รายการอ้างอิง

[1] Byerly, R.T. Sherman, D.E. and Benon, R.J. Frequency domain analysis of low frequency oscillations EPRI El-726, Part 2, Project 744-1(April 1978)

[2] C.C. Young. Equipment and system modelling for large-scale stability studies IEEE Trans PAS (June 1971) : pp. 99-109.

[3] C.D. Vournas, B.C. Papadias. Power system stabilization via parameter optimization application to the Hellenic interconnected system IEEE Trans. PAS Vol. PWRS-2 (August 1987) : pp. 615-623

[4] Chern-Lin Chen, Yuan-Yih Hsu. Coordinated synthesis of multimachine power system stabilizer using an efficient decentralized modal control (DMC) algorithm IEEE Trans . PAS Vol. PWRS-2 (August 1987) : pp. 543-551.

[5] E.F. Richards, A.B.R. Kumar. An opimal control law by eigenvalue assignment for improved dynamic stability in power systems. IEEE Trans. PAS Vol. PAS-101(June 1982) : pp. 1570-77

[6] E.V. Larsen, D.A. Swann. Applying power system stabilizers part I : General concepts IEEE Trans. PAS Vol. PAS-100 (June, 1981) : pp. 3017-24.

[7] E.V. Larsen, D.A. Swann. Applying power system stabilizers part III : Practical Considerations. IEEE Trans. PAS Vol. 100 (June 1981) : pp. 3034-41.

[8] F.P. Demello, C. Concordia. Concepts of synchronous machine stability as affected by excitation control. IEEE Trans. PAS Vol. PAS-88 (April 1969) : pp. 316-329

[9] F.P. Demello, P.J. Nolan, T.F. Laskowski, J.M. Undrill. Coordinated application of stabilizers in multimachine power systems IEEE Trans. PAS Vol. PAS-99 (May/June 1980) : pp. 892-901.

[10] G. Gross, C.F. Imparato, P.M.Look. A tool for comprehensive analysis of power system dynamic stability IEEE Trans. PAS Vol. PAS-101 (January 1982) : pp. 226-234.

[11] Glenn W. Stagg, Ahmed H. El-Abiad. Computer methods in power system analysis McGRAW-HILL (1968).

[12] H.B. Gooi, E.F. Hill, M.A. Mobarok, D.H. Thorne, T.M. Lee. Coordinated multi-machine stabilizers setting without eigenvalue drift IEEE Trans. PAS Vol. PAS-100 (1981) : pp. 3879-87.

[13] IEEE committee report. Computer representation of excitation systems. IEEE Trans. PAS Vol. PAS-87 (June 1968) : pp. 1460-64.

[14] IEEE committee report. Excitation system model for power system stability IEEE Trans. PAS Vol. PAS-100 (February 1981) : pp. 494-509.

[15] IEEE committee report. Proposed excitation system definitions for synchronous machine IEEE Trans. PAS Vol. PAS-88 (August 1969) : pp. 1248-58.

[16] John M. Undrill. Dynamic stability calculations for an arbitrary number of interconnected synchronous machines IEEE Trans. PAS Vol. PAS-87 (March 1968) : pp. 835-44.

[17] P.M. Anderson, A.A. Found Power system control and stability Iowa State University Press (1977).

[18] R.J. Fleming, M.A. Mohan. K. Parvatisam. Selection of parameters of stabilizers in multimachine power systems. IEEE Trans. PAS Vol. PAS-100 (May 1981) : pp. 2329-33.

[19] R.T. Byerly, R.J. Bennon, D.E. Sherman. Eigenvalue analysis of synchronizing power flow oscillations in large electric power systems IEEE Trans PAS Vol. PAS-101 (January 1982) : pp. 235-243.

[20] S. Abe, A. Doi. A new power system stabilizers synthesis in multimachine power systems IEEE Trans. PAS Vol. 102 (1983) : pp. 3910-18.

[21] S. Abe, A.Doi. Coordinated synthesis of power system stabilizers in multimachine power systems IEEE Trans. PAS Vol. PAS-103 (June 1984) : pp. 1473-79.

[22] S. Lefebvre. Tuning of stabilizers in multimachine power systems IEEE Trans. PAS Vol. 102 (1983) : pp. 290-299.

[23] The MATH works Inc. MATLAB, Reference Guide (August 1992).

[24] The MATH works Inc. MATLAB, User's Guide (February 1993).

[25] Wah-Chun, Yuan-Yih. An optimal variable structure stabilizer for power system stabilation. IEEE Trans. PAS Vol. PAS-102 (June 1983) : pp.1738-46

[26] William L. Brogan. Modern control theory Prentice-Hall (l99l).

[27] Yao-nan Yu, Hamdy A.M. Moussa. Optimal stabilization of a multi-machine system IEEE Trans. PAS Vol. PAS-91 (May/June l972).: pp 1174-82

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

หอสมุดกลาง สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก ก.

การใช้โปรแกรมสำหรับคำนวณหาสเตทสเปซและค่าตัวแปรของอุปกรณ์สเตบิไลเซอร์

โปรแกรมสำหรับคำนวณหาสเตทสเปซและค่าตัวแปรของอุปกรณ์สเตบิไลเซอร์ในวิทยานิพนธ์ฉบับนี้ เป็นโปรแกรมที่พัฒนาขึ้นมาจากการใช้งานโปรแกรมการคำนวณแม็ทแล็บ (MATLAB) เวอร์ชั่น 4.0 ของ The Math workss, Inc. ซึ่งเป็นโปรแกรมที่ทำงานภายใต้ระบบปฏิบัติการไมโครซอฟท์ วินโดว์ (Microsoft Windows) ดังนั้น ในการใช้งานโปรแกรมนี้ ผู้ใช้จะต้องเข้าสู่ระบบการทำงานของวินโดว์ก่อน แล้วเรียกใช้โปรแกรมแม็ทแล็บ แล้วจากโปรแกรมแม็ทแล็บนี้จะสามารถเรียกใช้โปรแกรมที่ได้พัฒนาในการใส่ข้อมูล, คำนวณหาสเตทสเปซ, คำนวณหาค่าตัวแปรของอุปกรณ์สเตบิไลเซอร์ และหาค่าเจาะจงของระบบไฟฟ้ากำลัง

ก.1 การใส่ข้อมูล

การใส่ข้อมูลต่าง ๆ ที่จำเป็น จะใส่ผ่านโปรแกรมเวิร์ดโปรเซสเซอร์ (Word Processor) ลงไปในไฟล์ข้อมูลที่ได้เตรียมไว้แล้ว โดยแนะนำให้ใช้โปรแกรมโน๊ตแพด (Notepad) ซึ่งเป็นโปรแกรมที่ระบบปฏิบัติการวินโดว์มีไว้ให้ใช้งานได้เลย และเป็นโปรแกรมที่มีขนาดเล็กสามารถเรียกใช้งานได้ง่ายไม่เสียเวลา

วิธีการใส่ข้อมูลทำได้โดยจากโปรแกรมโน๊ตแพด ให้เปิดไฟล์ข้อมูลที่ต้องการจะใส่ข้อมูลลงไป จากนั้น ทำการป้อนข้อมูลผ่านทางแป้นพิมพ์ลงไปในไฟล์ข้อมูล โดยให้มีรูปแบบ (format) ตามที่กำหนดไว้ใน หัวข้อ ก.2 เมื่อป้อนข้อมูลครบตามที่ต้องการแล้ว ให้ทำการบันทึก (save) ไฟล์ข้อมูลนั้น ๆ โดยให้ใช้ชื่อไฟล์ตามเดิม

ข้อมูลที่จะต้องป้อนให้โปรแกรมเพื่อใช้ในการคำนวณ จะประกอบด้วยข้อมูลของอุปกรณ์และข้อมูลของระบบไฟฟ้า ดังนี้.-

1. ข้อมูลเครื่องกำเนิดไฟฟ้า ป้อนข้อมูลผ่านไฟล์ tge.m
2. ข้อมูลระบบเอกไซเตชั่น ป้อนข้อมูลผ่านไฟล์ texc.m

3. ข้อมูลโหลดโฟลว์ (load flow) ของระบบไฟฟ้า ป้อนข้อมูลผ่านไฟล์ tbu.m

4. ข้อมูลบัสแอ็ดมิทแตทซ์เมตริกซ์ของระบบไฟฟ้าป้อนข้อมูลผ่านไฟล์ tybus.m

5. ข้อมูลของอุปกรณ์สเตบิไลเซอร์ ป้อนข้อมูลผ่านไฟล์ tpss.m

6. ข้อมูลค่าเจาะจงของระบบไฟฟ้ากำลังที่ต้องการป้อนข้อมูลผ่านไฟล์ tpole.m

7. ข้อมูลค่าเริ่มต้นของตัวแปรของอุปกรณ์สเตบิไลเซอร์ที่จะใช้ในการคำนวณตามวิธีของ นิวตัน-ราล์ฟสัน ป้อนข้อมูลผ่านไฟล์ tint.m

8. ข้อมูลจำนวนเครื่องกำเนิดไฟฟ้าที่จะติดตั้งอุปกรณ์สเตบิไลเซอร์ป้อนข้อมูลผ่านตัวแปร m

ก.2 รูปแบบของข้อมูล

ข้อมูลตามข้อ 1 - 6 ใน ก.1 นั้น จะถูกป้อนลงในไฟล์ข้อมูลให้อยู่ในรูปของเมตริกซ์ โดยจะต้องมีเครื่องหมาย "[" และ "]" ในตำแหน่งเริ่มต้นและสุดท้ายของเมตริกซ์ตามตัวอย่าง ดังนี้.-

```
int     =    [ 1     50.0     0.2
               2     50.0     0.2
               3     50.0     0.2 ]
```

ก.2.1 รูปแบบข้อมูลเครื่องกำเนิดไฟฟ้า

ข้อมูลจะต้องเรียงกันทีละแถวตามลำดับของเครื่องกำเนิดไฟฟ้าจากเครื่องที่ 1 ไปจนถึงเครื่องสุดท้าย โดยในแต่ละคอลัมน์จะเป็นข้อมูลจำเพาะต่าง ๆ ของเครื่องกำเนิดไฟฟ้าแต่ละเครื่องที่จะต้องป้อนให้กับโปรแกรมโดยมีรายละเอียดของชนิดข้อมูลตาม ตารางที่ ก.1

คอลัมน์ที่	ข้อมูล	หน่วย
1	หมายเลขเครื่องกำเนิดไฟฟ้า	---
2	ชนิดของแบบจำลอง	---
3	X''_d	เปอร์ยูนิตของ rated MVA, kV
4	X'_d	เปอร์ยูนิตของ rated MVA, kV
5	X_d	เปอร์ยูนิตของ rated MVA, kV
6	X''_q	เปอร์ยูนิตของ rated MVA, kV
7	X'_q	เปอร์ยูนิตของ rated MVA, kV
8	X_q	เปอร์ยูนิตของ rated MVA, kV
9	R_A	เปอร์ยูนิตของ rated MVA, kV
10	X_A	เปอร์ยูนิตของ rated MVA, kV
11	T''_{d0}	วินาที
12	T'_{d0}	วินาที
13	T''_{q0}	วินาที
14	T'_{q0}	วินาที
15	H	วินาที
16	D	---
17	Rated MVA	MVA
18	MVA BASE ของระบบ	MVA
19	Rated kV	kV
20	kV BASE ของระบบ	kV

ตารางที่ ก.1 รูปแบบข้อมูลของเครื่องกำเนิดไฟฟ้า

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ก.2.2 รูปแบบข้อมูลระบบเอกไซเตชั่น

ข้อมูลจะต้องเรียงกันทีละแถวตามลำดับของระบบเอกไซเตชั่นที่ต่อเข้ากับเครื่องกำเนิดไฟฟ้า ตามที่ได้กำหนดหมายเลขเอาไว้แล้ว ตอนใส่ข้อมูลของเครื่องกำเนิดไฟฟ้า โดยในแต่ละคอลัมน์จะเป็นข้อมูลจำเพาะต่าง ๆ ของระบบเอกไซเตชั่นแต่ละชุด โดยมีรายละเอียดตาม ตารางที่ ก.2

คอลัมน์ที่	ข้อมูล	หน่วย
1	หมายเลขระบบเอกไซเตชั่น	---
2	ชนิดของแบบจำลอง	---
3	K_A	เปอร์ยูนิต
4	T_A	วินาที
5	K_E	เปอร์ยูนิต
6	T_E	วินาที
7	K_F	เปอร์ยูนิต
8	T_F	วินาที

ตารางที่ ก.2 รูปแบบข้อมูลของระบบเอกไซเตชั่น

ก.2.3 รูปแบบข้อมูลของโหลดโฟลว์

ข้อมูลจะต้องเรียงกันทีละแถวตามลำดับของเครื่องกำเนิดไฟฟ้า ตามที่กำหนดไว้ในตอนใส่ข้อมูลของเครื่องกำเนิดไฟฟ้า ในแต่ละคอลัมน์จะเป็นข้อมูลต่าง ๆ ตามรายละเอียดใน ตารางที่ ก.3

คอลัมน์ที่	ข้อมูล	หน่วย
1	ชนิดของบัส	---
2	กำลังไฟฟ้าจริงที่ผลิต	MW
3	กำลังไฟฟ้ารีแอ็คตีฟที่ผลิต	MVAR
4	แรงดันที่ขั้วของเครื่องกำเนิดไฟฟ้า	เปอร์ยูนิตของระบบ
5	เฟสของแรงดัน	เรเดียน

ตารางที่ ก.3 รูปแบบข้อมูลของโหลดโฟลว์

ก.2.4 <u>รูปแบบข้อมูลของบัสแอ็ดมิทแตนซ์เมตริกซ์ของระบบไฟฟ้า</u>

บัสแอ็ดมิทแตนซ์เมตริกซ์ (Y BUS) ของระบบไฟฟ้าจะต้องเรียงลำดับหมายเลขของบัส (bus) ให้เรียงตามลำดับเดียวกับหมายเลขของเครื่องกำเนิดไฟฟ้าที่ต่อยู่กับบัสนั้น ๆ ก่อน จากนั้น บัสที่ไม่มีเครื่องกำเนิดไฟฟ้าต่ออยู่ สามารถเรียงลำดับอย่างไรก็ได้ จากนั้น ให้สร้างบัสแอ็ดมิทแตนซ์เมตริกซ์ตามที่ได้กำหนดหมายเลขบัสเอาไว้แล้ว โดยค่าแอ็ดมิทแตนซ์ทั้งหมดจะต้องมีหน่วยเป็นเปอร์ยูนิตของระบบ

ก.2.5 <u>รูปแบบข้อมูลของอุปกรณ์สเตบิไลเซอร์</u>

ข้อมูลจะต้องเรียงกันทีละแถวตามลำดับของอุปกรณ์สเตบิไลเซอร์ที่ต่อเข้ากับเครื่องกำเนิดไฟฟ้า ตามที่ได้กำหนดหมายเลขเอาไว้แล้ว โดยรายละเอียดของข้อมูลในแต่ละคอลัมน์จะแสดงไว้ใน ตารางที่ ก.4

คอลัมน์ที่	ข้อมูล	หน่วย
1	หมายเลขอุปกรณ์สเตบิไลเซอร์	---
2	T_1	วินาที
3	T_2	วินาที
4	k	เปอร์ยูนิต
5	t	วินาที

ตารางที่ ก.4 รูปแบบข้อมูลของอุปกรณ์สเตบิไลเซอร์

ในการหาเมตริกซ์ A และ B ของระบบก่อนที่จะคำนวณหาค่าตัวแปรของอุปกรณ์สเตบิไลเซอร์จากค่าเจาะจงที่กำหนด ถ้าไม่ต้องการให้อุปกรณ์ สเตบิไลเซอร์เครื่องใดต่อเข้ากับเครื่องกำเนิดไฟฟ้า ให้ป้อนข้อมูลในคอลัมน์ที่ 4 เป็น 0 (k = 0)

ก.2.6 <u>รูปแบบข้อมูลของค่าเจาะจงของระบบไฟฟ้าที่ต้องการ</u>

ค่าเจาะจงของระบบไฟฟ้าที่ต้องการ ก็คือ ตำแหน่งของขั้ว (pole) ของสมการลักษณะของสเตทสเปซ การใส่ข้อมูลจะใส่ข้อมูลเรียงตามลำดับแถวเดียว ดังนั้น ถ้ามีอุปกรณ์สเตบิไลเซอร์จำนวน m ชุด ต่อเข้ากับเครื่องไฟฟ้า ค่าเจาะจงที่จะป้อนให้กับโปรแกรมจะต้องมีจำนวน 2m ค่า

ก.2.7 <u>รูปแบบข้อมูลของค่าเริ่มต้นของอุปกรณ์สเตบิไลเซอร์ที่จะใช้ในการคำนวณตามวิธีของ นิวตัน-ราล์ฟสัน</u>

ข้อมูลจะต้องเรียงทีละแถวตามลำดับของอุปกรณ์สเตบิไลเซอร์ โดยรายละเอียดของข้อมูลในแต่ละคอลัมน์จะแสดงไว้ใน ตารางที่ ก.5

คอลัมน์ที่	ข้อมูล	หน่วย
1	หมายเลขอุปกรณ์สเตบิไลเซอร์	---
2	k	เปอร์ยูนิต
3	t	วินาที

ตารางที่ ก.5 รูปแบบข้อมูลของค่าเริ่มต้นอุปกรณ์สเตบิไลเซอร์

ซึ่งค่า k และ t ที่ใส่ในไฟล์นี้ จะมีความหมายต่างกับที่ใส่ในไฟล์ tpss.m โดยค่าที่ใส่ในไฟล์ tpss.m จะเป็นค่า k และ t ของอุปกรณ์สเตบิไลเซอร์ที่ได้ติดตั้งกับเครื่องกำเนิดไฟฟ้าเรียบร้อยแล้ว แล้วจะนำข้อมูลใน tpss.m ไปคำนวณหาเมตริกซ์ **A** และ **B** ของสเตทสเปซ เพื่อนำมาหาค่าเจาะจงของระบบ เพื่อจะได้วิเคราะห์ได้ว่าระบบไฟฟ้ากำลังมีเสถียรภาพหรือไม่ ถ้าไม่มีเสถียรภาพก็จะทำการปรับค่า k, t ของอุปกรณ์สเตบิไลเซอร์ตัวเดิม และตัวที่ได้ติดตั้งเพิ่มเข้าไปในระบบ โดยค่า k และ t ในไฟล์ tint.m จะเป็นเพียงค่าเริ่มต้นในการทดลองแทนค่าตามวิธีนิวตัน-ราล์ฟสัน เท่านั้น

ก.2.8 <u>รูปแบบข้อมูลจำนวนเครื่องกำเนิดไฟฟ้าทั้งหมดที่จะติดตั้งอุปกรณ์ สเตบิไลเซอร์</u>

ภายหลังจากที่วิเคราะห์จากสเตทสเปซแล้วว่า ระบบไม่มีเสถียรภาพ ดังนั้น จะทำการติดตั้งอุปกรณ์สเตบิไลเซอร์ให้กับเครื่องกำเนิดไฟฟ้า ข้อมูลนี้จะเป็นตัวกำหนดว่าจะติดตั้งอุปกรณ์สเตบิไลเซอร์ให้กับเครื่องกำเนิดไฟฟ้าจำนวนกี่เครื่อง ซึ่งจำนวนที่จะติดตั้งจะต้องเป็นครึ่งหนึ่งของค่าเจาะจงของระบบไฟฟ้าที่ต้องการ การป้อนข้อมูลจะป้อนในโปรแกรมแม็ทแล็บได้เลย โดยใช้ตัวแปร m โดยจากจอภาพในแม็ทแล็บจะมีเครื่องหมาย prompt ให้พิมพ์ "m = x" แล้วกด Return (x = จำนวนอุปกรณ์สเตบิไลเซอร์ที่ต้องการ) การใส่ข้อมูลนั้น จะสามารถใส่ข้อมูลของอุปกรณ์สเตบิไลเซอร์ และค่าเจาะจงของระบบที่ต้องการให้ครบตามจำนวนเครื่องกำเนิดไฟฟ้าเลยก็ได้ โดยโปรแกรมจะตรวจสอบจากค่าตัวแปร m ที่ป้อนข้อมูลเข้าไป แล้วจะเลือกจำนวนอุปกรณ์สเตบิไลเซอร์และจำนวนค่าเจาะจงให้สอดคล้องกับค่า m เอง โดยเครื่องกำเนิดไฟฟ้าที่จะติดตั้งอุปกรณ์สเตบิไลเซอร์จะเรียงลำดับตามที่ได้ใส่ข้อมูลไว้ เช่น ถ้า m = 2 เครื่องกำเนิดไฟฟ้าหมายเลข 1 และ 2 จะติดตั้งอุปกรณ์สเตบิไลเซอร์ แต่ถ้า m = 3 เครื่องกำเนิดไฟฟ้า หมายเลข 1, 2 และ 3 จะติดตั้งอุปกรณ์สเตบิไลเซอร์

ก.3 การแก้ไขข้อมูล

ถ้ามีการใส่ข้อมูลผิดพลาด ในขณะที่ยังไม่ได้บันทึกไฟล์ข้อมูลก็สามารถแก้ไขได้ทันทีเลย โดยการใช้โปรแกรมโน๊ตแพด แต่ถ้ามีการบันทึกข้อมูลลงไปแล้ว จะแก้ไขข้อมูลได้โดยการเปิดไฟล์ข้อมูลนั้น ๆ มาทำการแก้ไข และบันทึกเข้าไปใหม่โดยโปรแกรมโน๊ตแพดเช่นกัน ส่วนข้อมูลจำนวนอุปกรณ์สเตบิไลเซอร์ที่จะต่อเข้ากับระบบสามารถแก้ไขผ่านหน้าจอภาพจากโปรแกรมแม็ทแล็บได้เลย โดยการให้ค่าตัวแปร m ใหม่ตามที่ต้องการ

ก.4 การเรียกใช้โปรแกรม

หลังจากที่ได้ใส่ข้อมูลต่าง ๆ ลงไปในไฟล์ข้อมูลและทำการบันทึกเรียบร้อยแล้ว จากเครื่องหมาย prompt ของโปรแกรมแม็ทแล็บให้ทำตามขั้นตอน ดังนี้.-

1. พิมพ์ "tinput" แล้วกด Return ข้อมูลต่าง ๆ ที่ป้อนเข้าไปจะถูกนำไปสร้างเป็นเมตริกซ์ในโปรแกรมแม็ทแล็บ
2. พิมพ์ "aabb" แล้วกด Return โปรแกรมจะคำนวณหาสเตทสเปซของระบบไฟฟ้ากำลัง
3. พิมพ์ "eig (A)" แล้วกด Return โปรแกรมจะคำนวณค่าเจาะจงของระบบออกมา ซึ่งจะสามารถวิเคราะห์เสถียรภาพได้
4. พิมพ์ "m = x" (x = จำนวนอุปกรณ์สเตบิไลเซอร์ที่จะต่อกับเครื่องกำเนิดไฟฟ้า แล้วกด Return
5. พิมพ์ "tfind" แล้วกด Return จอภาพจะแสดงผลออกมาดังนี้.-

   "y =" (ค่าความแตกต่างของค่าที่คำนวณได้จากการแทนค่า k และ t กับค่าที่เป็นจริง)

   "kt =" (ค่า k และ t ล่าสุดที่คำนวณได้)

   "Press any key to continue or Press" Ctrl + C" to cancell."

ซึ่งเป็นการให้ผู้ใช้ได้ตรวจสอบก่อนว่าค่า k และ t เริ่มต้นที่กำหนดไว้นั้น จะให้ค่าความแตกต่างจากค่าเป็นจริงมากน้อยแค่ไหน ถ้า y ต่ำจะทำให้ใช้เวลาในการคำนวณน้อยลง และถ้าค่า y เริ่มต้นมากเกินไป (มากกว่า 10,000) ก็จะทำให้โปรแกรมใช้เวลาในการคำนวณมาก ดังนั้น ในขั้นตอนนี้ถ้าพิจารณาแล้วเห็นว่าค่า y มีค่ามากไป ก็ให้กด "Ctrl + C" เพื่อออกจากขั้นตอนนี้แล้วไปแก้ไขค่า k และ t ที่เป็นค่าเริ่มต้นใหม่ในไฟล์ tint.m แล้วบันทึกไฟล์ จากนั้น

ให้ทำตามขั้นตอนที่ 1 ใหม่อีกครั้ง แล้วทำตามขั้นตอนที่ 5 ได้เลย โดยไม่ต้องทำตามขั้นตอนที่ 2, 3 และ 4 และถ้าค่า y ไม่มากเกินไปแล้ว ก็ให้กดปุ่มใด ๆ ก็ได้เพื่อให้โปรแกรมทำการคำนวณหาค่า k และ t ที่จะทำให้ระบบมีค่าเจาะจงตามที่ต้องการ

6. พิมพ์ "taw" แล้วกด Return โปรแกรมจะคำนวณหาสเตทสเปซใหม่ตามจำนวนของอุปกรณ์สเตบิไลเซอร์ที่ต่อให้กับระบบ โดยมีค่า k และ t ตามที่คำนวณได้จากขั้นตอนที่ 5

7. พิมพ์ "eig (Aw)" โปรแกรมจะหาค่าเจาะจงของระบบตามสเตทสเปซที่คำนวณได้ขั้นตอนที่ 6 ซึ่งจะสามารถวิเคราะห์เสถียรภาพของระบบหลังจากที่ต่ออุปกรณ์สเตบิไลเซอร์คำนวณค่า k และ t ที่เหมาะสมแล้ว

8. ถ้าต้องการเปลี่ยนจำนวนอุปกรณ์สเตบิไลเซอร์ที่จะต่อเข้ากับระบบ (ค่าตัวแปร m) จะทำได้โดยการพิมพ์ "m = x" (x = จำนวนอุปกรณ์สเตบิไลเซอร์ที่จะต่อเข้ากับระบบ) จากนั้น ให้ทำตามขั้นตอนที่ 5, 6 และ 7 ได้เลย

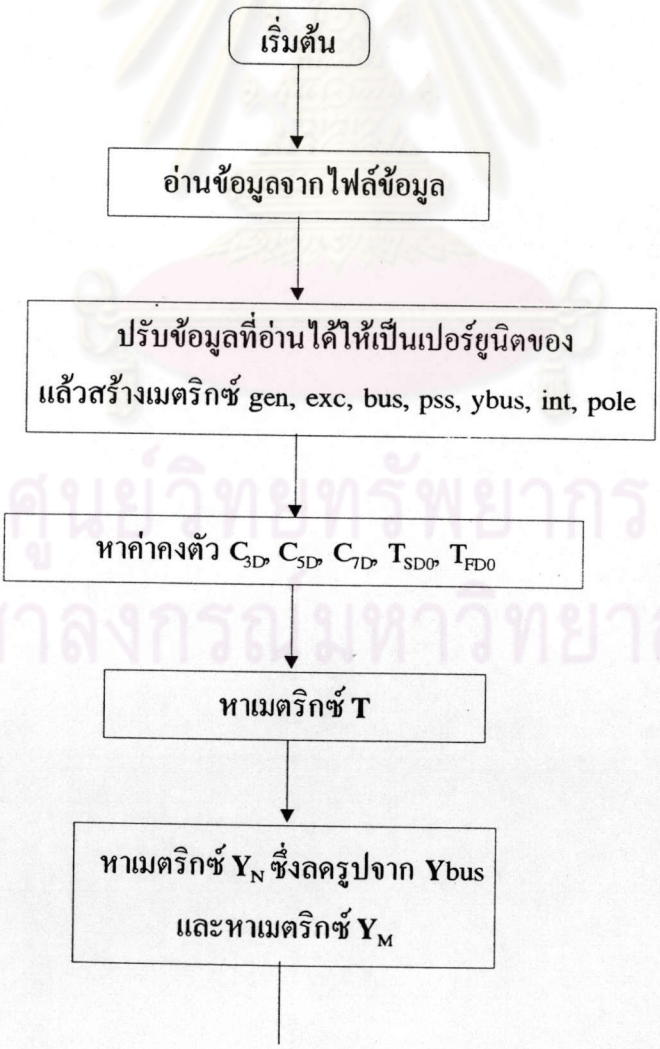
ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก ข.

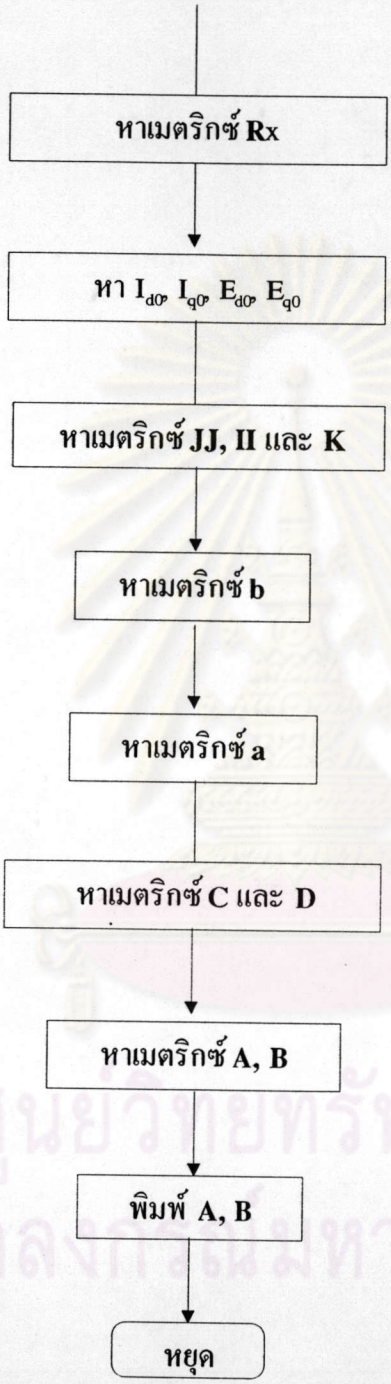
รายละเอียดโปรแกรมการคำนวณ

ข.1 โฟลว์ชาร์ทแสดงขั้นตอนการหาสเตทสเปซ

ในหัวข้อนี้จะแสดงโฟลว์ชาร์ทแสดงขั้นตอนตั้งแต่การเริ่มอ่านข้อมูล และนำข้อมูลที่อ่านได้ไปคำนวณหาสเตทสเปซ

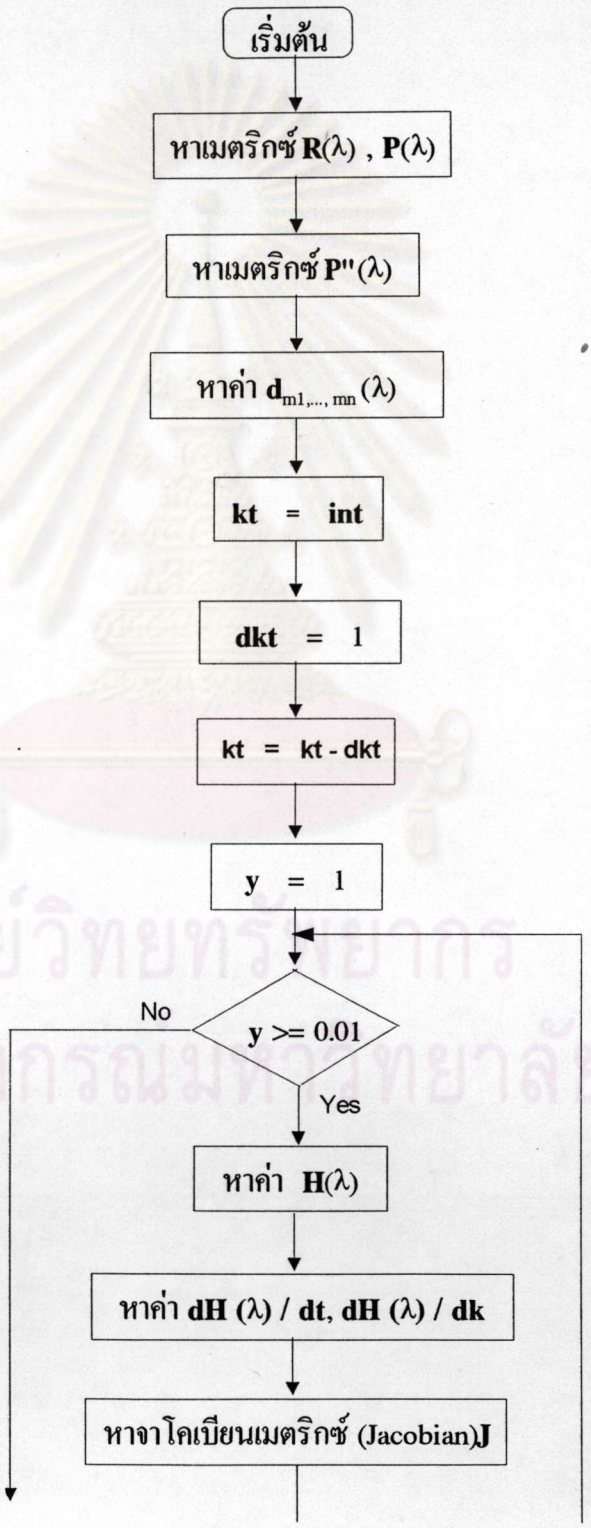


ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

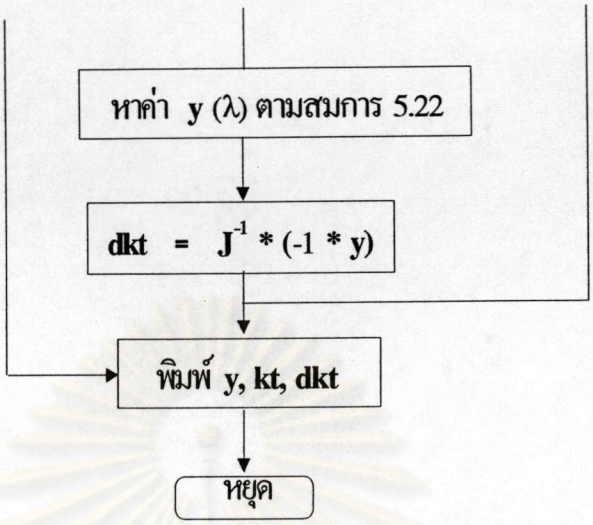


ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ข.2 โฟลว์ชาร์ทแสดงขั้นตอนการหาตัวแปรของอุปกรณ์สเตบิไลเซอร์



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก ค.

รายละเอียดของโปรแกรม

<u>รายละเอียดโปรแกรมการคำนวณ</u>

```
%tge.m is the data file which contain data of generators in system.
%Data are sorted in the column form [BUS , MODEL ,x"d , x'd  , xd , x"q ,
x'q, , xq , RA  , xA
%T"do , T'do , T"qo , T'qo  , H , D , MVAmach ,MVAsys , kVmach , kVsys].
%Impedance and 'H' are given in per unit of machine base.
%[BUS , MODEL ,x"d , x'd  , xd , x"q , x'q, , xq , RA  , xA  ,T"do ,T'do ,T"qo
, T'qo,  H  ,  D , MVAmach MVAsys , kVmach , kVsys]
ge= [
%texc.m. is the data file which contain data of exciters in system.
%Data are soted in the column form [BUS , MODEL , KA , TA , KE , TE ,
KF , TF]
%[BUS , MODEL ,  KA  , TA  ,  KE , TE  ,  KF , TF  ]
exc=[
%tbu.m is the data file which contain data of each bus in system.
%Data are sorted in the column form [TYPE , Pin ,Qin ,Pout ,Qout , V , b]
%[BUS , Pout  , Qout , V  , b   ]
bu =[
%tybus.m is the data file which contain bus addmitance matrix of the system.
%The upper rows are the node which connect to generator.
%Data must given in system base.
ybus=[
%tpss.m is the data file which contain data of power system stabilizers(PSS) in
system.
%Data are sorted in the column form [BUS , T1 , T2 , k , t ]
%[BUS , T1   , T2 , k , t ]
pss=[
%tint.m is a file which contain initial valve of k and t of PSS.
%Data in [bus k t]
%  [bus   k     t   ]
int=[
%tpole.m is the data file which contain required poles of system.
pole=[
%tgen.m is the file which calculate all impedance and 'H' in per unit of system
base.
gen = ge ;
for i = 1:n ;
```

```
        for j = 3:10 ;
                gen(i,j) = ge(i,j)*(ge(i,19)/ge(i,20))^2*ge(i,18)/ge(i,17) ;
        end ;
        gen(i,15) = ge(i,15)*ge(i,17)/ge(i,18) ;
end
gen
%tbus.m is the file which calculate all data in system base.
bus = bu ;
for i = 1:n ;
        for j = 2:3 ;
                bus(i,j) = bu(i,j)/gen(i,18) ;
        end
end
bus
%tcd.m is a file to determine direct axis parameters (CD) of generators.
%C3D,C5D,C7D, TSDO,TFDO are determined.
for i = 1:n ;
        if any(gen(i,2) == [2 4]) ;
                        cd(i,1) = gen(i,5)-gen(i,4) ;
                        cd(i,2) = 0 ;
                        cd(i,3) = 1 ;
                        cd (i,4) = 0 ;
                        cd(i,5) = 0 ;
                        cd(i,6) = gen(i,12) ;
        else ;
                        cd(i,1) = (gen(i,5)-gen(i,4))*(gen(i,3)-gen(i,10))/(gen
(i,4)-gen(i,10)) ;
                        cd(i,2) = (gen(i,5)-gen(i,10))*(gen(i,4)-gen(i,3))/(gen
(i,4)-gen(i,10)) ;
                        cd(i,3) = (gen(i,3)-gen(i,10))/(gen(i,4)-gen(i,10)) ;
                        cd (i,4) = (gen(i,4)-gen(i,3))/(gen(i,4)-gen(i,10)) + (gen
(i,3)-gen(i,10))/(gen(i,5)-gen(i,10)) ;
                        cd (i,5) = 0.5*(gen(i,12)+gen(i,11)-sqrt((gen(i,12)+gen
(i,11))^2-4*cd(i,1)*gen(i,11)*gen(i,12))) ;
                        cd (i,6) = 0.5*(gen(i,12)+gen(i,11)+sqrt((gen(i,12)+gen
(i,11))^2-4*cd(i,1)*gen(i,11)*gen(i,12))) ;
        end ;
end ;
cd
%tt.m is a file to detemine matrix T which used to convert generator referrence
to
%system referrence frame.
delta = zeros(n,n) ;
for i = 1:n ;
        delta(i,i)=atan((bus(i,2)*gen(i,8)-bus(i,3)*gen(i,9))/(bus(i,4)^2+bus
(i,3)*gen(i,8)+bus(i,2)*gen(i,9)))+bus(i,5) ;
        et(i,i) = exp(1j*delta(i,i)) ;
end ;
delta
et
```

```
for i = 1:n ;
        T(2*i-1,2*i-1) =  real(et(i,i)) ;
        T(2*i-1,2*i)   = -imag(et(i,i)) ;
        T(2*i,2*i-1)   =  imag(et(i,i)) ;
        T(2*i,2*i)     =  real(et(i,i)) ;
end ;
T
%tyrnm.m is a file to determine bus addmitance matrix
%which reduced the number of rows to equal quantity of genereators.
if size(ybus,1) == n ;
        yred = ybus
else
        y11 = ybus(1:n,1:n) ;
        y12 = ybus(1:n,(n+1):size(ybus,1)) ;
        y21 = ybus((n+1):size(ybus,2),1:n) ;
        y22 = ybus((n+1):size(ybus,1),(n+1):size(ybus,2)) ;
        yred = y11-y12*inv(y22)*y21
end

Yn=zeros(2*n,2*n) ;
for i = 1:n ;
        for j = 1:n ;
        Yn(2*i-1,2*j-1) =  real(yred(i,j)) ;
        Yn(2*i-1,2*j)   = -imag(yred(i,j)) ;
        Yn(2*i,2*j-1)   =  imag(yred(i,j)) ;
        Yn(2*i,2*j)     =  real(yred(i,j)) ;
        end ;
end;
Yn
Ym = T'*Yn*T
%trx.m is a file to calculate matrix Rx which used to determine Vdq.
Rx = zeros(2*n,2*n) ;
for i = 1:n ;
        if gen(i,2) == 2 ;
                Rx(2*i-1,2*i-1) = -gen(i,9) ;
                Rx(2*i-1,2*i)   =  gen(i,8) ;
                Rx(2*i,2*i-1)   = -gen(i,4) ;
                Rx(2*i,2*i)     = -gen(i,9) ;
        else
                if gen(i,2) == 4 ;
                  Rx(2*i-1,2*i-1) = -gen(i,9) ;
                  Rx(2*i-1,2*i)   =  gen(i,7) ;
                  Rx(2*i,2*i-1)   = -gen(i,4) ;
                  Rx(2*i,2*i)     = -gen(i,9) ;
                else
                  Rx(2*i-1,2*i-1) = -gen(i,9) ;
                  Rx(2*i-1,2*i)   =  gen(i,6) ;
                  Rx(2*i,2*i-1)   = -gen(i,3) ;
                  Rx(2*i,2*i)     = -gen(i,9) ;
                end
```

```
        end
end
Rx
%tiv0.m is a file to calculate initial I , E' and E" of generators.
for i = 1:n ;
        ang(i) = -1*delta(i,i)-atan(bus(i,3)/bus(i,2)) + bus(i,5) ;
        Ia(i) = bus(i,2)/bus(i,4)*cos(atan(bus(i,3)/bus(i,2))) ;
        I0(i) = Ia(i)*exp(1j*ang(i)) ;
end ;
Ia
I0
for i = 1:n ;
        V0(i) = bus(i,4)*exp(1j*(-delta(i,i)+bus(i,5))) ;
end ;
V0
Vdq = zeros(2*n,1) ;
Idq = zeros(2*n,1) ;
for i = 1:n ;
        Idq(2*i-1,1) = imag(I0(i)) ;
        Idq(2*i,1)   = real(I0(i)) ;
        Vdq(2*i-1,1) = imag(V0(i)) ;
        Vdq(2*i,1)   = real(V0(i)) ;
end
Edq = zeros(2*n,1) ;
Edq = Vdq-Rx*Idq ;
for i = 1:n ;
        if gen(i,2) == 2 ;
         Edq(2*i-1,1) = 0 ;
        else
        end
end

%tjki.m is a file to calculate matrix J K I of section 4.7.
JJ = zeros(2*n,n) ;
for i = 1:n ;
        JJ(2*i-1,i) =  Idq(2*i,1) ;
        JJ(2*i,i)   =  Idq(2*i-1,1) ;
end ;
JJ
II = zeros(2*n,n) ;
for i = 1:n ;
        II(2*i-1,i) = -Vdq(2*i-1,1)*sin(delta(i,i)) - Vdq(2*i,1)*cos(delta(i,i)) ;
        II(2*i,i)   =  Vdq(2*i-1,1)*cos(delta(i,i)) - Vdq(2*i,1)*sin(delta(i,i)) ;
end ;
II
K = T'*Yn*II+JJ
%tbi.m is a file to calculate submatrix of b matrix.
b1 = zeros(n,2*n) ;
b2 = zeros(n,2*n) ;
b3 = zeros(n,2*n) ;
```

```
b4 = zeros(n,2*n) ;
for i = 1:n ;
        b1(i,2*i-1) = -Edq(2*i-1,1)/(2*gen(i,15)) ;
        b1(i,2*i)   = -Edq(2*i,1)/(2*gen(i,15)) ;
end
b1
for i = 1:n ;
        if any(gen(i,2) == [2 4]) ;
           b3(i,2*i-1) = -cd(i,1)/gen(i,12) ;
        else
           b3(i,2*i-1) = (-cd(i,2)*cd(i,4)/cd(i,5))-(cd(i,2)*cd(i,3)/cd(i,6)) ;
        end
end
b3
KE = zeros(n,2*n) ;
for i= 1:n ;
        KE(i,2*i-1) = -exc(i,3)*Vdq(2*i-1,1)/(exc(i,4)*bus(i,4)) ;
        KE(i,2*i)   = -exc(i,3)*Vdq(2*i,1)/(exc(i,4)*bus(i,4)) ;
end
b5 = KE*Rx
%tai.m is a file to calculate submatrix of a matrix.
a11 = zeros(n,n) ;
a12 = zeros(n,n) ;
a13 = zeros(n,n) ;
a14 = zeros(n,n) ;
a15 = zeros(n,n) ;
a16 = zeros(n,n) ;
a17 = zeros(n,n) ;
a18 = zeros(n,n) ;
a21 = zeros(n,n) ;
a22 = zeros(n,n) ;
a23 = zeros(n,n) ;
a24 = zeros(n,n) ;
a25 = zeros(n,n) ;
a26 = zeros(n,n) ;
a27 = zeros(n,n) ;
a28 = zeros(n,n) ;
a31 = zeros(n,n) ;
a32 = zeros(n,n) ;
a33 = zeros(n,n) ;
a34 = zeros(n,n) ;
a35 = zeros(n,n) ;
a36 = zeros(n,n) ;
a37 = zeros(n,n) ;
a38 = zeros(n,n) ;
a41 = zeros(n,n) ;
a42 = zeros(n,n) ;
a43 = zeros(n,n) ;
a44 = zeros(n,n) ;
a45 = zeros(n,n) ;
```

```
a46 = zeros(n,n) ;
a47 = zeros(n,n) ;
a48 = zeros(n,n) ;
a51 = zeros(n,n) ;
a52 = zeros(n,n) ;
a53 = zeros(n,n) ;
a54 = zeros(n,n) ;
a55 = zeros(n,n) ;
a56 = zeros(n,n) ;
a57 = zeros(n,n) ;
a58 = zeros(n,n) ;
for i = 1:n ;
        a11(i,i) = -gen(i,16)/(2*gen(i,15)) ;
        a21(i,i) = 377 ;
        a13(i,i) = -Idq(2*i,1)/(2*gen(i,15)) ;
        a53(i,i) = -exc(i,3)*Vdq(2*i,1)/(exc(i,4)*bus(i,4)) ;
        a44(i,i) = -exc(i,5)/exc(i,6) ;
        a45(i,i) =  1/exc(i,6) ;
        a55(i,i) =  -1/exc(i,4) ;
        if pss(i,4) ~= 0 ;
        a58(i,i) =  exc(i,3)/exc(i,4) ;
        else
        end
        if any(gen(i,2) == [2 4]) ;
                a33(i,i) = -1/gen(i,12) ;
                a34(i,i) =  1/gen(i,12) ;
        else
                a34(i,i) = cd(i,3)/cd(i,6) ;
        end
end
%taa.m is a file to calculate aa matrix.
C  = inv(eye(2*n)-Ym*Rx)*K ;
D  = inv(eye(2*n)-Ym*Rx)*Ym ;
b  = [b1;b2;b3;b4;b5] ;
a1 = [a11;a21;a31;a41;a51] ;
a2 = [a12;a22;a32;a42;a52] ;
a3 = [a13;a23;a33;a43;a53] ;
a4 = [a14;a24;a34;a44;a54] ;
a5 = [a15;a25;a35;a45;a55] ;
a6 = [a16;a26;a36;a46;a56] ;
a7 = [a17;a27;a37;a47;a57] ;
a8 = [a18;a28;a38;a48;a58] ;
aa = [a1,a2+b*C,a3+b*D(:,2:2:2*n),a4,a5,a6,a7,a8]
%tab.m is a file to calculate A and B matrix.
for i = 1:n ;
        if exc(i,2) == 6 ;
                aa(3*n+i,:)     = aa(4*n+i,:) ;
                aa(3*n+i,4*n+i) = 0 ;
                aa(3*n+i,3*n+i) = exc(i,3)/exc(i,4) ;
        else
```

```
        end
end
A = zeros(8*n,8*n) ;
for i = 1:5*n ;
        for j = 1:5*n ;
                A(i,j) = aa(i,j) ;
        end
end
for i = 1:n
        if pss(i,4) ~= 0 ;
          for j = 1:8*n ;
                A(5*n+i,j) = A(i,j) ;
          end
          A(5*n+i,5*n+i) = A(5*n+i,5*n+i)-1/pss(i,2) ;
          for j = 1:8*n ;
                A(6*n+i,j) = pss(i,5)*A(5*n+i,j)/pss(i,3) ;
          end
          A(6*n+i,5*n+i) = A(6*n+i,5*n+i)+1/pss(i,3) ;
          A(6*n+i,6*n+i) = A(6*n+i,6*n+i)-1/pss(i,3) ;
          for j = 1:8*n ;
                A(7*n+i,j) = pss(i,4)*pss(i,5)*A(6*n+i,j)/pss(i,3) ;
          end
          A(7*n+i,6*n+i) = A(7*n+i,6*n+i)+pss(i,4)/pss(i,3) ;
          A(7*n+i,7*n+i) = A(7*n+i,7*n+i)-1/pss(i,3) ;
        else
        end
end
A
B = zeros(8*n,n) ;
for i = 1:n ;
        B(4*n+i,i) = exc(i,3)/exc(i,4) ;
        if exc(i,2) == 6  ;
          B(3*n+i,i) = exc(i,3)/exc(i,4) ;
        else
        end
end
B
%tsepa.m is a file to seperate A matrix to be 4 parts.
An  = A(1:5*n,1:5*n) ;
A11 = An(1:n,1:n) ;
A12 = An(1:n,(n+1):(5*n)) ;
A21 = An((n+1):(5*n),1:n) ;
A22 = An((n+1):(5*n),(n+1):(5*n)) ;
Bn  = B(1:5*n,n)
%tind.m is a file to determine permutation matrix.
bin=0;
for i =1:2^m-1 ;
j = i ;
x2 = i ;
x1 = 0 ;
```

```
z = 0 ;
        while all(x2 ~= [1 0]) | all(x1 ~= [1 0]) ;
                x1 = fix(j/2^z) ;
                x2 = rem(j,2^z) ;
                z = z+1 ;
                if x1 == 1 ;
                        bin(i,z) = 1 ;
                        z = 0 ;
                        j = x2 ;
                else ;
                end ;
        end ;
if x2 == 1 ;
        bin(i,1) = 1 ;
else ;
        bin(i,1) = 0 ;
end ;
end;
ind = 0 ;
for i=1:2^m-1 ;
        for j = 1:m ;
                if bin(i,j) == 1;
                  ind(i,j) = j;
                else
                  ind(i,j) = 0 ;
                end ;
        end ;
end ;
bin
ind
%td.m is a file to calculate determinant.
bbb = ones(n,n) ;
bb  = bbb ;
for k = 1:n  ;
        v = diag(B(4*n+1:5*n,:))' ;
        bbb(k,:) = v ;
        v = diag(B(3*n+1:4*n,:))' ;
        bb(k,:) = v ;
end;
for i=1:2*m  ;
        p = A12*inv(pole(i)*eye(4*n)-A22) ;
        ppp = (-1*bb).*p(:,2*n+1:3*n) + (-1*bbb).*p(:,3*n+1:4*n);
        R = pole(i)*eye(n)-A11-A12*inv(pole(i)*eye(4*n)-A22)*A21 ;
        dr(i)=det(R) ;
        for j = 1:2^m-1 ;
                Rr = R ;
                for k = 1:m ;
                        if ind(j,k) ~= 0 ;
                                Rr(:,ind(j,k)) = ppp(:,ind(j,k)) ;
                        else ;
```

```
                    end ;
                end ;
                d(i,j) = det(Rr) ;

        end;

end ;
d
dr
%th.m is a file to calculate H matrix.
H = zeros(2*n,n) ;
for i=1:(2*m) ;
        for j = 1:m ;
          H(i,j) = ((1+pole(i)*kt(j+m,1))/(1+pole(i)*pss(j,3)))^2*kt(j,1)*pole
(i)*pss(j,2)/(1+pole(i)*pss(j,2)) ;
        end ;
end ;
H;
%tdhp.m is a file to calculate derevative(dh/dk,dh/dt) matrix.
dhp = zeros(2*m,m) ;
for i=1:(2*m) ;
        for j = 1:m ;
                dhp(i,j) = H(i,j)/kt(j,1) ;
dhp(i,j+m) = kt(j,1)*pole(i)*pss(j,2)*2*pole(i)*(1+pole(i)*kt(j+m,1))/((1+pole
(i)*pss(j,2))*(1+pole(i)*pss(j,3))^2) ;
        end ;
end ;
dhp;
%tji.m is a file to calculate each row of jacobian matrix.
coe=zeros(2*m,m) ;
for i=1: 2*m ;
for j=1 : m ;
        mm = any(j*ones(m,2^m-1)==ind') ;
        for k = 1:2^m-1 ;
                h = 1 ;
          if mm(k) ~= 0 ;
                for q =1:m ;
                        if ind(k,q) ~= 0 ;
                            h = h*H(i,ind(k,q)) ;
                        else ;
                        end ;
                end ;
          else ;
              h = 0 ;
          end ;
        h = h/H(i,j) ;
        coe(i,j) = coe(i,j) +d(i,k)*h ;
        end ;
end ;
end ;
```

```
coe ;
%ty.m is a file to calculate solution of main equation for use in Newton
Raphson method.
for i=1 : 2*m ;
        R = pole(i)*eye(n)-A11-A12*inv(pole(i)*eye(4*n)-A22)*A21 ;
        p = A12*inv(pole(i)*eye(4*n)-A22) ;
        Bb = B(n+1:5*n,:)*diag(H(i,:)) ;
        y(i,1) = det(R-p*Bb) ;
end ;
y
%tkt.m is a file to calculate suitable value of k and t of pss.
dkt = ones(2*m,1) ;
y = ones(2*m,1) ;
kt=[int(1:m,2);int(1:m,3)] ;
kt = kt-dkt ;
count = 0 ;
while any(abs(-1*y) >= 0.1) ;
        count = count+1 ;
        kt = kt+dkt;
        th ;
        tdhp;
        tji ;
        coej = [coe,coe] ;
        J = coej.*dhp ;
        invJ=inv(J) ;
        ty
        dkt = inv(J)*(-1*y);
        kt
        if count == 1 ;
          disp('PRESS "ENTER" TO CONTINUE OR PRESS "Ctrl+C"
TO TERMINATE')
          pause
        else
        end
end ;
y
kt
dkt
%tinput.m is a file to take data from data file to be in matrixs.
tge
tgen
tbu
tbus
texc
tybus
tpss
tint
tpole
%aabb.m is a file to determine A matrix of power system before installed pss.
tcd
```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```
tt
tyrnm
trx
tiv0
tjki
tbi
tai
taa
tab
%tfind.m is a file to find the parameter of pss.
tsepa
tind
td
tkt
%taw.m is a file to calculate A matrix after determine new (k,t).
Aw = A ;
for i = 1:m
        Aw(4*n+i,7*n+i) = exc(i,3)/exc(i,4) ;
        if exc(i,2) == 6 ;
           Aw(3*n+i,7*n+i) = exc(i,3)/exc(i,4) ;
        else
        end
        for j = 1:8*n ;
                Aw(5*n+i,j) = Aw(i,j) ;
        end
        Aw(5*n+i,5*n+i) = Aw(5*n+i,5*n+i)-1/pss(i,2) ;
        for j = 1:8*n ;
                Aw(6*n+i,j) = kt(m+i,1)*Aw(5*n+i,j)/pss(i,3) ;
        end
        Aw(6*n+i,5*n+i) = Aw(6*n+i,5*n+i)+1/pss(i,3) ;
        Aw(6*n+i,6*n+i) = Aw(6*n+i,6*n+i)-1/pss(i,3) ;
        for j = 1:8*n ;
                Aw(7*n+i,j) = kt(i,1)*kt(m+i,1)*Aw(6*n+i,j)/pss(i,3) ;
        end
        Aw(7*n+i,6*n+i) = Aw(7*n+i,6*n+i)+kt(i,1)/pss(i,3) ;
        Aw(7*n+i,7*n+i) = Aw(7*n+i,7*n+i)-1/pss(i,3) ;
end
Aw
```

ประวัติผู้เขียน

นายอรุณ หัตถะรัชต์ เป็นบุตรของ นายเกียเข่ง แซ่ห่าน และ นางกิมลี้ แซ่โง้ว เกิดวันที่ 20 พฤษภาคม พ.ศ. 2513 ที่ แขวงป้อมปราบ เขตป้อมปราบ จังหวัด กรุงเทพมหานคร จบการศึกษาระดับอุดมศึกษา จาก สาขาวิศวกรรมไฟฟ้ากำลัง ภาควิชาไฟฟ้า คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย เมื่อปีการศึกษา 2533

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย