis the index became negative for an increase of 80% load from the new base case. At this load (5527.17MW) an eigenvalue with a maximum real part is calculated again. This time it picked up the eigenvalue C (which is the one that is actually crossing the imaginary axis first). In the previous case this eigenvalue was also selected for final tracing at the base load level of 6141.3MW. This implies the procedure proposed in Fig.4.8 guarantees the final identification of Hopf.

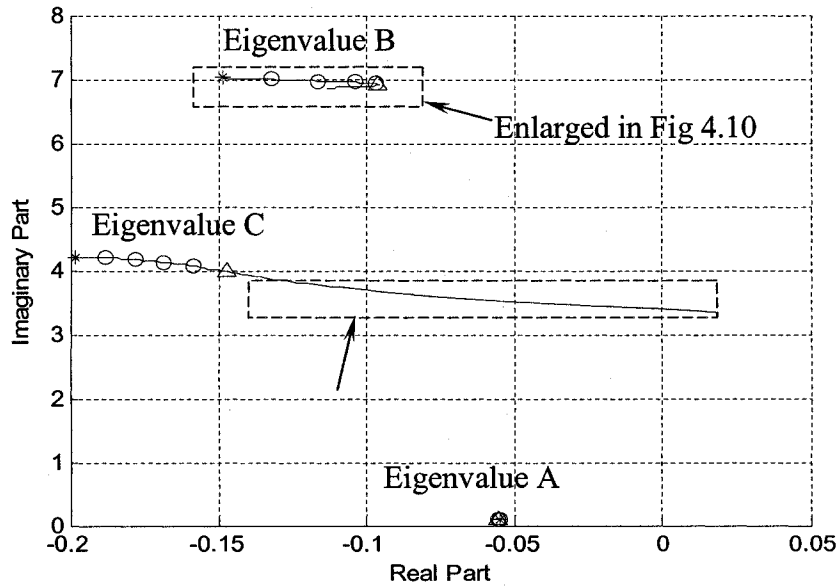


Figure 4.9 Eigenvalue A, B & C trajectories in complex plane for

The trajectories of eigenvalues A, B and C (the notation is same as in Fig. 4.1) for the new base case are shown in Fig 4.9. Part of Fig 4.9 is enlarged in Fig. 4.10, to show the details of the eigenvalue B's locus. The star represents the eigenvalue position at the new base value. Circles indicate the integration steps. The load increase step limit is the same as

the one used in Table 4.2. The triangle represents the eigenvalue's position at the fifth load increase step.

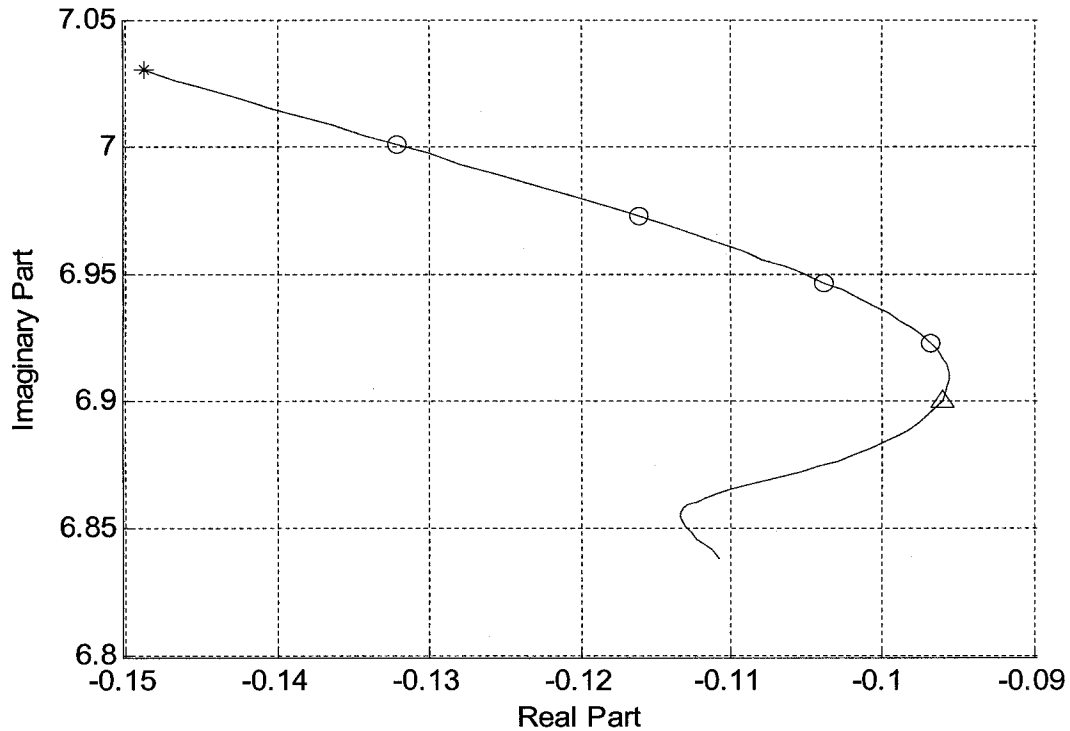


Figure 4.10 Eigenvalue B trajectory in complex plane

Since eigenvalue A and B both have negative index value (at the 5th step), eigenvalue C becomes the one we need to trace, according to the flow chart. From Fig 4.9 and our simulation result in Table 4.2, eigenvalue C will move toward the right and cross the imaginary axis.

To make sure the eigenvalue C crosses the imaginary axis first, we applied the maximum real part test at this load level where C crosses. We found out there is no other eigenvalue with real part greater than C. According to Figure 4.8, if there is another

eigenvalue which already crossed the imaginary axis before C, we have to apply backward tracing on this eigenvalue.

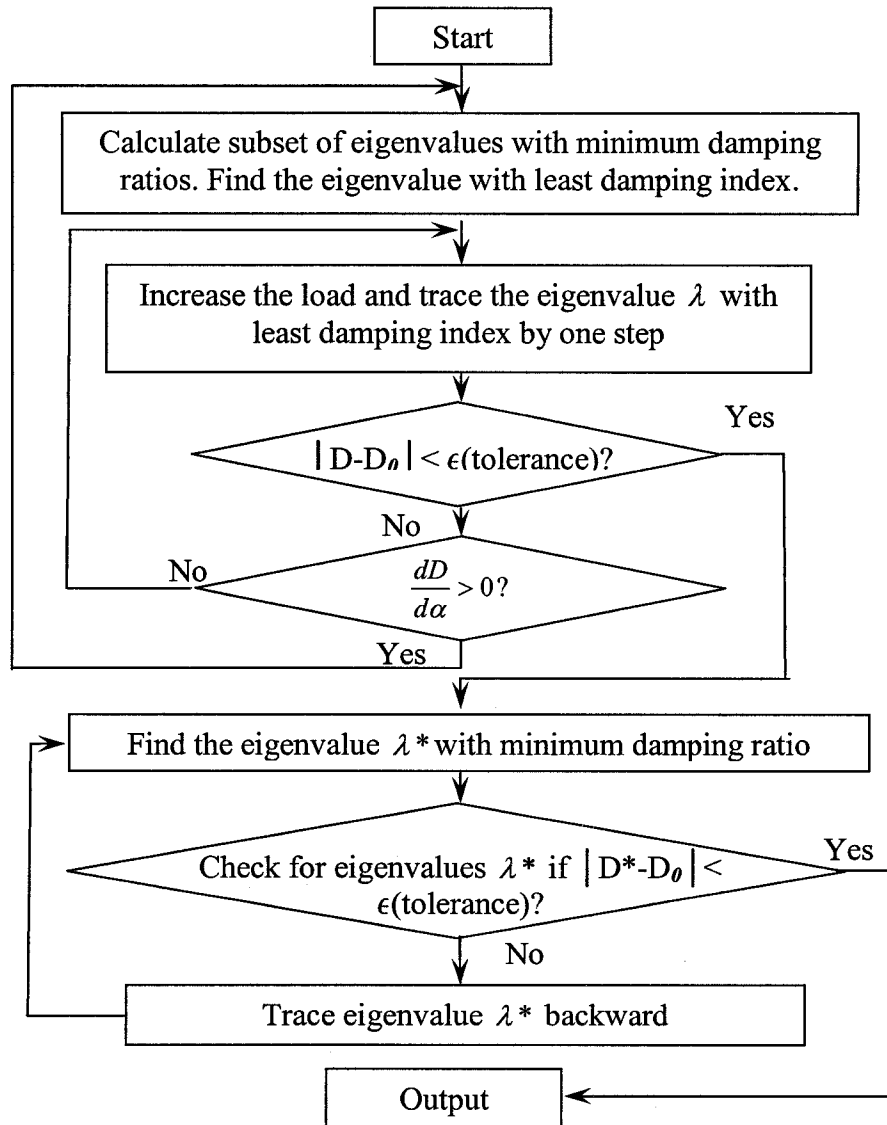


Fig 4.11 Flow chart for damping margin estimation

For the damping margin identification, we developed a very similar flow chart in Fig 4.11. But we need to compute the subset of eigenvalues with a minimum damping ratio at the beginning for the index comparison; compute a single eigenvalue with minimum

damping at last to verify the traced eigenvalue is the correct one. If the traced eigenvalue turns back, we need to start this process from this load level again.

#### 4.5.2 Consideration of Discrete Events

When the load level and transfer level increase in transmission systems, some discrete events could happen, like capacitor switching, transformer tap changing, hitting reactive power generation limits, etc. With some slight modifications the eigenvalue tracing can still function without any problems.

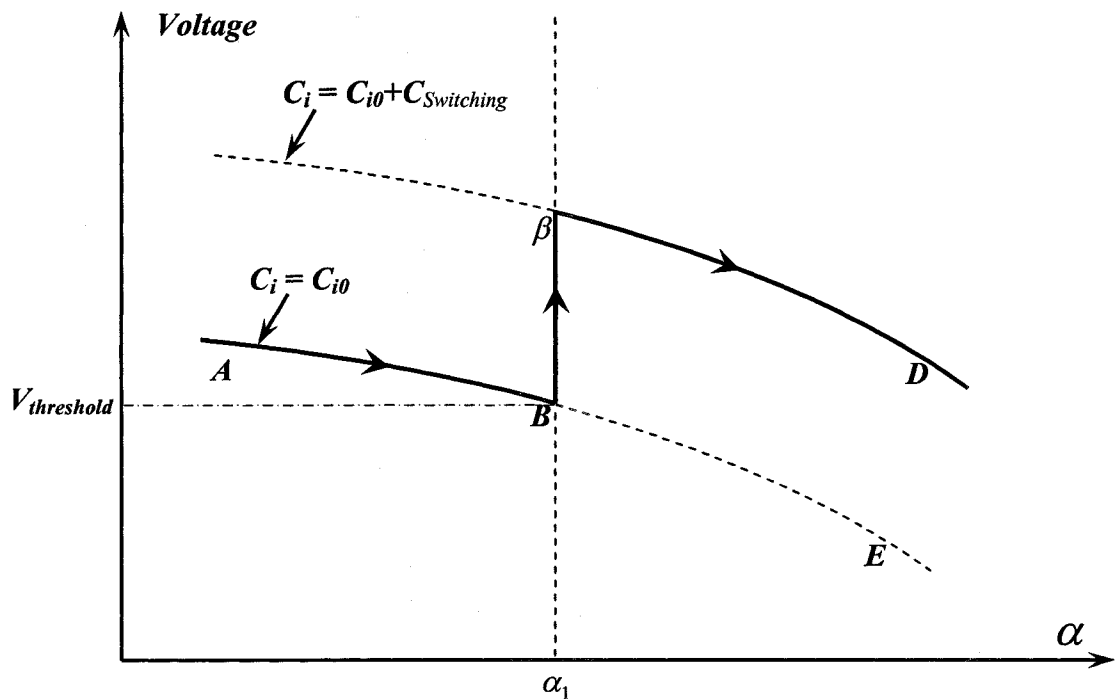


Figure 4.12 P-V curves due to capacitor switching

In the Figure 4.12, we demonstrate the capacitor switching action during the tracing. Here the horizontal axis is the  $\alpha$  value; vertical axis represents voltage. When the load or power transfer increases with  $\alpha$ , voltage will drop. When the voltage drops to the threshold

value  $V_{threshold}$ , capacitor  $C_{Switching}$  will be switched in at operating point B. Voltage will be raised to point C.

In the eigenvalue tracing process, the switching action is represented by equation (4.5.3). Before switching  $\gamma$  is zero.

$$\begin{cases} \dot{X} = F(X, Y, \alpha, \gamma); \\ 0 = G(X, Y, \alpha, \gamma); \end{cases} \quad (4.5.1)$$

$$\begin{cases} P_{Li} = (1 + K_{LPi}\alpha)P_{Li}^0; \\ Q_{Li} = (1 + K_{LQi}\alpha)Q_{Li}^0; \\ P_{gi} = (1 + K_{gi}\alpha)P_{gi}^0; \end{cases} \quad (4.5.2)$$

$$C_i = C_{i0} + \gamma K_{Ci} C_{switching} \quad (4.5.3)$$

When we trace the operating point from A to E in the above Figure 4.12, the  $\gamma$  stays at zero. Real and reactive load and generation keep increasing according to formula (4.5.2).

When we reach point B and find low voltage and the need for a capacitor to be switched in, we trace the eigenvalue from B to C. In this new tracing process, we let  $\alpha$  stop increasing, and treat it as a constant value.  $\gamma$  will be treated as a parameter to trace the eigenvalue. i.e., in formula (4.5.1),  $\alpha$  becomes a constant number. During the new tracing, all equations will be modified correspondingly (Now the all the derivatives are with respect to  $\gamma$ , not to  $\alpha$ .)

After  $C_i$  reaches  $C_{i0} + C_{Switching}$ , i.e., point C is obtained,  $\gamma$  will become a constant value again.  $C_i$  will be constant at  $C_{i0} + C_{Switching}$ . During tracing from C to D,  $\alpha$  becomes parameter again.

Figure 4.13 shows the eigenvalue's trajectory during this tracing process.

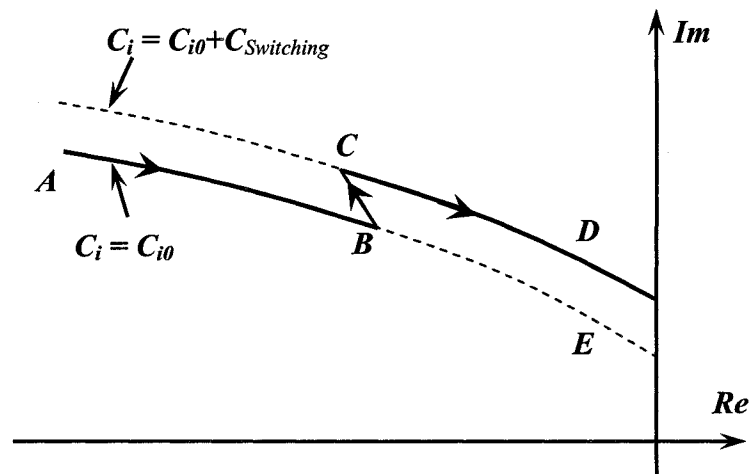


Figure 4.13 Traced eigenvalue trajectories with capacitor switching action

Similar reasoning can be applied for the other devices hitting the limit during the tracing. No eigenvalue computing is needed during integration from operating point A to D, and no accuracy is sacrificed.

## 4.6 Computational comparison

The following sections provide detailed computational comparisons with the secant method

### 4.6.1 Step length selection and accuracy

We simulated our case with second and fourth order integration methods with different step length maximum limits. We also tested the secant method with different initial step lengths. The correct oscillatory stability margin for the test case is 18.90605% load increase ( $\alpha = 0.1890605$ ), which is 1161.077MW. (6141.3MW is the total load in the base case.) Tables 4.4, 4.5 and 4.6 show the simulation result. Fig.4.14 summarizes these results.

The error in percentage is the percentage to oscillatory stability margin, (which is 1161.077MW).

In the secant method, to predict the step length, it needs two power flow solutions at different load levels and the corresponding eigenvalue. One power flow is chosen at the base case, the distance between the second operating point load level and the base case is called “Initial step length”. This step length has to be designated by users. After that, every step length will be predicted by the secant method. In Table 4.6, different initial step lengths were simulated. One can notice that the secant method can converge faster with an initial step length of 0.18 or 0.20 (This is due to the final answer which is 0.189 and the step length chosen is close to this value). The best case is when the initial step length is the final answer, where only one step is needed. If the user is not so lucky to jump close to the final answer in the first step, it takes an average of seven or eight steps to find the oscillatory stability margin.

From Table 1, when the step length limit is larger than 0.10, the error from the second order integration method becomes unacceptable. But the fourth order integration method result from Table 4.5 shows the error is pretty small even with a step length of 0.2. Load increase of 20% of the base case total load is more than enough in a practical system. In a stressed system, a step length more than 10% will require intermediate operating points to make the power flow converged, i.e. more than one power flow solution is needed for any method. If only considering a step length below or equal to 0.10, both the second and fourth order integration methods would find the final answer in a fewer number of steps. The



second order method could require less power flow solutions than the secant method with acceptable error.

It may appear that the secant method can take advantage of choosing a longer step length. To show how much benefit can be obtained for the secant method, we tested an initial step length of 0.40 load increase. As the result indicates, we didn't see any efficiency enhancement for the secant method from the large initial step length. The nonlinear characteristics of the system make the large step lengths perform poorly in the predictor.

For cases with a light base case load level, where an initial large step length can save time, we can use a hybrid method to start the search with the exact same way in the secant method. Once we find it's close to the final answer, then, we switch to our integration method, since the tangent predictor is always faster to converge in **local** search than the secant method.

We noticed that the fourth order integration method needs two power flow solutions for each step, one more than in the secant method, but, the fourth order integration method can find margin point with fewer steps to compensate for this disadvantage. If we don't consider the initial operating point calculation, for a 0.10 step length, the fourth order integration method needs four steps to find the final answer ( so, it needs eight power flow solutions). With the 0.10 initial step length, the secant method needs eight steps, which requires eight power flow solutions also. The numbers of power flow solutions are the same in both methods. With 0.10 step length, the second order method needs only 4 steps, which require only 4 power flow solutions.

**Table 4.4 Second order integration method simulation result**

Step Length Limit ( $\Delta\alpha$ )	Steps Needed	Error (MW)	Error (%)
0.04	7	1.307	0.11
0.05	6	2.446	0.21
0.06	7	7.065	0.61
0.07	6	7.374	0.64
0.08	6	15.84	1.36
0.09	7	20.69	1.78
0.1	4	19.51	1.68
0.11	5	32.35	2.79
0.12	5	53.15	4.58
0.13	6	85.42	7.36
0.14	6	130.7	11.25

**Table 4.5 Fourth order integration method simulation result**

Step Length Limit ( $\Delta\alpha$ )	Steps Needed	Error (MW)	Error (%)
0.04	7	0.176	0.015
0.06	5	0.155	0.013
0.08	5	0.162	0.014
0.10	4	0.025	0.002
0.12	4	0.079	0.007
0.14	4	0.342	0.029
0.16	4	0.764	0.066
0.18	3	1.445	0.124
0.1890605	1	0	0
0.20	3	2.722	0.234
0.22	4	4.949	0.426

**Table 4.6 Secant method simulation result**

Initial Step Length	0.04	0.06	0.08	0.10	0.12	0.14	0.16	0.18	0.1890605	0.2
Steps Needed	9	9	8	8	7	7	7	6	1	6

Initial Step Length	0.22	0.24	0.26	0.3	0.3	0.32	0.34	0.36	0.38	0.4
Steps Needed	7	7	7	7	8	8	8	8	9	9

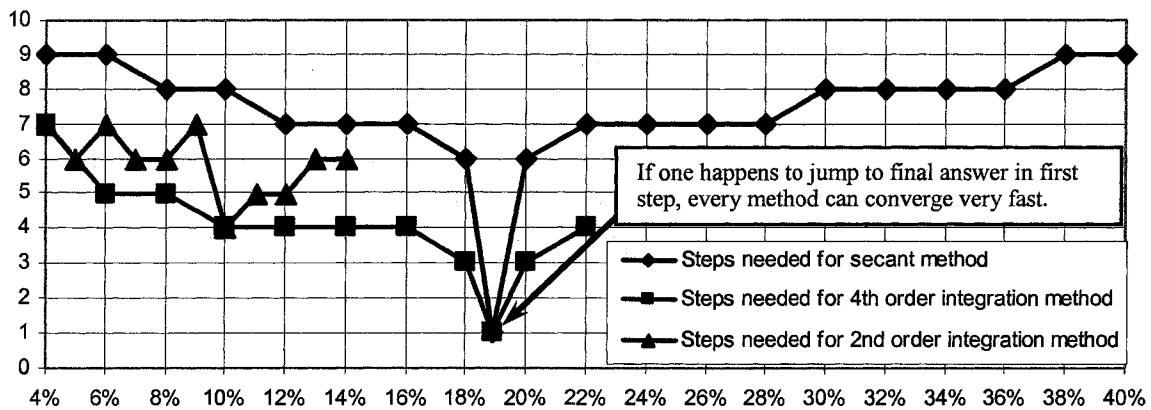


Figure 4.14. Steps needed for integration methods vs different initial step length for secant method

#### 4.6.2 Estimation of computational cost

For a full matrix with dimension  $N$ , LU factorization needs  $N^3$  times multiplication-additions. The forward and backward substitution costs  $N^2$  times multiplication-additions. In this situation, the fourth order integration method needs four LU factorizations; a single eigenvalue with a maximum real part needs only one LU factorization.

However in large scale power system, a Jacobian matrix is always sparse. The non-zero element for every column is influenced more by network topology and dynamic model than by system size. Thus, with an increase in system dimension, the number of non-zero elements will increase approximately by  $N$ , not with  $N^2$ .

According to William Tinney's paper [35], if the numbers of non-zero elements for each row doesn't increase with matrix dimension, the CPU cost will increase with  $N$ , not  $N^2$  (or  $N^3$ ) for both LU factorization and forward-backward substitution.

In our New England test case, according to Tinney's paper, for our system dimension of  $2(m+n+1) = 336$  linear equations, LU factorization needs 3787 times of multiplication-additions, forward and backward substitution needs 2087 times of multiplication-additions. For signal eigenvalue computation, forward and backward substitution needs 1838 times of multiplication-additions.

**Table 4.7: Computing cost comparison between different methods**

**Note:** For second order method, each step needs to solve two linear equations and one power flow.

	Number of Steps	Power flow solutions needed	Time for the computation of eigenvalue (unit is number of multiplication-additions )
Secant method	7	$7 \times 1 = 7$	$7 \times 20 \times 1838 = 257,320$
2nd integration	4	$4 \times 1 = 4$	$4 \times 2 \times (3787 + 2087) = 46,992$
4th integration	4	$4 \times 2 = 8$	$4 \times 4 \times (3787 + 2087) = 93,984$

From the above table 4.7, we can see how fast the second order method can be on both major parts of the calculation.

To estimate the computing cost for power flow, we assumed a Newton-Raphson method can converge within four iterations. Each iteration in a Newton-Raphson method consists of three major parts of computation: Mismatch calculation, calculation of a Jacobian matrix and solving of the linear equation.

Computing cost for solving linear equations: For the example above we need to solve  $2(m+n+1)$  linear equations (336 equations for the New England test systems) . The total cost is  $3787 + 2087 = 5874 \approx 17.5 \times 336 = 17.5 \times 2(m+n+1)$ . For DAE model with dimension  $(m+n)$ , we assume that solving linear equation would cost  $17.5(m+n)$  times of multiplication -additions

The other two parts of computing (calculating mismatch and Jacobian matrix) cost depends on the dynamic model used for DAE. If we assume these two parts cost as much as solving a linear equation, one power flow with four iterations will cost  $4 \times [17.5(m+n) + 17.5(m+n)] = 140(m+n) = 23,380$  times of multiplication - additions.

We can get the following total computing cost estimation.

**Table 4.8: Total computing cost comparison between different methods**

	Number of Steps	Power flow Solutions Needed	Time for Eigenvalue Information	Total Computing Cost
		*unit is number of multiplication - additions		
Secant method	7	$7 \times 23,380 = 163,660$	$7 \times 20 \times 1838 = 257,320$	420,980
2nd order integration	4	$4 \times 23,380 = 93,520$	$4 \times 2 \times (3787 + 2087) = 46,992$	140,512
4th order integration	4	$4 \times 2 \times 23,380 = 187,040$	$4 \times 4 \times (3787 + 2087) = 93,984$	281,024

In Table 4.8, the second order method is  $420,980/140,512 \approx 3.0$  times faster than the secant method. The fourth order method is  $420,980/281,024 \approx 1.5$  times faster. This speed comparison depends on the number of iterations needed to get a power flow solution.

#### 4.6.3 Robustness of the algorithms

In the above discussion we assumed on an average it takes 20 iterations to calculate a single eigenvalue. However it may take more than 20 iterations for large systems. To verify the robustness of the eigenvalue calculation for a large systems we randomly generated test systems. We calculated a single eigenvalue with a maximum real part of these test systems by MATLAB provided function "eigs.m". We found on an average, the number of iterations for these test matrixes will increase with increasing the test matrix dimension. The large

matrixes need much more than 20 times of iteration. The probability of this algorithm converging also increases with increasing the test matrix dimension.

Randomly generated test systems:

To make our randomly generated matrix more realistic, these matrixes have to be sparse. The non-zero elements on each row won't increase when the system size is increasing. Thus, the diagonal matrix is selected. Since, our matrix has to be a real matrix. We choose a two by two block on diagonal to represent a pair of conjugate complex eigenvalues.

If we have a complex eigenvalue  $\alpha \pm j\beta$ , where  $\beta \neq 0$ , then we can find a two by two real matrix  $\begin{bmatrix} \alpha & \beta \\ -\beta & \alpha \end{bmatrix}$  whose complex eigenvalues are  $\alpha \pm j\beta$ .

If we have two pairs of complex eigenvalues  $\alpha_1 \pm j\beta_1$  and  $\alpha_2 \pm j\beta_2$  and three single real eigenvalues  $\alpha_3, \alpha_4$  and  $\alpha_5$ , we can construct the following sparse real matrix with all of these eigenvalues.

$$\begin{bmatrix} \alpha_1 & \beta_1 & 0 & 0 & 0 & 0 & 0 \\ -\beta_1 & \alpha_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \alpha_2 & \beta_2 & 0 & 0 & 0 \\ 0 & 0 & -\beta_2 & \alpha_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \alpha_3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \alpha_4 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \alpha_5 \end{bmatrix}$$

In the test case, 41 real eigenvalues out of a total of 89 (46%) are real. The rest of the 54% eigenvalues are complex. All the real eigenvalues' mean is -11.2608, the standard deviation is 21.5945. All the complex eigenvalues' imaginary part mean is 4.2770, the standard deviation is 3.7219. The complex eigenvalues' real part mean is -2.0660, and the standard deviation is 3.7385

With this method, we randomly produced real eigenvalues and complex eigenvalues with the same statistical characteristics as above, and built matrices with different dimensions.

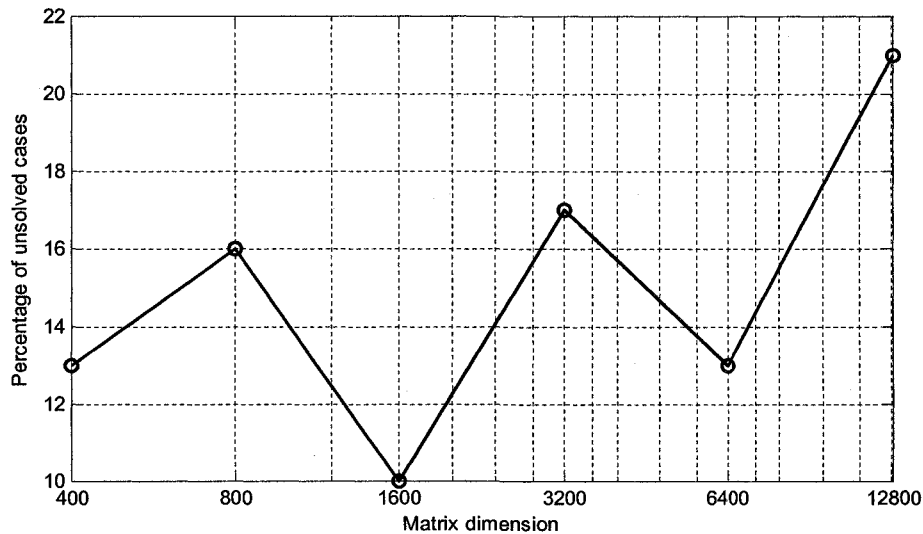


Figure 4.15. Percentage of unsolved cases with different size matrices

MATLAB single eigenvalue function “eigs.m” is applied to calculate a single eigenvalue with the maximum real part. The MATLAB default maximum iteration threshold is 300. We chose six different matrix dimensions: 400, 800, 1600, 3200, 6400 and 12800. For every dimension, 100 matrixes are randomly produced with the above statistical data. For a 12800 dimension matrix, 21 out of 100 cases can not converge within an iteration limit

(which is 300 iterations in our test). It appears, for large systems, conventional methods may face convergence problems.

In our New England test system Jacobian matrix computation, only three out of one hundred cases failed to converge within the iteration limit.

Figure 4.15 shows the percentage of unsolved cases for matrix dimensions from 400 to 12800. For all converged cases, we calculated the average number of iterations for matrices with different sizes. Results are shown in Figure 4.16.

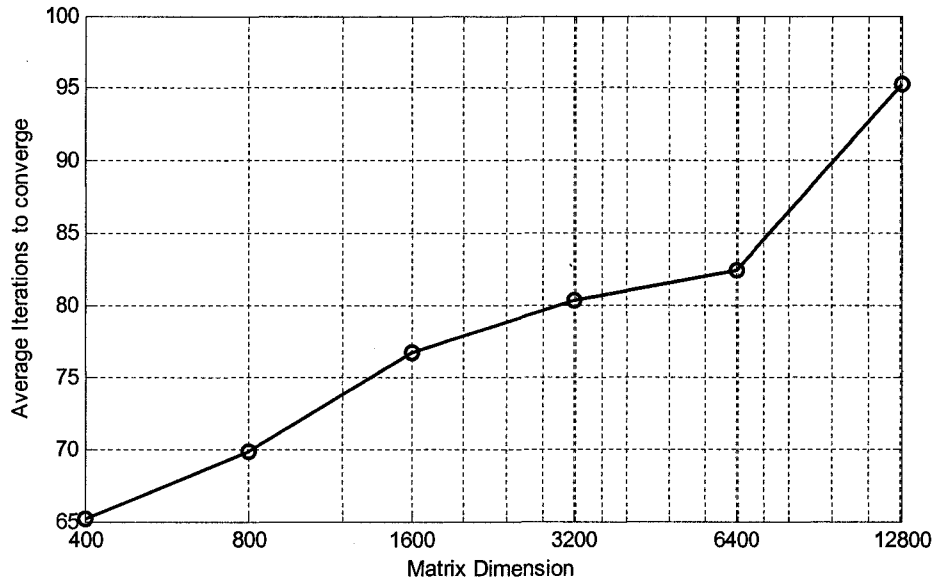


Figure 4.16. Average number of iterations needed to converge with different size matrices

The data for these two figures is given in Table 4.9.



Table 4.9: Numbers of iteration for different size matrixes

Matrix size	Unsolved cases per 100 tests	Average number of iterations for converged cases
400	13	65.20
800	16	69.87
1600	10	76.70
3200	17	80.33
6400	13	82.44
12800	21	95.29

These results indicate the potential disadvantage of the conventional secant method. In the New England 39-bus test case, we assumed 20 iterations to get a single eigenvalue. If the comparison is based on the iteration numbers in Table 4.9, the proposed method in this thesis would be much faster than the comparison result in the section 4.6.2.

## **Chapter 5 Oscillatory stability and damping ratio margin boundary tracing**

### **5.1 Oscillatory stability margin boundary tracing**

In the last chapter, an efficient eigenvalue-tracing algorithm is described for a fixed set of control parameters. When these control parameters change, the stability margin related to the Hopf bifurcation changes as well. The oscillatory stability margin boundary is confined by the Hopf bifurcation with a different set of control parameters. This boundary could be traced by augmenting the power system equilibrium with a characterization equation that defines the Hopf boundary. This characterization equation, together with the system equilibrium, defines the margin boundary.

In grid security assessment, control actions are needed to maintain a given oscillatory stability and damping margins, i.e., one can maximize the stability margin or damping margin by adjusting control parameters. The algorithm presented in this chapter can estimate the change in the oscillatory stability margin for any given large change in system parameters. Through nonlinear margin boundary tracing, one can easily estimate the margin for any given variation in the control parameter value. Discrete events, like a transmission line tripping, also can be parameterized as a continuous event, such that, one can apply this margin boundary tracing method in contingency analysis. This can reduce the computational time required for a large scale system contingency analysis.

Compared with margin sensitivity, margin boundary tracing provides an actual value of the margin by considering all the nonlinear factors. It can provide more accurate information than the margin sensitivity.

The oscillatory stability margin boundary can be traced by augmenting the power system equilibrium with a characterization equation. This characterization equation, together with the system equilibrium, defines the margin boundary.

In chapter 3 the set of equations (3.1.4) (3.1.5) and (3.1.6) define the Hopf boundary. Here (5.1.1) (equilibrium conditions) corresponds to (3.1.6) where an additional control parameter  $\beta$  is explicitly represented. In chapter 4,  $\beta$  is fixed and  $\alpha$  is varied. Here both  $\alpha$  and  $\beta$  are varied.

$$\begin{cases} 0 = F(X, Y, \alpha, \beta) \\ 0 = G(X, Y, \alpha, \beta) \end{cases} \quad (5.1.1)$$

The conditions related to (3.1.4) and (3.1.5) are combined to define characterization equations (5.1.2) which together with (5.1.1) defines oscillatory stability boundary.

$$C(X, Y, v_R, u_R, v_I, u_I, \alpha, \beta, s) = \begin{cases} F_X v_R + F_Y u_R + s v_I = 0 \\ G_X v_R + G_Y u_R = 0 \\ F_X v_I + F_Y u_I - s v_R = 0 \\ G_X v_I + G_Y u_I = 0 \\ v_{R(i)} - 1 = 0 \\ v_{I(i)} = 0 \end{cases} \quad (5.1.2)$$

As described in chapter 2 (section 2.2) the load and the control parameter variation are represented as:

$$\begin{aligned} \text{Load:} \quad P_{iL} &= P_{iL}^0 (1 + \alpha K_{iPL}); \\ Q_{iL} &= Q_{iL}^0 (1 + \alpha K_{iQL}); \end{aligned} \quad (5.1.3)$$

$$\text{Control:} \quad U_i = U_{i0} + \beta K C_i \quad (5.1.4)$$

Where  $U_{i0}$  indicates the initial configuration of control  $i$ . In the control parameter space  $U_i = (V_{refi}, K_{ai}, \dots)$  ( $i = 1, \dots, k$ , where  $k$  denotes the total number of possible controls in the system.), all the control parameters are changed with the parameter  $\beta$  change.

Different combinations of control actions can be achieved by assigning different ratio values to  $K C_i$ . This parameterization leads to two parameter variations:  $\alpha$  characterizing system loading conditions with respect to a specified loading scenario, and  $\beta$  characterizing control parameters with respect to a specified control scenario

$$\begin{cases} 0 = F(X, Y, \alpha, \beta) \\ 0 = G(X, Y, \alpha, \beta) \end{cases} \text{ and } C(X, Y, v_R, u_R, v_I, u_I, \alpha, \beta, s) = 0 \text{ has } 3(n+m)+2 \text{ equations}$$

and  $3(n+m)+3$  variables. With the pre-defined control scenario (5.1.4) the oscillatory stability margin can be identified with any given  $\beta$  value.

This entire margin boundary tracing process begins with the base-case Hopf bifurcation point, which is detected by the Hopf bifurcation identification method described in Chapter 4.

Fig. 5.1 illustrated this process. The point C0 is the initial operating point. With the method introduced in Chapter 4, one can identify oscillatory stability margin point A0, i.e., the first pair of conjugate complex eigenvalues is identified to cross the imaginary axis. During this process, the parameter  $\beta$  keeps its value at  $\beta_0$ .

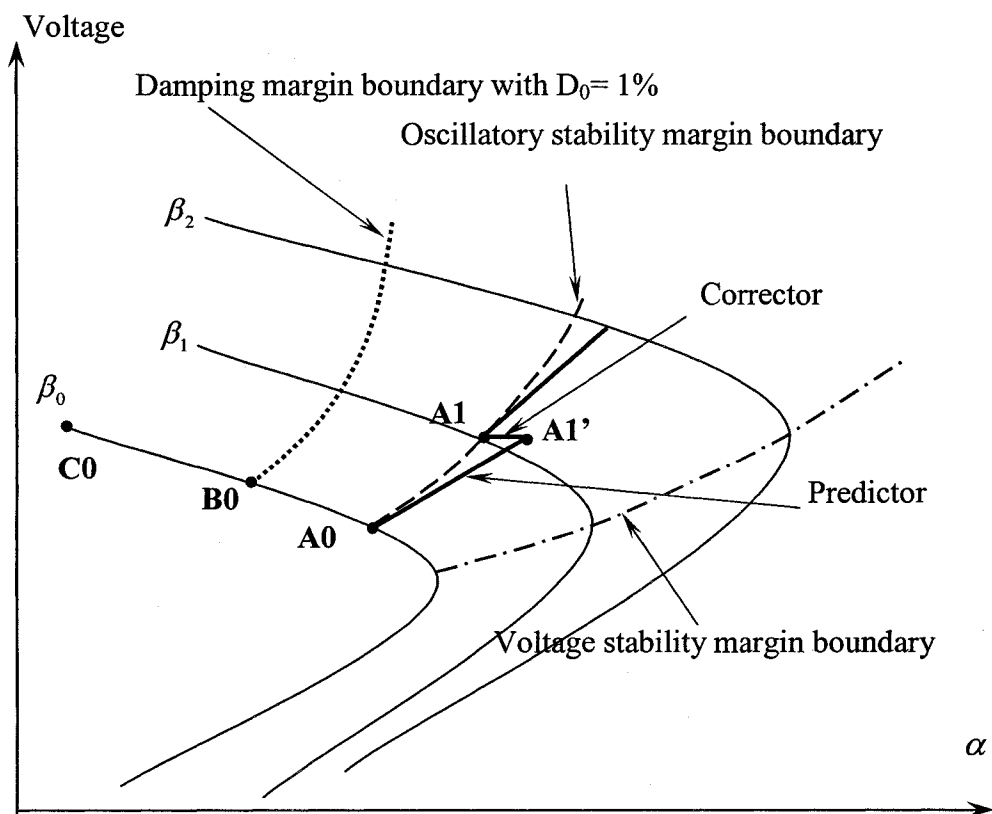


Figure 5.1 Illustration of the voltage stability, oscillatory stability and damping margin boundaries and their tracing process

Now, the margin boundary tracing process can start. All the information at operating point A0 should satisfy conditions (5.1.1) and (5.1.2).

### 5.1.1 Augmentation for boundary tracing

Combining (5.1.1) with (5.1.2) results in the following set of equations that define the Hopf boundary [36].

$$B(X, Y, v_R, u_R, v_I, u_I, \alpha, \beta, s) = \begin{bmatrix} F(X, Y, \alpha, \beta) \\ G(X, Y, \alpha, \beta) \\ C(X, Y, v_R, u_R, v_I, u_I, \alpha, \beta, s) \end{bmatrix} = 0 \quad (5.1.5)$$

In (5.1.5), there are  $3(m+n)+3$  unknown variables and  $3(m+n)+2$  equations. To get a specific boundary point, one has to assign a value to one of the variables by the following equation:

$$[X^T, Y^T, v_R^T, u_R^T, v_I^T, u_I^T, \alpha, \beta, s]e_k - \eta = 0 \text{ where } k \text{ denotes the index of the specified variable and } \eta \text{ is its value.}$$

The total augmented equations for margin boundary tracing are

$$\begin{aligned} \therefore 0 = H(X, Y, v_R, u_R, v_I, u_I, \alpha, \beta, s) &= \begin{bmatrix} B(X, Y, v_R, u_R, v_I, u_I, \alpha, \beta, s) \\ [X^T, Y^T, v_R^T, u_R^T, v_I^T, u_I^T, \alpha, \beta, s]e_k - \eta \end{bmatrix} \Rightarrow \\ 0 = H(X, Y, v_R, u_R, v_I, u_I, \alpha, \beta, s) &= \begin{bmatrix} F(X, Y, \alpha, \beta) \\ G(X, Y, \alpha, \beta) \\ c(X, Y, v_R, u_R, v_I, u_I, \alpha, \beta, s) \\ [X^T, Y^T, v_R^T, u_R^T, v_I^T, u_I^T, \alpha, \beta, s]e_k - \eta \end{bmatrix} \end{aligned} \quad (5.1.6)$$

Where  $e_k$  is the vector with all zero elements, except the  $k^{\text{th}}$  element equal to one.

$$\therefore \frac{\partial H(X, Y, v_R, u_R, v_I, u_I, \alpha, \beta, s)}{\partial (X, Y, v_R, u_R, v_I, u_I, \alpha, \beta, s)} = \begin{bmatrix} \frac{\partial F}{\partial X} & \frac{\partial F}{\partial Y} & \frac{\partial F}{\partial v_R} & \frac{\partial F}{\partial u_R} & \frac{\partial F}{\partial v_I} & \frac{\partial F}{\partial u_I} & \frac{\partial F}{\partial \alpha} & \frac{\partial F}{\partial \beta} & \frac{\partial F}{\partial s} \\ \frac{\partial G}{\partial X} & \frac{\partial G}{\partial Y} & \frac{\partial G}{\partial v_R} & \frac{\partial G}{\partial u_R} & \frac{\partial G}{\partial v_I} & \frac{\partial G}{\partial u_I} & \frac{\partial G}{\partial \alpha} & \frac{\partial G}{\partial \beta} & \frac{\partial G}{\partial s} \\ \frac{\partial c}{\partial X} & \frac{\partial c}{\partial Y} & \frac{\partial c}{\partial v_R} & \frac{\partial c}{\partial u_R} & \frac{\partial c}{\partial v_I} & \frac{\partial c}{\partial u_I} & \frac{\partial c}{\partial \alpha} & \frac{\partial c}{\partial \beta} & \frac{\partial c}{\partial s} \\ e_k \end{bmatrix} \quad (5.1.7)$$

Where  $\eta = [X_{pre}^T, Y_{pre}^T, v_{Rpre}^T, u_{Rpre}^T, v_{Ipre}^T, u_{Ipre}^T, \alpha_{pre}, \beta_{pre}, s_{pre}]e_k$  and is obtained from the

tangent vector calculated in the boundary predictor.

### 5.1.2 Boundary predictor

In order to get the oscillatory stability margin with  $\beta$  value  $\beta_I$ , which is the point A1 in Fig. 5.1, one can apply the predictor and corrector strategy. The margin boundary tangent which is shown in the Fig. 5.1 is denoted as  $[dX^T, dY^T, dv_R^T, du_R^T, dv_I^T, du_I^T, \alpha, \beta, s]^T$ , and can be obtained by solving

$$\frac{\partial H(X, Y, v_R, u_R, v_I, u_I, \alpha, \beta, s)}{\partial (X, Y, v_R, u_R, v_I, u_I, \alpha, \beta, s)} \begin{bmatrix} dX \\ dY \\ dv_R \\ du_R \\ dv_I \\ du_I \\ d\alpha \\ d\beta \\ ds \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \pm 1 \end{bmatrix} \quad (5.1.8)$$

After solving (5.1.8) for a tangent vector, the predicted values of the unknown variables can be obtained from (5.1.9). Where  $\delta$  is the step length.

$$\begin{bmatrix} X_{pre} \\ Y_{pre} \\ v_{Rpre} \\ u_{Rpre} \\ v_{Ipre} \\ u_{Ipre} \\ \alpha_{pre} \\ \beta_{pre} \\ s_{pre} \end{bmatrix} = \begin{bmatrix} X \\ Y \\ v_R \\ u_R \\ v_I \\ u_I \\ \alpha \\ \beta \\ s \end{bmatrix}^{[i]} + \delta \begin{bmatrix} dX \\ dY \\ dv_R \\ du_R \\ dv_I \\ du_I \\ d\alpha \\ d\beta \\ ds \end{bmatrix} \quad (5.1.9)$$

This predicted value is represented as point A1' in Fig 5.1, and it can be used to as an initial guess to converge upon the stability boundary by solving the non-linear algebraic equations (5.1.6) with the Newton-Raphson method.

### 5.1.3 Boundary corrector

The Newton-Raphson method can be employed to do the boundary correction as

$$\begin{bmatrix} X \\ Y \\ v_R \\ u_R \\ v_I \\ u_I \\ \alpha \\ \beta \\ s \end{bmatrix}^{new} = \begin{bmatrix} X \\ Y \\ v_R \\ u_R \\ v_I \\ u_I \\ \alpha \\ \beta \\ s \end{bmatrix} - \left( \frac{\partial H(X, Y, v_R, u_R, v_I, u_I, \alpha, \beta, s)}{\partial (X, Y, v_R, u_R, v_I, u_I, \alpha, \beta, s)} \right)^{-1} H(X, Y, v_R, u_R, v_I, u_I, \alpha, \beta, s) \quad (5.1.10)$$

Iterate until the mismatch is less than the tolerance. Finally one can obtain the solution which is the Hopf bifurcation point corresponding to  $\left[ X^T, Y^T, v_R^T, u_R^T, v_I^T, u_I^T, \alpha, \beta, s \right] e_k = \eta$ .

In the corrector, there are two ways to converge to the boundary curve. Fig. 5.2 shows these two strategies.

If one wants to trace the margin boundary curve with several specific  $\beta$  value points, the step length  $\delta$  in (5.1.9) needs to be chosen carefully by  $\delta = \frac{\beta_{i+1} - \beta_i}{d\beta_i}$ , where  $d\beta_i$  is the solution of (5.1.8).  $\beta_i$  is the value for current available margin point on boundary.  $\beta_{i+1}$  is the specific value for the next margin boundary point which is unknown. With this step length,



the predicted  $\beta$  value will be exactly  $\beta_{i+1}$ . By solving the non-linear algebraic equation (5.1.6), one can get the margin point with  $\beta = \beta_{i+1}$ . In this case, the index  $k$  in  $e_k$  should be  $3(n+m)+2$ .

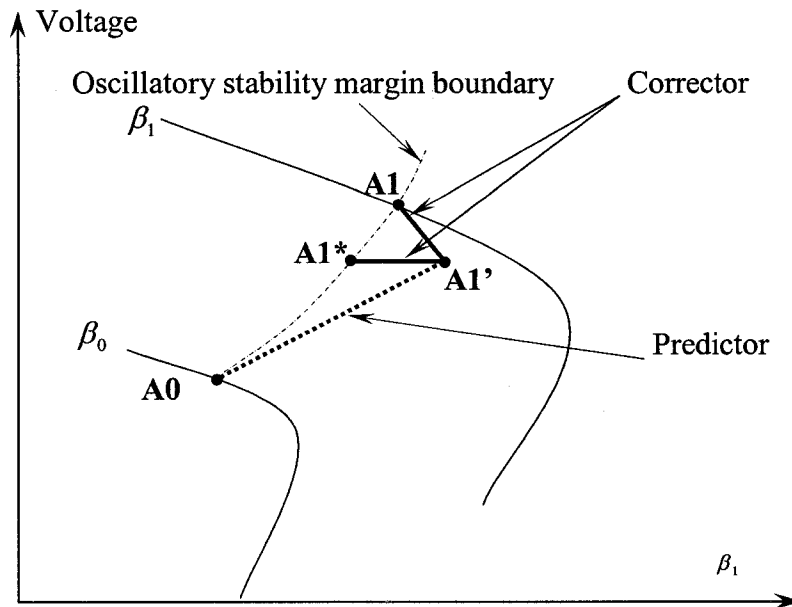


Figure 5.2 Illustration of the corrector convergence strategies

Thus, the margin point for a given  $\beta$  value can be traced. This corrector is shown in Fig 5.2 (from the point  $A1'$  to the point  $A1$ ).

There is another way to trace the margin boundary curve without controlling the intermediate point's  $\beta$  value. After calculating the tangent vector  $[dX^T, dY^T, dv_R^T, du_R^T, dv_I^T, du_I^T, \alpha, \beta, s]^T$ , check the element with maximum magnitude. For example, let  $dV_i$  be the maximum magnitude element ( $V_i$  is an element of  $Y$ , which is the  $i^{\text{th}}$  bus voltage magnitude). In the corrector, (5.1.6) will be modified to

$$\begin{cases} 0 = F(X, Y, \alpha, \beta); \\ 0 = G(X, Y, \alpha, \beta); \\ C(X, Y, v_R, u_R, v_I, u_I, \alpha, \beta, s) = 0; \\ V_i - V_i^{predicted} = 0; \end{cases} \quad (5.1.11)$$

When solving this algebraic equation with the Newton-Raphson method, the index  $k$  in (5.1.6) becomes the index of  $dV_i$  in vector  $[dX^T, dY^T, dv_R^T, du_R^T, dv_I^T, du_I^T, \alpha, \beta, s]^T$ .

This process is illustrated in Fig. 5.2 (From point A1' to point A1\*).

The only drawback of this corrector is that the  $\beta$  value is not totally controllable. The advantage of this approach is it is numerically stable. The Newton-Raphson method will converge faster compared to the previous selection of  $\beta$ . For the sample system tested in this thesis both methods are equally good. In the event that one encounters divergence problems with the first approach, the second approach for the corrector iteration can be applied to overcome this obstacle.

The following steps are involved in oscillatory stability margin boundary tracing:

- 1) with the initial state, which is the solution of (5.1.1) and (5.1.2), where  $\beta = \beta_0$ , solve the (5.1.8) to get the tangent of margin boundary.
- 2) with (5.1.9), predict the margin boundary solution.
- 3) solve (5.1.10) for the Newton-Raphson algorithm.
- 4) If  $\beta$  reaches the target value or any other stopping criteria, stop the tracing process. Otherwise, return to step 2).

The result of tracing is a set of Hopf bifurcation points for various values of  $\beta$ . Each  $\beta$  value can be related to the amount of corresponding control variable. The distance between two bifurcation points depends on the step length  $\delta$  in margin boundary tracing.

The sensitivity information is available as a by-product, like  $\frac{dX}{d\beta}$ ,  $\frac{dY}{d\beta}$ ,  $\frac{d\alpha}{d\beta}$ ,  $\frac{dv}{d\beta}$ ,  $\frac{du}{d\beta}$ ,

$$\frac{ds}{d\beta}$$

## 5.2 Damping ratio margin boundary tracing

The same procedure can be employed to trace the damping ratio margin boundary. Here, for a given damping ratio  $\xi_0$ , the real part of the dominant eigenvalue should be  $r = s \cdot \cot(\cos^{-1}(\xi_0))$ , where  $s$  is the imaginary part of dominant eigenvalue. So the cut function of the damping margin boundary becomes

$$C(X, Y, \alpha, \beta, s, v_R, v_I, u_R, u_I) = \begin{cases} F_X u_R + F_Y v_R + s u_I + s \cdot \cot(\cos^{-1}(\xi_0)) u_R = 0 \\ F_X u_I + F_Y v_I - s u_R + s \cdot \cot(\cos^{-1}(\xi_0)) u_I = 0 \\ G_X u_R + G_Y v_R = 0 \\ G_X u_I + G_Y v_I = 0 \\ u_{R(i)} - 1 = 0 \\ u_{I(j)} - 1 = 0 \end{cases} \quad (5.2.1)$$

Similar procedures described for oscillatory stability margin boundary can be applied to obtain a damping ratio margin boundary. The only difference is “C” which is replaced with (5.2.1).

In Fig. 5.1, the boundary tracing starts from point B0, this is obtained by the damping ratio margin identification method from Chapter 4. With the predictor and the corrector, the boundary curve can be obtained.

### **5.3 Simulation results**

#### **5.3.1 Oscillatory stability margin boundary tracing**

The algorithm was tested on a New England 39-bus system described in Chapter 4. Load consists of 50% constant power, 30% constant current and 20% constant impedance. Load on all buses will increase by the same percentage. Four cases are studied with different control scenarios. For all these cases, the starting conditions correspond to the margin point identified in chapter 4, where the oscillatory instability occurs at a load increase of 1161MW.

##### **Case 1:**

In this case, parameters  $K_a$  in the excitation system of all generators are decreased to 50% of their initial values.

The vertical axis in Fig 5.3 shows the oscillatory stability margin value. The horizontal axis shows the  $K_a$  value at all generators as a percentage of their original values. When the values of  $K_a$  are decreased by around 22%, the entire system oscillatory stability margin reaches its maximum value of 1230MW. The original  $K_a$  values have the margin of 1160MW. The margin is increased by 70MW. From the margin boundary tracing, the control parameter perturbation tolerance also can be obtained. For example, if all generators'  $K_a$  values are set to be 80% of original value, the margin can be guaranteed larger than 1200MW with 10 percent perturbation.

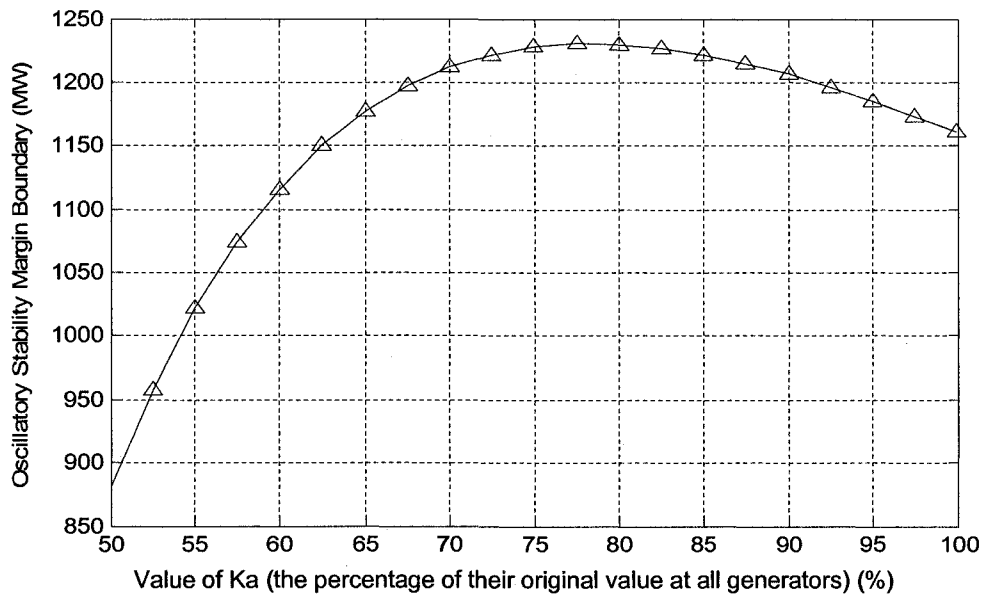


Figure 5.3 Oscillatory stability margin boundary with decrease of Ka at all generators

### Case 2:

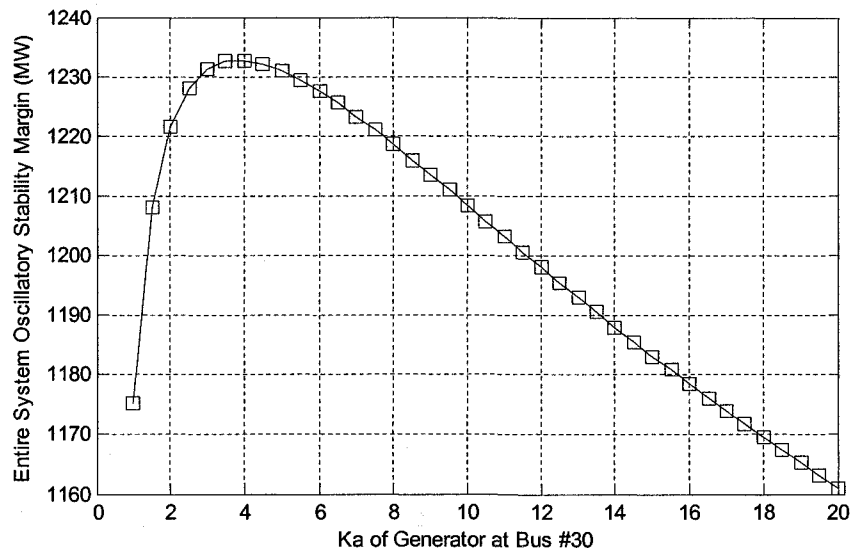


Figure 5.4 Oscillatory stability margins with Ka of generator at Bus #30

Only the  $K_a$  of the generator at Bus #30 is reduced. Fig5.4 shows the margin change. Entire system load margin reaches its maximum value of 1232MW when the  $K_a$  value of the generator at bus#30 decreases from 20 to 4. The load margin is improved by approximately 70MW.

### Case 3:

Three generators are chosen for adjusting the  $K_a$  value. The control scenario is:

$$K_{a_{30}} = K_{a_{30}}^{(0)}(1 - 0.95\beta);$$

$$K_{a_{31}} = K_{a_{31}}^{(0)}(1 + 8.0\beta);$$

$$K_{a_{36}} = K_{a_{36}}^{(0)}(1 - 0.5\beta);$$

Fig5.5 shows the variation of the margin. When the  $\beta$  value is 0.8, the oscillatory stability margin becomes 1357MW. The margin is improved by 197MW.

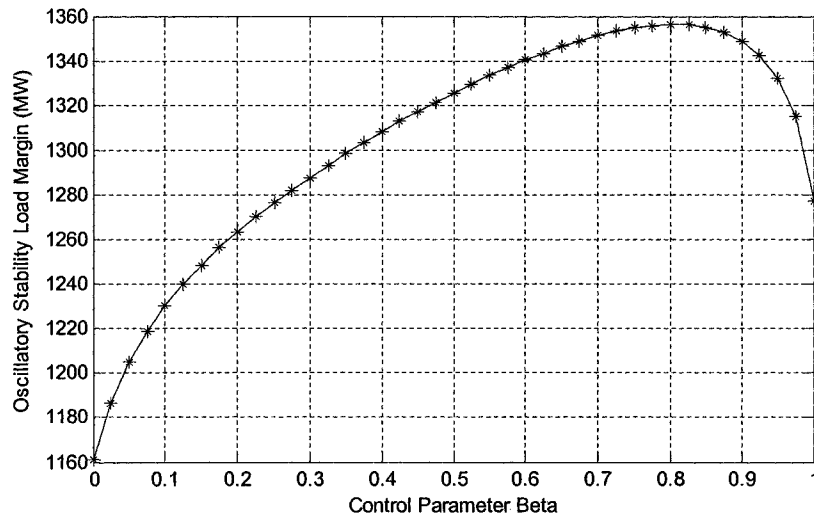


Figure5.5 Oscillatory stability margin boundary with  $K_a$  of three selected generators varying

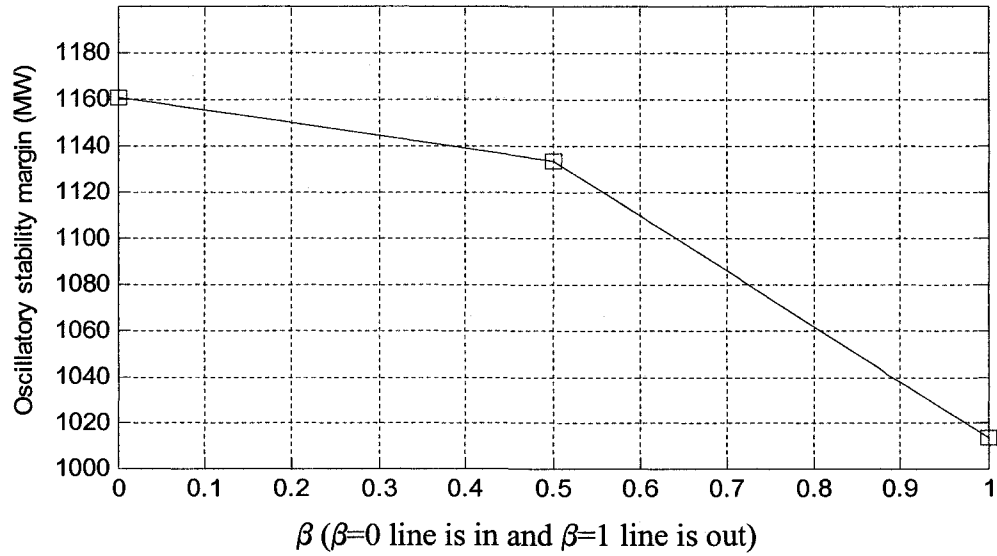
**Case 4:**

Figure 5.6 Oscillatory stability margin boundary with outage of line 3-18

This case corresponds to line outage. Here  $\beta = 0$  corresponds to the line is in and  $\beta = 1$  indicates the line is out.

Fig 5.6 shows the system load margin for post-contingency. The margin after line 3-18 out is 1013MW. In Fig5.6, the boundary can be traced in only two steps, which is denoted by squares. The total cost involved: solving of (5.1.1) and (5.1.2) twice by the Newton-Raphson method. There is no eigenvalue computing involved. Without the margin boundary tracing, a P-V curve has to be traced without this transmission line, in order to obtain the oscillatory stability margin for post-contingency.

### 5.3.2 Damping ratio margin boundary tracing

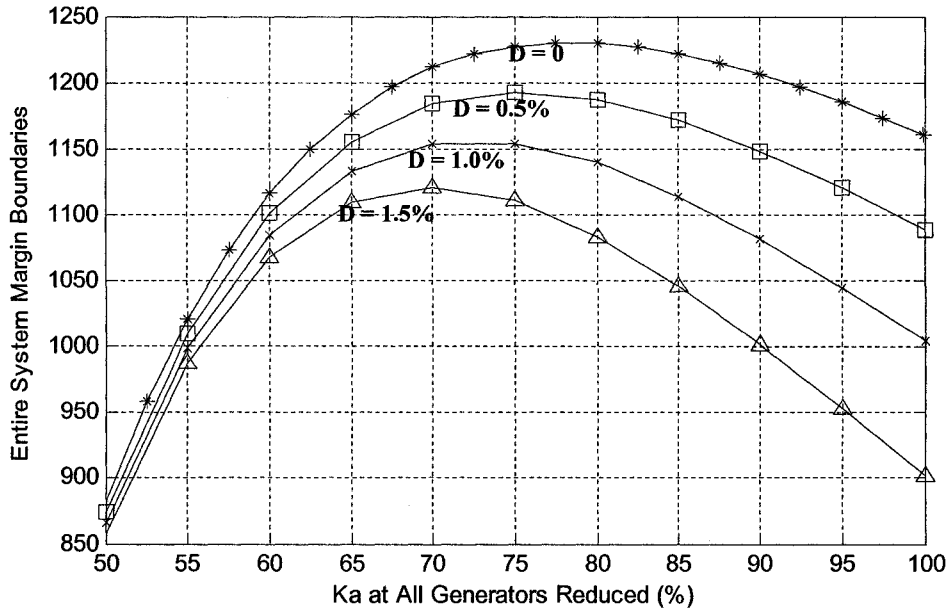


Figure 5.7 Oscillatory stability margin and damping margin with decrease of  $K_a$  at  $f$  all the generators

In Fig 5.7, the four curves are the oscillatory stability margin boundary ( $D = 0$ ), and the damping margin boundaries for  $D=0.5\%$ ,  $1.0\%$ , and  $1.5\%$ , from the top to the bottom of the figure.

Fig. 5.8 shows a three dimensional view of Fig.5.7. The  $D$  threshold values are changed in increments from zero to  $1.5\%$  with  $0.1\%$  increment intervals. The vertical axis denotes the entire system damping margin. The same color represents similar margin values. All the points under the surface are safe.



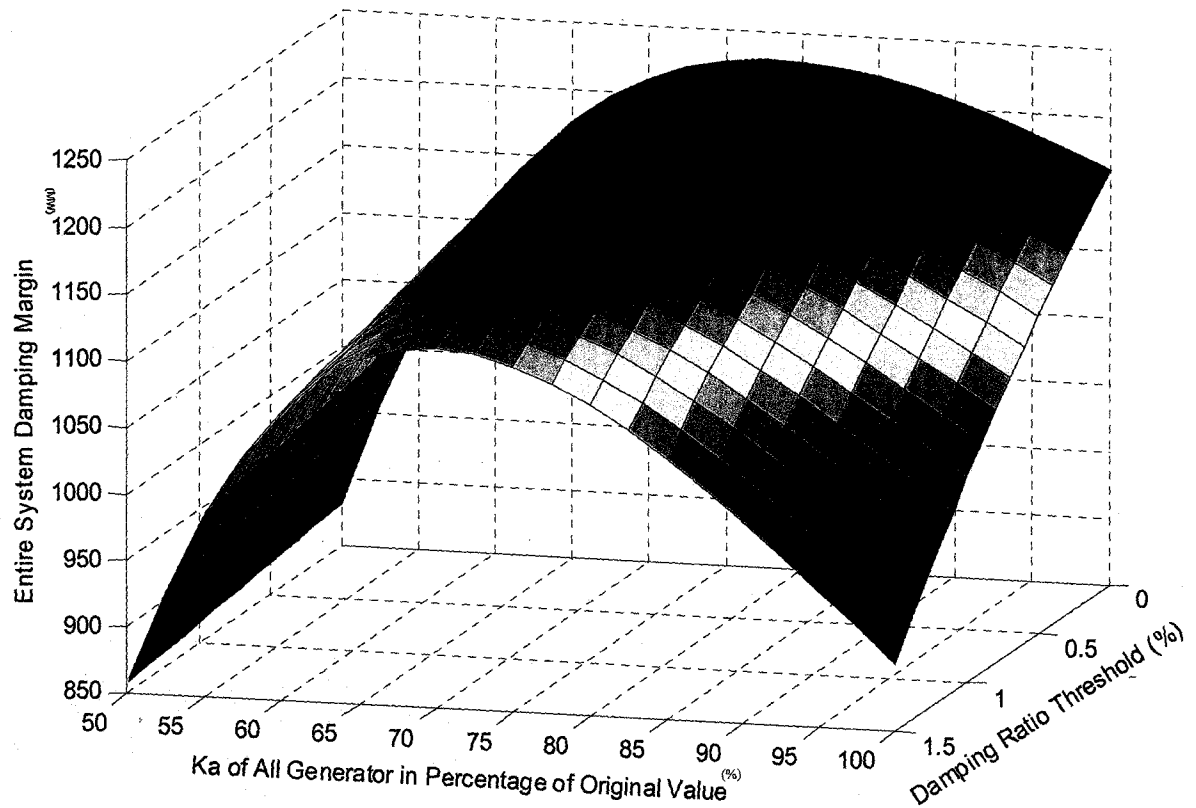


Figure 5.8 damping margin surface with change of  $K_a$  at all generators

If we want to achieve the maximum oscillatory margin (or damping margin with  $D=0$ ), according to this  $K_a$  change scenario, all generators  $K_a$  should be decreased to around 78% of their original value. If one wants to obtain the maximum damping margin boundary with  $D=1.5\%$ , the  $K_a$  should be reduced to around 70% of their original value.

From Fig. 5.8, one can see that even with the same control parameter scenario, different stability assessment criteria could result in a different optimal control parameter value. i.e., the optimal  $\beta$  value for maximizing oscillatory stability margin may not be the optimal for the damping margin.

Further in Fig. 5.8, with all  $K_a$ 's set at 100% of their initial values and with  $D=1.5\%$  the difference between the damping margin and oscillatory stability margin is around 260MW. However in other extreme where  $K_a$ 's are at 50% of their initial value and with the same  $D=1.5\%$ , the difference between the damping margin and the oscillatory stability margin is only 25MW. In this case even though there is adequate damping you may experience oscillatory instability problems.

Similarly Fig. 5.9 and Fig 5.10 show the three dimensional view for cases 2 and 3 respectively of section 5.3.

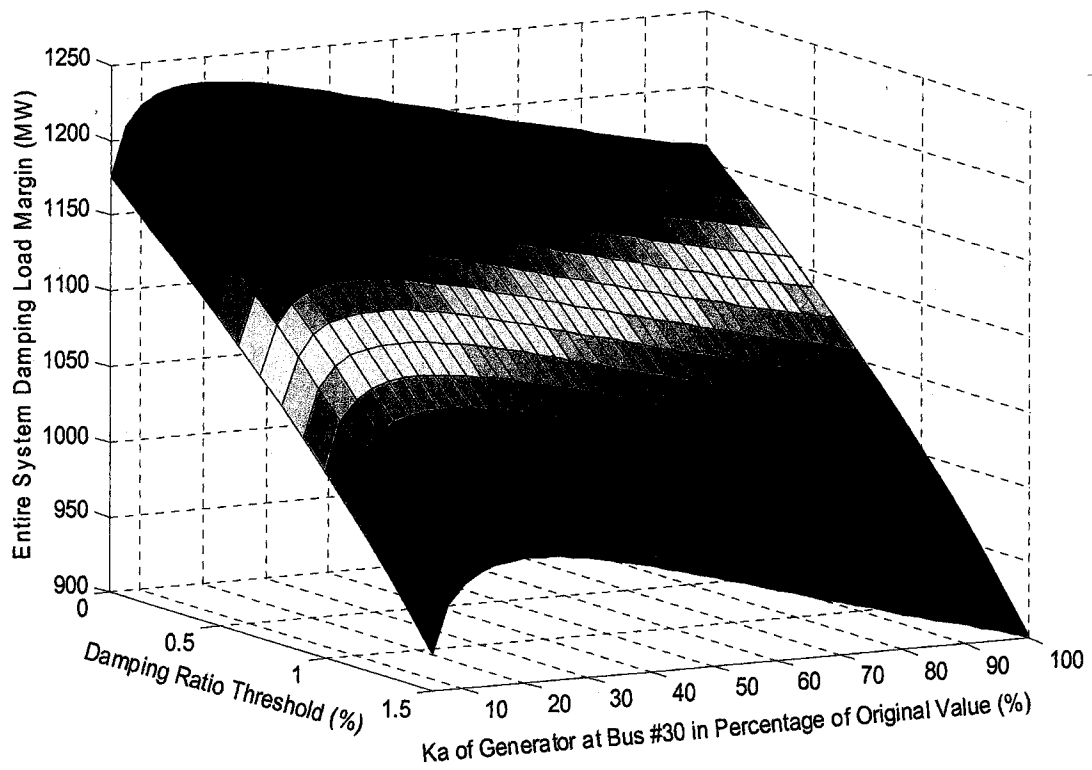


Figure 5.9 damping margin surface with  $K_a$  of generator at bus #30

In the Fig. 5.10, all the points on the curve B- A have the same damping margin of 1160MW. The curve basically provides the relationship between the damping ratio and  $\beta$  to maintain the same damping margin. This information is valuable to decide optimal control actions to maintain a specific damping or oscillatory stability margin for changing system conditions.

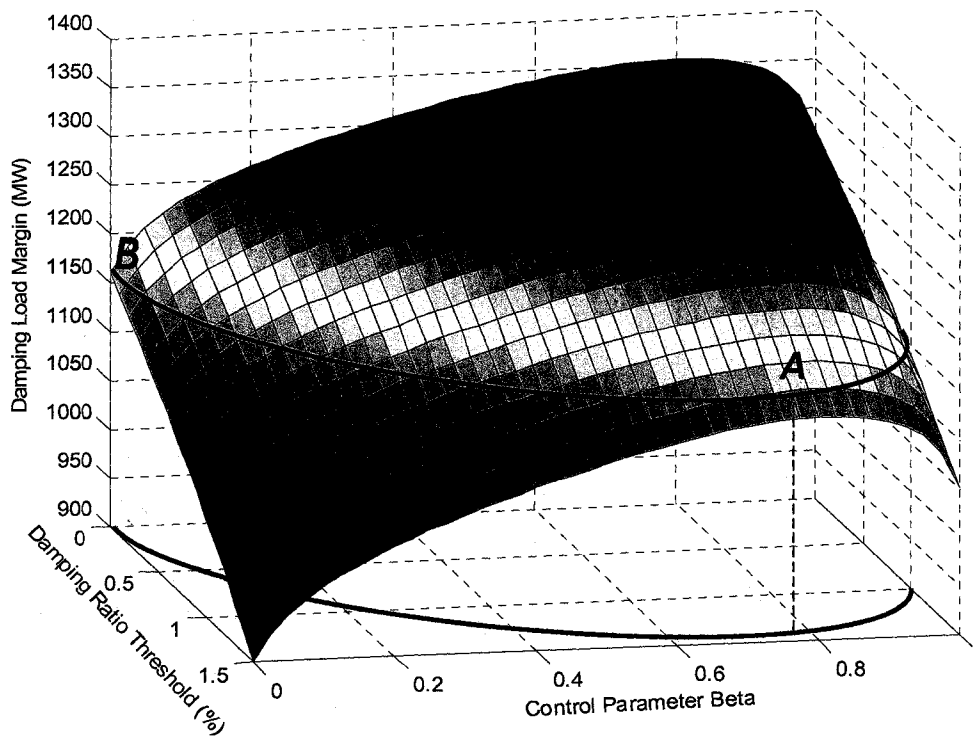


Figure 5.10 damping margin surface when  $K_a$  of three selected generators varying

Similar to oscillatory stability margin boundary tracing, the damping margin tracing technique can also compute the post-contingency boundary without tracing the whole post-contingency P-V curve. Fig 5.11 shows the simulation result for the damping margin with an outage of line 3-18. The damping threshold values from the top to the bottom are 0%, 0.5%, 1.0%, and 1.5%.

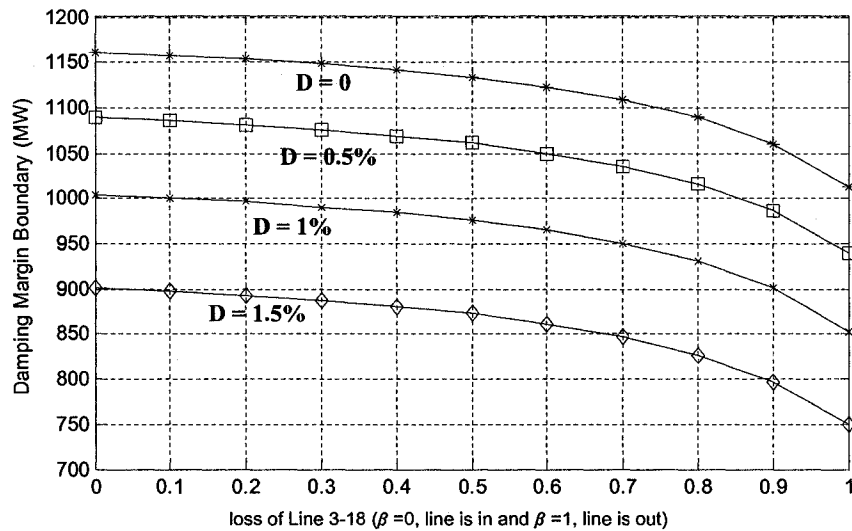


Figure 5.11 damping margin with outage of line 3-18

#### 5.4 Computational aspects

The above boundary tracing can also be obtained by repeatedly applying the eigenvalues tracing method described in chapter 4. The process is illustrated in Fig 5.12.

At base case, where  $\beta = \beta_0$ , an eigenvalue tracing method can be applied to identify the Hopf bifurcation point or damping margin point. The  $\alpha_0$  in Fig 5.12 represents the oscillatory stability margin with parameter  $\beta = \beta_0$ . This is the starting point of the margin boundary.

To identify the next margin point on the boundary, one can fix  $\alpha$  value as  $\alpha_0$ , and let  $\beta = \beta_1$ . After solving a power flow, one can get point  $B_1$ . At point  $B_1$ , one has to calculate the eigenvalue with a maximum real part to check whether or not the margin point is reached.

If not, the eigenvalue tracing method has to be applied to trace the P-V curve for the margin point with  $\beta = \beta_1$ .

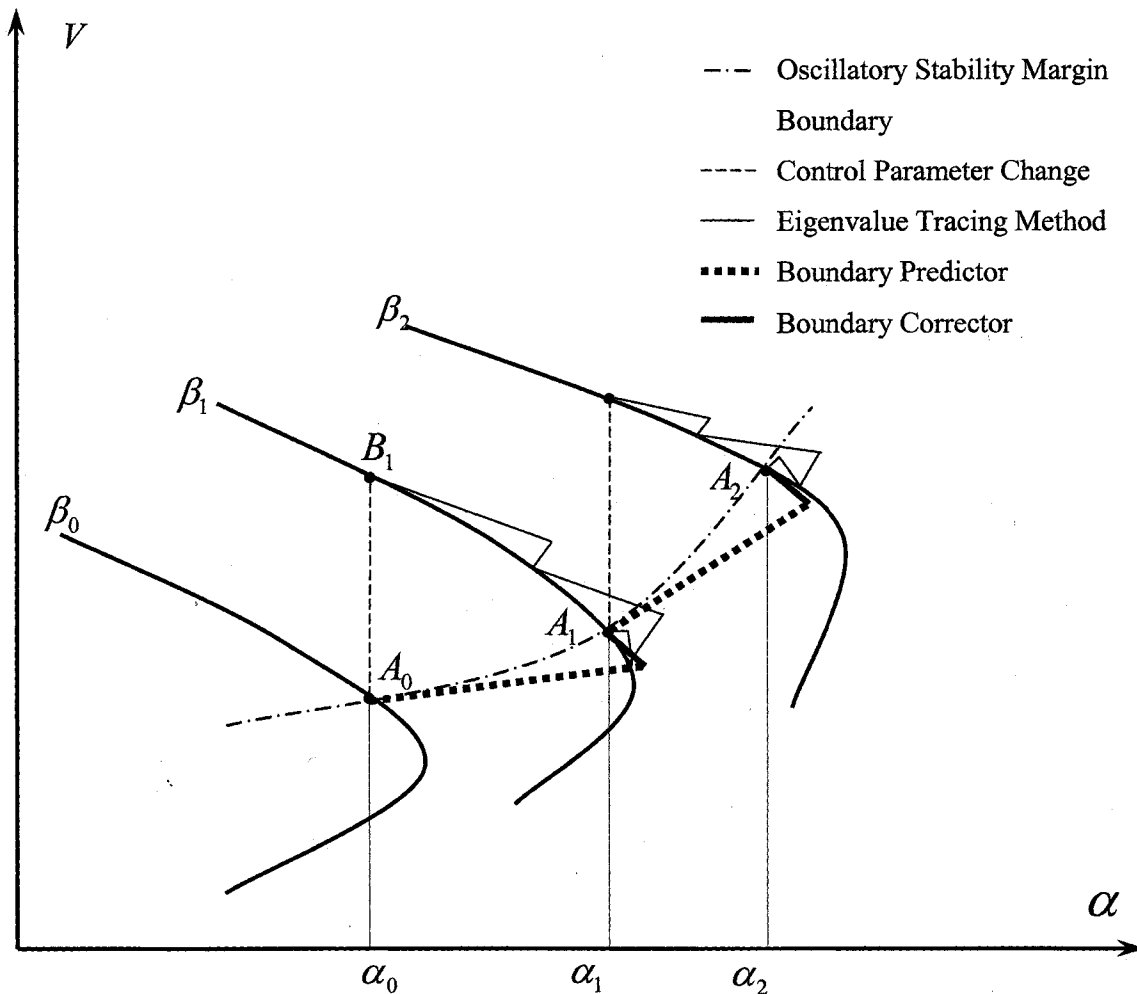


Figure 5.12 Oscillatory stability margin boundary tracing vs. eigenvalue tracing

In the tracing process from point  $B_1$  to  $A_1$ , multiple steps are needed. Each step needs to solve power flow and eigenvalue information.

For the proposed margin boundary tracing method, getting  $A_1$  from  $A_0$  will be faster than the above process. The boundary tangent can be obtained by solving linear equation



If the LU factorization of  $\begin{bmatrix} F_X & F_Y \\ G_X & G_Y \end{bmatrix}$  is known, there won't be too much computational cost to find the LU factorization of matrix for (5.4.1) or (5.4.2). i.e. solving this  $3(m+n+1)$  dimension linear equation (5.4.1) or (5.4.2) doesn't cost too much computational time.

How to handle a special case which may not appear in practical situations is presented in Appendix 3.

## Chapter 6 Conclusions

This dissertation proposes novel algorithms for power system oscillatory stability assessment. An integration based eigenvalue tracing method is proposed to trace any specified eigenvalue of interest. Also a margin boundary tracing algorithm that can trace not only the oscillatory stability margin boundary, but also damping margin boundary is proposed. The eigenvalues tracing method can trace any eigenvalues of interest. An eigenvalue index is proposed to rank the eigenvalues. This index is helpful for identifying the rate of change of movement and the direction of movement for these eigenvalues with respect to change in any parameter of interest. This approach is used to identify Hopf bifurcation. It is also extended to satisfy minimum damping margin constraints.

The salient points and contributions of this approach are:

- For the first time, the eigenvalue tracing algorithm is introduced into power system oscillatory stability assessment.
- In the oscillatory stability margin identification, existing methods are either difficult to converge (like the direct method), or comparably slower (like the secant method). In some base cases, where the eigenvalue with the maximum real part moves toward the left-hand side on complex plane, the secant method will predict a negative step length which will prevent this method from finding margin point. The proposed eigenvalue tracing method in this dissertation can overcome these disadvantages by introducing the eigenvalue index. This index can find an eigenvalue which is more likely to



cross the imaginary axis than the eigenvalue with the maximum real part. Similarly, the eigenvalue damping index can help to trace the right eigenvalue in damping margin identification

- The proposed method avoids repeated calculation of the dominant eigenvalue for changing operating conditions. It also takes into account the rate of change as well as the direction of the movement of the eigenvalue in step length prediction. Thus the entire process to identify oscillatory stability margin is accelerated.
- The method can be used to get both oscillatory stability and damping ratio information.
- Eigenvalue and Eigenvector sensitivities are by-products of this approach
- This method is faster and more robust than the secant method, especially for large scale systems.
- Can trace close eigenvalues without any numerical problems

The dissertation discusses the computational advantages of this algorithm in detail, and demonstrates the potential convergence problems with the secant method. The eigenvector derivatives can be used to obtain participation factor derivatives. The participation factors are generally indicative of the relative participations of the respective states in the corresponding modes (or eigenvalues), so the derivatives of these factors can further show how this association will change for any parameter variation of interest.

Without the proposed margin boundary (both oscillatory and damping) tracing algorithm one has to repeat tracing the P-V curve with a different parameter  $\beta$  value to get the oscillatory stability margin boundary and the damping margin boundary. To get one margin point on the boundary, one has to search multiple operating points in tracing one P-V curve. Each operating point requires solving a power flow with Newton- Raphson or other algorithms. The proposed method can speed-up this process with help from the boundary predictor and corrector.

The salient points and contribution of this approach are:

- Tangent vector can be used to predict local sensitivity information
- Can estimate the margin for any large change in system parameters faster than the existing methods.
- The approach provides the relevant information about the nonlinear characteristics between margin and control parameters, by which one can find not only the control parameter values to maximize the margin, but also the control parameter perturbation tolerance, which can help keep the system more robust
- For the first time, an algorithm to trace the damping margin boundary is proposed

The eigenvalues tracing and margin boundary tracing methodologies proposed in this thesis will make contribution to future on-line stability assessment tools for large scale power systems.

## References

- [1] Dmitry Kosterev, Bill Mittelstadt, Mike Viles, Brian Tuck, Jim Burns, Mike Kwok, Jorge Jardim, and Gerry Garnett, "Model validation and analysis of WSCC system oscillations following Alberta separation on August 4, 2000", Final Report, January 25, 2000, [http://www.transmission.bpa.gov/orgs/opi/Wide\\_Area/August4Report.doc](http://www.transmission.bpa.gov/orgs/opi/Wide_Area/August4Report.doc).
- [2] P. M. Anderson, "Power system oscillations summary of utility experience".
- [3] P. Kundur, "Power system stability and control", McGraw-Hill, New York, 1994.
- [4] M. Ilic, J. Zaborszky, "Dynamics and control of large electric power systems", John Wiley & Sons, New York, 2000.
- [5] L. H. Fink, ed. "Proceeding: Bulk power system dynamics and control IV restructuring", Santorini, Greece, August 23-28, 1998.
- [6] B. D. Hassard, N. D. Kazarinoff and Y. -H. Wan, "Theory and applications of Hopf bifurcation", Cambridge University Press, Cambridge, 1981.
- [7] I. Dobson, F. Alvarado and C.L DeMarco, "Sensitivity of Hopf bifurcations to power system parameters," Proceedings of the 31st IEEE Conference on Decision and Control, 1992 vol. 3, pp. 2928 – 2933.
- [8] C.D. Vournas, M.A. Pai, P.W. Sauer, "The effect of automatic voltage regulation on the bifurcation evolution in power systems," IEEE Transactions on Power Systems, vol. 11, Nov. 1996, pp. 1683-1688.

- [9] W. Zhu, R.R. Mohler, R. Spee, W.A. Mittelstadt, D. Maratukulam, "Hopf bifurcations in a SMIB power system with SSR," *IEEE Transactions on Power Systems*, vol. 11, Aug. 1996, pp. 1579- 1584.
- [10] V. Ajjarapu, B. Lee. " Bifurcation theory and its application to nonlinear dynamical phenomena in an electrical power system,"*IEEE Transactions on Power Systems*, vol. 7, pp. 424-431 Feb. 1992.
- [11] K. Kim, H. Schattler, V. Venkatasubramanian, J. Zaborszky, P. Hirsch, "Methods for calculating oscillations in large power systems," *IEEE Transactions on Power Systems*, vol. 12, Nov. 1997, pp. 1639-1648.
- [12] F. Howell, V. Venkatasubramanian, "Transient stability assessment with unstable limit cycle approximation," *IEEE Transactions on Power Systems*, vol. 14, May 1999, pp. 667 -677.
- [13] Zhihong Feng, Venkataramana Ajjarapu, and Bo Long, "Indentification of Voltage Collapse Through Direct Equilibrium Tracing", *IEEE Transactions on Power Systems*, vol. 15, no. 1, Feb2000. pp. 342-349.
- [14] Dmitry Kosterev, Bill Mittelstadt, Mike Viles, Brian Tuck, Jim Burns, Mike Kwok, Jorge Jardim, and Gerry Garnett, "Model Validation and Analysis of WSCC System Oscillations following Alberta Separation on August 4, 2000", Final Report, January 25, 2000, [www.transmission.bpa.gov/orgs/opi/Wide\\_Area/August4Report.doc](http://www.transmission.bpa.gov/orgs/opi/Wide_Area/August4Report.doc).
- [15] P.W. Sauer and M.A. Pai, "Power system steady-state stability and the load flow Jacobian," *IEEE Transaction on power systems*, PWRS-5, Nov. 1990, pp. 1374-1383.

[16] M. Kubicek, M. Marek, "Computational Methods in Bifurcation Theory and Dissipative Structures", Springer Series in Computational Physics, 1983 by Springer-Verlag New York Inc.

[17] Yuri V. Makarov, Zhao Yang Dong, and David J. Hill, "A general method for small signal stability analysis" IEEE Transaction on Power System, vol. 13, no. 3, August 1998, pp. 979-985.

[18] David J. Hill, and Zhao-Yang Dong, "Nonlinear Computation and Control for Small Disturbance Stability", Panel Session on Recent Applications of Small-Signal Stability Techniques, Proc. of IEEE Summer Meeting, Seattle, July 2000.

[19] Zhao-Yang Dong, Y. V. Makarov, and D. J. Hill, "Analysis of Small Signal Stability Margins Using Genetic Optimization", Electric Power Systems Research, 46 (1998) pp. 195-204.

[20] K. Kim, H. Schattler, V. Venkatasubramanian, J. Zaborszky, P. Hirsch, "Methods for calculating oscillations in large power systems," IEEE Transactions on Power Systems, vol. 12, Nov. 1997, pp. 1639-1648.

[21] George Angelidis and Adam Semlyen, "Improved methodologies for the calculation of critical eigenvalues in small signal stability analysis" IEEE Transaction on Power System, vol. 11, no. 3, August 1996, pp. 1209-1217.

[22] George Angelidis and Adam Semlyen, "Efficient calculation of critical eigenvalue clusters in the small signal stability analysis of large power systems", IEEE Transaction on Power System, vol. 10, no. 1, Feb 1995, pp. 427-432.

[23] Jorge M. Campagnolo, Nelson Martins and Djalma M. Falcao, "An efficient and robust eigenvalue method for small-signal stability assessment in parallel computers" IEEE Transaction on Power System, vol. 10, no. 1, Feb 1995, pp. 506-511.

[24] Leonardo T.G. Lima, Licio H. Bezerra, Carlos Tomei, and Nelson Martins, "New method for fast small-signal stability assessment of large scale power systems", IEEE Transaction on Power System, vol. 10, no. 4, Nov 1995, pp. 1979-1985.

[25] Nelson Martins, Leonardo T.G. Lima, and Jerminio J.C.P. Pinto, "Computing dominant poles of power system transfer function", IEEE Transaction on Power System, vol. 11, no. 1, Feb 1996, pp. 162-170.

[26] Jorge M. Campagnolo, Nelson Martins, Jose L.R. Pereira, Leonardo T.G. Lima, Herminio J.C.P. Pinto and Djalma M. Falcao, "Fast small-signal stability assessment using parallel processing", IEEE Transaction on Power System, vol. 9, no. 2, May 1994, pp. 949-956.

[27] Jorge M. Campagnolo, Nelson Martins and Djalma M. Falcao, "Refactored Bi-Iteration: A high performance eigensolution method for large power system matrices" IEEE Transaction on Power System, vol. 11, no. 3, August 1996, pp. 1228-1235.

[28] L. Wang, and A. Semlyen, "Application of sparse eigenvalue techniques to the small signal stability analysis of large power systems", IEEE Transactions on Power Systems, vol. 5 no. 2, May 1990, pp. 635 -642.

[29] Yuan Zhou; and V. Ajjarapu, "Identification and tracing of voltage and oscillatory stability margins", Power Engineering Society Winter Meeting, 2002. IEEE , vol. 1 , 27-31 Jan. 2002 pp. 250 -255.

[30] S. Gomes Jr., N. Martins, and C. Portela, "Computing small-signal stability boundaries for large-scale power systems", IEEE Transactions on Power Systems, , vol. 18, no. 2 , May 2003, pp. 747 -752.

[31] Yuan Zhou, and Venkataramana Ajjarapu, "A Fast Algorithm for Identification and Tracing of Voltage and Oscillatory Stability Margin Boundaries", Proceedings of the IEEE, Vol. 93, No. 5, May 2005.

[32] Robert Kalaba, Karl Spingarn, and Leigh Tesfatsion, "Variational Equations for the eigenvalues and eigenvectors of nonsymmetric matrices", Journal of Optimaization Theory and Applications: vol. 33, no. 1, Jan 1981, pp. 1-8.

[33] Robert Kalaba, Karl Spingarn, and Leigh Tesfatsion, "Individual tracking of an eigenvalue and eigenvector of parameterized matrix", Nonlinear Analysis, Theory, Methods & Application, vol. 5, no. 4, pp. 337-340.

[34] N. Martins, "The dominant pole spectrum eigensolver", IEEE Transactions on Power Systems, vol. 12, no. 1, Feb. 1997, pp. 245 -254.

[35] William F. Tinney, and John W. Walker, "Direct Solutions of Sparse Network Equations by Optimally Ordered Triangular Factorization", Proceedings of the IEEE, Nov 1967, pp. 1801-1809.

[36] T. Van. Cutsem, " An approach to corrective control of voltage instability using simulation and sensitivity," IEEE Transactions on Power Systems, PWRS-10, May 1995, pp. 616-622.







## NEW\_ENGLAND SYSTEM STABILITY RELATED PARAMETERS OF GENERATOR &amp; EXCITATION &amp; GOVERNOR

Generator transient parameter follows

Num	Gen_name	Xd	Xq	X'd	X'q	Rs	T'do	T'qo	Mg	Dg
30	BUS30	0.1000	0.0690	0.0310	0.0690	0.0002	10.2000	0.010	84.000	5.000
31	BUS31	0.2590	0.2820	0.0700	0.1700	0.0002	6.5600	1.5000	60.600	5.000
32	BUS32	0.2500	0.2370	0.0530	0.0880	0.0002	5.7000	1.5000	71.600	5.000
33	BUS33	0.2620	0.2580	0.0440	0.1660	0.0002	5.6900	1.5000	57.200	5.000
34	BUS34	0.6700	0.6200	0.1320	0.1660	0.0002	5.4000	0.4400	52.000	5.000
35	BUS35	0.2540	0.2410	0.0500	0.0810	0.0002	7.3000	0.4000	69.600	5.000
36	BUS36	0.2950	0.2920	0.0490	0.1860	0.0002	5.6600	1.5000	52.800	5.000
37	BUS37	0.2900	0.2800	0.0570	0.0910	0.0010	6.7000	0.4100	48.600	5.000
38	BUS38	0.2110	0.2050	0.0570	0.0590	0.0002	4.7900	1.9600	69.000	5.000
39	BUS39	0.0200	0.0190	0.0060	0.0080	0.0002	7.0000	0.7000	1000.000	10.000

-999

Generator control system ( excitor + AVR + governor ) parameter follows

Num	Gen_name	Ke	Te	Se	Ka	Ta	Kf	Tf	Tch	Tg	Rg
30	BUS30	1.0000	0.2500	0.0000	20.0000	0.0600	0.0400	1.0000	1.6000	0.2000	0.0500
31	BUS31	1.0000	0.4100	0.0000	40.0000	0.0500	0.0600	0.5000	54.1000	0.4500	0.0500
32	BUS32	1.0000	0.5000	0.0000	40.0000	0.0600	0.0800	1.0000	10.0000	3.0000	0.0500
33	BUS33	1.0000	0.5000	0.0000	40.0000	0.0600	0.0800	1.0000	10.1800	0.2400	0.0500
34	BUS34	1.0000	0.7900	0.0000	30.0000	0.0200	0.0300	1.0000	9.7900	0.1200	0.0500
35	BUS35	1.0000	0.4700	0.0000	40.0000	0.0200	0.0800	1.2500	10.0000	3.0000	0.0500
36	BUS36	1.0000	0.7300	0.0000	30.0000	0.0200	0.0300	1.0000	7.6800	0.2000	0.0500
37	BUS37	1.0000	0.5300	0.0000	40.0000	0.0200	0.0900	1.2600	7.0000	3.0000	0.0500
38	BUS38	1.0000	1.4000	0.0000	20.0000	0.0200	0.0300	1.0000	6.1000	0.3800	0.0500
39	BUS39	1.0000	1.0000	0.0000	20.0000	0.0200	0.0300	1.0000	10.0000	2.0000	0.0500

-999

## Appendix 2: Application of symbolic computing

In proposed eigenvalue tracing method, (4.1.12) needs to be solved.

$$\begin{bmatrix} \lambda_R I - F_X & -\lambda_I I & -F_Y & 0 & v_R & -v_I \\ \lambda_I I & \lambda_R I - F_X & 0 & -F_Y & v_I & v_R \\ -G_X & 0 & -G_Y & 0 & 0 & 0 \\ 0 & -G_X & 0 & -G_Y & 0 & 0 \\ v_R^T & -v_I^T & 0 & 0 & 0 & 0 \\ v_I^T & v_R^T & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{v}_R \\ \dot{v}_I \\ \dot{u}_R \\ \dot{u}_I \\ \dot{\lambda}_R \\ \dot{\lambda}_I \end{bmatrix} = \begin{bmatrix} \dot{F}_X v_R + \dot{F}_Y u_R \\ \dot{F}_X v_I + \dot{F}_Y u_I \\ \dot{G}_X v_R + \dot{G}_Y u_R \\ \dot{G}_X v_I + \dot{G}_Y u_I \\ 0 \\ 0 \end{bmatrix} \quad (4.1.12)$$

The derivative of Jacobian matrix on the right hand side of (4.1.12) is required. The expression of element on  $i$ th row and  $j$ th column of  $\dot{F}_Y$  is given in (4.1.13)

$$\dot{F}_{Y(ij)} = \frac{\partial F_{Y(ij)}}{\partial X} \frac{\partial X}{\partial \alpha} + \frac{\partial F_{Y(ij)}}{\partial Y} \frac{\partial Y}{\partial \alpha} + \frac{\partial F_{Y(ij)}}{\partial \alpha} \quad (4.1.13)$$

In the program code, the explicit expression of Jacobian derivative has to be included to speed up the code execution time.

To show the complexity of this expression, the code to compute just one element of

$\dot{F}_Y$  is shown below:

```
Fydot (jj*9+2, jj*2+2) =
Fydot (jj*9+2, jj*2+2)+Tangent (jj*9+1) * ( (Xdprime (genNo) * (Rs (genNo) *X (m+jj*2+1)
) *sin (-X (jj*9+1)+X (m+jj*2+2)) -Xqprime (genNo) *X (m+jj*2+1) *cos (-
X (jj*9+1)+X (m+jj*2+2)) ) / (Rs (genNo) ^2+Xdprime (genNo) *Xqprime (genNo) ) ^2 * (Rs (g
enNo) *X (jj*9+3) -X (jj*9+4) *Xdprime (genNo) -Rs (genNo) *X (m+jj*2+1) *cos (-
X (jj*9+1)+X (m+jj*2+2)) -Xdprime (genNo) *X (m+jj*2+1) *sin (-
X (jj*9+1)+X (m+jj*2+2)) ) +Xdprime (genNo) * (Rs (genNo) *X (m+jj*2+1) *cos (-
X (jj*9+1)+X (m+jj*2+2)) +Xqprime (genNo) *X (m+jj*2+1) *sin (-
X (jj*9+1)+X (m+jj*2+2)) ) / (Rs (genNo) ^2+Xdprime (genNo) *Xqprime (genNo) ) ^2 * (-
Rs (genNo) *X (m+jj*2+1) *sin (-
```

```

X(jj*9+1)+X(m+jj*2+2))+Xdprime(genNo)*X(m+jj*2+1)*cos(-
X(jj*9+1)+X(m+jj*2+2)))+Xdprime(genNo)*(-Rs(genNo)*X(m+jj*2+1)*cos(-
X(jj*9+1)+X(m+jj*2+2))-Xqprime(genNo)*X(m+jj*2+1)*sin(-
X(jj*9+1)+X(m+jj*2+2)))/(Rs(genNo)^2+Xdprime(genNo)*Xqprime(genNo))^2*(Rs(g
enNo)*X(m+jj*2+1)*sin(-X(jj*9+1)+X(m+jj*2+2)))-
Xdprime(genNo)*X(m+jj*2+1)*cos(-X(jj*9+1)+X(m+jj*2+2)))-X(jj*9+3)-
Xdprime(genNo)*(Rs(genNo)*X(jj*9+4)+X(jj*9+3)*Xqprime(genNo)+Rs(genNo)*X(m+
jj*2+1)*sin(-X(jj*9+1)+X(m+jj*2+2))-Xqprime(genNo)*X(m+jj*2+1)*cos(-
X(jj*9+1)+X(m+jj*2+2)))/(Rs(genNo)^2+Xdprime(genNo)*Xqprime(genNo)))*(-
Rs(genNo)*X(m+jj*2+1)*cos(-X(jj*9+1)+X(m+jj*2+2)))-
Xdprime(genNo)*X(m+jj*2+1)*sin(-
X(jj*9+1)+X(m+jj*2+2)))/(Rs(genNo)^2+Xdprime(genNo)*Xqprime(genNo))-
Xqprime(genNo)*(-Rs(genNo)*X(m+jj*2+1)*cos(-X(jj*9+1)+X(m+jj*2+2))-
Xdprime(genNo)*X(m+jj*2+1)*sin(-
X(jj*9+1)+X(m+jj*2+2)))/(Rs(genNo)^2+Xdprime(genNo)*Xqprime(genNo))^2*(Rs(g
enNo)*X(jj*9+4)+X(jj*9+3)*Xqprime(genNo)+Rs(genNo)*X(m+jj*2+1)*sin(-
X(jj*9+1)+X(m+jj*2+2))-Xqprime(genNo)*X(m+jj*2+1)*cos(-
X(jj*9+1)+X(m+jj*2+2)))-Xqprime(genNo)*(Rs(genNo)*X(m+jj*2+1)*sin(-
X(jj*9+1)+X(m+jj*2+2))-Xdprime(genNo)*X(m+jj*2+1)*cos(-
X(jj*9+1)+X(m+jj*2+2)))/(Rs(genNo)^2+Xdprime(genNo)*Xqprime(genNo))^2*(-
Rs(genNo)*X(m+jj*2+1)*cos(-X(jj*9+1)+X(m+jj*2+2)))-
Xqprime(genNo)*X(m+jj*2+1)*sin(-X(jj*9+1)+X(m+jj*2+2)))-Xqprime(genNo)*(-
Rs(genNo)*X(m+jj*2+1)*sin(-
X(jj*9+1)+X(m+jj*2+2))+Xdprime(genNo)*X(m+jj*2+1)*cos(-
X(jj*9+1)+X(m+jj*2+2)))/(Rs(genNo)^2+Xdprime(genNo)*Xqprime(genNo))^2*(Rs(g
enNo)*X(m+jj*2+1)*cos(-
X(jj*9+1)+X(m+jj*2+2))+Xqprime(genNo)*X(m+jj*2+1)*sin(-
X(jj*9+1)+X(m+jj*2+2)))-X(jj*9+4)+Xqprime(genNo)*(Rs(genNo)*X(jj*9+3)-
X(jj*9+4)*Xdprime(genNo)-Rs(genNo)*X(m+jj*2+1)*cos(-X(jj*9+1)+X(m+jj*2+2))-
Xdprime(genNo)*X(m+jj*2+1)*sin(-
X(jj*9+1)+X(m+jj*2+2)))/(Rs(genNo)^2+Xdprime(genNo)*Xqprime(genNo)))*(Rs(g
enNo)*X(m+jj*2+1)*sin(-X(jj*9+1)+X(m+jj*2+2)))-
Xqprime(genNo)*X(m+jj*2+1)*cos(-
X(jj*9+1)+X(m+jj*2+2)))/(Rs(genNo)^2+Xdprime(genNo)*Xqprime(genNo))/Mg(gen
No));

```

Without the symbolic computing, one has to derive similar expression for all

$$\text{elements of } \begin{bmatrix} \dot{F}_X & \dot{F}_Y \\ \dot{G}_X & \dot{G}_Y \end{bmatrix}.$$

It's very difficult for any one to get this result manually without any mistakes.

During the programming process of proposed eigenvalue tracing algorithm, we found a solution based on MATLAB symbolic computing functions.

Let's illustrate this process by an example

The MATLAB code consists of three lines:

```
syms x
y=x^3;
ydot = jacobian(y,x)
The execution result is:
```

```
ydot = 3*x^2
```

First line defines  $x$  as symbolic variable. The second line defines  $y=x^3$ . The third line calculate the  $\frac{dy}{dx}$  with MATLAB provided function “jacobian”. The result is  $3x^2$ .

In our programming for eigenvalue tracing, the explicit expression of derivative of Jacobian matrix is derived from MATLAB symbolic computing. This has helped make our code more mistake-proof and speed-up the programming process. In the future industry application of this algorithm, this symbolic technique is suggested to help produce source code.

### Appendix 3: Some special cases in oscillatory stability margin boundary tracing

In all simulated cases, the critical eigenvalue in first margin point always is the critical eigenvalue at every traced margin point during the entire tracing process.

For some very nonlinear system, theoretically there is the possibility that one pair of conjugate complex eigenvalues is the critical eigenvalue with  $[\beta_0, \beta_1]$ , and the other different pair of conjugate complex eigenvalues becomes the critical eigenvalue with  $[\beta_1, \beta_2]$ .

The critical eigenvalue could be switched as shown in Fig 1. The thick line from point A to D and C is actual margin boundary.

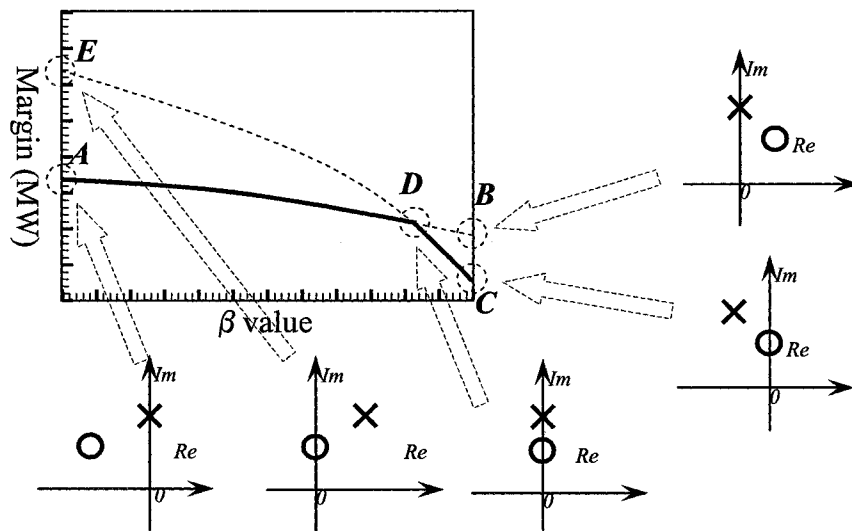


Figure 1 When Margin Boundary with Two Critical Eigenvalues

To handle this situation, we improved our procedure as follow:

- 1) With predictor and corrector, tracing margin boundary from point A to B

2) At the last traced margin point, compute system dominant eigenvalue, if no other eigenvalue has crossed imaginary axis, the problem is solved.

Otherwise, go to step 3).

3) With the eigenvalue tracing method presented in Chapter 4, find the actual oscillatory stability margin, (which is shown as point C in Fig 1). Trace margin boundary backward and/or forward with  $\beta$  value decreasing or increasing, until margin value equal to previous traced margin boundary with the same  $\beta$  value. (This process can be shown as from point C to point D in Fig 1.) Then, go to step 2).

## Acknowledgements

I would like to thank Dr. Venkataramana Ajarapu for his patience, encouragement, and long hours without which I could have never accomplished this task. There are no adequate words to convey my appreciation for all of the support.

I always need to express my gratitude to my parents, Junxian and Changli Wen, for instilling in me a determination to succeed and love of knowledge that were necessary foundations in my pursuit of this degree. The love and encouragement of my entire family kept me going through it all.

I also sincerely appreciate the assistance of Dr. Vijay Vittal, Dr. James D. McCalley, Dr. Degang J. Chen, and Dr. Wolfgang Kliemann as being my committee members and so much advice from you enlightens my way to research accomplishment.

I would finally like to thank all of my friends, especially, Elias (Leo) Villanueva, who paid abundant time, patience, and support.