

# สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

โปรแกรม คอมพิวเตอร์เพื่อศึกษาผลของการเชื่อมโยงด้วย  
ระบบไฟฟ้ากระแสตรงแรงดันสูงในระบบไฟฟ้ากำลัง

Computer Program for Study of HVDC Link

Effected in Electrical Power System



โดย

นายพฤทธิ ประเสริฐธรรม

นายมนัสวี ตันเจริญ

นายยุทธนา สังข์เฉย

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมไฟฟ้า

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2542

เลขหม.....  
เลขทะเบียน..... 36969  
วัน, เดือน, ปี..... 29 ส.ค. 2548

ปีการศึกษา 2542

โปรแกรมคอมพิวเตอร์เพื่อศึกษาผลการเชื่อมโยงด้วยระบบ  
ไฟฟ้ากระแสตรงแรงดันสูงในระบบไฟฟ้ากำลัง  
Computer Program for Study of HVDC Link  
Effected in Electrical Power System

โดย

|           |              |
|-----------|--------------|
| นายพฤทธิ  | ประเสริฐธรรม |
| นายมนัสวี | ตันเจริญ     |
| นายยุทธนา | สังข์เฉย     |

อาจารย์ที่ปรึกษา

รศ.ศิริวัฒน์ โปธิเวชกุล

ปริญญาานิพนธ์ปีการศึกษา 2542

ภาควิชาวิศวกรรมไฟฟ้า

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง โปรแกรมคอมพิวเตอร์เพื่อศึกษาผลของการเชื่อมโยงด้วยระบบไฟฟ้ากระแสตรงแรงดันสูง  
ในระบบไฟฟ้ากำลัง

ผู้จัดทำ

- |              |              |
|--------------|--------------|
| 1. นายพฤทธิ  | ประเสริฐธรรม |
| 2. นายมนัสวี | ตันเจริญ     |
| 3. นายยุทธนา | สังข์เฉย     |



อาจารย์ที่ปรึกษา

(รศ.ศิริวัฒน์ โปธิเวชกุล)

|

โปรแกรมคอมพิวเตอร์เพื่อศึกษาผลของการเชื่อมโยงด้วย  
ระบบไฟฟ้ากระแสตรงแรงดันสูงในระบบไฟฟ้ากำลัง

นายพฤษดิ์ ประเสริฐธรรม  
นายมนัสวี ตันเจริญ  
นายยุทธนา สังข์เฉย  
รศ.ศิริวัฒน์ โปธิเวชกุล อาจารย์ที่ปรึกษา  
ปีการศึกษา 2542

บทคัดย่อ

โครงการวิจัยนี้ เป็นการศึกษาการไหลของกำลังงานในระบบไฟฟ้ากำลังที่มีการเชื่อมโยงด้วยระบบไฟฟ้ากระแสตรงแรงดันสูง โดยศึกษาและวิเคราะห์เอชไอแอลโฟล (AC Load Flow) ด้วยวิธีนิวตัน-ราฟสัน (Newton-Raphson) , ฟาสต์ดีคัปเปิล (Fast-decouple) และวิเคราะห์เอชไอ-ดีซี โหลดโฟล (AC-DC Load Flow) พร้อมทั้งเปรียบเทียบขนาดและมุมของแรงดัน การไหลของกำลังไฟฟ้าจริงและกำลังไฟฟ้าปรากฏ ความสูญเสีย(loss)ที่เกิดขึ้นในระบบ เมื่อมีและไม่มี การเชื่อมต่อด้วยระบบไฟฟ้ากระแสตรงแรงดันสูง(HVDC Link) ในระบบไฟฟ้ากำลัง

ในโครงการนี้ได้จัดสร้างโปรแกรมคอมพิวเตอร์ขึ้น 2 โปรแกรม โดยโปรแกรมแรกคือ “เอชไอซีพาวเวอร์โพลเวอร์ชัน 1.0” สามารถคำนวณระบบไฟฟ้าในขอบเขตของ 118 บัส , 176 สาย และสายส่งไฟฟ้ากระแสตรงแรงดันสูง 1 สาย รวมทั้งมีโหมดการทำงานที่สามารถแสดงการคำนวณพาวเวอร์โพล ทีละขั้นได้ โปรแกรมที่สองคือ “โปรแกรมส่วนการจำลองระบบไฟฟ้า” สามารถแสดงระบบไฟฟ้าที่ทำการวิเคราะห์ในรูปแบบกราฟฟิก และแสดงผลของพาวเวอร์โพล เป็นกราฟฟิกด้วย

**Computer Program for Study of HVDC Link**  
**Effected in Electrical Power System**

|                    |                    |
|--------------------|--------------------|
| Prut               | Prasrettam         |
| Manussavee         | Tuncharoen         |
| Yuttana            | Sangchoei          |
| Assoc.Prof.Siriwat | Potivejkul Advisor |
| 1999               |                    |

**ABSTRACT**

This project studies about power flow in the conventional AC power which is added High Voltage Direct Current transmission line (HVDC link), AC-DC system. This project is composed of AC load flow analysis using Newton-Raphson and fast decoupled method and AC-DC load flow analysis as well as compare AC system to AC-DC system about voltage magnitude, voltage angle at fundamental at each buses , MW flow , MVAR flow from bus to bus and loss occurred in system.

This project consists of 2 Computer programs. First is "ACDC Power Flow v1.0".It can calculate the system in limit of 118 buses , 176 branches and 1 High Voltage Direct Current transmission line. It also contains debug mode for running power flow step by step. The second is "System Builder Program". It can show the power system and power flow result in graphic mode.

## สารบัญ

|   | หน้า |
|---|------|
| บทคัดย่อ.....   | I    |
| ABSTRACT.....   | II   |
| สารบัญภาพ.....  | III  |
| สารบัญตาราง.....  | V    |
| บทที่ 1 บทนำ.....   | 1    |
| 1.1 วัตถุประสงค์.....                                       | 1    |
| 1.2 ขอบเขตของโครงการ.....                                   | 1    |
| 1.3 ขั้นตอนดำเนินงาน.....                                   | 1    |
| บทที่ 2 ระบบไฟฟ้ากระแสตรงแรงดันสูง.....                     | 2    |
| 2.1 บทนำ.....   | 2    |
| 2.2 ข้อดีของระบบไฟฟ้ากระแสตรงแรงดันสูง.....                 | 2    |
| 2.3 ข้อเสียของระบบไฟฟ้ากระแสตรงแรงดันสูง.....               | 4    |
| 2.4 ส่วนประกอบของระบบไฟฟ้ากระแสตรงแรงดันสูง.....            | 4    |
| 2.5 ทฤษฎีระบบไฟฟ้ากระแสตรงแรงดันสูง.....                    | 5    |
| 2.6 แบบจำลองและวงจรเสมือนของระบบไฟฟ้ากระแสตรงแรงดันสูง..... | 9    |
| บทที่ 3 โหลดโพล.....  | 10   |
| 3.1. โหลดโพล.....   | 10   |
| 3.2 ชนิดของบัส.....   | 10   |
| 3.3. ข้อมูลในการศึกษาโหลดโพล.....                           | 11   |
| 3.4. การคำนวณโหลดโพลด้วยวิธีนิวตัน-ราฟสัน.....              | 12   |
| 3.5. การคำนวณโหลดโพลด้วยวิธีฟาสต์คัปเปิล.....               | 13   |
| บทที่ 4 เอชดีซีโพล.....                                     | 15   |
| 4.1 วงจรสมมูลของระบบกระแสตรงแรงดันสูง.....                  | 15   |
| 4.2 การคำนวณ.....   | 16   |
| บทที่ 5 โปรแกรม.....  | 19   |
| 5.1 การออกแบบ.....  | 19   |
| 5.2 การใช้งานโปรแกรม.....                                   | 22   |

|                               |    |
|-------------------------------|----|
| 5.3 ผลการทดสอบโปรแกรม.....    | 29 |
| 5.4 สรุปผลการทดลอง.....       | 42 |
| บทที่ 6 สรุปผลและวิจารณ์..... | 43 |
| ภาคผนวก                       |    |
| กิตติกรรมประกาศ               |    |
| เอกสารอ้างอิง                 |    |

## สารบัญญภาพ

หน้า

|  |    |
|--|----|
| รูป 2.1 ระบบไฟฟ้ากระแสตรงแรงดันสูงใช้เชื่อมที่ความถี่ต่างกัน.....          | 2  |
| รูป 2.2 กราฟความสามารถในการส่งจ่ายพลังงานกับระยะทาง.....                   | 3  |
| รูป 2.3 กราฟเปรียบเทียบค่าใช้จ่ายของระบบ AC และระบบ HVDC .....             | 3  |
| รูป 2.4 ส่วนประกอบของสถานีระบบไฟฟ้ากระแสตรงแรงดันสูง .....                 | 4  |
| รูป 2.5 การเชื่อมต่อระหว่างระบบไฟฟ้ากระแสสลับโดยระบบกระแสตรงแรงดันสูง..... | 5  |
| รูป 2.6 วงจร Three-phase full-wave controlled rectifier .....              | 6  |
| รูป 2.7 แรงดันของวงจร ที่มุนำกระแสต่างๆ.....                               | 7  |
| รูป 2.8 วงจรเสมือนของหน่วยรีคตีไฟเออร์และอินเวอร์เตอร์.....                | 9  |
| รูป 2.9 แบบจำลองของระบบไฟฟ้ากระแสตรงแรงดันสูง .....                        | 9  |
| รูป 2.10 วงจรเสมือนของระบบไฟฟ้ากระแสตรงแรงดันสูง .....                     | 9  |
| รูป 3.1 ชนิดของบัส.....  | 10 |
| รูป 3.2 วงจรสมมูลสายส่งระหว่างบัส i และบัส j.....                          | 11 |
| รูป 4.1 วงจรเสมือนคอนเวอร์เตอร์.....                                       | 15 |
| รูป 5.1 แผนผังการคำนวณโดยวิธีนิวตัน-ราฟสัน.....                            | 19 |
| รูป 5.2 แผนผังการคำนวณโดยวิธีฟาสต์ดีคัปเปิล.....                           | 20 |
| รูป 5.3 แผนผังการคำนวณ AC-DC โหลดโพล.....                                  | 21 |
| รูป 5.4 หน้าจอขณะเข้าสู่โปรแกรม .....                                      | 22 |
| รูป 5.5 แสดงการเปิดข้อมูลระบบไฟฟ้า.....                                    | 23 |
| รูป 5.6 แสดงข้อมูลบัสและสายส่ง.....  | 23 |
| รูป 5.7 แสดงวิธีการคำนวณโหลดโพล .....                                      | 24 |
| รูป 5.8 แสดงผลจากการคำนวณโหลดโพล.....                                      | 24 |
| รูป 5.9 แสดงผลการคำนวณความสูญเสียในสายส่ง .....                            | 25 |
| รูป 5.10 กราฟแสดงการดูเข้าของคำตอบ .....                                   | 25 |
| รูป 5.11 แสดงผลการคำนวณเอชี่ดีซีโหลดโพล.....                               | 26 |

|   |    |
|---|----|
| รูป 5.12แสดงโปรแกรมสร้างระบบไฟฟ้า .....   | 26 |
| รูป 5.13แสดงการเพิ่มพารามิเตอร์.....  | 27 |
| รูป 5.14แสดงผล Power Flow ผ่านโปรแกรมสร้างระบบไฟฟ้า .....                               | 28 |
| รูป 5.15ข้อมูลระบบไฟฟ้าเอช 14 บัสจากหนังสืออ้างอิง [4] .....                            | 29 |
| รูป 5.16ข้อมูลระบบไฟฟ้าเอชดีซี 14 บัสจากหนังสืออ้างอิง [4] .....                        | 30 |
| รูป 5.17ข้อมูลของส่วน HVDC Link จากหนังสืออ้างอิง [4].....                              | 31 |
| รูป 5.18ข้อมูลของส่วน HVDC Link จากหนังสืออ้างอิง [3].....                              | 31 |
| รูป 5.19แผนภูมิแสดงการเปรียบเทียบผลจากการเพิ่มระบบไฟฟ้ากระแสตรง<br>แรงดันสูง(MW) .....  | 36 |
| รูป 5.20แผนภูมิแสดงการเปรียบเทียบผลจากการเพิ่มระบบไฟฟ้ากระแสตรง<br>แรงดันสูง(MVAR)..... | 36 |
| รูป 5.21แผนภูมิแสดงการเปรียบเทียบผลจากการเพิ่มระบบไฟฟ้ากระแสตรง<br>แรงดันสูง(MVA) ..... | 37 |
| รูป 5.22 ระบบไฟฟ้า 14 บัสที่เชื่อม HVDC ที่บัส 2-3 .....                                | 39 |
| รูป 5.23 ระบบไฟฟ้า 14 บัสที่เชื่อม HVDC ที่บัส 2-4 .....                                | 40 |

## สารบัญตาราง

หน้า

|   |    |
|---|----|
| ตาราง 5.1 เปรียบเทียบผลจากโหนดโพลและข้อมูลอ้างอิง.....  | 32 |
| ตาราง 5.2 เปรียบเทียบแรงดันและมุมกับข้อมูลอ้างอิงเมื่อเพิ่มระบบไฟฟ้ากระแสตรง<br>แรงดันสูง ..... | 33 |
| ตาราง 5.3 แสดงผลกระทบต่อค่าแรงดันและมุมเมื่อเพิ่มระบบไฟฟ้ากระแสตรงแรงดันสูง.....                | 34 |
| ตาราง 5.4 เปรียบเทียบความสูญเสียจากการเพิ่มระบบไฟฟ้ากระแสตรงแรงดันสูง.....                      | 35 |

# บทที่ 1

## บทนำ

เนื่องจากความต้องการใช้กำลังไฟฟ้าในปัจจุบันมีมากขึ้น ทำให้ประเทศต้องซื้อพลังงานไฟฟ้าจากประเทศเพื่อนบ้าน โดยส่งผ่านระบบไฟฟ้ากระแสตรงแรงดันสูง เข้ามาเชื่อมต่อกับระบบไฟฟ้าในประเทศซึ่งเป็นระบบกระแสสลับ โครงการนี้จึงตั้งใจศึกษาการเปลี่ยนแปลงของระบบไฟฟ้าที่จะเกิดขึ้น เนื่องจากการใช้ระบบไฟฟ้ากระแสตรงแรงดันสูง ว่ามีผลกระทบต่อระบบเดิมอย่างไร

### 1.1 วัตถุประสงค์

1. ศึกษาการวิเคราะห์โหลดโฟล (Load Flow) ในระบบไฟฟ้ากระแสสลับ (AC System) และระบบไฟฟ้ากระแสสลับที่มีการเชื่อมต่อกับระบบไฟฟ้ากระแสตรงแรงดันสูง (AC-DC System)
2. เปรียบเทียบผลจากการเพิ่มระบบไฟฟ้ากระแสตรงแรงดันสูง (HVDC Link) เข้าไปในระบบ
3. สร้างโปรแกรมที่สามารถเป็นพื้นฐานในการพัฒนาโปรแกรมวิเคราะห์ระบบไฟฟ้ากำลัง ที่มีประสิทธิภาพเพิ่มขึ้น ต่อไปได้
4. ฝึกการทำงานร่วมกันเป็นกลุ่ม

### 1.2 ขอบเขตของโครงการ

1. สามารถคำนวณหาเอชไอโหลดโฟลโดยวิธีนิวตันราฟสัน และฟาสต์คัปเปิล
2. คำนวณหาเอชดีซีไอโหลดโฟลได้อย่างถูกต้อง โดยมีสายส่งกระแสตรงแรงดันสูงในระบบได้ 1 เส้น
3. สร้างระบบไฟฟ้ากำลังที่ต้องการศึกษาขึ้นเองได้

### 1.3 ขั้นตอนการดำเนินงาน

1. ศึกษาเอชไอโหลดโฟล
2. เขียนโปรแกรมเอชไอโหลดโฟล
3. ศึกษาเอชดีซีไอโหลดโฟล
4. เขียนโปรแกรมในส่วนกราฟฟิค

## บทที่ 2

### ระบบไฟฟ้ากระแสตรงแรงดันสูง (HVDC Link)

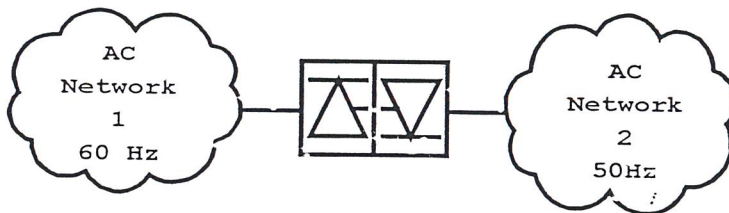
#### 2.1 บทนำ

ในอีกไม่นานนี้ ประเทศไทยเราจะมีการใช้ระบบไฟฟ้ากระแสตรงแรงดันสูงในระบบส่งจ่าย แต่ในต่างประเทศแล้วระบบไฟฟ้ากระแสตรงแรงดันสูง เป็นที่ยอมรับและถูกนำมาใช้นานแล้ว เนื่องจากข้อดีหลายๆ ด้านโดยเฉพาะทางด้านเศรษฐศาสตร์ ซึ่งเชื่อกันว่าระบบไฟฟ้าที่เหมาะสมทางด้านเศรษฐศาสตร์นั้น ประกอบด้วย การผลิต (Generation) เป็นระบบกระแสสลับ ส่งจ่ายทางสายส่ง (Transmission) โดยใช้ระบบกระแสตรงแรงดันสูง และใช้ระบบจำหน่าย (Distribution) เป็นระบบ กระแสสลับ

HVDC มาจากคำว่า High Voltage Direct Current ซึ่งเป็นการส่งจ่ายพลังงานไฟฟ้าในระบบกระแสตรง (แต่ก็ยังผลิตและจ่ายให้ตามบ้านเรือนเป็นระบบกระแสสลับ) ซึ่งมีการใช้คอนเวอร์เตอร์ มาแปลงชนิดแรงดันไฟฟ้า AC-DC และ DC-AC

#### 2.2 ข้อดีของระบบไฟฟ้ากระแสตรงแรงดันสูง

2.2.1 ระบบไฟฟ้ากระแสตรงมีลักษณะเป็น asynchronous link จึงสามารถนำมาใช้เป็นตัวเชื่อมระหว่างระบบไฟฟ้าที่มีความถี่ต่างกันได้หรือระบบไฟฟ้าที่มีความถี่เท่ากันแต่ไม่ได้ synchronous กัน



รูปที่ 2.1 ระบบไฟฟ้ากระแสตรงแรงดันสูงใช้เชื่อมระบบที่มีความถี่ต่างกัน

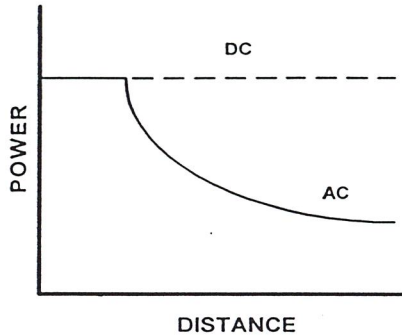
2.2.2 ระบบไฟฟ้ากระแสตรงมีผลของกระแส Charging น้อยกว่าระบบกระแสสลับ จึงสามารถส่งกำลังไฟฟ้าได้ระยะทางไกลกว่า

2.2.3 ระบบไฟฟ้ากระแสตรงมีผลของโคโรนา (corona loss) น้อยกว่าระบบกระแสสลับ ซึ่งผลของโคโรนานี้จะไปรบกวนสัญญาณวิทยุในระบบสื่อสาร

2.2.4 ไม่มี Reactive Power ในการสายส่งจึงมีประสิทธิภาพมากกว่าระบบไฟฟ้ากระแสสลับ

2.2.5 ขนาดของกระแสลัดวงจร (Fault) ที่เกิดขึ้นมีความรุนแรงต่ำกว่าระบบไฟฟ้ากระแสสลับ เนื่องจากกระแสลัดวงจรที่เกิดขึ้นไม่สามารถส่งผ่านระบบสายส่งกระแสตรง

2.2.6 ไม่มีปัญหาด้านเสถียรภาพ โดยความยาวของสายส่งของ HVDC Link ไม่มีขีดจำกัด เนื่องจากมีเพียงความต้านทานเพียงเล็กน้อยเท่านั้น ส่วนในระบบไฟฟ้ากระแสสลับกำลังการส่งจะขึ้นอยู่กับมุมต่าง (Angle difference) ระหว่างแรงดันที่ปลายทั้งสองด้าน ซึ่งจะเพิ่มขึ้นตามระยะทาง ทำให้กำลังส่งของระบบกระแสสลับเป็นฟังก์ชันกับระยะทาง



รูปที่ 2.2 กราฟความสามารถในการส่งจ่ายพลังงานกับระยะทาง

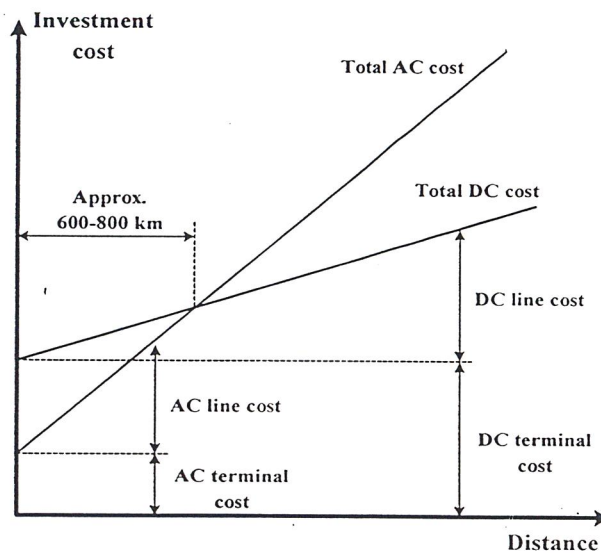
2.2.7 ไม่มีผลของ Skin Effect

2.2.8 สามารถใช้พื้นดินหรือพื้นน้ำเป็นส่วนหนึ่งของสายส่งได้

2.2.9 ในกรณีของสายส่งแบบ 2 ขั้ว (bipolar) ถ้าขั้วใดขั้วหนึ่งเกิดกระแสลัดวงจรขึ้น ขั้วที่เหลือสามารถที่จะส่งจ่ายได้โดยใช้ร่วมกับพื้นดินซึ่งสามารถส่งจ่ายกำลังได้ครึ่งหนึ่งของความจุสายส่งและคอนเวอร์เตอร์

2.2.10 ง่ายต่อการควบคุมกำลังไฟฟ้าโดยเราสามารถเลือกควบคุมแบบ กำลังงานส่งคงที่ , กำลังงานรับคงที่ , กระแสส่งคงที่ ซึ่งสามารถช่วยให้เราส่งจ่ายพลังงานได้อย่างปลอดภัย

2.2.11 โครงสร้างของระบบส่งจ่ายราคาถูกกว่า



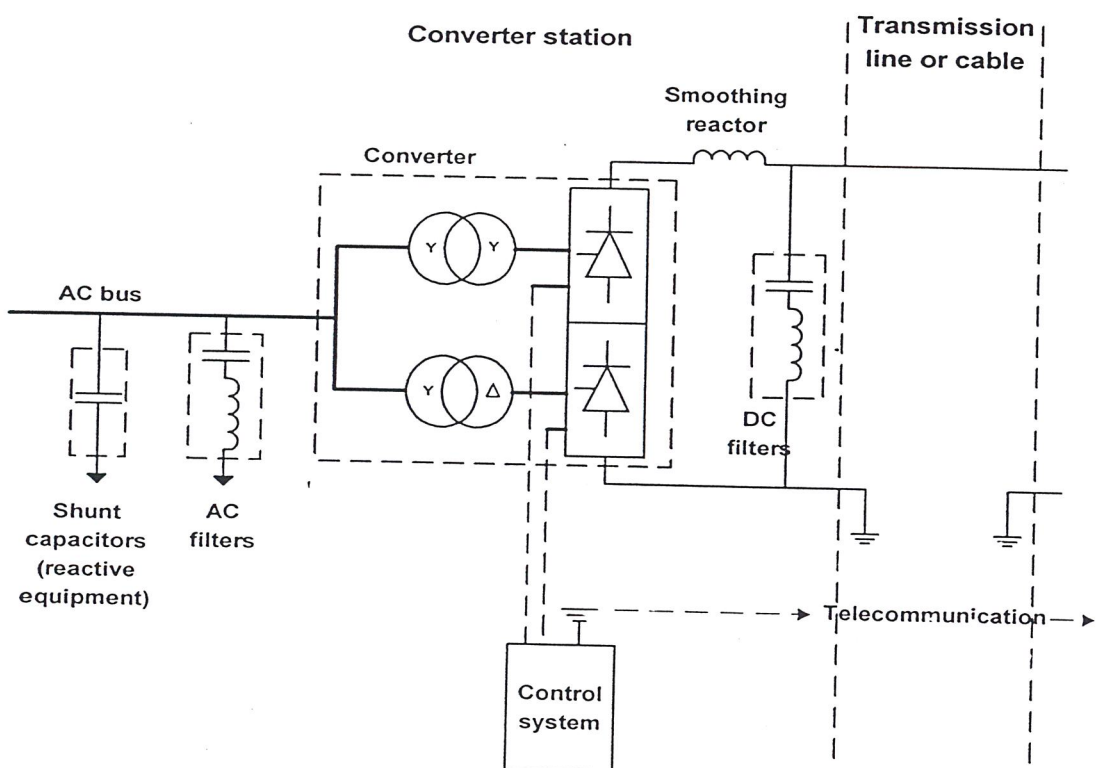
รูปที่ 2.3 กราฟเปรียบเทียบค่าใช้จ่ายของระบบ AC และระบบ HVDC

## 2.3 ข้อเสียของระบบไฟฟ้ากระแสตรงแรงดันสูง

- 2.3.1 คอนเวอร์เตอร์มีราคาสูง
- 2.3.2 คอนเวอร์เตอร์ต้องการ reactive power สูง
- 2.3.3 คอนเวอร์เตอร์รองรับการใช้โหลดเกินได้ไม่มาก
- 2.3.4 คอนเวอร์เตอร์ก่อให้เกิดฮาร์โมนิกส์ ดังนั้นจึงต้องมีฟิลเตอร์

## 2.4 ส่วนประกอบของระบบไฟฟ้ากระแสตรงแรงดันสูง

ส่วนประกอบหลักๆ ของระบบไฟฟ้ากระแสตรงแรงดันสูงแสดงดังรูป 2.4



รูปที่ 2.4 ส่วนประกอบของสถานีระบบไฟฟ้ากระแสตรงแรงดันสูง

### 2.4.1 คอนเวอร์เตอร์ (Converter)

ใช้ในการแปลงไฟ AC/DC ประกอบด้วยตัววาล์วบริดจ์ (valve bridges) และหม้อแปลงคอนเวอร์เตอร์ที่มีแทป ซึ่งวาล์วบริดจ์จะเป็นสวิตช์แรงดันสูงต่อกันเป็นลักษณะแบบ 6 พัลส์ หรือ 12 พัลส์ ส่วนหม้อแปลงทำหน้าที่รักษาระดับให้แรงดันเหมาะสมกับวาล์วบริดจ์

#### 2.4.2 สมูทติงรีแอคเตอร์ (Smoothing reactor)

จะเป็นรีแอคเตอร์ขนาดใหญ่ซึ่งมีค่าความเหนี่ยวนำสูงประมาณ 1 H ต่ออนุกรมกับขั้วแต่ละด้านของสถานีคอนเวอร์เตอร์

#### 2.4.3 ตัวกรองฮาร์โมนิกส์

เนื่องจากคอนเวอร์เตอร์จะสร้างแรงดันและกระแสฮาร์โมนิกส์ออกมาทั้งด้านเอซีและด้านดีซี ซึ่งฮาร์โมนิกส์จะเป็นสาเหตุให้เกิดความร้อนขึ้นในตัวเก็บประจุและเครื่องกำเนิดไฟฟ้าที่อยู่ใกล้เคียง นอกจากนี้ยังไปรบกวนระบบสื่อสารอีกด้วย จึงต้องใช้ตัวกรองทั้งด้านเอซี และดีซี

#### 2.4.4 แหล่งจ่าย Reactive power

คอนเวอร์เตอร์จะใช้ reactive power ประมาณ 50 % ของ reactive power ที่ใช้ในการส่งจ่าย ซึ่งมีปริมาณสูงมาก จึงต้องมีตัวจ่าย reactive power อยู่ใกล้คอนเวอร์เตอร์ สำหรับระบบเอซีใหญ่ๆ จะใช้ในรูปของ shunt capacitor นอกจากนี้ยังอาจอยู่ในรูปของ synchronous condenser หรือ static var-compensator นอกจากนี้ตัวกรองกระแสสลับ (AC filter) ยังช่วยเป็นตัวจ่าย reactive power ได้อีกด้วย

#### 2.4.5 สายส่งกระแสตรง

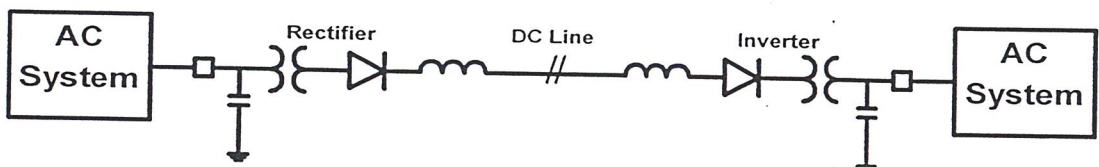
อาจจะอยู่ในรูปสายส่งโอเวอร์เฮด (Over head) หรือเคเบิล โดยถ้าไม่นับเรื่องของจำนวนตัวนำและพื้นที่ที่ต้องใช้แล้ว สายส่งกระแสตรงก็เหมือนกับสายส่งกระแสสลับ

#### 2.4.6 ระบบควบคุม

ใช้ในการควบคุมการส่งจ่ายของพลังงาน โดยควบคุมที่ตัวคอนเวอร์เตอร์

### 2.5 ทฤษฎีระบบไฟฟ้ากระแสตรงแรงดันสูง

ระบบไฟฟ้ากำลังรูปที่ 2.5 ประกอบด้วยระบบไฟฟ้ากระแสสลับ 2 ระบบ ถูกเชื่อมต่อโดยระบบไฟฟ้ากระแสตรงแรงดันสูง ซึ่งประกอบด้วย หน่วยเรกติไฟเออร์ (rectifier unit) สายส่งกระแสตรง และหน่วยอินเวอร์เตอร์ (inverter unit)

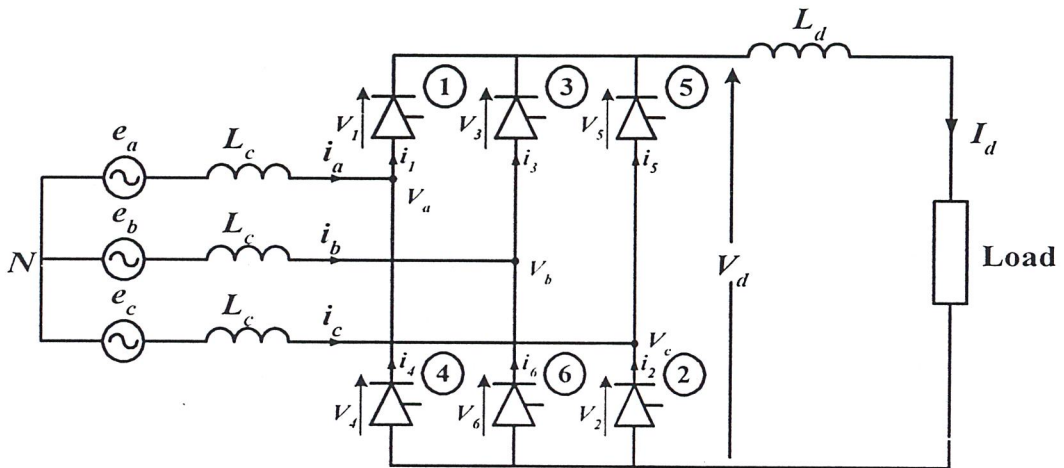


รูปที่ 2.5 การเชื่อมต่อระหว่างระบบไฟฟ้ากระแสสลับ

โดยระบบกระแสตรงแรงดันสูง

### 2.5.1 หน่วยเรกติไฟเออร์ (Rectifier Unit)

ทำหน้าที่เปลี่ยนระบบไฟฟ้ากระแสสลับให้เป็นระบบไฟฟ้ากระแสตรง โดยเราพิจารณาจาก วงจรบริดจ์คอนเวอร์เตอร์ (Bridge Converter Circuit) ลักษณะของ three-phase full-wave controlled rectifier circuit แสดงดังรูป 2.6



รูปที่ 2.6 วงจร Three-phase full-wave controlled rectifier

โดยที่ line-to-neutral voltage

$$e_a = E_m \cos(\omega t + 60^\circ) \quad (2.1)$$

$$e_b = E_m \cos(\omega t - 60^\circ) \quad (2.2)$$

$$e_c = E_m \cos(\omega t - 180^\circ) \quad (2.3)$$

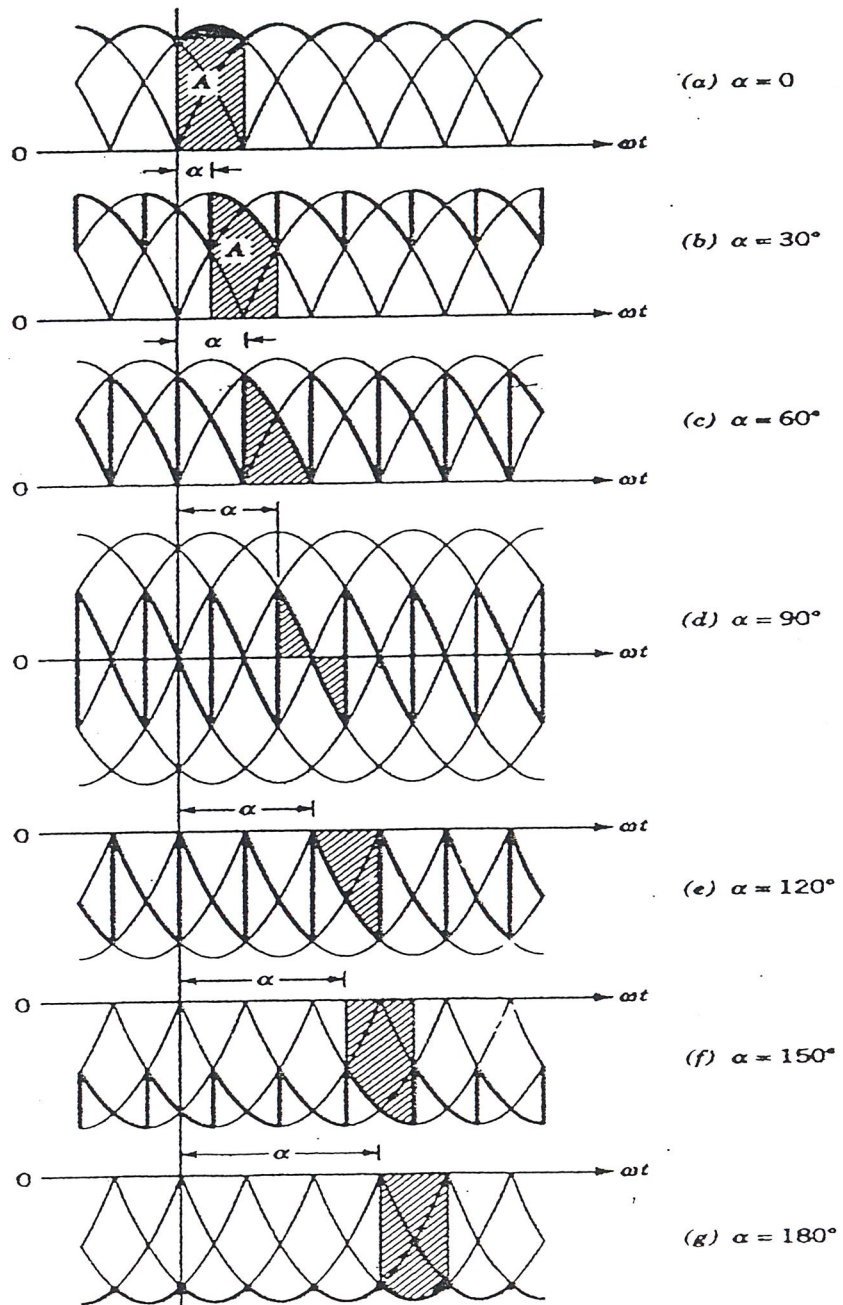
และ line-to-line voltage

$$e_{ab} = e_a - e_b = \sqrt{3} E_m \cos(\omega t + 90^\circ) \quad (2.4)$$

$$e_{bc} = e_b - e_c = \sqrt{3} E_m \cos(\omega t - 30^\circ) \quad (2.5)$$

$$e_{ca} = e_c - e_a = \sqrt{3} E_m \cos(\omega t + 210^\circ) \quad (2.6)$$

วงจรบริดจ์คอนเวอร์เตอร์จะให้แรงดันกระแสตรงออกมาตามมุมนำกระแส (firing angle  $\alpha$ ) ค่าต่างๆ กันแสดงดังรูป 2.7



รูปที่ 2.7 แรงดันของของวงจรที่มุมนำกระแสต่างๆ

เมื่อทำการวิเคราะห์ห้จะได้

แรงดันกระแสตรงสำหรับมุมนำกระแสเป็นศูนย์ ( $\alpha = 0$ );  $V_{d0}$  เป็น

$$\begin{aligned}
 V_{d0} &= \frac{3}{\pi} A_0 = \frac{3}{\pi} \int_{-\frac{\pi}{3}}^0 e_{ac} d\theta \\
 &= \frac{3}{\pi} \int_{-60^\circ}^0 \sqrt{3} E_m \cos(\theta + 30^\circ) d\theta \\
 &= \frac{3\sqrt{3}}{\pi} E_m \sin(\theta + 30^\circ) \Big|_{-60^\circ}^0 \\
 &= \frac{3\sqrt{3}}{\pi} E_m 2 \sin 30^\circ \\
 &= \frac{3\sqrt{3}}{\pi} E_m \\
 &= \frac{3\sqrt{6}}{\pi} E_{LN} \\
 V_{d0} &= K_1 E_{LN} \tag{2.7}
 \end{aligned}$$

โดยให้  $K_1 = \frac{3\sqrt{6}}{\pi}$

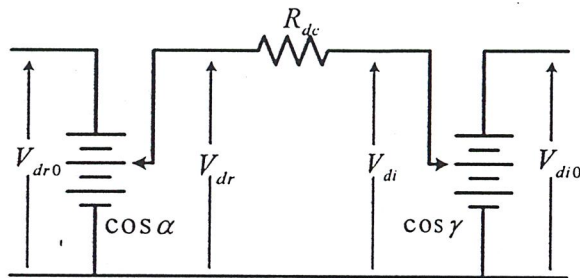
เมื่อกำหนดให้มุมนำกระแสเป็น  $\alpha$  เพราะฉะนั้นจะได้ค่า

$$\begin{aligned}
 V_d &= V_{d0} \int_{\alpha-60^\circ}^{\alpha} \cos(\theta + 30^\circ) d\theta \\
 &= V_{d0} (2 \sin 30^\circ) \cos \alpha \\
 V_d &= V_{d0} \cos \alpha \tag{2.8}
 \end{aligned}$$

ในทำนองเดียวกันเราจะได้  $I_{LN} = \frac{\sqrt{6}}{\pi} I_d$  (2.9)

2.5.2 หน่วยอินเวอร์เตอร์ (Inverter Unit)

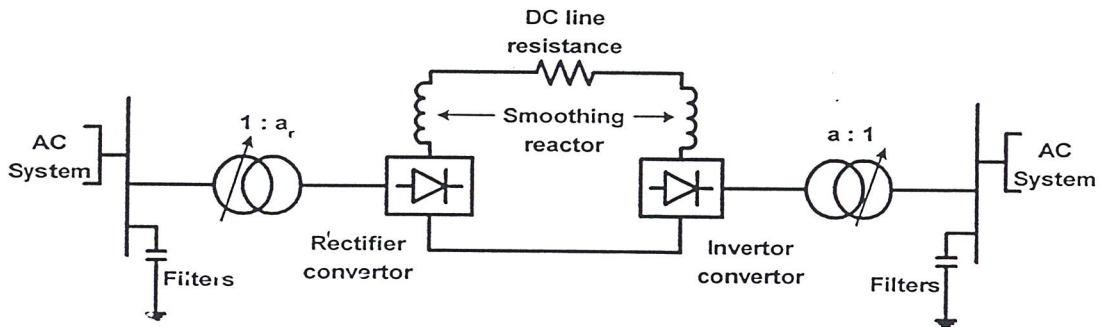
หน่วยนี้ทำหน้าที่เปลี่ยนระบบไฟฟ้ากระแสตรงให้เป็นกระแสสลับ ซึ่งใช้วงจรบริดจ์คอนเวอร์เตอร์เช่นเดียวกับหน่วยเรกติไฟเออร์ แต่ทำงานที่มุมนำกระแสมากกว่า 90 องศา ซึ่งวงจรเสมือนของทั้งเรกติไฟเออร์และอินเวอร์เตอร์แสดงดังรูป



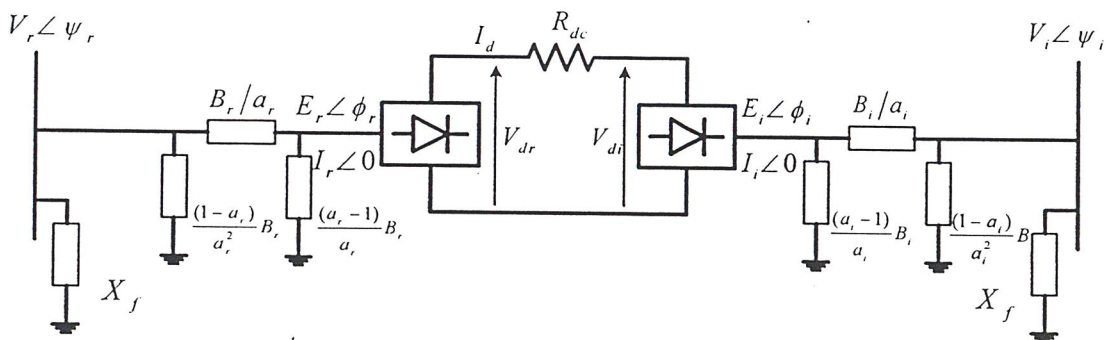
รูป 2.8 วงจรเสมือนของหน่วยเรกติไฟเออร์และอินเวอร์เตอร์

2.6 แบบจำลองและวงจรเสมือนของระบบไฟฟ้ากระแสตรงแรงดันสูง

แบบจำลองและวงจรเสมือนแสดงดังรูป 2.9 และ 2.10 ตามลำดับ



รูปที่ 2.9 แบบจำลองของระบบไฟฟ้ากระแสตรงแรงดันสูง



รูปที่ 2.10 วงจรเสมือนของระบบไฟฟ้ากระแสตรงแรงดันสูง

## บทที่ 3 โหลดโฟล

### 3.1 โหลดโฟล

การคำนวณโหลดโฟล (load flow) เป็นการศึกษาการไหลของกำลังไฟฟ้าและกระแสที่จุดต่างๆ ในระบบไฟฟ้าในสภาวะการทำงานปกติ ผลของโหลดโฟลจะประกอบด้วย ขนาดและมุมของแรงดันไฟฟ้าที่บัสต่างๆ กระแสและกำลังไฟฟ้าที่ไหลในระบบสายส่ง นำไปพิจารณาค่าต่างๆ ให้มีขนาดเหมาะสม เพื่อใช้ในการวางแผนจ่ายไฟฟ้า รวมถึงการขยายงานระบบไฟฟ้า

วิธีการแก้ปัญหาโหลดโฟลมีด้วยกันหลายวิธี แต่ในโครงงานนี้จะศึกษาอยู่ 2 วิธี ได้แก่วิธีนิวตัน-ราฟสัน (Newton-Raphson) และวิธี ฟาสต์ดีคัปเปิล (Fast Decoupled)

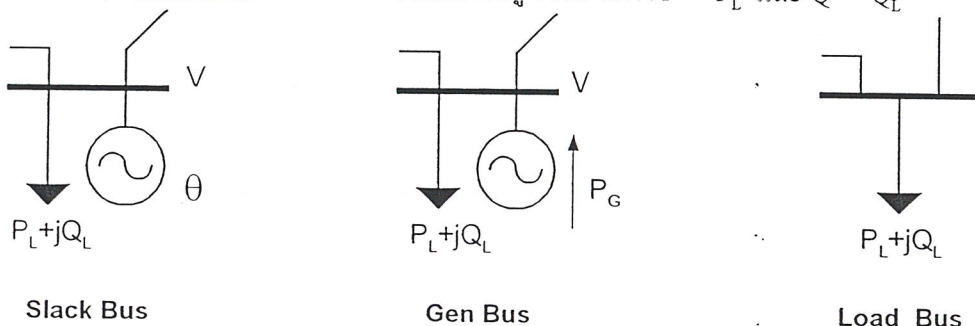
### 3.2 ชนิดของบัส

ในการศึกษาโหลดโฟลเราต้องทราบชนิดของบัสก่อนเพื่อจะได้ทราบว่าบัสแต่ละชนิดมีพารามิเตอร์ใดบ้าง โดยพารามิเตอร์ต่างๆ ของแต่ละบัสประกอบไปด้วย ขนาดและมุมของแรงดันไฟฟ้า กำลังไฟฟ้าจริง และกำลังไฟฟ้าปรากฏ ( $V, \theta, P, Q$ ) โดยทั่วไปแต่ละบัสจะทราบอยู่ 2 ค่า ส่วนอีก 2 ค่าหาจากการคำนวณโหลดโฟล โดยบัสในระบบไฟฟ้ากำลังแบ่งเป็น 3 ชนิดคือ

1) Slack bus (Swing bus , Infinite bus) จะใช้เป็นบัสอ้างอิง โดยมีขนาดแรงดันคงที่มีมุมคงที่ (ปกติกำหนดเป็น 0) บัสนี้จะเป็นแหล่งกำเนิดไฟฟ้าขนาดใหญ่ จะมีพารามิเตอร์ที่คงที่คือ  $V$  และ  $\theta$

2) Generator bus เป็นบัสที่มีเครื่องกำเนิดไฟฟ้า (Generator) ต่ออยู่หรือมีทั้ง เครื่องกำเนิดไฟฟ้าและโหลดต่ออยู่ บัสนี้สามารถควบคุมแรงดันให้คงที่โดยใช้ AVR (Automatic Voltage Regulator) มี 2 ประเภทคือ PV bus : มีค่า  $V, P$  คงที่ และ PQ bus : มีค่า  $P, Q$  คงที่

3) Load Bus เป็นบัสที่มีโหลดต่ออยู่ จะทราบค่า  $P = -P_L$  และ  $Q = -Q_L$



รูปที่ 3.1 แสดงชนิดของบัส

3.3 ข้อมูลในการศึกษาโหลดโพล

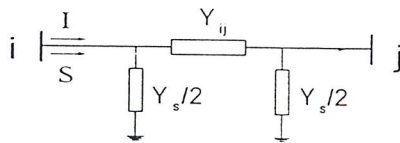
3.3.1 ข้อมูลเพื่อใช้ในการคำนวณ

- 1) ข้อมูลระบบเพื่อใช้หา  $Y_{bus}$  หรือ  $Z_{bus}$  ของระบบประกอบไปด้วย
  - one line diagram
  - อิมพีแดนซ์หรือแอดมิตแตนซ์ของสายส่ง
  - ขนาดและอิมพีแดนซ์ของหม้อแปลงและเครื่องกำเนิดไฟฟ้าในระบบ
  - ขนาดคาปาซิเตอร์ที่ใช้ในระบบ
- 2) ข้อมูลแสดงการทำงานของระบบ
  - PQ bus (load bus)
  - PV bus (gen. Bus)
  - V ของ slack bus

3.3.2 ข้อมูลที่ต้องการจากการคำนวณโหลดโพล

- 1) แรงดันไฟฟ้าแต่ละบัส เป็นข้อมูลอันดับแรกที่สำคัญมาก
- 2) กระแสในสายส่ง สามารถหากระแสระหว่างบัส  $i$  กับบัส  $j$  โดยใช้ค่าแรงดันไฟฟ้าที่ได้จากข้อ 1 มาคำนวณโดยใช้สมการ  $I_{ij} = (V_i - V_j) Y_{ij} + V_i \frac{Y_s}{2}$

$$I_{ij} = (V_i - V_j) Y_{ij} + V_i \frac{Y_s}{2} \tag{3.1}$$



รูปที่ 3.2 แสดงวงจรสมมุติฐานสายส่งระหว่างบัส  $i$  และบัส  $j$

- 3) กำลังไฟฟ้าในสายส่ง กำลังไฟฟ้าที่ไหลเข้าในระบบสามารถหาได้จากสมการ

$$S = P + jQ = V_i I_i^* \tag{3.2}$$

$$= V_i \sum_{j=0}^n Y_{ij}^* V_j \tag{3.3}$$

### 3.4 การคำนวณโหลดโพลด้วยวิธีนิวตัน-ราฟสัน

วิธีของนิวตัน-ราฟสันใช้ทฤษฎีของอนุกรมเทเลอร์ (Taylor series) โดยใช้ค่าที่กำหนดไว้ในแต่ละบัส เป็นตัวเปรียบเทียบเพื่อไปหาค่าแรงดันไฟฟ้าของแต่ละบัสโดยใช้สมการ

$$[\Delta B] = [J] [\Delta V] \quad (3.4)$$

$[\Delta B]$  - เมตริกซ์ผลต่างของค่าที่กำหนดให้กับค่าที่คำนวณได้ในแต่ละบัส

$[J]$  - จาคอบีเยนเมตริกซ์

$[\Delta V]$  - เมตริกซ์ผลต่างแรงดันไฟฟ้า

โดยในการคำนวณนี้จะใช้เป็นลักษณะชุดสมการดังนี้

$$\begin{bmatrix} \Delta P^{p-1} \\ \Delta Q^{p-1} \end{bmatrix} = \begin{bmatrix} H^{p-1} & N^{p-1} \\ J^{p-1} & L^{p-1} \end{bmatrix} \begin{bmatrix} \Delta \theta^p \\ \frac{\Delta V^p}{V^{p-1}} \end{bmatrix} \quad (3.5)$$

Jacobian matrix

$$\text{โดย} \quad \Delta P = P_{spec} - P_{cal} \quad (3.6)$$

$$\Delta Q = Q_{spec} - Q_{cal} \quad (3.7)$$

และเราสามารถหา  $P_{cal}$  และ  $Q_{cal}$  ได้จาก

$$P_{cal} = \sum_{i=1}^j V_i V_j (G_{ij} \cos \theta_{ij} + B_{ij} \sin \theta_{ij}) \quad (3.8)$$

$$Q_{cal} = \sum_{i=1}^j V_i V_j (G_{ij} \sin \theta_{ij} - B_{ij} \cos \theta_{ij}) \quad (3.9)$$

$$\text{โดยที่} \quad \theta_{ij} = \theta_i - \theta_j$$

และหาจาโคเบียนเมตริกซ์ได้จาก

$$H_{ij} = \frac{\partial P_i}{\partial \theta_j} = V_i V_j (G_{ij} \sin \theta_{ij} - B_{ij} \cos \theta_{ij}) \quad (3.10)$$

$$N_{ij} = V_j \frac{\partial P_i}{\partial V_j} = V_i V_j (G_{ij} \cos \theta_{ij} + B_{ij} \sin \theta_{ij}) \quad (3.11)$$

$$J_{ij} = \frac{\partial Q_i}{\partial \theta_j} = -V_i V_j (G_{ij} \cos \theta_{ij} + B_{ij} \sin \theta_{ij}) \quad (3.12)$$

$$E_{ij} = V_j \frac{\partial Q_i}{\partial V_j} = V_i V_j (G_{ij} \sin \theta_{ij} - B_{ij} \cos \theta_{ij}) \quad (3.13)$$

และสำหรับ  $i=j$

$$H_{ii} = \frac{\partial P_i}{\partial \theta_i} = -Q_i - B_{ii}V_i^2 \quad (3.14)$$

$$N_{ii} = V_i \frac{\partial P_i}{\partial V_i} = P_i + G_{ii}V_i^2 \quad (3.15)$$

$$J_{ii} = \frac{\partial Q_i}{\partial \theta_i} = P_i - G_{ii}V_i^2 \quad (3.16)$$

$$L_{ii} = V_i \frac{\partial Q_i}{\partial V_i} = Q_i - B_{ii}V_i^2 \quad (3.17)$$

### 3.5 การคำนวณโหลดโพลด้วยวิธีฟาสต์ดีคัปเปิล

จากสมการหลักของนิวตันราฟสัน

$$\begin{bmatrix} \Delta P \\ \Delta Q \end{bmatrix} = \begin{bmatrix} H & N \\ J & L \end{bmatrix} \begin{bmatrix} \Delta \theta \\ \frac{\Delta V}{V} \end{bmatrix} \quad (3.18)$$

การเปลี่ยนแปลงของ  $\frac{\Delta V}{V}$  มีผลต่อ  $\Delta P$  น้อยมาก

การเปลี่ยนแปลงของ  $\Delta \theta$  มีผลต่อ  $\Delta Q$  น้อยมาก

จะได้สมการหลักใหม่เป็น

$$\begin{bmatrix} \Delta P \\ \Delta Q \end{bmatrix} = \begin{bmatrix} H & 0 \\ 0 & L \end{bmatrix} \begin{bmatrix} \Delta \theta \\ \frac{\Delta V}{V} \end{bmatrix} \quad (3.19)$$

จากสมการของ H และ L

- $V_i, V_{ij} = 1$  p.u.
- $G_{ij} \ll B_{ij}$  (สายส่งในปัจจุบัน  $R \ll X$ )
- $\cos(\theta_i - \theta_j) = 1$  p.u.
- $Q_i \ll B_{ii}V_i^2$

$$\text{จะได้} \quad [\Delta P] = [V \cdot \bar{B} \cdot V] [\Delta \theta] \quad (3.20)$$

$$[\Delta Q] = [V \cdot \bar{B} \cdot V] \left[ \frac{\Delta V}{V} \right] \quad (3.21)$$

โดยที่ 
$$\bar{B}_{ij} = -B_{ij} \quad (3.22)$$

$$\bar{B}_{ii} = \sum_{i=1}^j B_{ij} \quad (3.23)$$

เพื่อเพิ่มประสิทธิภาพของวิธีการทำได้โดย

1. ตัดองค์ประกอบซึ่งมีผลต่อ reactive power มากกว่า real power ออก  $[\bar{B}]$  จากสมการ (19) ได้แก่ shunt reactor, off-nominal in-phase transformer tap
2. ตัดองค์ประกอบซึ่งมีผลต่อการชิวเฟสของแรงดันออกจาก  $[\bar{B}]$  ในสมการ (20) ได้แก่ phase shifter
3. จัดสมการ(19),(20) ใหม่โดยย้ายค่า  $V$  ที่อยู่ข้างซ้ายไปยังด้าน power mismatch และกำหนดค่า  $V$  ซึ่งอยู่ทางขวาเป็น 1.0 p.u.

เพราะฉะนั้น จะได้สมการของ ฟาสต์ดีคัปเปิลโหนดโพล คือ

$$[\Delta P/V] = [B'] [\Delta \theta] \quad (3.24)$$

$$[\Delta Q/V] = [B''] [\Delta V] \quad (3.25)$$

โดยที่

$$B'_{ij} = -\frac{1}{X_{ij}}$$

$$B'_{ii} = \sum_{i=1}^j \frac{1}{X_{ij}}$$

$$B''_{ij} = -B_{ij}$$

$$B''_{ii} = \sum_{i=1}^j B_{ij}$$

## บทที่ 4

### เอชดีซีโพลดิฟล

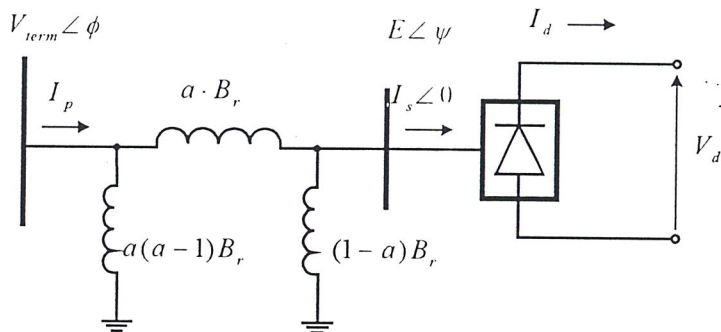
การทำ ACDC loadflow พัฒนามาจากวิธี fast decouple โดยการเพิ่มส่วนของระบบไฟฟ้ากระแสตรงเข้าไป แล้วใช้อัลกอริทึมเดิมในการหาคำตอบ โดยส่วนที่เพิ่มเข้ามา เป็นฟังก์ชันของ โวลเตจของบัสที่ต่อกับคอนเวอร์เตอร์และ ตัวแปรจากระบบกระแสตรงดังนี้

$$\bar{R}(\bar{x}, V_{term})_k = 0 \quad (4.1)$$

#### 4.1 แบบจำลองระบบไฟฟ้ากระแสตรง

ตัวแปร  $x$  ที่ภาวะปกติของคอนเวอร์เตอร์ จำเป็นต้องสมมติเงื่อนไขต่าง ๆ ดังนี้

1. โวลเตจของไฟฟ้ากระแสสลับต้องสมดุลและเป็นสัญญาณรูปไซน์
2. คอนเวอร์เตอร์ต้องให้โวลเตจที่สมดุล
3. กระแสไฟตรงและแรงดันต้องเป็นไฟที่เรียบมาก
4. หม้อแปลงไม่มีการสูญเสีย และไม่คิดแมกเนไทซิง แอทมิตแตนซ์



รูปที่ 4.1 วงจรเสมือนคอนเวอร์เตอร์

จากรูป 4.1

- $V$  คือ โวลเตจที่คอนเวอร์เตอร์บัสบาร์  
 $E$  คือ โวลเตจที่หม้อแปลงด้านทุติยภูมิ  
 $I_p, I_s$  คือ กระแสฟลักซ์ตามทอลที่หม้อแปลงด้านปฐมภูมิและด้านทุติยภูมิตามลำดับ

ลำดับ

- $\alpha$  คือ มุมทริก  
 $a$  คือ อัตราส่วนหม้อแปลง  
 $V_d$  คือ โวลเตจเฉลี่ย  
 $I_d$  คือ กระแสตรง

ตัวแปร  $\bar{x}$  ที่เกิดจากคอนเวอร์เตอร์แต่ละด้าน แสดงได้ดังนี้

$$\bar{x} = [V_d, I_d, \alpha, \cos \alpha, \phi]^T \tag{4.2}$$

จากสมการที่ 4.1 ซึ่งเป็นส่วนที่เพิ่มเข้ามา มีรายละเอียดดังนี้

จากเรคตีไฟเออร์

$$R(1) = V_d - k_1 \alpha V_{term} \cos \phi \tag{4.3}$$

$$R(2) = V_d - k_1 \alpha V_{term} \cos \alpha + \frac{3}{\pi} I_d X_c \tag{4.4}$$

$$R(3) = V_{dm} + V_{dn} - R_{dc} I_{dc} \tag{4.5}$$

$$R(4) = \cos \alpha - \cos \alpha^{sp} \tag{4.6}$$

$$R(5) = V_d I_d - P_d^{sp} \tag{4.7}$$

จากอินเวอร์เตอร์

$$R(1) = V_d - k_1 \alpha V_{term} \cos \phi \tag{4.8}$$

$$R(2) = V_d - k_1 \alpha V_{term} \cos(\pi - \gamma) - \frac{3}{\pi} I_d X_c \tag{4.9}$$

$$R(3) = V_{dm} + V_{dn} - R_{dc} I_{dc} \tag{4.10}$$

$$R(4) = \cos(\pi - \gamma) - \cos(\pi - \gamma^{sp}) \tag{4.11}$$

$$R(5) = V_d - V_d^{sp} \tag{4.12}$$

สำหรับ อินเวอร์เตอร์  $V_d$  จะมีค่าเป็นลบ

โดยที่

$$Q_{term}(dc) = V_{term} K_1 \alpha I_d \sin \phi \tag{4.13}$$

$$P_{term}(dc) = V_{term} k_1 \alpha I_d \cos \phi \tag{4.14}$$

$$P_{term}(dc) = V_d I_d \tag{4.15}$$

#### 4.2 การคำนวณ

นำสมการจากฟาสต์คัปเปิลมาพัฒนา โดยเพิ่มส่วนระบบไฟฟ้ากระแสตรงเข้าไป แสดงได้ดังนี้

|                                    |   |      |       |            |        |                               |
|------------------------------------|---|------|-------|------------|--------|-------------------------------|
| $\frac{\Delta \bar{P}}{\bar{V}}$   | = | $B'$ |       |            |        | $\frac{\Delta \bar{\theta}}$  |
| $\frac{\Delta P_{term}}{V_{term}}$ |   |      |       |            | $AA'$  | $\frac{\Delta \theta_{term}}$ |
| $\frac{\Delta \bar{Q}}{\bar{V}}$   |   |      | $B''$ |            |        | $\frac{\Delta \bar{V}}$       |
| $\frac{\Delta Q_{term}}{V_{term}}$ |   |      |       | $B''_{ii}$ | $AA''$ | $\frac{\Delta V_{term}}$      |
| $\bar{R}$                          |   |      |       | $BB''$     | $A$    | $\frac{\dot{\Delta x}}$       |

โดยที่

$$DD = 0 \quad (4.16)$$

$$[AA'] = \frac{1}{V_{term}} \left[ \frac{\partial P_{term}(dc)}{\partial \bar{x}} \right]$$

$$= \begin{bmatrix} \Delta\phi & \Delta\alpha & \Delta I_d & \Delta \cos\alpha & \Delta V_d \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 & V_d/V_{term} & 0 & 0 \end{bmatrix} \quad (4.17)$$

$$[AA''] = \frac{1}{V_{term}} \left[ \frac{\partial Q_{term}(dc)}{\partial \bar{x}} \right]$$

$$= \begin{bmatrix} K_1 a I_d \cos\phi & K_1 I_d \sin\phi & K_1 a \sin\phi & 0 & 0 \end{bmatrix} \quad (4.18)$$

$$[BB''] = \frac{\partial \bar{R}}{\partial V_{term}} \quad (4.19)$$

$$= [-k_1 a I_d \cos\phi, -k_1 a \cos\alpha, 0, 0, 0, -k_1 a \cos\phi, -k_1 a \cos(\pi - \gamma), 0, 0, 0]^T$$

$$[A] = \frac{\partial \bar{R}}{\partial \bar{x}} \quad (4.20)$$

$$=$$

|                           |                            |                     |                   |       |                           |                            |                      |                   |   |
|---------------------------|----------------------------|---------------------|-------------------|-------|---------------------------|----------------------------|----------------------|-------------------|---|
| $K_1 a V_{term} \sin\phi$ | $-K_1 V_{term} \cos\phi$   | 0                   | 0                 | 1     | 0                         | 0                          | 0                    | 0                 | 0 |
| 0                         | $-K_1 V_{term} \cos\alpha$ | $\frac{3}{\pi} X_c$ | $-K_1 a V_{term}$ | 1     | 0                         | 0                          | 0                    | 0                 | 0 |
| 0                         | 0                          | $-R_{dc}$           | 0                 | 1     | 0                         | 0                          | 0                    | 0                 | 1 |
| 0                         | 0                          | 0                   | 1                 | 0     | 0                         | 0                          | 0                    | 0                 | 0 |
| 0                         | 0                          | $V_d$               | 0                 | $I_d$ | 0                         | 0                          | 0                    | 0                 | 0 |
| 0                         | 0                          | 0                   | 0                 | 0     | $K_1 a V_{term} \sin\phi$ | $-K_1 V_{term} \cos\phi$   | 0                    | 0                 | 1 |
| 0                         | 0                          | 0                   | 0                 | 0     | 0                         | $-K_1 V_{term} \cos\alpha$ | $-\frac{3}{\pi} X_c$ | $-K_1 a V_{term}$ | 1 |
| 0                         | 0                          | 0                   | 0                 | 1     | 0                         | 0                          | $-R_{dc}$            | 0                 | 1 |
| 0                         | 0                          | 0                   | 0                 | 0     | 0                         | 0                          | 0                    | 1                 | 0 |
| 0                         | 0                          | 0                   | 0                 | 0     | 0                         | 0                          | 0                    | 0                 | 1 |

$$B''_{ii} = \frac{1}{V_{term}} \frac{\partial Q_{term}(ac)}{\partial V_{term}} + \frac{1}{V_{term}} \left[ \frac{\partial Q_{term}(dc)}{\partial V_{term}} \right] \quad (4.21)$$

$$= B''_{ii}(ac) + \frac{k_1 a I_d \sin\phi}{V_{term}}$$

ในทวนคี่ซิมิตแมชและตัวแปรทั้งหมดสามารถแยกเป็นสองส่วน เหมือนกับสมการฟาสต์คัปเปิลของระบบกระแสกลับได้ ดังนี้

$$\begin{array}{|c|} \hline \Delta \bar{P} / \bar{V} \\ \hline \Delta P_{term} / V_{term} \\ \hline \bar{R} \\ \hline \end{array} = \begin{array}{|c|c|} \hline B' & \\ \hline & AA' \\ \hline & A \\ \hline \end{array} \begin{array}{|c|} \hline \Delta \bar{\theta} \\ \hline \Delta \theta_{term} \\ \hline \Delta \bar{x} \\ \hline \end{array}$$

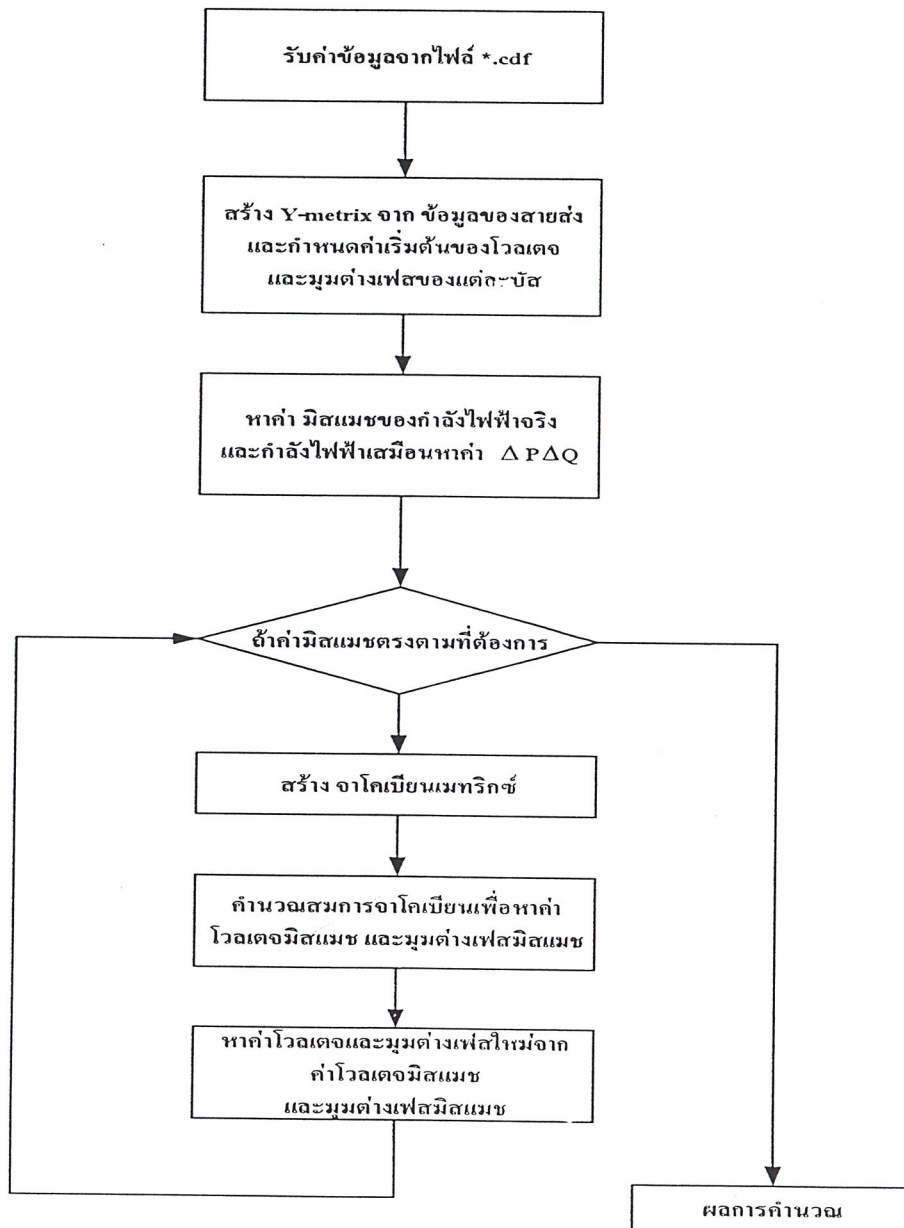
$$\begin{array}{|c|} \hline \Delta \bar{Q} / \bar{V} \\ \hline \Delta Q_{term} / V_{term} \\ \hline \bar{R} \\ \hline \end{array} = \begin{array}{|c|c|} \hline B'' & \\ \hline & B''_{ii} \quad AA'' \\ \hline & BB'' \quad A \\ \hline \end{array} \begin{array}{|c|} \hline \Delta \bar{V} \\ \hline \Delta V_{term} \\ \hline \Delta \bar{x} \\ \hline \end{array}$$

ลำดับในการคำนวณแสดงดังรูปที่ 5.3

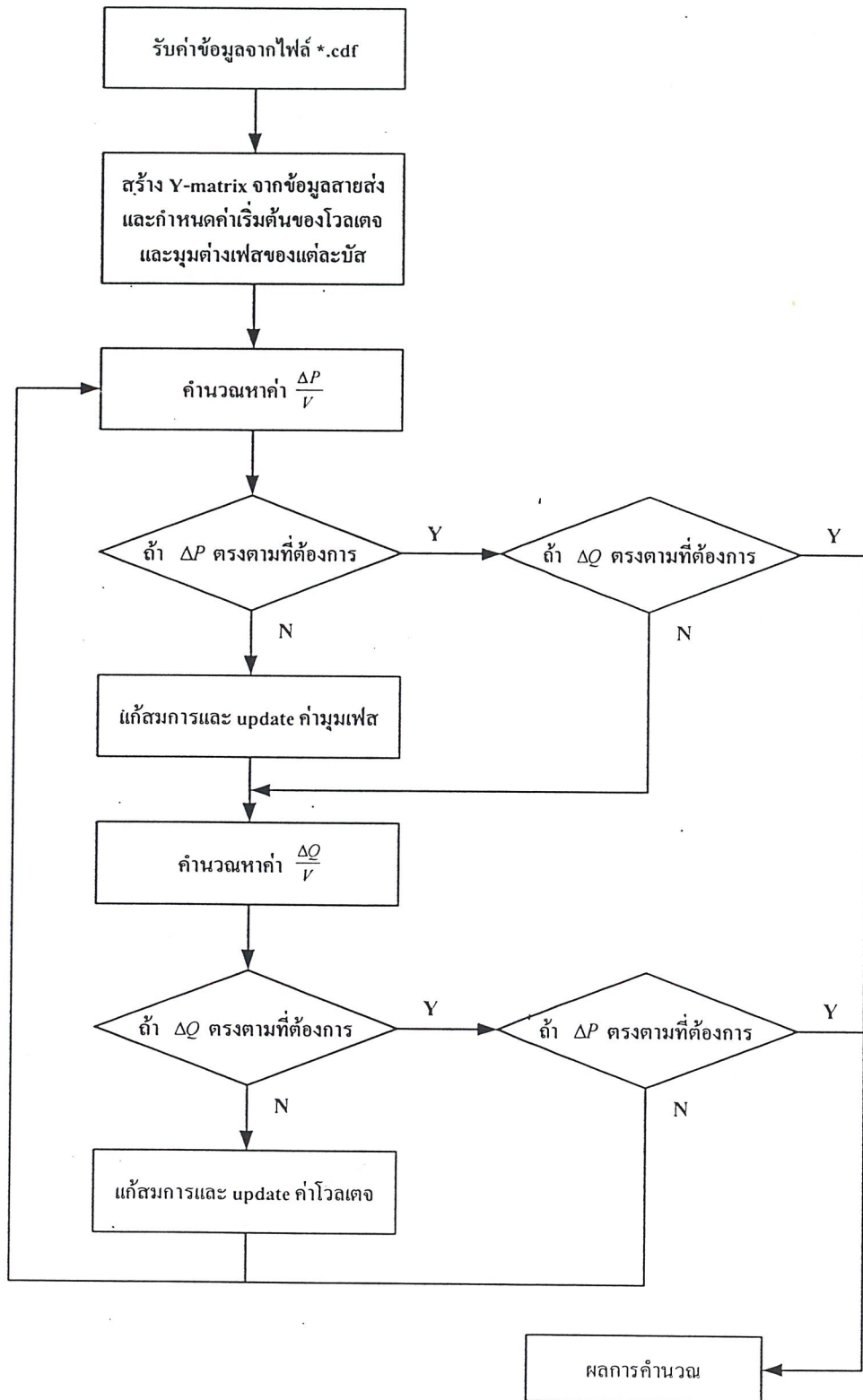
## บทที่ 5 โปรแกรม

### 5.1 การออกแบบโปรแกรม

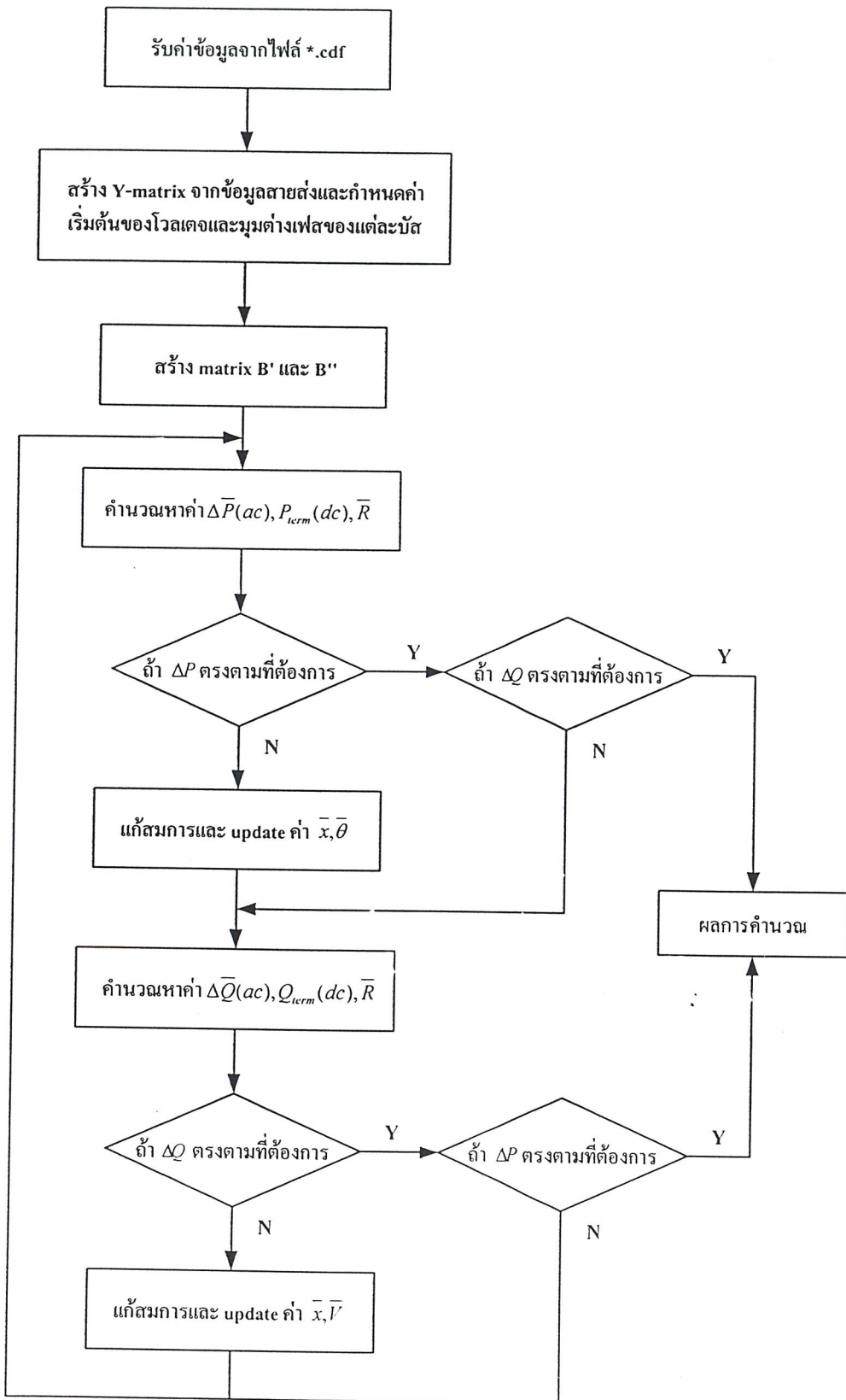
ใช้โปรแกรมเดลไฟล์ (Delphi) เป็นเครื่องมือในการพัฒนาโปรแกรม และโปรแกรม ถูกเขียนเป็นลักษณะโปรแกรมเชิงวัตถุ (Object Oriented Program, OOP) เพื่อให้ง่ายต่อการแก้ไข โปรแกรมมีลำดับขั้นตอนการทำงานในวิธีต่างๆ ดังต่อไปนี้



รูปที่ 5.1 แสดงแผนผังการคำนวณ โดยวิธีนิวตัน-ราฟสัน



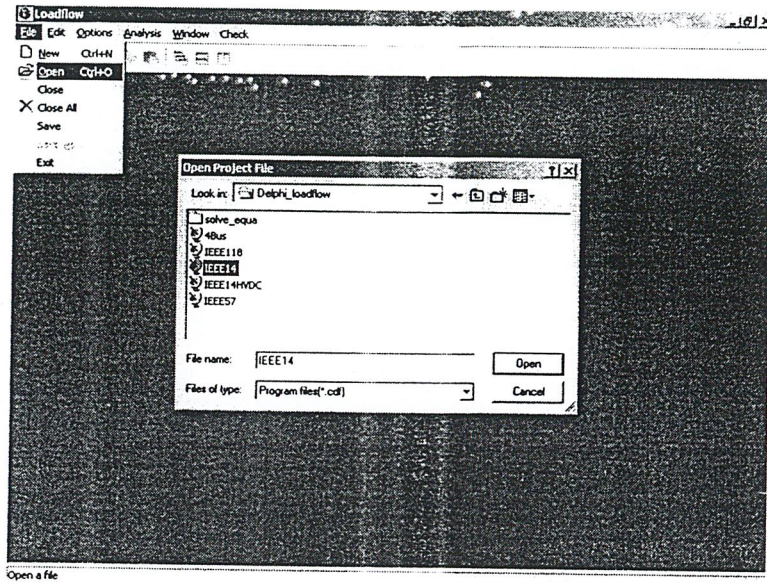
รูปที่ 5.2 แสดงแผนผังการคำนวณโดยวิธีฟาสต์ดีคัปเปิล



รูปที่ 5.3 แสดงแผนผังการคำนวณของวิธี AC-DC โหลดโฟล



2. เลือก File | Open จากเมนูบาร์ของโปรแกรม เพื่อเปิดไฟล์ \*.cdf ซึ่งเป็นข้อมูลของระบบไฟฟ้าที่จะใช้ในการคำนวณ



รูปที่ 5.5 แสดงการเปิดข้อมูลระบบไฟฟ้า

3. ถ้าต้องการดูข้อมูลของบัสและสายส่ง สามารถทำได้โดยเลือก Options | Show Bus Data และ Show Branch Data ตามลำดับ

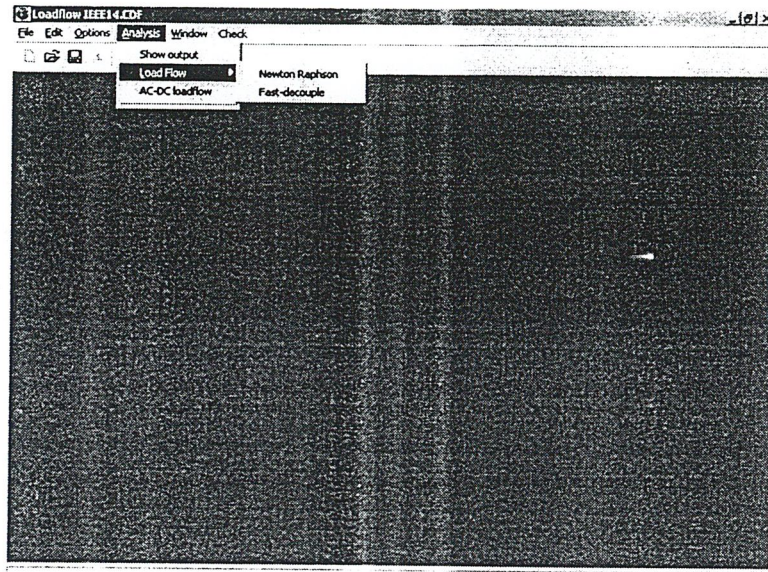
| Bus | Name     | Area number | Loss zone number | Type | Final voltage | Final angle | Load MW | Load MVAR | Gen MW |
|-----|----------|-------------|------------------|------|---------------|-------------|---------|-----------|--------|
| 1   | Bus 1 HV | 1           | 1                | 3    | 1.06          | 0           | 0       | 0         | 232.4  |
| 2   | Bus 2 HV | 1           | 1                | 2    | 1.045         | -4.98       | 21.7    | 12.7      | 40     |
| 3   | Bus 3 HV | 1           | 1                | 2    | 1.01          | -12.72      | 94.2    | 19        | 0      |
| 4   | Bus 4 HV | 1           | 1                | 0    | 1.015         | -10.33      | 47.6    | -3.9      | 0      |
| 5   | Bus 5 HV | 1           | 1                | 0    | 1.02          | -8.78       | 7.6     | 1.6       | 0      |
| 6   | Bus 6 LV | 1           | 1                | 2    | 1.07          | -14.22      | 11.2    | 7.5       | 0      |
| 7   | Bus 7 2V | 1           | 1                | 0    | 1.062         | -13.37      | 0       | 0         | 0      |

| Line | From Bus | To Bus | Area | Loss zone | Circuit | Type | Resistance | Reactance | Line charging | MVA rating No 1 | MVA rating |
|------|----------|--------|------|-----------|---------|------|------------|-----------|---------------|-----------------|------------|
| 1    | 1        | 2      | 1    | 1         | 1       | 0    | 0.01938    | 0.05917   | 0.0264        | 0               | 0          |
| 2    | 1        | 5      | 1    | 1         | 1       | 0    | 0.05462    | 0.22304   | 0.0246        | 0               | 0          |
| 3    | 2        | 3      | 1    | 1         | 1       | 0    | 0.04635    | 0.19797   | 0.0219        | 0               | 0          |
| 4    | 2        | 4      | 1    | 1         | 1       | 0    | 0.05811    | 0.17632   | 0.0187        | 0               | 0          |
| 5    | 2        | 5      | 1    | 1         | 1       | 0    | 0.05695    | 0.17388   | 0.017         | 0               | 0          |
| 6    | 3        | 4      | 1    | 1         | 1       | 0    | 0.06701    | 0.17103   | 0.0173        | 0               | 0          |
| 7    | 4        | 5      | 1    | 1         | 1       | 0    | 0.01325    | 0.04211   | 0.0064        | 0               | 0          |

รูปที่ 5.6 แสดงข้อมูลบัสและสายส่ง

4. ในการคำนวณเอชไอแอลโพล เลือกไปที่ Analysis | Load Flow ซึ่งสามารถเลือกกว่าจะใช้วิธีนิวตันราฟสันหรือวิธีฟาสต์ดีคัปเปิล และในกรณีเอชไอแอลโพลเลือกที่ Analysis | AC-DC loadflow



รูปที่ 5.7 แสดงวิธีการคำนวณโหลดโพล

5. เมื่อคำนวณเสร็จแล้วจะแสดงผล Power Flow ออกมาคังรูป และสามารถดูค่า Transmission losses ได้ นอกจากนี้ยังแสดงกราฟของมิตแมชต์ (mismatch) ในการคำนวณ ส่วนในกรณีของระบบเอชไอแอลโพล จะแสดงส่วนข้อมูลของHVDC Link ด้วย

Output Power Flow Analysis  
Loss flow HVDC link Graph mismatch

Power Flow Analysis P\_iter6 Q\_iter5  
Time used 0.05 sec

| From Bus | Name     | Volt Mag | Volt Angle | MW Gen  | MVAR Gen | MW Load | MVAR Load | To Bus | MW Flow  | MVAR Flow | MVA Flow |
|----------|----------|----------|------------|---------|----------|---------|-----------|--------|----------|-----------|----------|
| 1        | Bus 1 HV | 1.060    | 0.000      | 232.657 | -22.420  | 0.000   | 0.000     | 2      | 157.659  | -20.586   | 158.998  |
|          |          |          |            |         |          |         |           | 5      | 75.011   | -1.827    | 75.033   |
| 2        | Bus 2 HV | 1.045    | 5.068      | 40.000  | 11.224   | 21.700  | 12.700    | 1      | -153.319 | 27.969    | 155.852  |
|          |          |          |            |         |          |         |           | 3      | 72.518   | 3.631     | 72.608   |
|          |          |          |            |         |          |         |           | 4      | 59.612   | -27.521   | 65.658   |
|          |          |          |            |         |          |         |           | 5      | 39.489   | -5.570    | 39.880   |
| 3        | Bus 3 HV | 1.010    | -12.671    | 0.000   | -1.990   | 14.200  | 19.000    | 2      | -70.240  | 1.343     | 70.253   |
|          |          |          |            |         |          |         |           | 4      | -23.960  | -22.331   | 32.753   |
| 4        | Bus 4    | 1.061    | -11.217    | 0.000   | 0.000    | 47.600  | 0.500     | 2      | -57.376  | 30.100    | 64.820   |
|          |          |          |            |         |          |         |           | 3      | 24.615   | 20.290    | 31.900   |
|          |          |          |            |         |          |         |           | 7      | 27.577   | 2.775     | 27.716   |
|          |          |          |            |         |          |         |           | 9      | 15.915   | 4.710     | 16.597   |
| 5        | Bus 5    | 1.022    | 8.772      | 0.000   | 0.000    | 1.600   | 1.600     | 1      | -72.304  | 7.624     | 72.705   |
|          |          |          |            |         |          |         |           | 2      | -38.609  | 4.406     | 36.918   |
|          |          |          |            |         |          |         |           | 6      | 44.774   | 18.468    | 49.433   |
| 6        | Bus 6 LV | 1.070    | 14.237     | 0.006   | 0.037    | 11.200  | 7.900     | 5      | -44.774  | -13.645   | 46.807   |
|          |          |          |            |         |          |         |           | 11     | 7.092    | 0.761     | 7.492    |

รูปที่ 5.8 แสดงผลจากการคำนวณโหลดโพล

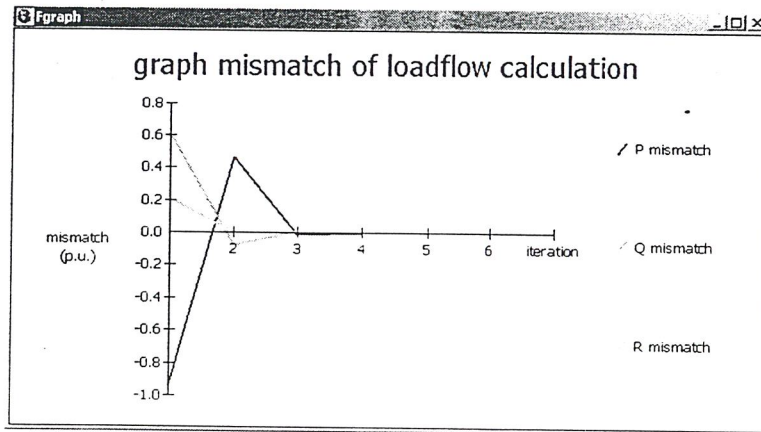
Output Transmission Losses Analysis  
loadflow HVDC link

Transmission Losses Analysis

P\_iter 6 Q\_iter 5

| From Bus | Name     | Volt Mag | Volt Angle | MW Gen  | MVAR Gen | MW Load | MVAR Load | To Bus | MW Losses | Mvar Losses |
|----------|----------|----------|------------|---------|----------|---------|-----------|--------|-----------|-------------|
| 1        | Bus 1 HV | 1.060    | 0.000      | 232.657 | -22.420  | 0.000   | 0.000     | 2      | 4.341     | 13.253      |
|          |          |          |            |         |          |         |           | 5      | 2.706     | 11.171      |
| 2        | Bus 2 HV | 1.045    | 5.006      | 40.000  | 11.224   | 21.700  | 12.700    | 1      | 4.341     | 13.253      |
|          |          |          |            |         |          |         |           | 3      | 2.279     | 9.600       |
|          |          |          |            |         |          |         |           | 4      | 2.236     | 6.786       |
|          |          |          |            |         |          |         |           | 5      | 0.820     | 2.505       |
| 3        | Bus 3 HV | 1.010    | -12.671    | 0.000   | -1.990   | 94.200  | 19.000    | 2      | 2.279     | 9.600       |
|          |          |          |            |         |          |         |           | 4      | 0.655     | 1.672       |
| 4        | Bus 4    | 1.061    | -11.217    | 0.000   | 0.000    | 47.800  | -3.900    | 2      | 2.236     | 6.786       |
|          |          |          |            |         |          |         |           | 3      | 0.655     | 1.672       |
|          |          |          |            |         |          |         |           | 7      | 0.000     | 1.355       |
|          |          |          |            |         |          |         |           | 9      | -0.000    | 1.278       |
| 5        | Bus 5    | 1.032    | 8.772      | 0.000   | 0.000    | 7.600   | 1.600     | 1      | 2.706     | 11.171      |
|          |          |          |            |         |          |         |           | 2      | 0.820     | 2.505       |
|          |          |          |            |         |          |         |           | 6      | 0.000     | 4.823       |

รูปที่ 5.9 แสดงผลจากการคำนวณการสูญเสียในสายส่ง



รูปที่ 5.10 แสดงการการลู่เข้าของคำตอบ

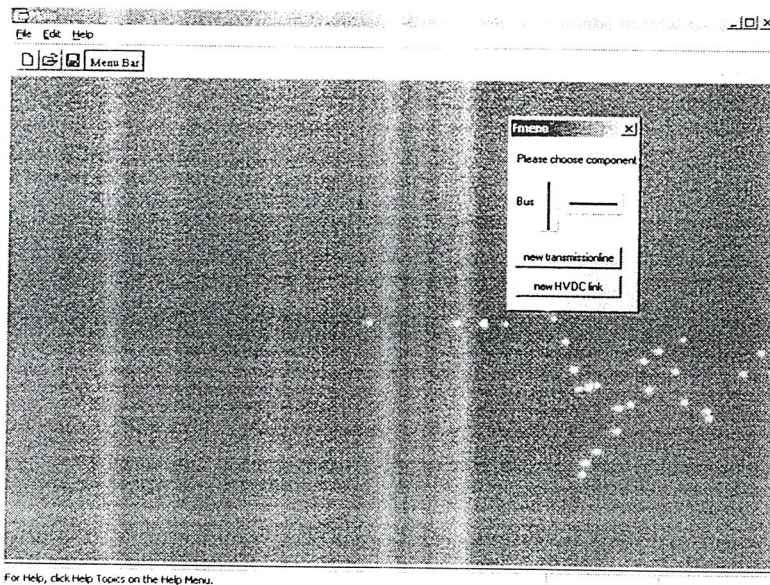
| From Bus | To Bus | type | Volt.DC | MW Flow | MVAR Flow | Direct Current | I(gansf) |
|----------|--------|------|---------|---------|-----------|----------------|----------|
| 5        | 4      | Rec  | 1.2902  | 0.5860  | 0.1880    | 0.4542         | 0.972    |
| 4        | 5      | Inv  | 1.2887  | -0.5853 | 0.1673    | 0.4542         | 0.93E    |

รูป 5.11 รูปแสดงผลจากการคำนวณเอชวีดีซีโหลดโฟล

6. นอกจากนี้ยังมีส่วนในการตรวจเช็คการคำนวณโหลดโฟลในแต่ละรอบได้ด้วยโดยเลือกที่ Check ซึ่งจะปรากฏทูลบาร์ให้ใช้ในการคำนวณ

โปรแกรมส่วนสร้างระบบไฟฟ้าจำลอง

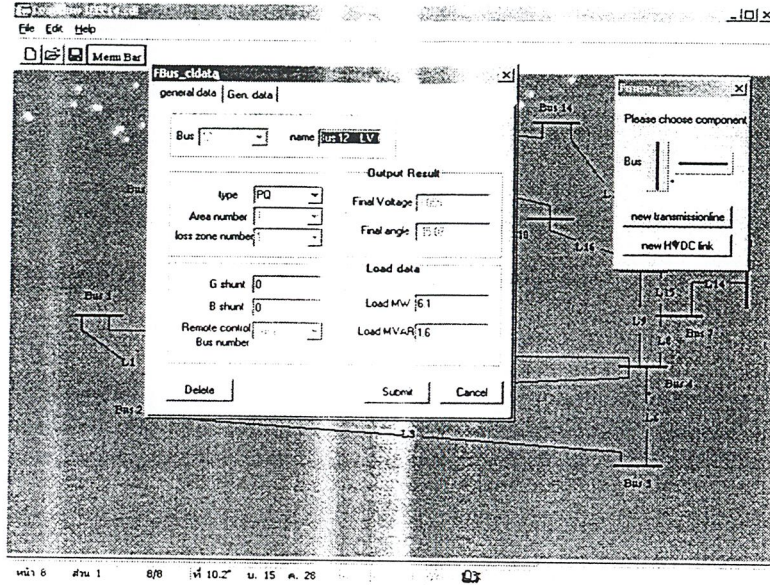
โปรแกรมส่วนสร้างระบบไฟฟ้า เมื่อคลิกที่ปุ่มเมนูบาร์ จะมีไดอะล็อกบ็อกสำหรับใช้สร้างระบบไฟฟ้าดังรูปที่ 5.12



รูปที่ 5.12 แสดงโปรแกรมสร้างระบบไฟฟ้า

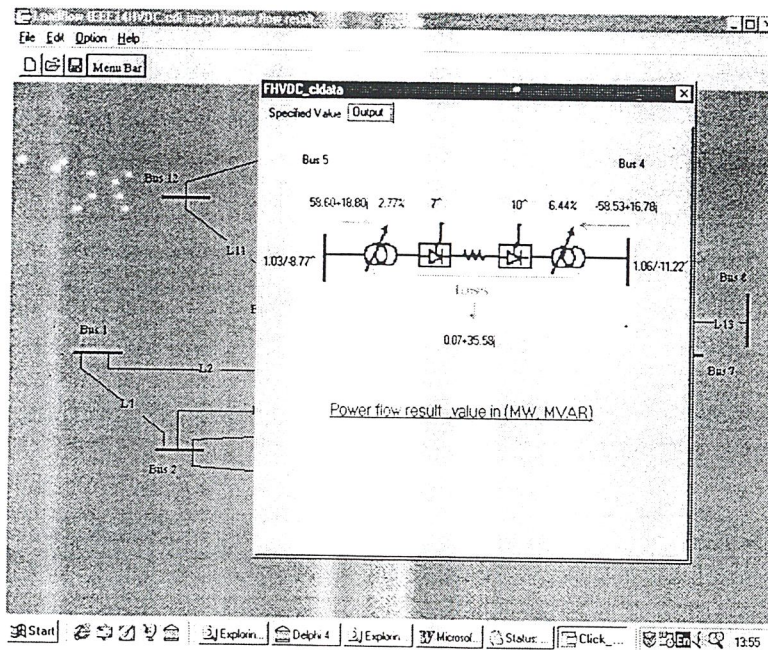
- วาดบัสได้โดยคลิกที่ปุ่ม Bus แล้วคลิกในตำแหน่งที่ต้องการ
- การวาดสายส่งให้คลิกที่ปุ่ม new transmissionline แล้วคลิกขวาที่บัสที่ต้องการ แล้วลากไปยังอีกบัส ถ้าต้องการให้บัสอยู่ในแนวนอนหรือแนวตั้งให้กดปุ่ม H,V ที่คีย์บอร์ด ก่อนปล่อยเมาส์ ตามลำดับ
- การวาดสายส่งกระแสตรงแรงดันสูงให้กดที่ ปุ่ม new HVDC link

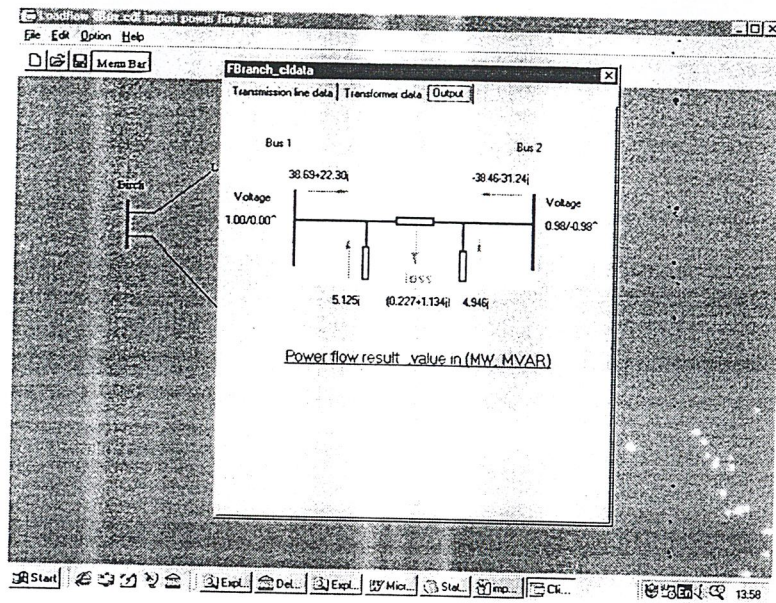
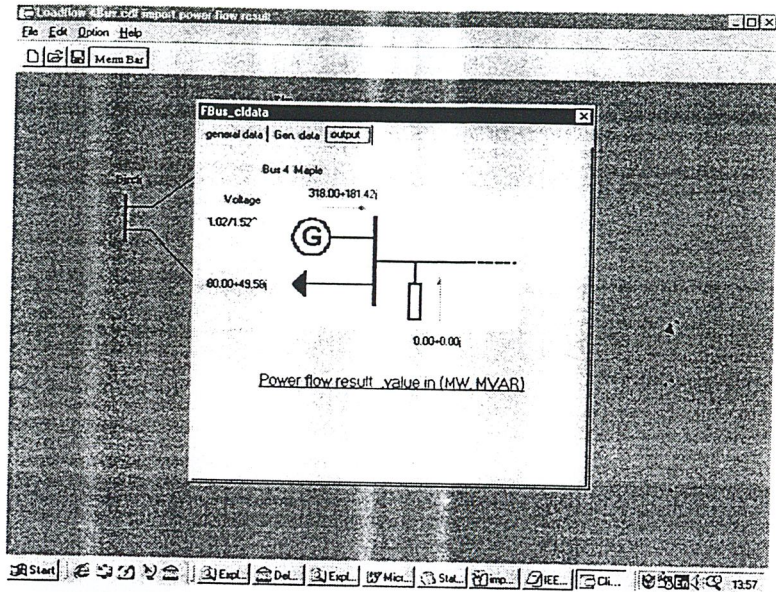
-การใส่พารามิเตอร์ ที่บัสและสายส่งทำได้โดยดับเบิลคลิกที่ BUS และ L แล้วจะปรากฏไดอะล็อกบ็อกดังรูปที่ 5.13



รูปที่ 5.13 แสดงการเพิ่มพารามิเตอร์

- ในโปรแกรมสร้างระบบไฟฟ้าจำลอง สามารถจะแสดงผลของพาวเวอร์โฟลด์ได้โดยการอิมพอร์ตไฟล์ (\*.output) ซึ่งเป็นไฟล์ที่เก็บค่าเอาท์พุทจากโปรแกรม ACDC Load flow ได้

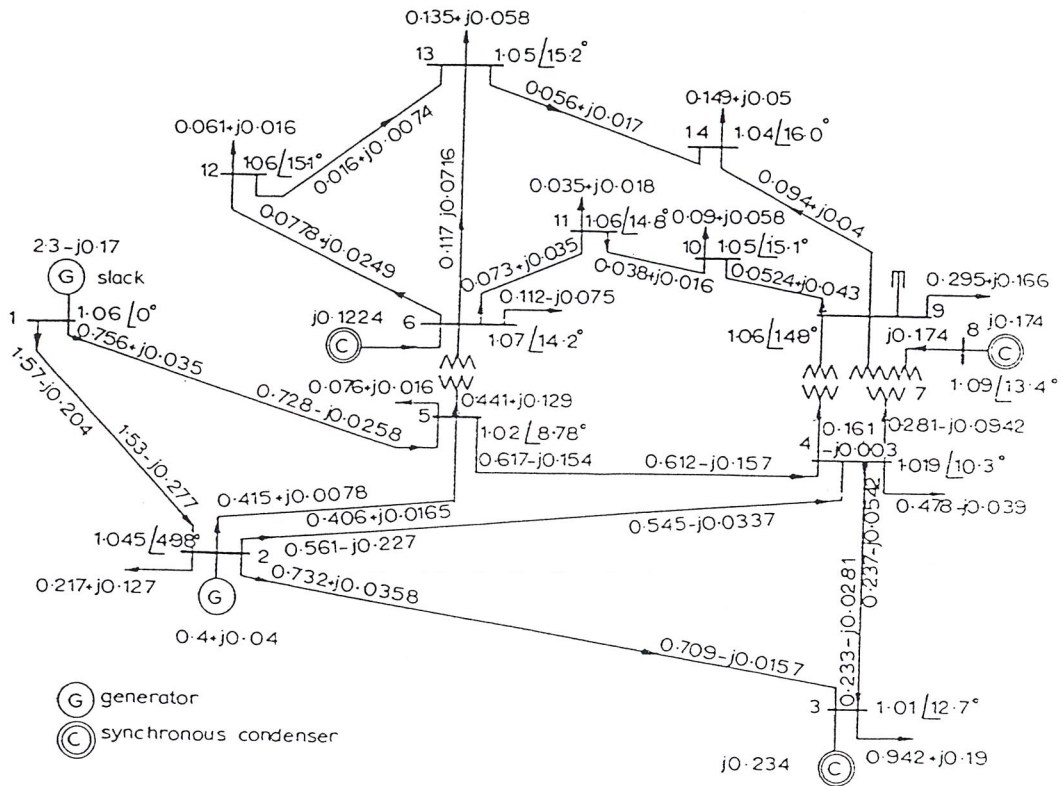




รูปที่ 5.14 แสดงผล Power Flow ผ่านโปรแกรมสร้างระบบไฟฟ้า

5.3 ผลจากการทดสอบโปรแกรม

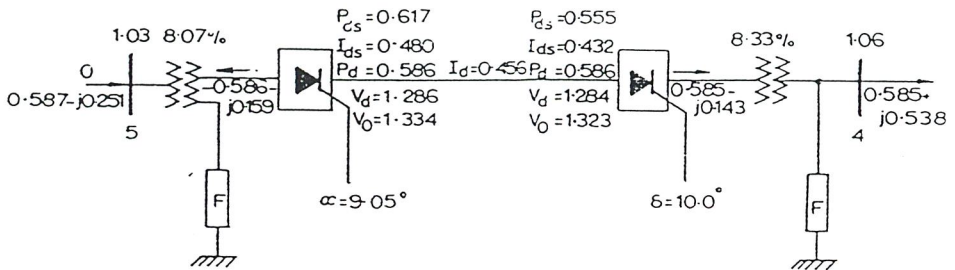
5.3.1. ทำการทดสอบความถูกต้องของโปรแกรมโดยการเปรียบเทียบกับข้อมูลอ้างอิง ข้อมูลอ้างอิงจาก [3] , [4] แสดงดังรูปที่ 5.15 – 5.18



Sample system, including loads, generation and results of the a.c. load flow

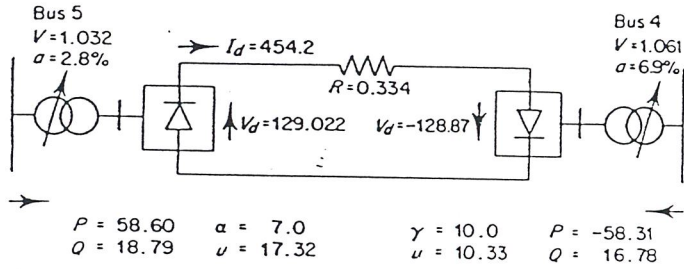
รูปที่ 5.15 ข้อมูลระบบไฟฟ้าเอช 14 บัสจากหนังสืออ้างอิง [4]





D.C. parameters resulting from the a.c.-d.c. load flow:

รูปที่ 5.17 ข้อมูลของส่วน HVDC Link จากหนังสืออ้างอิง [4]



All angles are in degrees. D.C. voltages and current are in kV and Amp respectively. D.C. resistance is in ohms. A.C. powers ( $P, Q$ ) are in MW and MVARs.

รูปที่ 5.18 ข้อมูลของส่วน HVDC Link จากหนังสืออ้างอิง [3]

เปรียบเทียบผลจากการทดสอบโปรแกรมโดยใช้ระบบ IEEE14 ทั้งวิธีนิวตันราฟสัน และวิธีฟาสต์คัปเปิล กับข้อมูลอ้างอิง ได้ผลดังตารางที่ 5.1

| บัส | แรงดัน<br>(อ้างอิง) | แรงดัน<br>(นิวตัน) | %ความ<br>แตกต่าง | แรงดัน<br>(ฟาส) | %ความ<br>แตกต่าง | มุม<br>(อ้างอิง) | มุม<br>(นิวตัน) | %ความ<br>แตกต่าง | มุม<br>(ฟาส) | %ความ<br>แตกต่าง |
|-----|---------------------|--------------------|------------------|-----------------|------------------|------------------|-----------------|------------------|--------------|------------------|
| 1   | 1.06                | 1.06               | 0                | 1.06            | 0                | 0                | 0               | 0                | 0            | 0                |
| 2   | 1.045               | 1.045              | 0                | 1.045           | 0                | -4.98            | -4.981          | -0.02            | -4.981       | -0.02            |
| 3   | 1.01                | 1.01               | 0                | 1.01            | 0                | -12.72           | -12.718         | 0.02             | -12.718      | 0.02             |
| 4   | 1.019               | 1.019              | 0                | 1.019           | 0                | -10.33           | -10.324         | 0.06             | -10.324      | 0.06             |
| 5   | 1.02                | 1.02               | 0                | 1.02            | 0                | -8.78            | -8.783          | -0.03            | -8.782       | -0.02            |
| 6   | 1.07                | 1.07               | 0                | 1.07            | 0                | -14.22           | -14.223         | -0.02            | -14.222      | -0.01            |
| 7   | 1.062               | 1.062              | 0                | 1.062           | 0                | -13.37           | -13.368         | 0.01             | -13.368      | 0.01             |
| 8   | 1.09                | 1.09               | 0                | 1.09            | 0                | -13.36           | -13.368         | -0.06            | -13.368      | -0.06            |
| 9   | 1.056               | 1.056              | 0                | 1.056           | 0                | -14.94           | -14.947         | -0.05            | -14.946      | -0.04            |
| 10  | 1.051               | 1.051              | 0                | 1.051           | 0                | -15.1            | -15.104         | -0.03            | -15.104      | -0.03            |
| 11  | 1.057               | 1.057              | 0                | 1.057           | 0                | -14.79           | -14.795         | -0.03            | -14.795      | -0.03            |
| 12  | 1.055               | 1.055              | 0                | 1.055           | 0                | -15.07           | -15.077         | -0.05            | -15.077      | -0.05            |
| 13  | 1.05                | 1.05               | 0                | 1.05            | 0                | -15.16           | -15.159         | 0.01             | -15.159      | 0.01             |
| 14  | 1.036               | 1.036              | 0                | 1.036           | 0                | -16.04           | -16.039         | 0.01             | -16.039      | 0.01             |

ตารางที่ 5.1 เปรียบเทียบผลจากการโหลดไฟลและข้อมูลอ้างอิง

5.3.2. ทดสอบความถูกต้องของโปรแกรม เมื่อเพิ่มระบบไฟฟ้ากระแสตรงแรงดันสูง  
เข้าไปในระบบไฟฟ้ากระแสสลับ

เมื่อเพิ่มระบบไฟฟ้ากระแสตรงแรงดันสูงเข้าไป นำโปรแกรมทดสอบ แล้วเปรียบเทียบค่าที่ได้ กับข้อมูลที่ใช้อ้างอิง ได้ผลดังตารางที่

| บัส | แรงดัน<br>จากข้อมูล | แรงดัน<br>โปรแกรม | % ความ<br>แตกต่าง | มุมจาก<br>ข้อมูล | มุมจาก<br>โปรแกรม | % ความ<br>แตกต่าง |
|-----|---------------------|-------------------|-------------------|------------------|-------------------|-------------------|
| 1   | 1.06                | 1.06              | 0.00              | 0                | 0                 | 0.00              |
| 2   | 1.045               | 1.045             | 0.00              | -5.02            | -5.008            | 0.24              |
| 3   | 1.01                | 1.01              | 0.00              | -12.7            | -12.671           | 0.23              |
| 4   | 1.06                | 1.061             | -0.09             | -11.2            | -11.217           | -0.15             |
| 5   | 1.03                | 1.032             | -0.19             | -8.74            | -8.772            | -0.37             |
| 6   | 1.07                | 1.07              | 0.00              | -14.2            | -14.237           | -0.26             |
| 7   | ไม่มีข้อมูล         | 1.081             | -                 | ไม่มีข้อมูล      | -14.035           | -                 |
| 8   | ไม่มีข้อมูล         | 1.09              | -                 | ไม่มีข้อมูล      | -14.035           | -                 |
| 9   | 1.07                | 1.074             | -0.37             | -15.5            | -15.533           | -0.21             |
| 10  | 1.07                | 1.066             | 0.37              | -15.6            | -15.595           | 0.03              |
| 11  | 1.06                | 1.064             | -0.38             | -15.1            | -15.063           | 0.25              |
| 12  | 1.06                | 1.057             | 0.28              | -15.1            | -15.119           | -0.13             |
| 13  | 1.05                | 1.053             | -0.29             | -15.3            | -15.258           | 0.27              |
| 14  | 1.05                | 1.047             | 0.29              | -16.4            | -16.397           | 0.02              |

ตารางที่ 5.2 เปรียบเทียบแรงดันและมุมกับข้อมูลอ้างอิงเมื่อเพิ่มระบบไฟฟ้ากระแสตรงแรงดันสูง

5.3.3. เปรียบเทียบผลกระทบจากการเพิ่มระบบไฟฟ้ากระแสตรงแรงดันสูงเข้าไป  
เมื่อเพิ่มระบบไฟฟ้ากระแสตรงแรงดันสูงเข้าไปในระบบไฟฟ้ากระแสสลับจะทำให้  
แรงดันและมุมที่บัสต่าง ๆ เปลี่ยนแปลงไป แสดงผลได้ดังตารางที่ 5.3

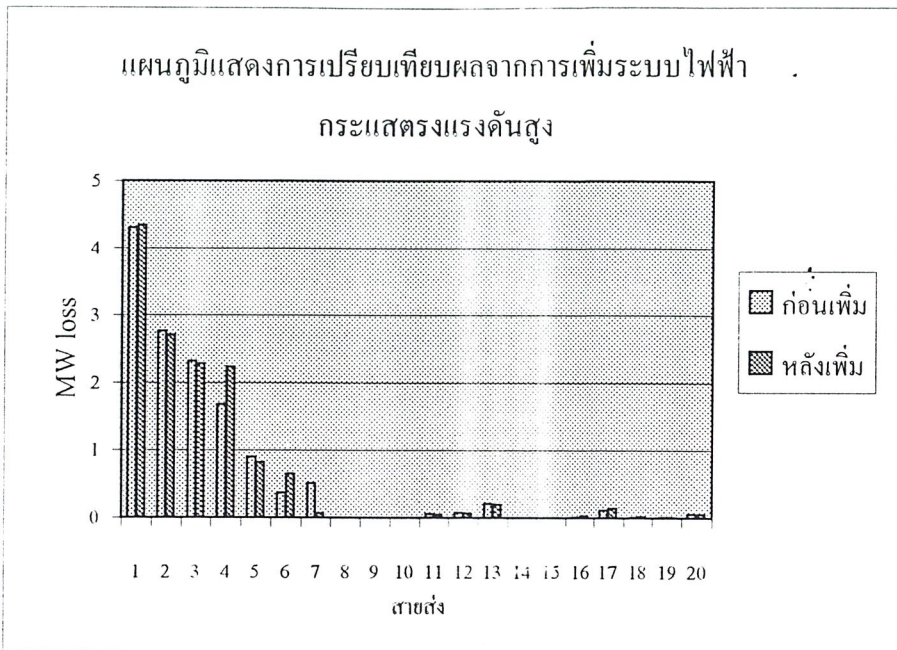
| บัส | แรงดัน<br>(หลังเพิ่ม) | แรงดัน<br>(ก่อนเพิ่ม) | % ความ<br>แตกต่าง | มุม<br>(หลังเพิ่ม) | มุม<br>(ก่อนเพิ่ม) | % ความ<br>แตกต่าง |
|-----|-----------------------|-----------------------|-------------------|--------------------|--------------------|-------------------|
| 1   | 1.06                  | 1.06                  | 0.00              | 0                  | 0                  | 0.00              |
| 2   | 1.045                 | 1.045                 | 0.00              | -5.008             | -4.981             | 0.54              |
| 3   | 1.01                  | 1.01                  | 0.00              | -12.671            | -12.718            | -0.37             |
| 4   | 1.061                 | 1.019                 | 3.96              | -11.217            | -10.324            | 7.96              |
| 5   | 1.032                 | 1.02                  | 1.16              | -8.772             | -8.782             | -0.11             |
| 6   | 1.07                  | 1.07                  | 0.00              | -14.237            | -14.222            | 0.11              |
| 7   | 1.081                 | 1.062                 | 1.76              | -14.035            | -13.368            | 4.75              |
| 8   | 1.09                  | 1.09                  | 0.00              | -14.035            | -13.368            | 4.75              |
| 9   | 1.074                 | 1.056                 | 1.68              | -15.533            | -14.946            | 3.78              |
| 10  | 1.066                 | 1.051                 | 1.41              | -15.595            | -15.104            | 3.15              |
| 11  | 1.064                 | 1.057                 | 0.66              | -15.063            | -14.795            | 1.78              |
| 12  | 1.057                 | 1.055                 | 0.19              | -15.119            | -15.077            | 0.28              |
| 13  | 1.053                 | 1.05                  | 0.28              | -15.258            | -15.159            | 0.65              |
| 14  | 1.047                 | 1.036                 | 1.05              | -16.397            | -16.039            | 2.18              |

ตารางที่ 5.3 แสดงผลกระทบต่อค่าแรงดันและมุม เมื่อเพิ่มระบบไฟฟ้ากระแสตรงแรงดันสูง

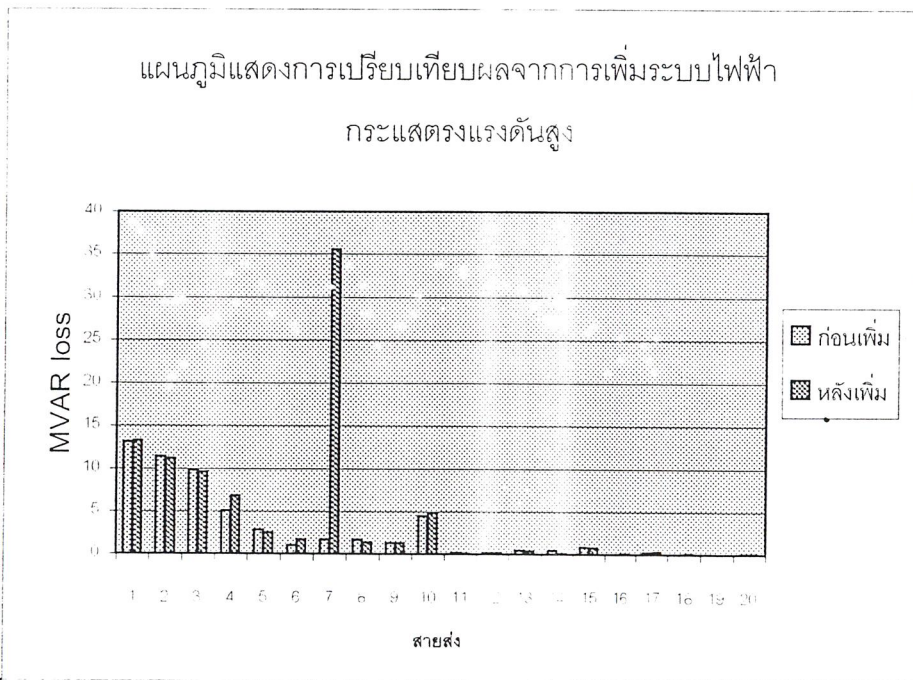
เมื่อเพิ่มระบบไฟฟ้ากระแสตรงแรงดันสูงเข้าไปในระบบทำให้ ความสูญเสียในสายส่ง  
ระหว่างบัสต่าง ๆ เปลี่ยนแปลงไป ได้ผลตามตารางที่ 5.4

| บัสเริ่มต้น | บัสสิ้นสุด | MW loss   |           | MVAR loss |           | MVA Loss  |           |
|-------------|------------|-----------|-----------|-----------|-----------|-----------|-----------|
|             |            | ก่อนเพิ่ม | หลังเพิ่ม | หลังเพิ่ม | ก่อนเพิ่ม | ก่อนเพิ่ม | หลังเพิ่ม |
| 1           | 2          | 4.295     | 4.341     | 13.112    | 13.253    | 13.797    | 13.946    |
| 1           | 5          | 2.764     | 2.706     | 11.409    | 11.171    | 11.739    | 11.494    |
| 2           | 3          | 2.32      | 2.279     | 9.774     | 9.6       | 10.046    | 9.866     |
| 2           | 4          | 1.677     | 2.236     | 5.089     | 6.786     | 5.358     | 7.145     |
| 2           | 5          | 0.902     | 0.82      | 2.755     | 2.505     | 2.899     | 2.636     |
| 3           | 4          | 0.371     | 0.655     | 0.948     | 1.672     | 1.018     | 1.795     |
| 4           | 5          | 0.516     | 0.07      | 1.629     | 35.58     | 1.709     | 35.58     |
| 4           | 7          | 0         | 0         | 1.692     | 1.365     | 1.692     | 1.365     |
| 4           | 9          | 0         | 0         | 1.303     | 1.278     | 1.303     | 1.278     |
| 5           | 6          | 0         | 0         | 4.428     | 4.823     | 4.458     | 4.823     |
| 6           | 11         | 0.055     | 0.052     | 0.114     | 0.109     | 0.127     | 0.121     |
| 6           | 12         | 0.072     | 0.068     | 0.149     | 0.142     | 0.165     | 0.157     |
| 6           | 13         | 0.211     | 0.2       | 0.416     | 0.395     | 0.467     | 0.443     |
| 7           | 8          | 0         | 0         | 0.477     | 0.047     | 0.477     | 0.047     |
| 7           | 9          | 0         | 0         | 0.802     | 0.762     | 0.802     | 0.762     |
| 9           | 10         | 0.013     | 0.026     | 0.035     | 0.07      | 0.037     | 0.075     |
| 9           | 14         | 0.117     | 0.14      | 0.248     | 0.299     | 0.274     | 0.33      |
| 10          | 11         | 0.012     | 0.019     | 0.029     | 0.044     | 0.031     | 0.048     |
| 12          | 13         | 0.006     | 0.005     | 0.006     | 0.004     | 0.008     | 0.007     |
| 13          | 14         | 0.054     | 0.053     | 0.109     | 0.109     | 0.122     | 0.121     |

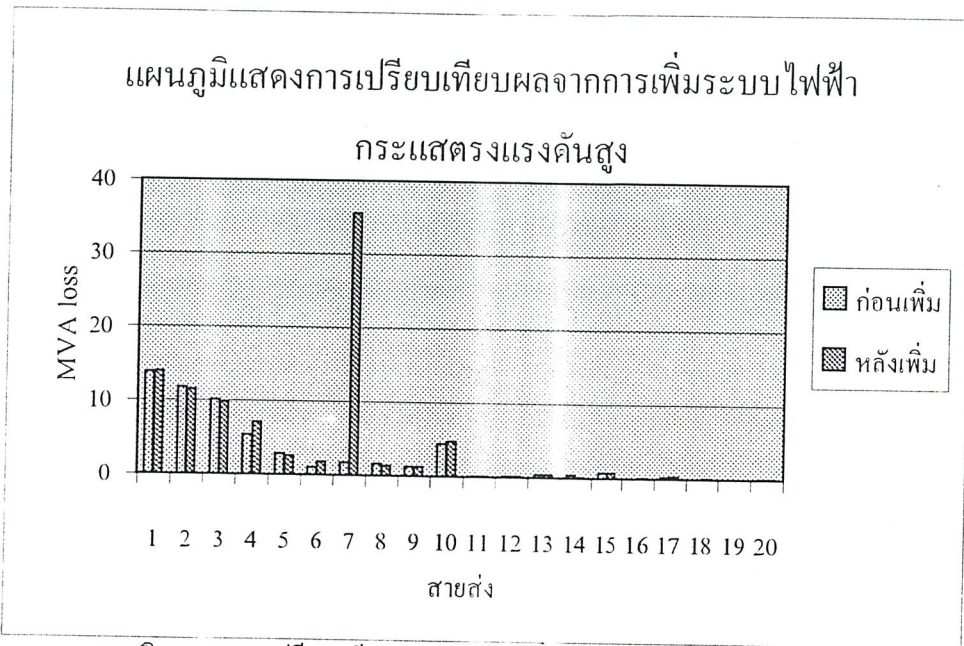
ตารางที่ 5.4 เปรียบเทียบความสูญเสียจากการเพิ่มระบบไฟฟ้ากระแสตรงแรงดันสูง



รูปที่ 5.19 แผนภูมิแสดงการเปรียบเทียบผลจากการเพิ่มระบบไฟฟ้ากระแสตรงแรงดันสูง(MW)



รูปที่ 5.20 แผนภูมิแสดงการเปรียบเทียบผลจากการเพิ่มระบบไฟฟ้ากระแสตรงแรงดันสูง(MVAR)



รูปที่ 5.21 แผนภูมิแสดงการเปรียบเทียบผลจากการเพิ่มระบบไฟฟ้ากระแสตรงแรงดันสูง(MVA)

เปรียบเทียบระบบ AC กับ ACDC (All Value in MW , MVAR)

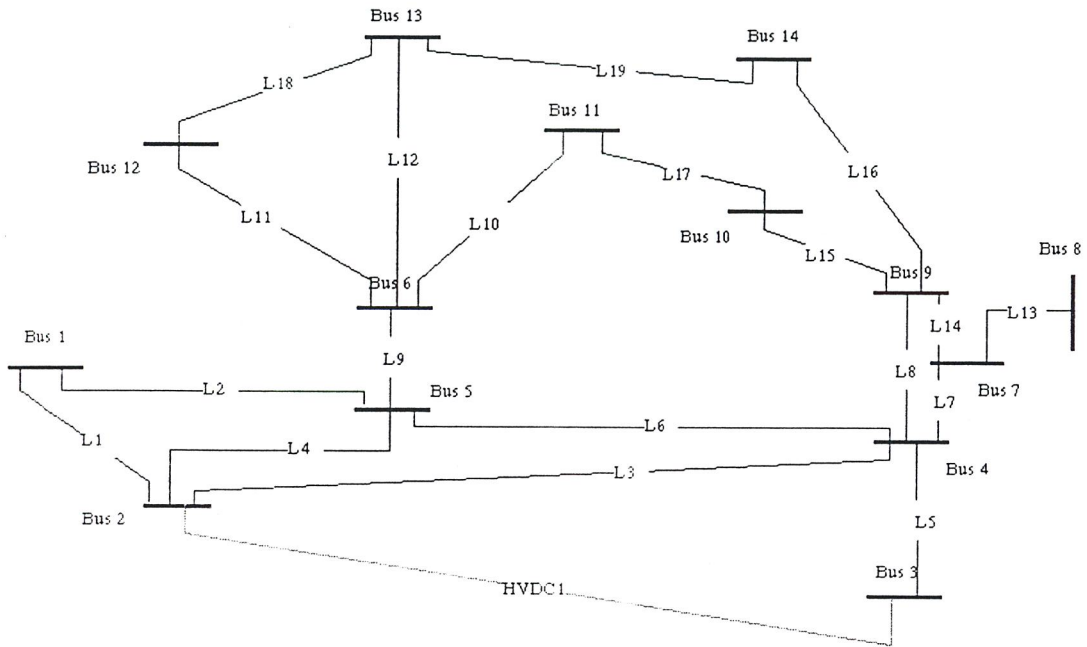
ระบบ AC 14 Bus (จากโปรแกรม)

|   |       |         |    |         |      |
|---|-------|---------|----|---------|------|
| - ผลรวมกำลังจากเจนเนอเรเตอร์            |       | 272.30  | MW | 78.25   | MVAR |
| - ผลรวมของโหลด                          |       | 259     | MW | 73.5    | MVAR |
| - ผลรวมความสูญเสียในสายส่ง              |       | 13.39   | MW | 54.50   | MVAR |
| - ผลรวม MVAR จากไลน์ชาร์จิง             |       |         |    | 28.30   | MVAR |
| - ผลรวม MVAR จากไฮต์โวลเตจคาร์ปาซิเตอร์ |       |         |    | 21.20   | MVAR |
| - เจนเนอเรเตอร์                         | บัส 1 | 232.304 | MW | -16.899 | MVAR |
|   | บัส 2 | 40      | MW | 42.317  | MVAR |
| - ชิงโครไนส์คอนแดนเซอร์                 | บัส 3 |         |    | 23.358  | MVAR |
|   | บัส 6 |         |    | 12.168  | MVAR |
|   | บัส 8 |         |    | 17.316  | MVAR |

ระบบ ACDC14 Bus (จากโปรแกรม)

|   |       |         |    |         |      |
|---|-------|---------|----|---------|------|
| - ผลรวมกำลังจากเจนเนอเรเตอร์            |       | 272.66  | MW | -7.51   | MVAR |
| - ผลรวมของโหลด                          |       | 259     | MW | 73.5    | MVAR |
| - ผลรวมความสูญเสียในสายส่ง              |       | 13.67   | MW | 90.01   | MVAR |
| - ผลรวม MVAR จากไลน์ชาร์จิง             |       |         |    | 27.38   | MVAR |
| - ผลรวม MVAR จากไฮต์โวลเตจคาร์ปาซิเตอร์ |       |         |    | 143.63  | MAVR |
| - เจนเนอเรเตอร์                         | บัส 1 | 232.657 | MW | -22.420 | MVAR |
|   | บัส 2 | 40      | MW | 11.224  | MVAR |
| - ชิงโครไนส์คอนแดนเซอร์                 | บัส 3 |         |    | -1.990  | MVAR |
|   | บัส 6 |         |    | 0.037   | MVAR |
|   | บัส 8 |         |    | 5.636   | MVAR |

ทำการทดลองโดยเปลี่ยนสายส่งกระแสตรงแรงดันสูงไปเชื่อมต่อที่บัสอื่นๆ ตามตัวอย่างดังนี้  
 -ใช้เชื่อมระหว่างบัส 2 - 3



รูปที่ 5.22 ระบบไฟฟ้า 14 บัสที่เชื่อม HVDC ที่บัส 2-3

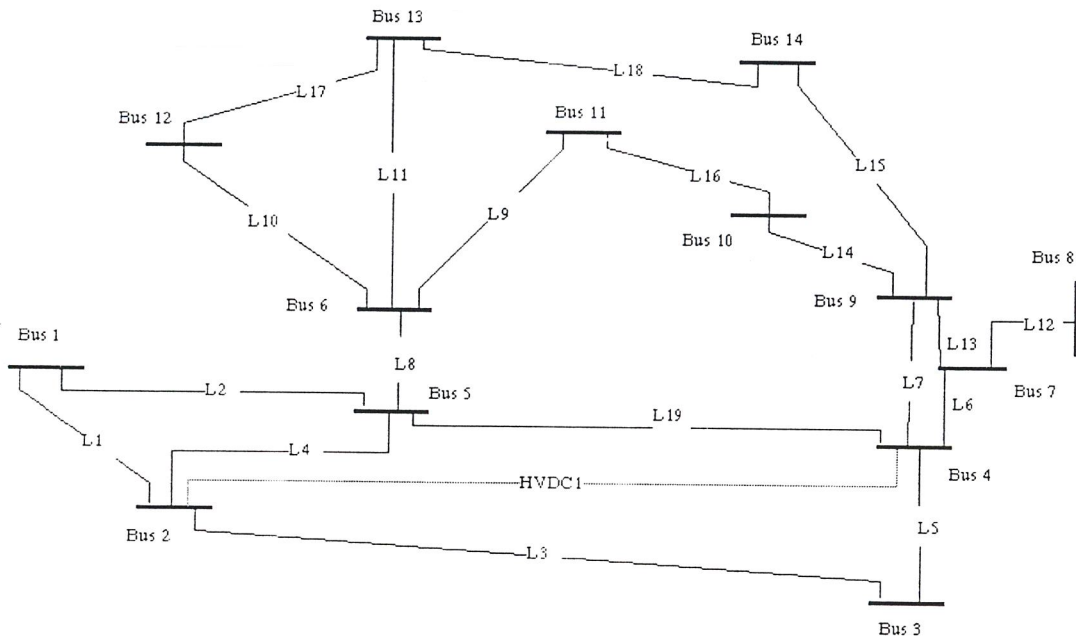
โดยมีการควบคุมระบบส่งจ่ายกระแสตรงแรงดันสูงดังต่อไปนี้

|                                    | Rectifier (Bus 2) | Inverter (Bus 3) |
|------------------------------------|-------------------|------------------|
| Transformer reactances, p.u.       | 0.07275           | 0.126            |
| Commutation reactance $X_c$ , p.u. | 0.126             | 0.07275          |
| Filter admittance $B_f$ , p.u.     | 0.42              | 0.432            |
| <u>Control parameters</u>          |                   |                  |
| DC link power setting ,p.u.        | 0.586             | -                |
| Minimum delay angle $\alpha$ ,deg  | 7                 | -                |
| Minimum extinction $\delta$ ,deg   | -                 | 10               |
| DC link voltage                    | -                 | 1.284            |
| Resistance of DC. Line             | 0.00334           |                  |

จากการคำนวณพาวเวอร์โพลได้ค่าต่างๆ ของระบบดังนี้

|  |       |            |              |
|--|-------|------------|--------------|
| - ผลรวมกำลังจากเจนเนอเรเตอร์             |       | 271.40 MW  | 22.98 MVAR   |
| - ผลรวมของโหลด                           |       | 259 MW     | 73.5 MVAR    |
| - ผลรวมความสูญเสียในสายส่ง               |       | 12.41 MW   | 84.27 MVAR   |
| - ผลรวม MVAR จากไลน์ชาร์จิง              |       |            | 22.66 MVAR   |
| - ผลรวม MVAR จากไฮโดรเจนเตคคาร์ปาซิเตอร์ |       |            | 111.12 MAVR  |
| - เจนเนอเรเตอร์                          | บัส 1 | 231.397 MW | -15.946 MVAR |
|  | บัส 2 | 40 MW      | 8.37 MVAR    |
| - ชิงโครไนส์คอนแดนเซอร์                  | บัส 3 |            | 0.498 MVAR   |
|  | บัส 6 |            | 12.58 MVAR   |
|  | บัส 8 |            | 17.547 MVAR  |

- ใช้เชื่อมระหว่างบัส 2-4



รูปที่ 5.22 ระบบไฟฟ้า 14 บัสที่เชื่อม HVDC ที่บัส 2-4

โดยมีการควบคุมระบบส่งจ่ายกระแสตรงแรงดันสูงดังต่อไปนี้

|                                    | Rectifier (Bus 2) | Inverter (Bus 4) |
|------------------------------------|-------------------|------------------|
| Transformer reactances, p.u.       | 0.07275           | 0.126            |
| Commutation reactance $X_c$ , p.u. | 0.126             | 0.07275          |
| Filter admittance $B_f$ , p.u.     | 0.45              | 0.42             |
| <u>Control parameters</u>          |                   |                  |
| DC. link power setting ,p.u.       | 0.586             | -                |
| Minimum delay angle $\alpha$ ,deg  | 7                 | -                |
| Minimum extinction $\delta$ ,deg   | -                 | 10               |
| DC link voltage                    | -                 | 1.284            |
| Resistance of DC. Line             | 0.00334           |                  |

จากการคำนวณพาวเวอร์โฟลได้ค่าต่างๆ ของระบบดังนี้

|   |           |             |
|---|-----------|-------------|
| - ผลรวมกำลังจากเจนเนอเรเตอร์            | 270.46 MW | 17.23 MVAR  |
| - ผลรวมของโหลด                          | 259 MW    | 73.5 MVAR   |
| - ผลรวมความสูญเสียในสายส่ง              | 11.55 MW  | 83.96 MVAR  |
| - ผลรวม MVAR จากไลน์ชาร์จิจึง           |           | 24.50 MVAR  |
| - ผลรวม MVAR จากไฮต์โวลเตทคาร์ปาซิเตอร์ |           | 115.70 MAVR |
| - เจนเนอเรเตอร์                         | บัส 1     | 230.457 MW  |
|   | บัส 2     | 40 MW       |
| - ซิงโครนัสคอนเดนเซอร์                  | บัส 3     | 13.195 MVAR |
|   | บัส 6     | 4.728 MVAR  |
|   | บัส 8     | 12.549 MVAR |

#### 5.4 สรุปผลการทดลอง

1. ผลจากโปรแกรม การหาโหลดโพลทั้งนิวตันราฟสัน และฟาสต์คัปเปิล และเอชอีดีซี โหลด โพล ให้ผลที่ใกล้เคียง กับข้อมูลที่ใช้อ้างอิงมาก แสดงถึงความถูกต้องของ โปรแกรมอยู่ในขั้นที่น่าพอใจ

2. เมื่อเพิ่มระบบไฟฟ้ากระแสตรงแรงดันสูงเข้าไปในระบบไฟฟ้ากระแสสลับแล้ว มีผลทำให้ค่าแรงดันและมุมที่บัสต่างๆ เปลี่ยนไป ในหลาย ๆ บัส ไม่ใช่เฉพาะบัสที่ติดกับสายส่งกระแสตรงแรงดันสูง รวมทั้งค่าความสูญเสียที่สายส่งเส้นต่างๆ ดังแสดงในรูปที่ 5.19-5.21

## บทที่ 6

### สรุปผลและวิจารณ์

โปรแกรมที่พัฒนาขึ้นนี้ในการทำโปรเจกต์ครั้งนี้ แบ่งได้เป็น 2 ส่วนใหญ่ คือ ส่วนการสร้างแบบจำลองระบบไฟฟ้า และส่วนคำนวณ โดยส่วนการสร้างแบบจำลองระบบไฟฟ้า สามารถให้ผู้ใช้สามารถสร้างระบบไฟฟ้าที่ต้องการศึกษาได้เอง นอกจากที่ระบบที่มีอยู่แล้ว โดยเขียนระบบบัสและป้อนข้อมูลที่ต้องการ สำหรับสายส่งกระแสตรงแรงดันสูงมีในระบบได้หนึ่งเส้น ทำให้สามารถใช้งานได้สะดวก และเกิดความยืดหยุ่นในการใช้งาน

ในส่วนคำนวณสามารถคำนวณได้ทั้ง เอชโพลด์โพลและ เอชดีซีโพลด์โพล ให้ผลการคำนวณที่ใกล้เคียงกับค่าจากแหล่งอ้างอิงมากโดย ระบบไฟฟ้ากระแสสลับ 14 บัส ใช้ข้อมูลจาก IEEE14 ในการเปรียบเทียบ และระบบไฟฟ้ากระแสตรงแรงดันสูง ใช้ ข้อมูลจาก [4] ดังแสดงในตารางที่ 5.1 และ 5.2

สายส่งกระแสตรงแรงดันสูงมีข้อดีหลายประการดังแสดงไว้ในบทที่ 2 แต่ว่าจากผลของโปรแกรมเอชดีซีพาวเวอร์โพลก็แสดงให้เห็นว่า เมื่อระบบไฟฟ้ามีการเพิ่มสายส่งกระแสตรงแรงดันสูงเข้าในระบบ จะทำให้เกิดความสูญเสีย MVAR ปริมาณมากในตัวคอนเวอร์เตอร์ ซึ่งจำเป็นจะต้องมีการชดเชยด้วยวิธีต่างๆ เช่น เพิ่ม ตัวเก็บประจุขนาน หรือ เพิ่มปริมาณ MVAR จาก Synchronous Condensor ซึ่งทำให้เสียค่าใช้จ่ายเพิ่มขึ้นอีกส่วนหนึ่ง สำหรับการตัดสินใจว่าจะใช้สายส่งกระแสตรงแรงดันสูงในพื้นที่ใดนั้น ต้องมีการพิจารณาอย่างรอบคอบถึงความจำเป็น สภาพเศรษฐกิจและความคุ้มค่าในการลงทุน

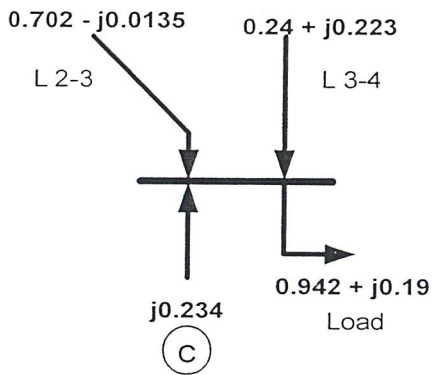
โปรแกรมนี้ใช้การเขียนแบบออบเจกโอเรเนตท(OOP) ซึ่งมีข้อดีคือ นำกลับมาใช้ใหม่ได้(Re-use) สามารถที่นำคอมโพเนนต์ที่เคยสร้างไว้แล้วกลับมาใช้ได้ ซึ่งทำให้ผู้ที่สนใจจะพัฒนาโปรแกรม ให้มีความสามารถและประสิทธิภาพมากยิ่งขึ้น ช่วยลดเวลาในการปรับปรุงและพัฒนาโปรแกรมได้

ปัญหาที่พบ

ข้อมูลที่ใช้อ้างอิงสำหรับระบบไฟฟ้าที่มีการเชื่อมโยงด้วยระบบไฟฟ้ากระแสตรงแรงดันสูงมีน้อยทำให้การศึกษามีความล่าช้า และทำให้เปรียบเทียบผลจากข้อมูลกับผลที่ได้จากการทดสอบโปรแกรมมีไม่กี่กรณี นอกจากนี้ข้อมูลจาก 2 แหล่งที่ใช้ระบบ AEP 14 บัสในการทดสอบเหมือนกัน ( จากหนังสือ[3] และบทความ [4] ) ซึ่งผู้เขียนเป็นคนเดียวกัน ( Prof.Jos Arillaga ) แต่ใช้คนละวิธีการในการคำนวณ แล้วปรากฏว่าผลลัพธ์บางค่าก็ไม่ตรงกัน ดังแสดงในรูปที่ 5.17 และ 5.18 ซึ่งในโครงการนี้ใช้วิธีการคำนวณจากหนังสือ[3] ซึ่งตีพิมพ์ที่หลัง และข้อมูลบางส่วนก็มีความผิดพลาดดังต่อไปนี้

ผล ACDC Power Flow จาก IEE (นำมาแสดงบางส่วน)

• Bus3

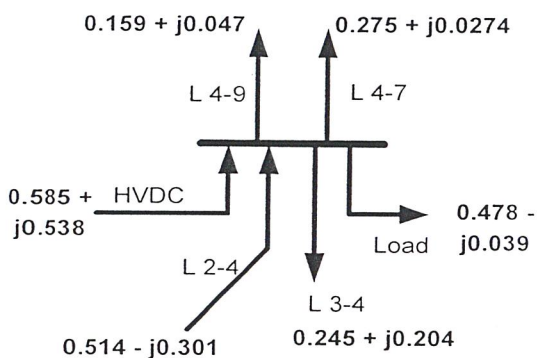


$$\text{Total P} = 0.942 - 0.702 - 0.24 = 0$$

$$\text{Total Q} = 0.234 - 0.19 + 0.0135 + 0.223 = 0.2805$$

(ตามปกติ Total P, Total Q ต้องเท่ากับ 0)

• Bus 4



$$\text{Total P} = 0.585 + 0.514 - 0.245 - 0.478 -$$

$$0.275 - 0.159 = -0.058$$

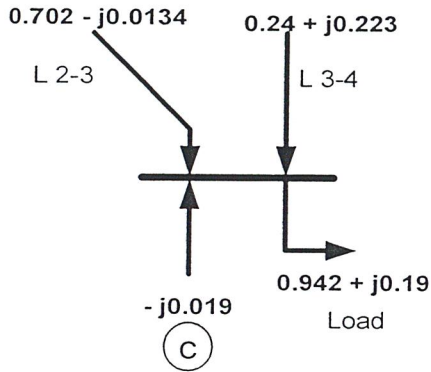
$$\text{Total Q} = 0.538 - 0.301 - 0.204 + 0.039 -$$

$$0.0274 - 0.047 = -0.0024$$

(ตามปกติ Total P, Total Q ต้องเท่ากับ 0)

ឆ្លង ACDC Power Flow ទិន្នន័យ Program

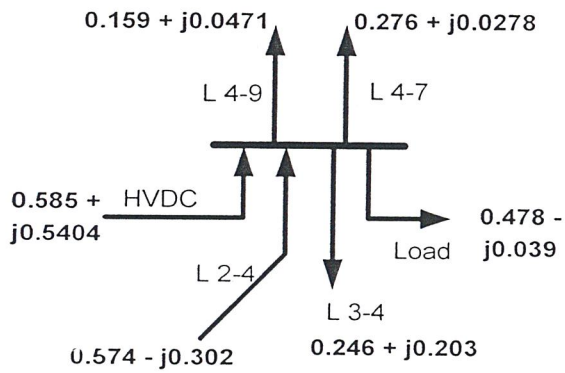
• Bus 3



$$\begin{aligned} \text{Total P} &= 0.942 - 0.24 - 0.702 \\ &= 0 \end{aligned}$$

$$\begin{aligned} \text{Total Q} &= 0.19 + 0.0134 - 0.223 + 0.019 \\ &\approx 0 \end{aligned}$$

• Bus 4



$$\begin{aligned} \text{Total P} &= 0.585 + 0.574 - 0.246 - 0.478 - \\ &\quad 0.276 - 0.159 \\ &= 0 \end{aligned}$$

$$\begin{aligned} \text{Total Q} &= 0.5404 - 0.302 - 0.203 + 0.039 \\ &\quad - 0.0278 - 0.0471 \\ &\approx 0 \end{aligned}$$

ภาคผนวก

## วิธีลดรูปเมตริกซ์แบบแก้เขียน-อีลิมีเนชัน (Gaussian elimination)

ในการแก้ปัญหาด้านระบบไฟฟ้ากำลังจะครอบคลุมพื้นที่กว้างใหญ่ ซึ่งประกอบไปด้วยสถานีย่อยหลายร้อยสถานี, เครื่องกำเนิดไฟฟ้า, ศูนย์กลางโหลด (Load center) ทำให้ต้องแก้สมการที่ประกอบไปด้วยเมตริกซ์ขนาดใหญ่ ในวิธีการเชิงเลขของคอมพิวเตอร์จะพยายามแก้ปัญหาโดยหลีกเลี่ยงการอินเวอร์สเมตริกซ์โดยตรงเพราะจะใช้เวลาและหน่วยความจำมาก โดยใช้วิธีการลดรูปแบบต่างๆ ซึ่งวิธีการที่จะนำเสนอในรายงานนี้ เป็นหนึ่งในวิธีที่มีประสิทธิภาพสูงทั้งด้านเวลาและหน่วยความจำ

แก้เขียน-อีลิมีเนชัน เป็นวิธีการที่ประกอบด้วย 2 ส่วน ส่วนแรกคือการลดรูปเมตริกซ์ (Forward Reduction) ให้อยู่ในรูปอัปเพอร์ทริแองกูลาร์ ส่วนที่สองเป็นการหาค่าคำตอบที่ต้องการ โดยการแทนค่ากลับ (Back Substitute) ซึ่งจะอธิบายโดยใช้สมการ Y-matrix ขนาด 3x3

### 1. การลดรูปเมตริกซ์ (Forward Reduction)

จากสมการ Y-matrix ซึ่งเรารู้ค่า  $Y$  และค่า  $I$  ต้องการหาค่า  $V$

$$\begin{bmatrix} Y_{11} & Y_{12} & Y_{13} \\ Y_{21} & Y_{22} & Y_{23} \\ Y_{31} & Y_{32} & Y_{33} \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ I_3 \end{bmatrix}$$

หารบรรทัดที่ 1 ด้วย  $Y_{11}$  ได้บรรทัดที่ 1 ใหม่เป็น

$$\begin{bmatrix} 1 & Y'_{12} & Y'_{13} \\ Y_{21} & Y_{22} & Y_{23} \\ Y_{31} & Y_{32} & Y_{33} \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix} = \begin{bmatrix} I'_1 \\ I_2 \\ I_3 \end{bmatrix}$$

นำบรรทัดที่ 1 คูณด้วย  $Y_{21}$  แล้วนำไปลบออกจากบรรทัดที่ 2 ได้เป็น

$$\begin{bmatrix} 1 & Y'_{12} & Y'_{13} \\ 0 & Y'_{22} & Y'_{23} \\ Y_{31} & Y_{32} & Y_{33} \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix} = \begin{bmatrix} I'_1 \\ I'_2 \\ I_3 \end{bmatrix}$$

จากนั้นหารบรรทัดที่ 2 ด้วย  $Y'_{22}$  ได้เป็น

$$\begin{bmatrix} 1 & Y'_{12} & Y'_{13} \\ 0 & 1 & Y''_{23} \\ Y_{31} & Y_{32} & Y_{33} \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix} = \begin{bmatrix} I'_1 \\ I''_2 \\ I_3 \end{bmatrix}$$

ทำเช่นเดียวกันในแถวที่ 3 โดยนำบรรทัดที่ 1 คูณด้วย  $Y_{31}$  แล้วนำมาลบออกจากบรรทัดที่ 3

$$\begin{bmatrix} 1 & Y'_{12} & Y'_{13} \\ 0 & 1 & Y''_{23} \\ 0 & Y'_{32} & Y'_{33} \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix} = \begin{bmatrix} I'_1 \\ I''_2 \\ I'_3 \end{bmatrix}$$

จากนั้นนำบรรทัดที่ 2 คูณด้วย  $Y'_{32}$  แล้วนำไปลบออกจากบรรทัดที่ 3

$$\begin{bmatrix} 1 & Y'_{12} & Y'_{13} \\ 0 & 1 & Y''_{23} \\ 0 & 0 & Y'''_{33} \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix} = \begin{bmatrix} I'_1 \\ I''_2 \\ I'''_3 \end{bmatrix}$$

จากนั้นหารบรรทัดที่ 3 ด้วย  $Y'''_{33}$  จะได้เป็น

$$\begin{bmatrix} 1 & Y'_{12} & Y'_{13} \\ 0 & 1 & Y''_{23} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix} = \begin{bmatrix} I'_1 \\ I''_2 \\ I'''_3 \end{bmatrix}$$

เสร็จสิ้นในขั้นตอนการลดรูปเมตริกซ์

## 2. การแทนค่ากลับ (Back Substitute)

ทำการแก้สมการดังกล่าวโดยจะหาค่า  $V$  ได้ดังนี้

$$V_3 = I'''_3$$

$$V_2 = I''_2 - V_3 Y''_{23}$$

$$V_1 = I'_1 - V_2 Y'_{12} - V_3 Y'_{13}$$

```

1: unit Main;
2:
3: interface
4:
5: uses Windows, SysUtils, Classes, Graphics, Forms, Controls, Menus,
6:     StdCtrls, Dialogs, Buttons, Messages, ExtCtrls, ComCtrls, StdActns,
7:     ActnList, ToolWin, ImgList;
8:
9: type
10:  TMainForm = class(TForm)
11:     MainMenu: TMainMenu;
12:     Window1: TMenuItem;
13:     Check: TMenuItem;
14:     WindowCascadeItem: TMenuItem;
15:     WindowTileItem: TMenuItem;
16:     WindowArrangeItem: TMenuItem;
17:     OpenDialog: TOpenDialog;
18:     Edit1: TMenuItem;
19:     CutItem: TMenuItem;
20:     CopyItem: TMenuItem;
21:     PasteItem: TMenuItem;
22:     WindowMinimizeItem: TMenuItem;
23:     StatusBar: TStatusBar;
24:     ActionList1: TActionList;
25:     EditCut1: TEditCut;
26:     EditCopy1: TEditCopy;
27:     EditPaste1: TEditPaste;
28:     FileNew1: TAction;
29:     FileSave1: TAction;
30:     FileExit1: TAction;
31:     FileOpen1: TAction;
32:     FileSaveAs1: TAction;
33:     WindowCascadel: TWindowCascade;
34:     WindowTileHorizontal1: TWindowTileHorizontal;
35:     WindowArrangeAll1: TWindowArrange;
36:     WindowMinimizeAll1: TWindowMinimizeAll;
37:     HelpAbout1: TAction;
38:     FileClose1: TWindowClose;
39:     WindowTileVertical1: TWindowTileVertical;
40:     WindowTileItem2: TMenuItem;
41:     ToolBar2: TToolBar;
42:     ToolButton1: TToolButton;
43:     ToolButton2: TToolButton;
44:     ToolButton3: TToolButton;
45:     ToolButton4: TToolButton;
46:     ToolButton5: TToolButton;
47:     ToolButton6: TToolButton;
48:     ToolButton9: TToolButton;
49:     ToolButton7: TToolButton;
50:     ToolButton8: TToolButton;
51:     ToolButton10: TToolButton;
52:     ToolButton11: TToolButton;
53:     ImageList1: TImageList;
54:     Analysis1: TMenuItem;
55:     LoadFlow1: TMenuItem;
56:     ACDC: TMenuItem;
57:     Option1: TMenuItem;
58:     SaveDialog: TSaveDialog;
59:     ShowBusData: TMenuItem;
60:     ShowBranchData: TMenuItem;
61:     File1: TMenuItem;
62:     New1: TMenuItem;
63:     Open1: TMenuItem;
64:     CloseAll1: TMenuItem;
65:     Save1: TMenuItem;
66:     Saveas1: TMenuItem;
67:     NewtonRaphson1: TMenuItem;
68:     Fastdecouple1: TMenuItem;
69:     Exit1: TMenuItem;
70:     Showoutput1: TMenuItem;
71:     export1: TMenuItem;
72:     ShowConverterData1: TMenuItem;
73:     Help1: TMenuItem;
74:     About1: TMenuItem;
75:     procedure FileNew1Execute(Sender: TObject);

```

```

76:     procedure FileOpen1Execute(Sender: TObject);
77:     procedure FileExit1Execute(Sender: TObject);
78:     procedure ShowBusDataClick(Sender: TObject);
79:     procedure ShowBranchDataClick(Sender: TObject);
80:     procedure FileSaveAsItemClick(Sender: TObject);
81:     procedure CheckClick(Sender: TObject);
82:     procedure NewtonRaphson1Click(Sender: TObject);
83:     procedure Fastdecouple1Click(Sender: TObject);
84:     procedure CloseAll1Click(Sender: TObject);
85:     procedure ACDCClick(Sender: TObject);
86:     procedure Showoutput1Click(Sender: TObject);
87:     procedure export1Click(Sender: TObject);
88:     procedure Button1Click(Sender: TObject);
89:     procedure ShowConverterData1Click(Sender: TObject);
90:     procedure About(Sender: TObject);
91:     procedure ONFormClose(Sender: TObject; var Action: TCloseAction);
92:     procedure About1Click(Sender: TObject);
93:     procedure Print1Click(Sender: TObject);
94:
95:     private
96:     { Private declarations }
97:     public
98:     { Public declarations }
99:     end;
100:
101: var
102:   MainForm : TMainForm;
103: {-----}
104: implementation
105:
106:
107: {$R *.DFM}
108:
109: uses about, Bus, Branch, Output1, Output2, Check, complete_data, data, output
110: , fast_solve_ac, newton_solve_ac, solve_acdc, Output3, GraphMismatch, export_files,
111:   HVLink;
112:
113: var nBus, nBranch, iter: integer;
114:
115: function rts(numreal: real): string;
116: begin
117:   str(numreal:1:3, result);
118: end;
119:
120: procedure TMainForm.FileNew1Execute(Sender: TObject);
121: var nbus1, nbranch1: string;
122:     i, j: integer;
123: begin
124:   nBus1:=inputbox('','Enter bus number','none');
125:   nBranch1:=inputbox('','Enter branch number','none');
126:   nBus:=strtoint(nBus1);
127:   nBranch:=strtoint(nBranch1);
128:
129:   Fbus_data.BusGrid.RowCount:=nBus+1;
130:   Fbranch_data.BranchGrid.RowCount:=nBranch+1;
131:
132:   Fbus_data.FormStyle := fsMDIChild;
133:   Fbranch_data.FormStyle := fsMDIChild;
134:
135:   for i:=0 to 17 do
136:     for j:=1 to nBus do
137:       FBus_data.Busgrid.cells[i, j]:=floattostr(0.0);
138:
139:     for i:=0 to 21 do
140:       for j:=1 to nbranch do
141:         FBranch_data.Branchgrid.cells[i, j]:=floattostr(0.0);
142:
143:         i:=0;
144:         for j:=1 to nbranch do
145:           FBranch_data.Branchgrid.cells[i, j]:=inttostr(j);
146:
147:           i:=0;
148:           for j:=1 to nBus do
149:             FBus_data.Busgrid.cells[i, j]:=inttostr(j);
150:

```

```

151:   i:=1;
152:   for j:=1 to nBus do
153:     FBus_data.Busgrid.cells[i,j]:='Bus';
154:   end;
155:
156: procedure TMainForm.FileOpen1Execute(Sender: TObject);
157: begin
158: Application.CreateForm(TFcheck, Fcheck);
159: Application.CreateForm(TFloadflow, Floadflow);
160: Application.CreateForm(TFlossflow, Flossflow);
161: Application.CreateForm(TFBranch_data, FBranch_data);
162: Application.CreateForm(TFBus_data, FBus_data);
163: Application.CreateForm(TFHVflow, FHVflow);
164: Application.CreateForm(TFgraph, Fgraph);
165: load_flow:=Tload_flow.create;
166: loss_flow:=Tloss_flow.create;
167: datat:=Tdata.create;
168: c_data:=Tcomplete_data.create;
169:   _datat.getfromfile;
170:   datat.writeon(FBus_data,FBranch_data,FHVlink_data);
171: end;
172:
173: procedure TMainForm.FileExit1Execute(Sender: TObject);
174: begin
175: Mainform.close();
176: end;
177:
178: procedure TMainForm.ShowBusDataClick(Sender: TObject);
179: begin
180: if datat<>nil then
181: Fbus_data.formstyle:=fsMDIchild
182: else
183: showmessage('you did not load any cdf file');
184:
185: end;
186:
187: procedure TMainForm.ShowBranchDataClick(Sender: TObject);
188: begin
189: if datat<>nil then
190: Fbranch_data.formstyle:=fsMDIchild
191: else
192: showmessage('you did not load any cdf file');
193:
194: end;
195:
196: procedure TMainForm.FileSaveAsItemClick(Sender: TObject);
197: begin
198: datat:=Tdata.create;
199: datat.savetofile(FBus_data,Fbranch_data,nBus,nBranch);
200: datat.destroy;
201: end;
202:
203: procedure TMainForm.CheckClick(Sender: TObject);
204: begin
205:   try
206:     datat.writeon(FBus_data,FBranch_data,FHVlink_data);
207:     load_flow.getdata(datat);
208:     loss_flow.getdata(datat);
209:     c_data.build_calBus_list(datat);
210:     c_data.y.build(datat);
211:     caption:=caption+' execute';
212:     Fcheck.show;
213:   except
214:   else
215:     MessageDlg('can not open this form.',mtError,[mbOk],0);
216:   end;
217: end;
218:
219: procedure TMainForm.NewtonRaphson1Click(Sender: TObject);
220: var begintime,endtime:Tdatetime;
221:     hour,min,sec,msec,hour1,min1,sec1,msec1:word;
222:     count:real;
223:     i:integer;
224: begin
225:   try

```

```

226: iter:=1;
227:
228: begintime:=now;
229: Decodetime (begintime, hour, Min, Sec, Msec);
230:
231:   datat.writeon(FBus_data, FBranch_data, FHVlink_data);
232:   load_flow.getdata(datat);
233:   loss_flow.getdata(datat);
234:   c_data.build_calBus_list(datat);
235:   c_data.y.build(datat);
236:
237: newton_equa_ac:=Tnewton_equa_ac.create;
238: newton_equa_ac.build_equa(c_data);
239:
240: while (newton_equa_ac.maxmismatch>=0.00010) do
241: begin
242: newton_equa_ac.Newton(c_data, iter);
243: newton_equa_ac.solvenewton(c_data);
244: iter:=iter+1;
245: end;
246:
247: endtime:=now;
248: Decodetime (endtime, hour1, min1, secl, msec1);
249:   c_data.update;
250:   load_flow.getoutput(c_data, loss_flow);
251:   load_flow.writeon(Floadflow.stringgrid1);
252:   loss_flow.writeon(FlossFlow.stringgrid1);
253:   Floadflow.Label3.caption:='iter '+inttostr(iter-1);
254:   count := (secl*1000+msec1)-(sec*1000+msec);
255:   Floadflow.label6.caption:=' '+floattostr(count/1000)+' sec.';
256:   Flossflow.Label3.caption:=inttostr(iter-1);
257:   caption:=caption+' execute';
258:   // garaph
259:   Fgraph.graph1.numsets:=2;
260:   Fgraph.graph1.numpoints:=6;
261:   for i:=1 to 6 do
262:   begin
263:   Fgraph.graph1.thisset:=1;
264:   Fgraph.graph1.thispoint:=i;
265:   Fgraph.graph1.graphData:=newton_equa_ac.Pmaxmismatch[i].mag;
266:   end;
267:   for i:=1 to 6 do
268:   begin
269:   Fgraph.graph1.thisset:=2;
270:   Fgraph.graph1.thispoint:=i;
271:   Fgraph.graph1.graphData:=newton_equa_ac.Qmaxmismatch[i].mag;
272:   end;
273:   ////////////
274:   Floadflow.show;
275:   except
276:   else
277:   MessageDlg('can not run loadflow.', mtError, [mbOk], 0);
278:   end;
279:
280: newton_equa_ac.destroy;
281: end;
282:
283: procedure TMainForm.Fastdecouple1Click(Sender: TObject);
284: var begintime, endtime:Tdatetime;
285:     hour, min, sec, msec, hour1, min1, secl, msec1:word;
286:     count:real;
287:     iterP, iterQ, i:integer;
288:     label endloop;
289:
290: begin
291:   try
292:     begintime:=now;
293:     Decodetime (begintime, hour, Min, Sec, Msec);
294:
295:     datat.writeon(FBus_data, FBranch_data, FhVlink_data);
296:     load_flow.getdata(datat);
297:     loss_flow.getdata(datat);
298:     c_data.build_calBus_list(datat);
299:     c_data.y.build(datat);
300:

```

```

301: fast_equa_ac:=Tfast_equa_ac.create;
302: fast_equa_ac.build_equa(c_data);
303:
304: iterP:=1;
305: iterQ:=1;
306:
307: while (fast_equa_ac.maxmismatch>=0.0001) do
308: begin
309: fast_equa_ac.fast1(c_data,iterP);
310: fast_equa_ac.solvefast1(c_data);
311: iterP:=iterP+1;
312: if (fast_equa_ac.maxmismatch<=0.001) then
313: goto endloop;
314: fast_equa_ac.fast2(c_data,iterQ);
315: fast_equa_ac.solvefast2(c_data);
316: iterQ:=iterQ+1;
317: if (fast_equa_ac.maxmismatch<=0.001) then
318: goto endloop;
319: end;
320: endloop:
321:
322: endtime:=now;
323: Decodetime(endtime,hour1,min1,sec1,msec1);
324:
325: c_data.update;
326: load_flow.getoutput(c_data,loss_flow);
327: load_flow.writeon(Floadflow.stringgrid1);
328: loss_flow.writeon(Flossflow.stringgrid1);
329: Floadflow.Label3.caption:='P_iter '+inttostr(iterP)+' Q_iter '+inttostr(iterQ);
330: count := (sec1*1000+msec1)-(sec*1000+msec);
331: Floadflow.label6.caption:=' '+floattostr(count/1000)+' sec.';
332: Flossflow.label3.caption:='P_iter '+inttostr(iterP)+' Q_iter '+inttostr(iterQ);
333: caption:=caption+' execute';
334: // graph
335: Fgraph.graph1.numsets:=2;
336: Fgraph.graph1.numpoints:=7;
337: for i:=1 to 7 do
338: begin
339: Fgraph.graph1.thisset:=1;
340: Fgraph.graph1.thispoint:=i;
341: Fgraph.graph1.graphData:=fast_equa_ac.Pmaxmismatch[i].mag;
342: end;
343: for i:=1 to 7 do
344: begin
345: Fgraph.graph1.thisset:=2;
346: Fgraph.graph1.thispoint:=i;
347: Fgraph.graph1.graphData:=fast_equa_ac.Qmaxmismatch[i].mag;
348: end;
349: ///////////////
350: Floadflow.showmodal;
351: except
352: else
353: MessageDlg('can not run loadflow.',mtError,[mbOk],0);
354: end;
355: fast_equa_ac.destroy;
356: end;
357:
358: procedure TMainForm.CloseAll1Click(Sender: TObject);
359: begin
360: c_data.free;
361: datat.free;
362: load_flow.free;
363: loss_flow.free;
364: caption:='Loadflow';
365: end;
366:
367: // ACDC load flow jao~
368: procedure TMainForm.ACDCClick(Sender: TObject);
369: var begintime,endtime:Tdatetime;
370: hour,min,sec,msec,hour1,min1,sec1,msec1:word;
371: count:real;
372: iterP,iterQ,i:integer;
373: label endloop;
374:
375: begin

```

```

376:   try
377:   begintime:=now;
378:   Decodetime(begintime, hour, Min, Sec, Msec);
379:
380:   datat.writeon(FBus_data, FBranch_data, Fhmlink_data);
381:   load_flow.getdata(datat);
382:   loss_flow.getdata(datat);
383:   c_data.build_calBus_list(datat);
384:   c_data.y.build(datat);
385:
386:   equa_acdc:=Tequa_acdc.create;
387:   equa_acdc.build_fastequa(c_data);
388:
389:   iterP:=1;
390:   iterQ:=1;
391:
392:   while (equa_acdc.maxmismatch>=0.0001) do
393:   begin
394:   equa_acdc.acdc1(c_data, iterP);
395:   equa_acdc.solveacdc1(c_data);
396:   iterP:=iterP+1;
397:   if (abs(equa_acdc.maxmismatch)<=0.001) then
398:   goto endloop;
399:   equa_acdc.acdc2(c_data, iterQ);
400:   equa_acdc.solveacdc2(c_data);
401:   iterQ:=iterQ+1;
402:   if (abs(equa_acdc.maxmismatch)<=0.001) then
403:   goto endloop;
404:   end;
405:   endloop:
406:
407:   endtime:=now;
408:   Decodetime(endtime, hour1, min1, sec1, msec1);
409:
410:   c_data.update;
411:   load_flow.getoutput(c_data, loss_flow);
412:   FHVflow.getoutput(c_data);
413:   load_flow.writeon(Floadflow.stringgrid1);
414:   loss_flow.writeon(Flossflow.stringgrid1);
415:   Floadflow.Label3.caption:='P_iter '+inttostr(iterP)+' Q_iter '+inttostr(iterQ);
416:   count := (sec1*1000+msec1)-(sec*1000+msec);
417:   Floadflow.label6.caption:=' '+floattostr(count/1000)+' sec.';
418:   Flossflow.label3.caption:='P_iter '+inttostr(iterP)+' Q_iter '+inttostr(iterQ);
419:   caption:=caption+' execute';
420:   // graph
421:   Fgraph.graph1.numsets:=3;
422:   Fgraph.graph1.numpoints:=7;
423:   for i:=1 to 6 do
424:   begin
425:   Fgraph.graph1.thisset:=1;
426:   Fgraph.graph1.thispoint:=i;
427:   Fgraph.graph1.graphData:=equa_acdc.Pmaxmismatch[i].mag;
428:   end;
429:   for i:=1 to 6 do
430:   begin
431:   Fgraph.graph1.thisset:=2;
432:   Fgraph.graph1.thispoint:=i;
433:   Fgraph.graph1.graphData:=equa_acdc.Qmaxmismatch[i].mag;
434:   end;
435:   for i:=1 to 6 do
436:   begin
437:   Fgraph.graph1.thisset:=3;
438:   Fgraph.graph1.thispoint:=i;
439:   Fgraph.graph1.graphData:=equa_acdc.Rmaxmismatch[i].mag;
440:   end;
441:   ///////////////////////////////////
442:   Floadflow.showmodal;
443:   except
444:   else
445:   MessageDlg('can not run loadflow.', mtError, [mbOk], 0);
446:   end;
447:   equa_acdc.destroy;
448:   end;
449:   procedure TMainForm.Showoutput1Click(Sender: TObject);
450:   begin

```

```
451: Floodflow.showmodal;
452: end;
453:
454:
455: procedure TMainForm.export1Click(Sender: TObject);
456: begin
457:   try
458:     c_data.exporttofile();
459:     load_flow.exporttofile(loss_flow);
460:     savetofile(export_file);
461:     showMessage('export file already');
462:   except
463:   else
464:     MessageDlg('can not export file output.',mtError,[mbOk],0);
465:   end;
466: end;
467:
468:
469:
470: procedure TMainForm.Button1Click(Sender: TObject);
471: begin
472:   getfromfile(export_file);
473: end;
474:
475: procedure TMainForm.ShowConverterData1Click(Sender: TObject);
476: begin
477:   if datat<>nil then
478:     FHVlink_data.formstyle:=fsmdichild
479:   else
480:     showMessage('you did not load any cdf file');
481:
482: end;
483:
484: procedure TMainForm.About(Sender: TObject);
485: begin
486:   Aboutbox.visible:=true;
487: end;
488:
489: procedure TMainForm.ONFormClose(Sender: TObject; var Action: TCloseAction);
490: begin
491:   Aboutbox.close;
492: end;
493:
494: procedure TMainForm.About1Click(Sender: TObject);
495: begin
496:   Aboutbox.showmodal;
497: end;
498:
499: procedure TMainForm.Print1Click(Sender: TObject);
500: begin
501:   Print;
502: end;
503:
504: end.
```

```

1: unit data;
2:
3: interface
4: uses SysUtils, classes, bus, branch, Hvlink;
5: const maxbus=118;
6:     maxline=200;
7:     maxconv=20;
8: type
9:   TBusData = class
10:     Bus, Area_no, losszone_no, typed, remote_Bus: integer;
11:     name:string[15];
12:     Final_ag, Final_volt, LoadMW, LoadMVAR, GenMW, GenMVAR, BaseKV, Des_volt, MaxMvar,
13:     MinMvar, Gshunt, Bshunt: real;
14:   constructor create;
15:   destructor Destroy; override;
16: end;
17:   TBranchdata = class
18:     min_volt, max_volt :real;
19:     line, F_bus, T_bus, Area, Losszone, circuit, MVA_1, MVA_2,
20:     MVA_3, controlbus :integer;
21:     typed, resistance, reactance, side, line_charg, Tr_ratio, Tr_ag,
22:     Mintap, Maxtap, stepsize :real;
23:   constructor create;
24:   destructor Destroy; override;
25: end;
26:
27:   Tconverter_data = class
28:     typed, name:string[6];
29:     angle, cosfir, tap, Id, Vd, Pd, Qd,
30:     B, X, Rdc, cosfirsp, Vdsp, Pdsp, Bf, firsp:real;
31:     line, F_Bus, T_Bus:integer;
32:   constructor create;
33:   destructor destroy; override;
34: end;
35:
36:   Tdata = class
37:     Bus:array[1..maxbus] of TBusdata;
38:     Branch:array[1..maxline] of TBranchdata;
39:     conv:array[1..maxconv] of Tconverter_data;
40:     title1:string[49];
41:     title2:string;
42:     title3:string[37];
43:     stop:real;
44:     nBus, nBranch, nConv:integer;
45:   constructor create;
46:   destructor Destroy; override;
47:   procedure getfromfile;
48:   procedure savetofile(a:TFbus_data; B:TFbranch_data; nBus, nBranch:integer);
49:   procedure writeon(a:TFbus_data; B:TFbranch_data; C:TFHVlink_data);
50: end;
51:
52: var datat : Tdata;
53: implementation
54: uses Main;
55:
56: constructor Tconverter_data.create;
57: begin
58: inherited create;
59: typed:=' '; name:=' ';
60: angle:=0; cosfir:=0; tap:=0; Id:=0; Vd:=0;
61: Pd:=0; Qd:=0; B:=0; X:=0; F_Bus:=0;
62: T_Bus:=0;
63: end;
64:
65: destructor Tconverter_data.destroy;
66: begin
67: typed:=''; name:='';
68: angle:=0; cosfir:=0; tap:=0; Id:=0; Vd:=0;
69: Pd:=0; Qd:=0; B:=0; X:=0; F_Bus:=0;
70: T_Bus:=0;
71: inherited destroy;
72: end;
73:
74:
75: constructor TBusdata.create;

```

```

76: begin
77: inherited create;
78: Bus:=0;Area_no:=0;losszone_no:=0;
79: typed:=0;remote_Bus:=0;name:='';
80: Final_ag:=0;Final_volt:=0;LoadMW:=0;
81: LoadMVAR:=0;GenMW:=0;GenMVAR:=0;BaseKV:=0;
82: Des_volt:=0;MaxMvar:=0;
83: MinMvar:=0;Gshunt:=0;Bshunt:=0;
84: end;
85: destructor TBusdata.destroy;
86: begin
87: Bus:=0;Area_no:=0;losszone_no:=0;
88: typed:=0;remote_Bus:=0;name:='';
89: Final_ag:=0;Final_volt:=0;LoadMW:=0;
90: LoadMVAR:=0;GenMW:=0;GenMVAR:=0;BaseKV:=0;
91: Des_volt:=0;MaxMvar:=0;
92: MinMvar:=0;Gshunt:=0;Bshunt:=0;
93: inherited destroy;
94: end;
95:
96: constructor TBranchdata.create;
97: begin
98: line:=0;F_bus:=0;T_bus:=0;Area:=0;
99: Losszone:=0;circuit:=0;
100: line_charg:=0;MVA_1:=0;MVA_2:=0;
101: MVA_3:=0;controlbus:=0;
102: end;
103: destructor TBranchdata.destroy;
104: begin
105: line:=0;F_bus:=0;T_bus:=0;Area:=0;
106: Losszone:=0;circuit:=0;
107: line_charg:=0;MVA_1:=0;MVA_2:=0;
108: MVA_3:=0;controlbus:=0;
109: end;
110:
111: constructor Tdata.create;
112: var i:integer;
113: begin
114: inherited create;
115: for i:=1 to maxbus do
116: Bus[i]:=TBusdata.create;
117: for i:=1 to maxline do
118: Branch[i]:=TBranchdata.create;
119: for i:=1 to maxconv do
120: conv[i]:=Tconverter_data.create;
121: end;
122: destructor Tdata.destroy;
123: var i:integer;
124: begin
125: for i:=1 to maxbus do
126: Bus[i].destroy;
127: for i:=1 to maxline do
128: Branch[i].destroy;
129: inherited Destroy;
130: end;
131:
132:
133: procedure Tdata.getfromfile;
134: var F:textfile;
135:     i,nC:integer;
136:     c1,c2,A1,A3,A4,A5,A18:integer;
137:     A2,c3:string[15];
138:     A6,A7,A8,A9,A10,A11,A12,A13,A14,A15,A16,A17:real;
139:     B1,B2,B3,B4,B5,B6,B10,B11,B12,B13,B14 : Integer;
140:     B7,B8,B9,B15,B16,B17,B18,B19,B20,B21,c4,c5,c6,c7,c8,c9,c10 : Real;
141: begin
142: if MainForm.opendialog.execute then
143: BEGIN
144: MainForm.caption:='Loadflow'+ ' '+ExtractFilename(MainForm.opendialog.FileName);
145: Assignfile(F,MainForm.opendialog.filename);
146: Reset(F);
147: // read bus data
148: Read(F,title1); Readln(F,nBus); Readln(F,title2);
149:     for i:=1 to nBus do
150:     begin

```

```

151:   read(F,A1);   read(F,A2);   read(F,A3);   read(F,A4);
152:   read(F,A5);   read(F,A6);   read(F,A7);   read(F,A8);
153:   read(F,A9);   read(F,A10);  read(F,A11);  read(F,A12);
154:   read(F,A13);  read(F,A14);  read(F,A15);  read(F,A16);
155:   read(F,A17);  readln(F,A18);
156:   Delete(A2,1,1);
157:   with Bus[i] do begin
158:     Bus:=A1;      name:=A2;      Area_no:=A3;  losszone_no:=A4;
159:     typed:=A5;    Final_volt:=A6; Final_ag:=A7; LoadMW:=A8;
160:     LoadMVAR:=A9; GenMW:=A10;   GenMVAR:=A11; BaseKV:=A12;
161:     des_volt:=A13;MaxMVAR:=A14; MinMVAR:=A15; Gshunt:=A16;
162:     Bshunt:=A17;  remote_Bus:=A18; end;
163:   end;
164:
165: // read branch data
166: readln(F,stop); read(F,title3); readln(F,nBranch);
167: for i:=1 to nBranch do
168:   begin
169:     Read(F,B1);  Read(F,B2);   Read(F,B3);  Read(F,B4);
170:     Read(F,B5);  Read(F,B6);   Read(F,B7);  Read(F,B8);
171:     Read(F,B9);  Read(F,B10);  Read(F,B11); Read(F,B12);
172:     Read(F,B13); Read(F,B14);  Read(F,B15); Read(F,B16);
173:     Read(F,B17); Read(F,B18);  Read(F,B19); Read(F,B20);
174:     Readln(F,B21);
175:     with Branch[i] do begin
176:       line:=i;
177:       F_bus:=B1;      T_bus:=B2;      Area:=B3;
178:       losszone:=B4;   circuit:=B5;    typed:=B6;
179:       resistance:=B7; reactance:=B8;   line_charg:=B9/2;
180:       MVA_1:=B10;    MVA_2:=B11;    MVA_3:=B12;
181:       controlBus:=B13; side:=B14;      Tr_ratio:=B15;
182:       Tr_ag:=B16;    Mintap:=B17;    Maxtap :=B18;
183:       stepsize:=B19; Min_volt:=B20;  Max_volt:=B21;
184:     end;
185:   end;
186:
187: // read HVDC data
188: readln(F,stop); read(F,title3); readln(F,nC);
189: nconv:=nC;
190: for i:=1 to nC do
191:   begin
192:     read(F,c1);   read(F,c2);   read(F,c3);
193:     read(F,c4);   read(F,c5);   read(F,c6);
194:     read(F,c7);   read(F,c8);   read(F,c9);
195:     read(F,c10);
196:     delete(c3,1,1);
197:     delete(c3,4,11);
198:     with conv[i] do
199:       begin
200:         F_bus:=c1;   T_bus:=c2;   typed:=c3;   B:=c4;
201:         X:=c5;      Bf:=c6;   Rdc:=c7;    Vdsp:=c8;
202:         Pdsp:=c9;   firsp:=c10;
203:       end;
204:     end;
205:     readln(F,stop);
206:     closefile(F);
207:   END;
208: end;
209:
210:
211:
212:
213: procedure Tdata.writeon(a:TFbus_data;b:TFbranch_data;c:TFHVlink_data);
214: var i:integer;
215: begin
216:   a.Busgrid.rowcount:=nBus+2;
217:   for i:=1 to nBus do
218:     begin
219:       with Bus[i] do begin
220:         with a.Busgrid do begin
221:           cells[0,i]:=inttostr(Bus);
222:           cells[2,i]:=inttostr(Area_no); cells[3,i]:=inttostr(losszone_no);
223:           cells[4,i]:=inttostr(typed);  cells[5,i]:=floattostr(Final_volt);
224:           cells[6,i]:=floattostr(Final_ag); cells[7,i]:=floattostr(LoadMW);
225:           cells[8,i]:=floattostr(LoadMVAR); cells[9,i]:=floattostr(GenMW);

```

```

226:   cells[10,i]:=floattostr(GenMVAR); cells[11,i]:=floattostr(BaseKV);
227:   cells[12,i]:=floattostr(des_volt); cells[13,i]:=floattostr(MaxMVAR);
228:   cells[14,i]:=floattostr(MinMVAR); cells[15,i]:=floattostr(Gshunt);
229:   cells[16,i]:=floattostr(Bshunt); cells[17,i]:=inttostr(remote_Bus);
230: end; end;
231: a.Busgrid.cells[1,i]:=Bus[i].name;
232: end;
233:   b.Branchgrid.rowcount:=nBranch+2;
234:   for i:=1 to nBranch do
235:     begin
236:       with b.Branchgrid do begin
237:         with Branch[i] do begin
238:           cells[0,i]:=inttostr(line); cells[1,i]:=inttostr(F_bus);
239:           cells[2,i]:=inttostr(T_bus); cells[3,i]:=inttostr(Area);
240:           cells[4,i]:=inttostr(losszone); cells[5,i]:=inttostr(circuit);
241:           cells[6,i]:=floattostr(typed); cells[7,i]:=floattostr(resistance);
242:           cells[8,i]:=floattostr(reactance); cells[9,i]:=floattostr(line_charg);
243:           cells[10,i]:=inttostr(MVA_1); cells[11,i]:=inttostr(MVA_2);
244:           cells[12,i]:=inttostr(MVA_3); cells[13,i]:=inttostr(controlBus);
245:           cells[14,i]:=floattostr(side); cells[15,i]:=floattostr(Tr_ratio);
246:           cells[16,i]:=floattostr(Tr_ag); cells[17,i]:=floattostr(Mintap);
247:           cells[18,i]:=floattostr(Maxtap); cells[19,i]:=floattostr(stepsize);
248:           cells[20,i]:=floattostr(Min_volt); cells[21,i]:=floattostr(Max_volt); end;
249:         end;
250:       end;
251:   c.grid.rowcount:=nconv+2;
252:   for i:=1 to nconv do
253:     with c.grid do
254:       with conv[i] do begin
255:         cells[0,i]:=inttostr(i); cells[1,i]:=inttostr(F_bus);
256:         cells[2,i]:=inttostr(T_bus); cells[3,i]:=typed;
257:         cells[4,i]:=floattostr(tap);
258:         cells[5,i]:=floattostr(firsp); cells[6,i]:=floattostr(Bf);
259:         cells[7,i]:=floattostr(X); cells[8,i]:=floattostr(B);
260:         cells[9,i]:=floattostr(Rdc); cells[10,i]:=floattostr(Pdsp);
261:         cells[11,i]:=floattostr(Vdsp);
262:       end;
263:
264:       if nconv>0 then
265:         begin
266:           Mainform.Loadflow1.enabled:=false;
267:           Mainform.ACDC.enabled:=true;
268:         end;
269:
270:       if nconv=0 then
271:         begin
272:           Mainform.ACDC.enabled:=false;
273:           Mainform.Loadflow1.enabled:=true;
274:         end;
275:
276:     end;
277:
278: procedure Tdata.savetofile(A:TFbus_data;B:Tfbranch_data;nBus,nBranch:integer);
279: var F:textfile;
280:     i:integer;
281:     A1,A3,A4,A5,A18:integer;
282:     A2:string[15];
283:     A6,A7,A8,A9,A10,A11,A12,A13,A14,A15,A16,A17:real;
284:     B1,B2,B3,B4,B5,B6,B10,B11,B12,B13,B14 : Integer;
285:     B7,B8,B9,B15,B16,B17,B18,B19,B20,B21 : Real;
286:     title11,title12,title13,title21,title22:string;
287: begin
288: if MainForm.savedialog.execute then
289: BEGIN
290: MainForm.caption:='HarmNet'+ ' '+ExtractFilename(MainForm.opendialog.FileName);
291: title11 := 'ARCHIVE';
292: title12 := 'BUS';
293: title13 := 'BUS DATA FOLLOWS';
294: title21 := 'BRANCH DATA FOLLOWS';
295: title22 := 'ITEMS';
296:
297: AssignFile(F,MainForm.SaveDialog.FileName);
298: Rewrite(F);
299: Write(F,title11);
300: Write(F,nBus,' ');

```

```

301: Writeln(F,title12);
302: Writeln(F,title13);
303:
304:   for i:=1 to nBus do
305:     begin
306:       with a.Busgrid do
307:         begin
308:           A1:=strtoint(cells[0,i]); A2:=cells[1,i]+'          ';
309:           A3:=strtoint(cells[2,i]); A4:=strtoint(cells[3,i]);
310:           A5:=strtoint(cells[4,i]); A6:=strtofloat(cells[5,i]);
311:           A7:=strtofloat(cells[6,i]); A8:=strtofloat(cells[7,i]);
312:           A9:=strtofloat(cells[8,i]); A10:=strtofloat(cells[9,i]);
313:           A11:=strtofloat(cells[10,i]); A12:=strtofloat(cells[11,i]);
314:           A13:=strtofloat(cells[12,i]); A14:=strtofloat(cells[13,i]);
315:           A15:=strtofloat(cells[14,i]); A16:=strtofloat(cells[15,i]);
316:           A17:=strtofloat(cells[16,i]); A18:=strtoint(cells[17,i]);
317:         end;
318:         Write(F,A1,' ');Write(F,A2,' ');
319:         Write(F,A3,' ');Write(F,A4,' ');Write(F,A5,' ');
320:         Write(F,A6:7:3,' ');Write(F,A7:7:3,' ');Write(F,A8:7:3,' ');
321:         Write(F,A9:7:3,' ');Write(F,A10:7:3,' ');Write(F,A11:7:3,' ');
322:         Write(F,A12:7:3,' ');Write(F,A13:7:3,' ');Write(F,A14:7:3,' ');
323:         Write(F,A15:7:3,' ');Write(F,A16:7:3,' ');Write(F,A17:7:3,' ');
324:         Writeln(F,A18,' ');
325:       end;
326:
327:       Writeln(F,-999);
328:       Write(F,title21);
329:       Write(F,nBranch,' ');
330:       Writeln(F,title22);
331:
332:     for i:=1 to nBranch do
333:       begin
334:         with b.Branchgrid do
335:           begin
336:             B1:=strtoint(cells[1,i]);
337:             B2:=strtoint(cells[2,i]); B3:=strtoint(cells[3,i]);
338:             B4:=strtoint(cells[4,i]); B5:=strtoint(cells[5,i]);
339:             B6:=strtoint(cells[6,i]); B7:=strtofloat(cells[7,i]);
340:             B8:=strtofloat(cells[8,i]); B9:=strtofloat(cells[9,i]);
341:             B10:=strtoint(cells[10,i]); B11:=strtoint(cells[11,i]);
342:             B12:=strtoint(cells[12,i]); B13:=strtoint(cells[13,i]);
343:             B14:=strtoint(cells[14,i]); B15:=strtofloat(cells[15,i]);
344:             B16:=strtofloat(cells[16,i]); B17:=strtofloat(cells[17,i]);
345:             B18:=strtofloat(cells[18,i]); B19:=strtofloat(cells[19,i]);
346:             B20:=strtofloat(cells[20,i]); B21:=strtofloat(cells[21,i]);
347:           end;
348:
349:           Write(F,B1,' ');Write(F,B2,' ');Write(F,B3,' ');Write(F,B4,' ');
350:           Write(F,B5,' ');Write(F,B6,' ');Write(F,B7:7:5,' ');
351:           Write(F,B8:7:5,' ');Write(F,B9:7:5,' ');Write(F,B10,' ');
352:           Write(F,B11,' ');Write(F,B12,' ');Write(F,B13,' ');
353:           Write(F,B14,' ');Write(F,B15:7:3,' ');Write(F,B16:7:3,' ');
354:           Write(F,B17:7:3,' ');Write(F,B18:7:3,' ');Write(F,B19:7:3,' ');
355:           Write(F,B20:7:3,' ');Writeln(F,B21:7:3,' ');
356:         end;
357:       writeln(F,-999);
358:     closefile(F);
359:   END;
360: end;
361: end.
362:
363:

```

```

1: unit complete_data;
2:
3: interface
4: uses SysUtils, classes, data, main, math;
5: const
6:     maxconv=20;
7:     dpr=57.29577951;
8: type
9:
10: Tcal_Bus = class
11:     Psp, Qsp, Pgen, Qgen, Pload, Qload,
12:     E, angle, Pcal, Qcal, Ireal, Iimag, total_Gshunt, total_Bshunt: real;
13:     typed: string[2];
14:     name : string[8];
15:     haveconv: boolean;
16:     constructor create;
17:     destructor destroy; override;
18: end;
19:
20: {Tcomplex = class
21:     r : real;
22:     i : real;
23:     constructor create;
24:     destructor destroy; override;
25: end;}
26:
27: Ty_metrix = class
28:     G : array[1..maxbus, 1..maxbus] of real;
29:     B : array[1..maxbus, 1..maxbus] of real;
30:     length: integer;
31:     procedure build(a: Tdata);
32:     procedure buildB1(a: Tdata);
33:     constructor create;
34:     destructor destroy; override;
35: end;
36:
37:
38:
39: Tcomplete_data = class
40:     y : Ty_metrix;
41:     Bus: array[1..maxbus] of Tcal_Bus;
42:     conv: array[1..maxconv] of Tconverter_data;
43:     nBus, nConv: integer;
44:     procedure build_calBus_list(a: Tdata);
45:     procedure update;
46:     constructor create;
47:     destructor destroy; override;
48:     function nCtoB(const a: integer): integer;      {number of Conv to bus}
49:     function BtonC(const a: integer): integer;      {Bus to number of Conv}
50:     function GcBs(const k, m: integer): real;      {Gkm*coskm+Bkm*sinkm}
51:     function Gs_Bc(const k, m: integer): real;     {Gkm*sinkm-Bkm*coskm}
52:     {call after get o/p }
53:     procedure exporttofile;
54: end;
55:
56: var     c_data : Tcomplete_data;
57:
58: {-----}
59: implementation
60: uses export_files;
61:
62:
63: constructor Tcal_Bus.create;
64: begin
65: inherited create;
66: Psp:=0; Qsp:=0; Pgen:=0; Qgen:=0;
67: Pload:=0; Qload:=0; Ireal:=0; Iimag:=0;
68: E:=1;
69: angle:=0;
70: typed:='';
71: name:='';
72: haveconv:=false;
73: end;
74: destructor Tcal_Bus.destroy;
75: begin

```

```

76: Psp:=0;Qsp:=0;Pgen:=0;Qgen:=0;
77: Pload:=0;Qload:=0;E:=0;Ireal:=0;Imag:=0;
78: angle:=0;
79: typed:='';
80: name:='';
81: haveconv:=false;
82: inherited destroy;
83: end;
84:
85:
86: constructor Ty_metrix.create;
87: var i,j:integer;
88: begin
89: inherited create;
90: for i:=1 to maxbus do
91: for j:=1 to maxbus do
92: begin
93: G[i,j]:=0.0;
94: B[i,j]:=0.0;
95: end;
96: end;
97: destructor Ty_metrix.destroy;
98: var i,j:integer;
99: begin
100: for i:=1 to maxbus do
101: for j:=1 to maxbus do
102: begin
103: G[i,j]:=0.0;
104: B[i,j]:=0.0;
105: end;
106: inherited destroy;
107: end;
108:
109: constructor Tcomplete_data.create;
110: var i:integer;
111: begin
112: inherited create;
113:
114: y:=Ty_metrix.create;
115: for i:=1 to maxbus do
116: Bus[i]:=Tcal_Bus.create;
117: for i:=1 to maxconv do
118: conv[i]:=Tconverter_data.create;
119:
120:
121: end;
122:
123:
124: destructor Tcomplete_data.destroy;
125: var i:integer;
126: begin
127: y.destroy;
128: for i:=1 to maxbus do
129: Bus[i].destroy;
130: for i:=1 to maxconv do
131: conv[i].destroy;
132: nBus:=0; nConv:=0;
133: inherited destroy;
134: end;
135:
136: { finish constructor and destructor }
137:
138: procedure Ty_metrix.build(a:Tdata);
139: var i,j,k :integer;
140: gline,bline,z,tr_real,tr_imag,t:real;
141: begin
142: length:=a.nbus;
143: {for k:=1 to bran.nline do
144: if bran.flow[k].T bus > bran.flow[k].F bus then}
145: for k:=1 to a.nbranch do
146: begin
147: {line:=bran.flow[k].line;}
148: with a.branch[k] do begin
149: {with a.branch[line] do begin}
150: z:=resistance*resistance+reactance*reactance;

```

```

151:     if resistance=0 then gline:=0.0 else gline:=resistance/z;
152:     if reactance=0 then bline:=0.0 else bline:=-reactance/z;
153:
154:     if (Tr_ratio=0.0) and (Tr_ag=0.0) then
155:     begin
156:     i:=F_bus;
157:     j:=T_bus;
158:
159:     G[i,i]:=G[i,i]+gline;
160:     G[i,j]:=G[i,j]-gline;
161:     G[j,i]:=G[j,i]-gline;
162:     G[j,j]:=G[j,j]+gline;
163:
164:     B[i,i]:=B[i,i]+bline+line_charg;
165:     B[i,j]:=B[i,j]-bline;
166:     B[j,i]:=B[j,i]-bline;
167:     B[j,j]:=B[j,j]+bline+line_charg;
168:     end
169:     else
170:     begin
171:     t:=1/Tr_ratio;
172:     tr_real:=t*cos(Tr_ag);
173:     tr_imag:=t*sin(Tr_ag);
174:
175:     i:=F_bus;
176:     j:=T_bus;
177:
178:     G[i,i]:=G[i,i]+gline*t*t;
179:     G[i,j]:=G[i,j]-(gline*tr_real + tr_imag*bline);
180:     G[j,i]:=G[j,i]-(gline*tr_real - tr_imag*bline);
181:     G[j,j]:=G[j,j]+gline;
182:
183:     B[i,i]:=B[i,i]+bline*t*t+line_charg;
184:     B[i,j]:=B[i,j]-(tr_real*bline - tr_imag*gline);
185:     B[j,i]:=B[j,i]-(tr_real*bline + tr_imag*gline);
186:     B[j,j]:=B[j,j]+bline+line_charg;
187:     end;
188:
189:     end;
190: end;
191:
192:     for i := 1 to a.nBus do
193:     begin
194:     G[i,i] := G[i,i] + a.Bus[i].Gshunt;
195:     B[i,i] := B[i,i] + a.Bus[i].Bshunt;
196:     end;
197:
198:     for k:=1 to a.nconv do
199:     begin
200:     i:=a.conv[k].F_bus;
201:     B[i,i]:=B[i,i]+a.conv[k].Bf;
202:     end;
203: end;
204:
205:     { caution : do it after receive converter data }
206: procedure Tcomplete_data.build_calBus_list(a:Tdata);
207: var i,j,k: integer;
208: begin
209: nBus:=a.nBus;
210: for i:=1 to a.nBus do
211: begin
212: case a.Bus[i].typed of
213: 3 : begin
214: Bus[i].typed:='SL';
215: Bus[i].name:=a.Bus[i].name;
216: Bus[i].E:=a.Bus[i].Des_volt;
217: Bus[i].angle:=0;
218: Bus[i].Pload:=a.Bus[i].loadMW;
219: Bus[i].Qload:=a.Bus[i].loadMVAR;
220: Bus[i].Pgen:=a.Bus[i].GenMW;
221: Bus[i].Qgen:=a.Bus[i].GenMVAR;
222: k:=BtonC(i);
223: if k<>0 then Bus[i].haveconv:=true;
224: end;
225: 2 : begin

```

```

226:         Bus[i].typed:='PV';
227:         Bus[i].name:=a.Bus[i].name;
228:         Bus[i].E:=a.Bus[i].Des_volt;
229:         Bus[i].Pload:=a.Bus[i].loadMW;
230:         Bus[i].Qload:=a.Bus[i].loadMVAR;
231:         Bus[i].Pgen:=a.Bus[i].GenMW;
232:         Bus[i].Qgen:=a.Bus[i].GenMVAR;
233:         Bus[i].Psp:=(Bus[i].Pgen-Bus[i].Pload)/100;
234:         k:=BtonC(i);
235:         if k<>0 then Bus[i].haveconv:=true;
236:             {for j:=1 to nconv do
237:                 if i=conv[j].Bus then Bus[i].haveconv:=true;}
238:             end;
239:     0 : begin
240:         Bus[i].typed:='PQ';
241:         Bus[i].name:=a.Bus[i].name;
242:         Bus[i].Pload:=a.Bus[i].loadMW;
243:         Bus[i].Qload:=a.Bus[i].loadMVAR;
244:         Bus[i].Pgen:=a.Bus[i].GenMW;
245:         Bus[i].Qgen:=a.Bus[i].GenMVAR;
246:         Bus[i].Psp:=(Bus[i].Pgen-Bus[i].Pload)/100;
247:         Bus[i].Qsp:=(Bus[i].Qgen-Bus[i].Qload)/100;
248:         k:=BtonC(i);
249:         if k<>0 then Bus[i].haveconv:=true;
250:             { for j:=1 to nconv do
251:                 if i=conv[j].Bus then Bus[i].haveconv:=true;}
252:             end;
253:         end;
254:     end;
255:
256: // receive converter data and build initial
257: nconv:=a.nconv;
258: for i:=1 to a.nconv do
259: begin
260: conv[i]:=a.conv[i];
261: conv[i].tap:=1;
262: conv[i].Id:=1;
263: conv[i].Qd:=1;
264: j:=conv[i].F_bus;
265: Bus[j].haveconv:=true;
266:
267: if conv[i].typed='Rec' then
268: begin
269: conv[i].cosfir:=1;
270: conv[i].Vd:=1;
271: conv[i].Pd:=1;
272: conv[i].angle:=5/dpr;
273: conv[i].cosfirsp:=cos(a.conv[i].firsp/dpr);
274: end
275: else
276: begin
277: conv[i].cosfir:=-1;
278: conv[i].Vd:=-1;
279: conv[i].Pd:=-1;
280: conv[i].angle:=175/dpr;
281: conv[i].cosfirsp:=-cos(a.conv[i].firsp/dpr);
282: conv[i].Vdsp:=-conv[i].Vdsp;
283: end;
284: end;
285:
286: // <<<< test input converter data >>>> ja~
287: { Bus[4].haveconv:=true;
288:   Bus[5].haveconv:=true;
289:   nConv:=2;
290:   Conv[1].typed:='Rec';
291:   Conv[1].F Bus:=5; conv[1].T Bus:=4;
292:   with conv[1] do begin
293:     angle:=5/dpr;
294:     cosfir:=1;
295:     tap:=1;
296:     Id:=1;
297:     Vd:=1;
298:     Pd:=1;
299:     Qd:=1;
300:     B:=0.07275;

```

```

301: X:=0.126;
302: Rdc:=0.00334;
303: cosfirsp:=cos(7/dpr);
304: Pdsp:=0.586;
305: end;
306: Conv[2].typed:='Inv';
307: Conv[2].F_Bus:=4; conv[2].T_Bus:=5;
308: with conv[2] do begin
309: angle:=175/dpr;
310: cosfir:=-1;
311: tap:=1;
312: Id:=1;
313: Vd:=-1;
314: Pd:=-1;
315: Qd:=1;
316: B:=0.126;
317: X:=0.07275;
318: Rdc:=0.00334;
319: cosfirsp:=-cos(10/dpr);
320: Vdsp:=-1.2887
321: end;
322: {-----find total Gshunt total Bshunt-----}
323:
324: for k:=1 to a.nbranch do
325: begin
326: with a.branch[k] do begin
327: i:=F_bus;
328: j:=T_bus;
329: Bus[i].total_Bshunt:=Bus[i].total_Bshunt+line_charg;
330: Bus[j].total_Bshunt:=Bus[j].total_Bshunt+line_charg;
331: end;
332: end;
333:
334: for k:=1 to a.nBus do
335: begin
336: Bus[k].total_Gshunt:=Bus[k].total_Gshunt+a.Bus[k].Gshunt;
337: Bus[k].total_Bshunt:=Bus[k].total_Bshunt+a.Bus[k].Bshunt;
338: end;
339:
340: for k:=1 to a.nconv do
341: begin
342: i:=a.conv[k].F_bus;
343: Bus[i].total_Bshunt:=Bus[i].total_Bshunt+a.conv[k].Bf;
344: end;
345: end;
346:
347: {-----}
348:
349: function Tcomplete_data.BtonC(const a:integer):integer; {Bus to number of Conv}
350: var i:integer;
351: begin
352: i:=0;
353: repeat
354: i:=i+1;
355: until (Conv[i].F_Bus=a) or (i=nconv+1);
356: result:=i;
357: if i=nconv+1 then result:=0;
358: end;
359:
360: function Tcomplete_data.nCtoB(const a:integer):integer; {number of Conv to
361: var i:integer;
362: begin
363: i:=0;
364: repeat
365: i:=i+1;
366: until (Conv[a].F_Bus=i) or (i=nBus+1);
367: result:=i;
368: if i=nBus+1 then result:=0;
369: end;
370:
371: function Tcomplete_data.GcBs(const k,m:integer):real; {Gkm*coskm+Bkm*sinkm}
372: var anglekm:real;
373: begin
374: anglekm:=Bus[k].angle-Bus[m].angle;
375: result:=Y.G[k,m]*cos(anglekm)+Y.B[k,m]*sin(anglekm);

```

```

376: end;
377:
378: function Tcomplete_data.Gs_Bc(const k,m:integer):real; {Gkm*sinkm-Bkm*coskm}
379: var anglekm:real;
380: begin
381: anglekm:=Bus[k].angle-Bus[m].angle;
382: result:=Y.G[k,m]*sin(anglekm)-Y.B[k,m]*cos(anglekm);
383: end;
384:
385: procedure Ty_metrix.buildBl(a:Tdata);
386: var i,j,k :integer;
387:
388: begin
389:     length:=a.nbus;
390:     for k:=1 to a.nbranch do
391:     begin
392:     with a.branch[k] do begin
393:     i:=F_bus;
394:     j:=T_bus;
395:     B[i,i]:=B[i,i]+(-1/reactance);
396:     B[i,j]:=B[i,j]-(-1/reactance);
397:     B[j,i]:=B[j,i]-(-1/reactance);
398:     B[j,j]:=B[j,j]+(-1/reactance);
399:     end;
400:     end;
401: end;
402:
403: procedure Tcomplete_data.update;
404: var i,k:integer;
405:
406: begin
407:     {find Pgen Qgen}
408:
409: for i:=1 to nBus do
410:
411:     if Bus[i].typed='SL' then
412:     begin
413:         if Bus[i].haveconv then
414:         begin
415:             k:=BtonC(i);
416:             Bus[i].Pgen:=100*(Bus[i].Pcal+conv[k].Pd)+Bus[i].Pload;
417:             Bus[i].Qgen:=100*(Bus[i].Qcal+conv[k].Qd)+Bus[i].Qload;
418:         end
419:         else
420:         begin
421:             Bus[i].Pgen:=100*(Bus[i].Pcal)+Bus[i].Pload;
422:             Bus[i].Qgen:=100*(Bus[i].Qcal)+Bus[i].Qload;
423:         end;
424:     end
425:
426:     else if bus[i].typed='PV' then
427:     begin
428:         if Bus[i].haveconv then
429:         begin
430:             k:=BtonC(i);
431:             Bus[i].Qgen:=100*(Bus[i].Qcal+conv[k].Qd)+Bus[i].Qload;
432:         end
433:         else
434:         begin
435:             Bus[i].Qgen:=100*(Bus[i].Qcal)+Bus[i].Qload;
436:         end;
437:     end;
438: end;
439:
440: procedure Tcomplete_data.exporttofile;
441: var i:integer;
442: begin
443: export_file.nBus:=nBus;
444: export_file.nBranch:=datat.nBranch;
445: export_file.nconv:=nconv;
446:     for i:=1 to nBus do
447:     begin
448:     export_file.Bus[i].E:=Bus[i].E;
449:     export_file.Bus[i].angle:=Bus[i].angle*dpr;
450:     export_file.Bus[i].GenMW:=(Bus[i].Pgen/100);

```

```
451:     export_file.Bus[i].GenMVAR:=Bus[i].Qgen/100;
452:     export_file.Bus[i].LoadMW:=Bus[i].Pload/100;
453:     export_file.Bus[i].LoadMVAR:=Bus[i].Qload/100;
454:     end;
455: for i:=1 to nconv do
456: begin
457: export_file.conv[i].F_bus:=conv[i].F_bus;
458: export_file.conv[i].T_bus:=conv[i].T_bus;
459: export_file.conv[i].flowMW:=conv[i].Pd;
460: export_file.conv[i].flowMVAR:=conv[i].Qd;
461: export_file.conv[i].tap:=conv[i].tap;
462: export_file.conv[i].fir:=arccos(conv[i].cosfir);
463: end;
464:
465: end;
466:
467: end.
```

```

1: unit newton_solve_ac;
2:
3: interface
4: uses SysUtils, classes, complete_data, data;
5: const maxiter=100;
6:
7: type
8:     jacobian=array[1..maxbus*2+maxconv*5, 1..maxbus*2+maxconv*5] of real;
9:
10:    Tcommon_dX=class
11:        mag : real;
12:        typed:string[8];
13:        Bus:integer;
14:        constructor create;
15:        destructor destroy; override;
16:        end;
17:
18:    TdB_1 = class
19:        dP: array[1..maxBus] of Tcommon_dX;
20:        dQ: array[1..maxbus] of Tcommon_dX;
21:        nP, nQ: integer;
22:        procedure build_dB_1(a:Tcomplete_data);
23:        constructor create;
24:        destructor destroy; override;
25:    end;
26:
27:    Tnewton_equa_ac = class
28:        dB : array[1..maxbus*2] of Tcommon_dX;
29:        dX : array[1..maxbus*2] of Tcommon_dX;
30:        jac : jacobian;
31:        nP, nQ, ndB: integer;
32:        Pmaxmismatch: array[1..maxiter] of Tcommon_dX;
33:        Qmaxmismatch: array[1..maxiter] of Tcommon_dX;
34:        maxmismatch: real;
35:        constructor create;
36:        destructor destroy; override;
37:        procedure build_equa(a:Tcomplete_data);
38:        procedure solvenewton(a:Tcomplete_data);
39:        procedure Newton(a:Tcomplete_data; iter: integer);
40:    end;
41:
42:
43:
44: var newton_equa_ac:Tnewton_equa_ac;
45:
46: implementation
47:
48: constructor Tcommon_dX.create;
49: begin
50: inherited create;
51: mag:=0.0; typed:=''; Bus:=0;
52: end;
53: destructor Tcommon_dX.destroy;
54: begin
55: mag:=0.0; typed:=''; Bus:=0;
56: inherited destroy;
57: end;
58:
59: constructor TdB_1.create;
60: var i: integer;
61: begin
62: inherited create;
63:     for i:=1 to maxbus do
64:         begin
65:             dP[i]:=Tcommon_dX.create;
66:             dQ[i]:=Tcommon_dX.create;
67:         end;
68:     nP:=0;    nQ:=0;
69: end;
70:
71: destructor TdB_1.destroy;
72: var i: integer;
73: begin
74:     for i:=1 to maxbus do
75:         begin

```

```

76:     dP[i].destroy;
77:     dQ[i].destroy;
78:     end;
79: nP:=0; nQ:=0;
80: inherited destroy;
81: end;
82:
83: constructor Tnewton_equa_ac.create;
84: var i:integer;
85: begin
86:     inherited create;
87:     for i:=1 to maxbus*2 do
88:         begin
89:             dB[i]:=Tcommon_dX.create;
90:             dX[i]:=Tcommon_dX.create;
91:
92:
93:         end;
94:     for i:=1 to maxiter do
95:         begin
96:             Pmaxmismatch[i]:=Tcommon_dX.create;
97:             Qmaxmismatch[i]:=Tcommon_dX.create;
98:         end;
99:         maxmismatch:=1;
100:        nP:=0; nQ:=0; ndB:=0;
101:
102:    end;
103:
104: destructor Tnewton_equa_ac.destroy;
105: var i:integer;
106: begin
107:     for i:=1 to maxbus*2 do
108:         begin
109:             dB[i].destroy;
110:             dX[i].destroy;
111:
112:
113:         end;
114:     for i:=1 to maxiter do
115:         begin
116:             Pmaxmismatch[i].destroy;
117:             Qmaxmismatch[i].destroy;
118:         end;
119:     inherited destroy;
120: end;
121:
122:
123: { end of constructor and destructor }
124:
125:
126:
127: procedure TdB_1.build_dB_1(a:Tcomplete_data);
128: var i,j,m : integer;
129: begin
130:     i:=1; j:=1;
131:     for m:=1 to a.nBus do
132:         begin
133:             if (a.Bus[m].typed='PQ') and not(a.Bus[m].haveconv) then
134:                 begin
135:                     dP[i].typed:='dP';
136:                     dP[i].Bus:=m;
137:                     dQ[j].typed:='dQ';
138:                     dQ[j].Bus:=m;
139:                     i:=i+1; j:=j+1;
140:                 end
141:                 else if (a.Bus[m].typed='PV') and not(a.Bus[m].haveconv) then
142:                     begin
143:                         dP[i].typed:='dP';
144:                         dP[i].Bus:=m;
145:                         i:=i+1;
146:                     end;
147:                 end;
148:         nP:=i-1; nQ:=j-1;
149:     end;
150:

```

```

151: procedure Tnewton_equa_ac.build_equa(a:Tcomplete_data);
152: var i,j:integer;
153: d:TdB_1;
154: begin
155:   d:=TdB_1.create;
156:   d.build_dB_1(a);
157:   j:=1;
158:   for i:=1 to d.nP do           { dP }
159:     begin
160:       dB[j].typed:='dP';
161:       dB[j].Bus:=d.dP[i].Bus;
162:       dX[j].typed:='dag';
163:       dX[j].Bus:=d.dP[i].Bus;
164:       j:=j+1;
165:     end;
166:
167:
168:     for i:=1 to d.nQ do         { dQ }
169:       begin
170:         dB[j].typed:='dQ';
171:         dB[j].Bus:=d.dQ[i].Bus;
172:         dX[j].typed:='dE/E';
173:         dX[j].Bus:=d.dQ[i].Bus;
174:         j:=j+1;
175:       end;
176:
177:   nP:=d.nP;
178:   nQ:=d.nQ;
179:   ndB:=j-1;
180:   d.destroy;
181: end;
182:
183: procedure Tnewton_equa_ac.Newton(a:Tcomplete_data;iter:integer);   {get .mag }
184: var i,j ,k,m:integer;
185:   ak,bk,angle:real;
186: begin
187: {-----dB-----}
188: for k:=1 to a.nBus do
189: begin
190:   a.Bus[k].Pcal:=0;
191:   a.Bus[k].Qcal:=0;
192:   a.Bus[k].Ireal:=0;
193:   a.Bus[k].Iimag:=0;
194:   for m:=1 to a.nBus do
195:     begin
196:       angle:=a.Bus[m].angle;
197:       ak:=a.Bus[m].E*cos(angle)*a.Y.G[k,m]-a.Bus[m].E*sin(angle)*a.Y.B[k,m];
198:       bk:=a.Bus[m].E*cos(angle)*a.Y.B[k,m]+a.Bus[m].E*sin(angle)*a.Y.G[k,m];
199:       a.Bus[k].Ireal:=a.Bus[k].Ireal+ak;
200:       a.Bus[k].Iimag:=a.Bus[k].Iimag+bk;
201:       {a.Bus[k].Pcal:=a.Bus[k].Pcal + a.Bus[k].E*a.Bus[m].E*a.GcBs(k,m);
202:       a.Bus[k].Qcal:=a.Bus[k].Qcal + a.Bus[k].E*a.Bus[m].E*a.Gs Bc(k,m);}
203:     end;
204:     angle:=a.Bus[k].angle;
205:     a.Bus[k].Pcal:=a.Bus[k].E*cos(angle)*a.Bus[k].Ireal+
206:     a.Bus[k].E*sin(angle)*a.Bus[k].Iimag;
207:     a.Bus[k].Qcal:=a.Bus[k].E*sin(angle)*a.Bus[k].Ireal-
208:     a.Bus[k].E*cos(angle)*a.Bus[k].Iimag;
209:   end;
210: {----dP----}
211: Pmaxmismatch[iter].mag:=0;
212: Qmaxmismatch[iter].mag:=0;
213: for i:=1 to nP do
214: begin
215: k:=dB[i].Bus;
216: dB[i].mag:=a.Bus[k].Psp-a.Bus[k].Pcal;
217: if abs(dB[i].mag) > abs(Pmaxmismatch[iter].mag) then
218: begin
219: Pmaxmismatch[iter].mag:=dB[i].mag;
220: Pmaxmismatch[iter].Bus:=k;
221: end;
222: end;
223:
224:
225: {-----dQ-----}

```

```

226:     for i:= nP+1 to nP+nQ do
227:     begin
228:     k:=dB[i].Bus;
229:     dB[i].mag:=a.Bus[k].Qsp-a.Bus[k].Qcal;
230:     if abs(dB[i].mag) > abs(Qmaxmismatch[iter].mag) then
231:     begin
232:     Qmaxmismatch[iter].mag:=dB[i].mag;
233:     Qmaxmismatch[iter].Bus:=k;
234:     end;
235:     end;
236:
237:     if abs(Pmaxmismatch[iter].mag)>abs(Qmaxmismatch[iter].mag) then
238:     maxmismatch:=abs(Pmaxmismatch[iter].mag) else
239:     maxmismatch:=abs(Qmaxmismatch[iter].mag);
240:
241:
242: {-----jacobian-----}
243: {-----H-----}
244: for i:=1 to nP do
245:     for j:=1 to nP do
246:     begin
247:     k:=dB[i].Bus;
248:     m:=dX[j].Bus;
249:     if k<>m then
250:     jac[i,j]:=a.Bus[k].E*a.Bus[m].E*a.Gs_Bc(k,m)
251:     else if k=m then
252:     jac[i,j]:=-a.Bus[k].Qcal-a.Y.B[k,k]*a.Bus[k].E*a.Bus[k].E;
253:     end;
254: {-----N-----}
255: for i:=1 to nP do
256:     for j:=nP+1 to nP+nQ do
257:     begin
258:     k:=dB[i].Bus;
259:     m:=dX[j].Bus;
260:     if k<>m then
261:     jac[i,j]:=a.Bus[k].E*a.Bus[m].E*a.GcBs(k,m)
262:     else if k=m then
263:     jac[i,j]:=a.Bus[k].Pcal+a.Y.G[k,k]*a.Bus[k].E*a.Bus[k].E;
264:     end;
265: {-----J-----}
266: for i:=nP+1 to nP+nQ do
267:     for j:=1 to nP do
268:     begin
269:     k:=dB[i].Bus;
270:     m:=dX[j].Bus;
271:     if k<>m then
272:     jac[i,j]:=-a.Bus[k].E*a.Bus[m].E*a.GcBs(k,m)
273:     else if k=m then
274:     jac[i,j]:=a.Bus[k].Pcal-a.Y.G[k,k]*a.Bus[k].E*a.Bus[k].E;
275:     end;
276: {-----L-----}
277: for i:=nP+1 to nP+nQ do
278:     for j:=nP+1 to nP+nQ do
279:     begin
280:     k:=dB[i].Bus;
281:     m:=dX[j].Bus;
282:     if k<>m then
283:     jac[i,j]:=a.Bus[k].E*a.Bus[m].E*a.Gs_Bc(k,m)
284:     else if k=m then
285:     jac[i,j]:=a.Bus[k].Qcal-a.Y.B[k,k]*a.Bus[k].E*a.Bus[k].E;
286:     end;
287:
288: end;
289:
290:
291:
292: procedure Tnewton_equa_ac.solvenewton(a:Tcomplete_data);
293: var i,j, k:integer;
294:     value:real;
295: begin
296:
297: {triangulated}
298:     for i:=1 to ndB do
299:         for j:=1 to ndB do
300:             if i=j then

```

```

301:         begin
302:         value:=jac[i,i];
303:         dB[i].mag:=dB[i].mag/value;
304:         for k:=1 to ndB do
305:         jac[i,k]:=jac[i,k]/value;
306:         end
307:         else if j<i then
308:         begin
309:             value:=jac[i,j];
310:             dB[i].mag:=dB[i].mag-dB[j].mag*value;
311:             for k:=1 to ndB do
312:             jac[i,k]:=jac[i,k]-jac[j,k]*value;
313:         end;
314:         {back substituted}
315:         for i:= ndB downto 1 do
316:         begin
317:         dX[i].mag:=dB[i].mag;
318:         if i< ndB then
319:         for j:=i+1 to ndB do
320:         dX[i].mag:=dX[i].mag-dX[j].mag*jac[i,j];
321:         end;
322:         {update data}
323:         for i:=1 to nP do
324:         begin
325:         j:=dX[i].Bus;
326:         a.Bus[j].angle:=a.Bus[j].angle+dX[i].mag;
327:         end;
328:         for i:=nP+1 to nP+nQ do
329:         begin
330:         j:=dX[i].Bus;
331:         a.Bus[j].E:=a.Bus[j].E*(1-dX[i].mag);
332:         end;
333:     end;
334:
335:
336: end.

```

```

1: unit fast_solve_ac;
2:
3: interface
4: uses SysUtils, classes, complete_data, data, newton_solve_ac;
5: type
6:         Tfast_equa_ac = class
7:             dB1, dB2 : array[1..maxbus*2] of Tcommon_dX;
8:             dX1, dX2 : array[1..maxbus*2] of Tcommon_dX;
9:             B1, B2, tri_B1, tri_B2, to_dB1, to_dB2 : jacobian;
10:            nP, nQ: integer;
11:            Pmaxmismatch: array[1..maxiter] of Tcommon_dX;
12:            Qmaxmismatch: array[1..maxiter] of Tcommon_dX;
13:            nowmax: integer;
14:            maxmismatch: real;
15:            constructor create;
16:            destructor destroy; override;
17:            procedure build_equa(a: Tcomplete_data);
18:            procedure fast1(a: Tcomplete_data; iter: integer);
19:            procedure fast2(a: Tcomplete_data; iter: integer);
20:            procedure solvefast1(a: Tcomplete_data);
21:            procedure solvefast2(a: Tcomplete_data);
22:        end;
23:
24: var fast_equa_ac : Tfast_equa_ac;
25: implementation
26:
27: constructor Tfast_equa_ac.create;
28: var i: integer;
29: begin
30:     inherited create;
31:     maxmismatch:=1;
32:     for i:=1 to maxbus*2 do
33:     begin
34:         dB1[i]:=Tcommon_dX.create;
35:         dX1[i]:=Tcommon_dX.create;
36:         dB2[i]:=Tcommon_dX.create;
37:         dX2[i]:=Tcommon_dX.create;
38:     end;
39:     for i:=1 to maxiter do
40:     begin
41:         Pmaxmismatch[i]:=Tcommon_dX.create;
42:         Qmaxmismatch[i]:=Tcommon_dX.create;
43:     end;
44: end;
45:
46: destructor Tfast_equa_ac.destroy;
47: var i: integer;
48: begin
49:     for i:=1 to maxbus*2 do
50:     begin
51:         dB2[i].destroy;
52:         dX2[i].destroy;
53:         dB1[i].destroy;
54:         dX1[i].destroy;
55:     end;
56:     for i:=1 to maxiter do
57:     begin
58:         Pmaxmismatch[i].destroy;
59:         Qmaxmismatch[i].destroy;
60:     end;
61: inherited destroy;
62: end;
63:
64: {-----end of constructor destructor-----}
65:
66: procedure Tfast_equa_ac.build_equa(a: Tcomplete_data);
67: var i, j, k, m: integer;
68:     d: TdB_1;
69:     value: real;
70:     B_: Ty_metrix;
71: begin
72:     d:=TdB_1.create;
73:     d.build_dB_1(a);
74:     j:=1;
75:     for i:=1 to d.nP do         { dP }

```

```

76:         begin
77:         dB1[j].typed:='dP/E';
78:         dB1[j].Bus:=d.dP[i].Bus;
79:         dX1[j].typed:='dag';
80:         dX1[j].Bus:=d.dP[i].Bus;
81:         j:=j+1;
82:         end;
83:     nP:=d.nP;
84:     j:=1;
85:     for i:=1 to d.nQ do           { dQ }
86:     begin
87:     dB2[j].typed:='dQ/E';
88:     dB2[j].Bus:=d.dQ[i].Bus;
89:     dX2[j].typed:='dE';
90:     dX2[j].Bus:=d.dQ[i].Bus;
91:     j:=j+1;
92:     end;
93:     nQ:=d.nQ;
94:     d.destroy;
95:
96: {-----jac-----}
97: {-----B1-----}
98: B_:=Ty_metrix.create;
99: B_.buildB1(datat);
100: for i:=1 to nP do
101: for j:=1 to nP do
102: begin
103: k:=dB1[i].Bus;
104: m:=dX1[j].Bus;
105: B1[i,j]:=-B_.B[k,m];
106: end;
107:
108: {triangulated B1}
109: tri_B1:=B1;
110: for i:=1 to nP do
111:     for j:=1 to nP do
112:         if i=j then
113:             begin
114:                 value:=tri_B1[i,i];
115:                 to_dB1[i,i]:=tri_B1[i,i];
116:                 for k:=1 to nP do
117:                     tri_B1[i,k]:=tri_B1[i,k]/value;
118:                 end
119:             else if j<i then
120:                 begin
121:                     value:=tri_B1[i,j];
122:                     to_dB1[i,j]:=tri_B1[i,j];
123:                     for k:=1 to nP do
124:                         tri_B1[i,k]:=tri_B1[i,k]-tri_B1[j,k]*value;
125:                     end;
126: B_.destroy;
127: {-----B2-----}
128: for i:=1 to nQ do
129: for j:=1 to nQ do
130: begin
131: k:=dB2[i].Bus;
132: m:=dX2[j].Bus;
133: B2[i,j]:=-a.Y.B[k,m];
134: end;
135: {triangulated B2}
136: tri_B2:=B2;
137: for i:=1 to nQ do
138:     for j:=1 to nQ do
139:         if i=j then
140:             begin
141:                 value:=tri_B2[i,i];
142:                 to_dB2[i,i]:=tri_B2[i,i];
143:                 for k:=1 to nQ do
144:                     tri_B2[i,k]:=tri_B2[i,k]/value;
145:                 end
146:             else if j<i then
147:                 begin
148:                     value:=tri_B2[i,j];
149:                     to_dB2[i,j]:=tri_B2[i,j];
150:                     for k:=1 to nQ do

```

```

151:         tri_B2[i,k]:=tri_B2[i,k]-tri_B2[j,k]*value;
152:     end;
153:
154: end;
155:
156: procedure Tfast_equa_ac.fast1(a:Tcomplete_data;iter:integer);
157: var i ,k,m:integer;
158:     ak,bk,angle:real;
159: begin
160:
161: for k:=1 to a.nBus do
162: begin
163:     a.Bus[k].Pcal:=0;
164:     a.Bus[k].Qcal:=0;
165:     a.Bus[k].Ireal:=0;
166:     a.Bus[k].Iimag:=0;
167:     for m:=1 to a.nBus do
168:     begin
169:         angle:=a.Bus[m].angle;
170:         ak:=a.Bus[m].E*cos(angle)*a.Y.G[k,m]-a.Bus[m].E*sin(angle)*a.Y.B[k,m];
171:         bk:=a.Bus[m].E*cos(angle)*a.Y.B[k,m]+a.Bus[m].E*sin(angle)*a.Y.G[k,m];
172:         a.Bus[k].Ireal:=a.Bus[k].Ireal+ak;
173:         a.Bus[k].Iimag:=a.Bus[k].Iimag+bk;
174:     end;
175:     angle:=a.Bus[k].angle;
176:     a.Bus[k].Pcal:=a.Bus[k].E*cos(angle)*a.Bus[k].Ireal+
177:     a.Bus[k].E*sin(angle)*a.Bus[k].Iimag;
178:     a.Bus[k].Qcal:=a.Bus[k].E*sin(angle)*a.Bus[k].Ireal-
179:     a.Bus[k].E*cos(angle)*a.Bus[k].Iimag;
180: end;
181: {-----dB-----}
182: {----dP----}
183: Pmaxmismatch[iter].mag:=0;
184: for i:=1 to nP do
185: begin
186: k:=dB1[i].Bus;
187: dB1[i].mag:=(a.Bus[k].Psp-a.Bus[k].Pcal)/a.Bus[k].E;
188:
189:     if abs(dB1[i].mag*a.Bus[k].E) > abs(Pmaxmismatch[iter].mag) then
190:     begin
191:         Pmaxmismatch[iter].mag:=dB1[i].mag*a.Bus[k].E;
192:         Pmaxmismatch[iter].Bus:=k;
193:     end;
194:
195:     if abs(Pmaxmismatch[iter].mag)<maxmismatch then
196:     if nowmax=1 then
197:         maxmismatch:=abs(Pmaxmismatch[iter].mag);
198:
199: end;
200: end;
201:
202:
203: procedure Tfast_equa_ac.fast2(a:Tcomplete_data;iter:integer);
204: {get .mag ,dB1,}
205: var i ,k,m:integer;
206:     ak,bk,angle:real;
207: begin
208: for k:=1 to a.nBus do
209: begin
210:     a.Bus[k].Pcal:=0;
211:     a.Bus[k].Qcal:=0;
212:     a.Bus[k].Ireal:=0;
213:     a.Bus[k].Iimag:=0;
214:     for m:=1 to a.nBus do
215:     begin
216:         angle:=a.Bus[m].angle;
217:         ak:=a.Bus[m].E*cos(angle)*a.Y.G[k,m]-a.Bus[m].E*sin(angle)*a.Y.B[k,m];
218:         bk:=a.Bus[m].E*cos(angle)*a.Y.B[k,m]+a.Bus[m].E*sin(angle)*a.Y.G[k,m];
219:         a.Bus[k].Ireal:=a.Bus[k].Ireal+ak;
220:         a.Bus[k].Iimag:=a.Bus[k].Iimag+bk;
221:     end;
222:     angle:=a.Bus[k].angle;
223:     a.Bus[k].Pcal:=a.Bus[k].E*cos(angle)*a.Bus[k].Ireal+
224:     a.Bus[k].E*sin(angle)*a.Bus[k].Iimag;
225:     a.Bus[k].Qcal:=a.Bus[k].E*sin(angle)*a.Bus[k].Ireal-

```

```

226:     a.Bus[k].E*cos(angle)*a.Bus[k].Iimag;
227: end;
228: {-----dB1-----}
229: {-----dQ-----}
230:     Qmaxmismatch[iter].mag:=0;
231:     for i:= 1 to nQ do
232:     begin
233:         k:=dB2[i].Bus;
234:         dB2[i].mag:=(a.Bus[k].Qsp-a.Bus[k].Qcal)/a.Bus[k].E;
235:
236:         if abs(dB2[i].mag)*a.Bus[k].E > abs(Qmaxmismatch[iter].mag) then
237:         begin
238:             Qmaxmismatch[iter].mag:=dB2[i].mag*a.Bus[k].E;
239:             Qmaxmismatch[iter].Bus:=k;
240:         end;
241:     end;
242:
243:     if abs(Pmaxmismatch[iter].mag)>abs(Qmaxmismatch[iter].mag) then
244:     begin
245:         maxmismatch:=abs(Pmaxmismatch[iter].mag);
246:         nowmax:=1;
247:     end
248:     else
249:     begin
250:         maxmismatch:=abs(Qmaxmismatch[iter].mag);
251:         nowmax:=2;
252:     end;
253: end;
254:
255:
256: procedure Tfast_equa_ac.solvefast1(a:Tcomplete_data);
257: var i,j:integer;
258:     value:real;
259:
260: begin
261: {triangulated dB1 }
262:
263:     for i:=1 to nP do
264:         for j:=1 to nP do
265:             if i=j then
266:             begin
267:                 value:=to_dB1[i,i];
268:                 dB1[i].mag:=dB1[i].mag/value;
269:             end
270:             else if j<i then
271:             begin
272:                 value:=to_dB1[i,j];
273:                 dB1[i].mag:=dB1[i].mag-dB1[j].mag*value;
274:             end;
275: {back substituted dX}
276:         for i:= nP downto 1 do
277:         begin
278:             dX1[i].mag:=dB1[i].mag;
279:             if i< nP then
280:                 for j:=i+1 to nP do
281:                     dX1[i].mag:=dX1[i].mag-dX1[j].mag*tri_B1[i,j];
282:             end;
283:             {update data dX}
284:             for i:=1 to nP do
285:             begin
286:                 j:=dX1[i].Bus;
287:                 a.Bus[j].angle:=a.Bus[j].angle+dX1[i].mag;
288:             end;
289:         end;
290:
291: procedure Tfast_equa_ac.solvefast2(a:Tcomplete_data);
292: var i,j:integer;
293:     value:real;
294: begin
295: {triangulated dB1 }
296:
297:     for i:=1 to nQ do
298:         for j:=1 to nQ do
299:             if i=j then
300:             begin

```

```

301:         value:=to_dB2[i,i];
302:         dB2[i].mag:=dB2[i].mag/value;
303:         end
304:         else if j<i then
305:         begin
306:             value:=to_dB2[i,j];
307:             dB2[i].mag:=dB2[i].mag-dB2[j].mag*value;
308:         end;
309:
310:         {back substituted dX2}
311:         for i:= nQ downto 1 do
312:         begin
313:             dX2[i].mag:=dB2[i].mag;
314:             if i< nQ then
315:                 for j:=i+1 to nQ do
316:                     dX2[i].mag:=dX2[i].mag-dX2[j].mag*tri_B2[i,j];
317:                 end;
318:
319:                 { update data dX2 }
320:                 for i:=1 to nQ do
321:                 begin
322:                     j:=dX2[i].Bus;
323:                     a.Bus[j].E:=a.Bus[j].E+dX2[i].mag;
324:                 end;
325:             end;
326:
327:
328:         end.

```

```

1: unit solve_acdc;
2:
3: interface
4: uses SysUtils, classes, complete_data, data, newton_solve_ac;
5: const
6:     k_1=1.350474474;
7:     k_2=0.954929658;
8:
9: type
10:     Tequa_acdc = class
11:         dB1, dB2 : array[1..maxbus*2+maxconv*5] of Tcommon_dX;
12:         dX1, dX2 : array[1..maxbus*2+maxconv*5] of Tcommon_dX;
13:         jac1, jac2: jacobian;
14:         nP, nPterm, nQ, nQterm, nR, nconv, ndB1, ndB2: integer;
15:         Pmaxmismatch: array[1..maxiter] of Tcommon_dX;
16:         Qmaxmismatch: array[1..maxiter] of Tcommon_dX;
17:         Rmaxmismatch: array[1..maxiter] of Tcommon_dX;
18:         nowmax: integer;
19:         maxmismatch, dB1max, dB2max: real;
20:         constructor create;
21:         destructor destroy; override;
22:         procedure build_dB_dX(a:Tcomplete_data);
23:         procedure build_fastequa(a:Tcomplete_data);
24:         procedure solveACDC2(a:Tcomplete_data);
25:         procedure solveACDC1(a:Tcomplete_data);
26:         procedure ACDC2(a:Tcomplete_data; iter: integer);
27:         procedure ACDC1(a:Tcomplete_data; iter: integer);
28:     end;
29: var equa_acdc: Tequa_acdc;
30:
31:
32: {-----}
33: implementation
34:
35: constructor Tequa_acdc.create;
36: var i: integer;
37: begin
38:     inherited create;
39:     for i:=1 to maxbus*2+maxconv*5 do
40:         begin
41:             dB2[i]:=Tcommon_dX.create;
42:             dX2[i]:=Tcommon_dX.create;
43:             dB1[i]:=Tcommon_dX.create;
44:             dX1[i]:=Tcommon_dX.create;
45:         end;
46:
47:         for i:=1 to maxiter do
48:             begin
49:                 Pmaxmismatch[i]:=Tcommon_dX.create;
50:                 Qmaxmismatch[i]:=Tcommon_dX.create;
51:                 Rmaxmismatch[i]:=Tcommon_dX.create;
52:             end;
53:         maxmismatch:=1;
54:     end;
55:
56: destructor Tequa_acdc.destroy;
57: var i: integer;
58: begin
59:     for i:=1 to maxbus*2+maxconv*5 do
60:         begin
61:             dB2[i].destroy;
62:             dX2[i].destroy;
63:             dB1[i].destroy;
64:             dX1[i].destroy;
65:         end;
66:
67:         for i:=1 to maxiter do
68:             begin
69:                 Pmaxmismatch[i].destroy;
70:                 Qmaxmismatch[i].destroy;
71:                 Rmaxmismatch[i].destroy;
72:             end;
73:     inherited destroy;
74: end;
75:

```

```

76:
77:
78:
79:
80:
81: {finished constructor and destructor}
82:
83:
84:
85:
86:
87: procedure Tequa_acdc.build_dB_dX(a:Tcomplete_data);
88: var i,j,k,c_Qterm,c_Pterm:integer;
89:     d:TdB_1;
90: begin
91:     d:=TdB_1.create;
92:     d.build_dB_1(a);
93:     j:=1;
94:     for i:=1 to d.nP do           { dP }
95:     begin
96:         dB1[j].typed:='dP';
97:         dB1[j].Bus:=d.dP[i].Bus;
98:         dX1[j].typed:='dag';
99:         dX1[j].Bus:=d.dP[i].Bus;
100:        j:=j+1;
101:    end;
102:    c_Pterm:=0;
103:    for i:=1 to a.nConv do         {dP term }
104:    begin
105:        k:=a.nCtoB(i);
106:        if (a.Bus[k].typed='PQ') or (a.Bus[k].typed='PV') then
107:        begin
108:            dB1[j].typed:='dPterm';
109:            dB1[j].Bus:=k;
110:            dX1[j].typed:='dagterm';
111:            dX1[j].Bus:=k;
112:            j:=j+1;
113:            c_Pterm:=c_Pterm+1;
114:        end;
115:    end;
116:
117:
118:    for i:=1 to d.nQ do           { dQ }
119:    begin
120:        dB1[j].typed:='dQ';
121:        dB1[j].Bus:=d.dQ[i].Bus;
122:        dX1[j].typed:='dE/E';
123:        dX1[j].Bus:=d.dQ[i].Bus;
124:        j:=j+1;
125:    end;
126:
127:    c_Qterm:=0;
128:    for i:=1 to a.nConv do         {dQterm }
129:    begin
130:        k:=a.nCtoB(i);
131:        if a.Bus[k].typed='PQ' then
132:        begin
133:            dB1[j].typed:='dQterm';
134:            dB1[j].Bus:=k;
135:            dX1[j].typed:='dEterm';
136:            dX1[j].Bus:=k;
137:            j:=j+1;
138:            c_Qterm:=c_Qterm+1;
139:        end;
140:    end;
141:
142:    for i:=1 to a.nconv do         {dR}
143:    begin
144:        k:=a.nCtoB(i);
145:        dB1[j].typed:='R1';    dX1[j].typed:='dVd';
146:        dB1[j].Bus:=k;        dX1[j].Bus:=k;
147:        j:=j+1;
148:        dB1[j].typed:='R2';    dX1[j].typed:='dagdc';
149:        dB1[j].Bus:=k;        dX1[j].Bus:=k;
150:        j:=j+1;

```

```

151:         dB1[j].typed:='R3';    dX1[j].typed:='dcosfir';
152:         dB1[j].Bus:=k;        dX1[j].Bus:=k;
153:         j:=j+1;
154:         dB1[j].typed:='R4';    dX1[j].typed:='da';
155:         dB1[j].Bus:=k;        dX1[j].Bus:=k;
156:         j:=j+1;
157:         dB1[j].typed:='R5';    dX1[j].typed:='dId';
158:         dB1[j].Bus:=k;        dX1[j].Bus:=k;
159:         j:=j+1;
160:     end;
161:     nP:=d.nP;
162:     nQ:=d.nQ;
163:     ndB1:=j-1;
164:     nR:=5*a.nconv;
165:     nconv:=a.nconv;
166:     nPterm:=c_Pterm;
167:     nQterm:=c_Qterm;
168:     d.destroy;
169: end;
170:
171:
172: procedure Tequa_acdc.build_fastequa(a:Tcomplete_data);
173: var i,j,k,m, c_Qterm,c_Pterm:integer;
174:     d:TdB_1;
175:     ch_equa:boolean;
176: begin
177:     d:=TdB_1.create;
178:     d.build_dB_1(a);
179:     j:=1;
180:     for i:=1 to d.nP do           { dP }
181:     begin
182:         dB1[j].typed:='dP/E';
183:         dB1[j].Bus:=d.dP[i].Bus;
184:         dX1[j].typed:='dag';
185:         dX1[j].Bus:=d.dP[i].Bus;
186:         j:=j+1;
187:     end;
188:
189:     c_Pterm:=0;
190:     for i:=1 to a.nConv do       {dP term }
191:     begin
192:         k:=a.nCtoB(i);
193:         if (a.Bus[k].typed='PQ') or (a.Bus[k].typed='PV') then
194:         begin
195:             // check to ensure having non symmetrical equation
196:             ch_equa:=true;
197:             for m:=1 to j do
198:             begin
199:                 if dB1[m].typed='dPterm/E' then
200:                 if dB1[m].Bus=k then
201:                 ch_equa:=false;
202:             end;
203:             if ch_equa then
204:             begin
205:                 dB1[j].typed:='dPterm/E';
206:                 dB1[j].Bus:=k;
207:                 dX1[j].typed:='dagterm';
208:                 dX1[j].Bus:=k;
209:                 j:=j+1;
210:                 c_Pterm:=c_Pterm+1;
211:             end;
212:         end;
213:     end;
214:
215:     for i:=1 to a.nconv do       {dR}
216:     begin
217:         k:=a.nCtoB(i);
218:         dB1[j].typed:='R1';    dX1[j].typed:='dagdc';
219:         dB1[j].Bus:=k;        dX1[j].Bus:=k;
220:         j:=j+1;
221:         dB1[j].typed:='R2';    dX1[j].typed:='da';
222:         dB1[j].Bus:=k;        dX1[j].Bus:=k;
223:         j:=j+1;
224:         dB1[j].typed:='R3';    dX1[j].typed:='dId';
225:         dB1[j].Bus:=k;        dX1[j].Bus:=k;

```

```

226:         j:=j+1;
227:         dB1[j].typed:='R4';      dX1[j].typed:='dcosfir'; {tap}
228:         dB1[j].Bus:=k;          dX1[j].Bus:=k;
229:         j:=j+1;
230:         dB1[j].typed:='R5';      dX1[j].typed:='dVd';
231:         dB1[j].Bus:=k;          dX1[j].Bus:=k;
232:         j:=j+1;
233:     end;
234:     nP:=d.nP;
235:     nPterm:=c_Pterm;
236:     ndB1:=j-1;
237:     nR:=5*a.nconv;
238:     nconv:=a.nconv;
239:
240:     j:=1;
241:     for i:=1 to d.nQ do           { dQ }
242:     begin
243:         dB2[j].typed:='dQ/E';
244:         dB2[j].Bus:=d.dQ[i].Bus;
245:         dX2[j].typed:='dE';
246:         dX2[j].Bus:=d.dQ[i].Bus;
247:         j:=j+1;
248:     end;
249:
250:     c_Qterm:=0;
251:     for i:=1 to a.nConv do       {dQterm }
252:     begin
253:         k:=a.nCtoB(i);
254:         if a.Bus[k].typed='PQ' then
255:         begin
256:             ch_equa:=true;
257:             for m:=1 to j do
258:             begin
259:                 if dB2[m].typed='dQterm/E' then
260:                 if dB2[m].Bus=k then
261:                 ch_equa:=false;
262:             end;
263:             if ch_equa then
264:             begin
265:                 dB2[j].typed:='dQterm/E';
266:                 dB2[j].Bus:=k;
267:                 dX2[j].typed:='dEterm';
268:                 dX2[j].Bus:=k;
269:                 j:=j+1;
270:                 c_Qterm:=c_Qterm+1;
271:             end;
272:         end;
273:     end;
274:
275:     for i:=1 to a.nconv do       {dR}
276:     begin
277:         k:=a.nCtoB(i);
278:         dB2[j].typed:='R1';      dX2[j].typed:='dagdc';
279:         dB2[j].Bus:=k;          dX2[j].Bus:=k;
280:         j:=j+1;
281:         dB2[j].typed:='R2';      dX2[j].typed:='da';
282:         dB2[j].Bus:=k;          dX2[j].Bus:=k;
283:         j:=j+1;
284:         dB2[j].typed:='R3';      dX2[j].typed:='dId';
285:         dB2[j].Bus:=k;          dX2[j].Bus:=k;
286:         j:=j+1;
287:         dB2[j].typed:='R4';      dX2[j].typed:='dcosfir';
288:         dB2[j].Bus:=k;          dX2[j].Bus:=k;
289:         j:=j+1;
290:         dB2[j].typed:='R5';      dX2[j].typed:='dVd';
291:         dB2[j].Bus:=k;          dX2[j].Bus:=k;
292:         j:=j+1;
293:     end;
294:     nQ:=d.nQ;
295:     nQterm:=c_Qterm;
296:     ndB2:=j-1;
297:     d.destroy;
298: end;
299:
300:

```

```

301:
302: procedure Tequa_acdc.ACDC1(a:Tcomplete_data;iter:integer);
303: var i,j ,k,m,kl,ml,h:integer;
304:     ak,bk,ang:real;
305:     B_:TY_metrix;
306: begin
307: {-----dB1-----}
308: for k:=1 to a.nBus do
309: begin
310:     a.Bus[k].Pcal:=0;
311:     a.Bus[k].Qcal:=0;
312:     a.Bus[k].Ireal:=0;
313:     a.Bus[k].Iimag:=0;
314:     for m:=1 to a.nBus do
315:     begin
316: //     ang:=a.Bus[m].angle;
317: //     ak:=a.Bus[m].E*cos(ang)*a.Y.G[k,m]-a.Bus[m].E*sin(ang)*a.Y.B[k,m];
318: //     bk:=a.Bus[m].E*cos(ang)*a.Y.B[k,m]+a.Bus[m].E*sin(ang)*a.Y.G[k,m];
319: //     a.Bus[k].Ireal:=a.Bus[k].Ireal+ak;
320: //     a.Bus[k].Iimag:=a.Bus[k].Iimag+bk;
321: a.Bus[k].Pcal:=a.Bus[k].Pcal + a.Bus[k].E*a.Bus[m].E*a.GcBs(k,m);
322: a.Bus[k].Qcal:=a.Bus[k].Qcal + a.Bus[k].E*a.Bus[m].E*a.Gs_Bc(k,m);
323:     end;
324: //     ang:=a.Bus[k].angle;
325: //     a.Bus[k].Pcal:=a.Bus[k].E*cos(ang)*a.Bus[k].Ireal+
326: //         a.Bus[k].E*sin(ang)*a.Bus[k].Iimag;
327: //     a.Bus[k].Qcal:=a.Bus[k].E*sin(ang)*a.Bus[k].Ireal-
328: //         a.Bus[k].E*cos(ang)*a.Bus[k].Iimag;
329:     end;
330: {----dP----}
331: Pmaxmismatch[iter].mag:=0;
332: Rmaxmismatch[iter].mag:=0;
333: for i:=1 to nP do
334: begin
335: k:=dB1[i].Bus;
336: dB1[i].mag:=(a.Bus[k].Psp-a.Bus[k].Pcal)/a.Bus[k].E;
337:     if abs(dB1[i].mag) > abs(Pmaxmismatch[iter].mag) then
338:     begin
339: Pmaxmismatch[iter].mag:=dB1[i].mag;
340: Pmaxmismatch[iter].Bus:=k;
341:     end;
342: end;
343:
344: {-----dPterm-----}
345: for i:=nP+1 to nP+nPterm do
346: begin
347: k:=dB1[i].Bus;
348: m:=a.BtonC(k);
349:
350: with a.conv[m] do
351: begin
352: Pd:=Vd*Id;
353: Qd:=a.Bus[k].E*k1*tap*Id*sin(angle);
354: end;
355:
356: dB1[i].mag:=(a.Bus[k].Psp-a.Bus[k].Pcal-a.conv[m].Pd)/a.Bus[k].E;
357:     if abs(dB1[i].mag) > abs(Pmaxmismatch[iter].mag) then
358:     begin
359: Pmaxmismatch[iter].mag:=dB1[i].mag;
360: Pmaxmismatch[iter].Bus:=k;
361:     end;
362: end;
363: {-----dR-----}
364: i:=nP+nPterm+1;
365: for m:=1 to nconv do
366: begin
367: k:=a.nCtoB(m);
368: j:=a.conv[m].T_Bus;
369: j:=a.BtonC(j);
370: with a.conv[m] do
371: begin
372: // R1
373: dB1[i].mag:=k1*tap*a.Bus[k].E*cos(angle)-Vd;
374: if abs(dB1[i].mag) > abs(Rmaxmismatch[iter].mag) then
375:

```

```

376:     Rmaxmismatch[iter].mag:=dB1[i].mag;
377:     Rmaxmismatch[iter].Bus:=k;
378:     end;
379:     i:=i+1;
380:     // R2
381:     if typed='Rec' then dB1[i].mag:=k_1*tap*a.Bus[k].E*cosfir-Vd-k_2*Id*X
382:     else dB1[i].mag:=k_1*tap*a.Bus[k].E*cosfir+k_2*Id*X-Vd;
383:     if abs(dB1[i].mag) > abs(Rmaxmismatch[iter].mag) then
384:     begin
385:     Rmaxmismatch[iter].mag:=dB1[i].mag;
386:     Rmaxmismatch[iter].Bus:=k;
387:     end;
388:     i:=i+1;
389:     // R3
390:     dB1[i].mag:=-Vd-a.conv[j].Vd+Rdc*Id;
391:     if abs(dB1[i].mag) > abs(Rmaxmismatch[iter].mag) then
392:     begin
393:     Rmaxmismatch[iter].mag:=dB1[i].mag;
394:     Rmaxmismatch[iter].Bus:=k;
395:     end;
396:     i:=i+1;
397:     // R4
398:     dB1[i].mag:=cosfirsp-cosfir;
399:     if abs(dB1[i].mag) > abs(Rmaxmismatch[iter].mag).then
400:     begin
401:     Rmaxmismatch[iter].mag:=dB1[i].mag;
402:     Rmaxmismatch[iter].Bus:=k;
403:     end;
404:     i:=i+1;
405:     // R5
406:     if typed='Rec' then
407:     begin
408:     dB1[i].mag:=Pdsp-Vd*Id;
409:     Pd:=Vd*Id;
410:     end
411:     else dB1[i].mag:=Vdsp-Vd;
412:     if abs(dB1[i].mag) > abs(Rmaxmismatch[iter].mag) then
413:     begin
414:     Rmaxmismatch[iter].mag:=dB1[i].mag;
415:     Rmaxmismatch[iter].Bus:=k;
416:     end;
417:     i:=i+1;
418:     end;
419:     end;
420:
421: // mismatch
422: if abs(Pmaxmismatch[iter].mag) > abs(Rmaxmismatch[iter].mag) then
423: dB1max:=Pmaxmismatch[iter].mag
424: else
425: dB1max:=Rmaxmismatch[iter].mag;
426:
427: if abs(dB1max)<maxmismatch then
428: if nowmax=1 then
429: maxmismatch:=abs(dB1max);
430: {-----jacobian-----}
431:
432: {-----B'-----}
433: B_:=Ty_metrix.create;
434: B_.buildB1(datat);
435: for i:=1 to nP+nPterm do
436: for j:=1 to nP+nPterm do
437: begin
438: k:=dB1[i].Bus;
439: m:=dX1[j].Bus;
440: jacl[i,j]:=-B_.B[k,m];
441: end;
442:
443:
444: {-----AA'-----}
445: for i:=nP+1 to nP+nPterm do
446: for j:=nP+nPterm+1 to nP+nPterm+nConv*5 do
447: begin
448: k:=dB1[i].Bus;
449: kl:=a.BtonC(k);
450: h:=dX1[j].Bus;

```

```

451:     jacl[i,j]:=0;
452:     if k=h then
453:       if dX1[j].typed='dId' then
454:         jacl[i,j]:=a.conv[k1].Vd/a.Bus[k].E
455:       else if dX1[j].typed='dVd' then
456:         jacl[i,j]:=a.conv[k1].Id/a.Bus[k].E;
457:     end;
458:     {-----A-----}
459:     for kl:=1 to nconv do
460:       for i:=nP+nPterm+(kl-1)*5+1 to nP+nPterm+kl*5 do
461:         for j:=nP+nPterm+(kl-1)*5+1 to nP+nPterm+kl*5 do
462:           begin
463:             k:=a.nCtoB(kl);
464:             m:=a.conv[k1].T_Bus;
465:             jacl[i,j]:=0;
466:             with a.conv[k1] do begin
467:               {P1}if dB1[i].typed='R1' then
468:                 if dX1[j].typed='dagdc' then
469:                   jacl[i,j]:=k_1*tap*a.Bus[k].E*sin(angle)
470:                 else if dX1[j].typed='da' then
471:                   jacl[i,j]:=-k_1*a.Bus[k].E*cos(angle)
472:                 else if dX1[j].typed='dVd' then
473:                   jacl[i,j]:=1;
474:
475:               {P2}if dB1[i].typed='R2' then
476:                 if typed='Rec' then
477:                   begin
478:                     if dX1[j].typed='da' then
479:                       jacl[i,j]:=-k_1*a.Bus[k].E*cosfir
480:                     else if dX1[j].typed='dId' then
481:                       jacl[i,j]:=k_2*X
482:                     else if dX1[j].typed='dcosfir' then
483:                       jacl[i,j]:=-k_1*tap*a.Bus[k].E
484:                     else if dX1[j].typed='dVd' then
485:                       jacl[i,j]:=1;
486:                   end
487:                 else
488:                   begin
489:                     if dX1[j].typed='da' then
490:                       jacl[i,j]:=-k_1*a.Bus[k].E*cosfir
491:                     else if dX1[j].typed='dId' then
492:                       jacl[i,j]:=-k_2*X
493:                     else if dX1[j].typed='dcosfir' then
494:                       jacl[i,j]:=-k_1*tap*a.Bus[k].E
495:                     else if dX1[j].typed='dVd' then
496:                       jacl[i,j]:=1;
497:                   end;
498:
499:               {P3}if dB1[i].typed='R3' then
500:                 begin
501:                   if (dX1[j].typed='dId') then
502:                     jacl[i,j]:=-Rdc
503:                   else if dX1[j].typed='dVd' then
504:                     jacl[i,j]:=1;
505:                   for h:=nP+nPterm+1 to ndB1 do
506:                     if (dX1[h].typed='dVd') and (dX1[h].Bus=m) then
507:                       jacl[i,h]:=1;
508:                 end;
509:
510:               {P4}if dB1[i].typed='R4' then
511:                 if dX1[j].typed='dcosfir' then
512:                   jacl[i,j]:=1;
513:
514:               {P5}if dB1[i].typed='R5' then
515:                 if typed='Rec' then
516:                   begin
517:                     if dX1[j].typed='dId' then
518:                       jacl[i,j]:=Vd
519:                     else if dX1[j].typed='dVd' then
520:                       jacl[i,j]:=Id;
521:                   end
522:                 else
523:                   if dX1[j].typed='dVd' then
524:                     jacl[i,j]:=1;
525:                 end;

```

```

526: end;
527:
528: for i:=1 to nP do
529: for j:=nP+nPterm+1 to nP+nPterm+nconv*5 do
530: jacl[i,j]:=0;
531:
532: for i:=nP+nPterm+1 to nP+nPterm+nconv*5 do
533: for j:=1 to nP+nPterm do
534: jacl[i,j]:=0;
535:
536: end;
537:
538: procedure Tequa_acdc.ACDC2(a:Tcomplete_data;iter:integer);
539: var i,j ,k,m,k1,m1,h:integer;
540:     ak,bk,ang:real;
541: begin
542: {-----dB2-----}
543: for k:=1 to a.nBus do
544: begin
545:     a.Bus[k].Pcal:=0;
546:     a.Bus[k].Qcal:=0;
547:     a.Bus[k].Ireal:=0;
548:     a.Bus[k].Iimag:=0;
549:     for m:=1 to a.nBus do
550:     begin
551:         ang:=a.Bus[m].angle;
552:         ak:=a.Bus[m].E*cos(ang)*a.Y.G[k,m]-a.Bus[m].E*sin(ang)*a.Y.B[k,m];
553:         bk:=a.Bus[m].E*cos(ang)*a.Y.B[k,m]+a.Bus[m].E*sin(ang)*a.Y.G[k,m];
554:         a.Bus[k].Ireal:=a.Bus[k].Ireal+ak;
555:         a.Bus[k].Iimag:=a.Bus[k].Iimag+bk;
556:         {a.Bus[k].Pcal:=a.Bus[k].Pcal + a.Bus[k].E*a.Bus[m].E*a.GcBs(k,m) ;
557:         a.Bus[k].Qcal:=a.Bus[k].Qcal + a.Bus[k].E*a.Bus[m].E*a.Gs Bc(k,m) ;}
558:     end;
559:     ang:=a.Bus[k].angle;
560:     a.Bus[k].Pcal:=a.Bus[k].E*cos(ang)*a.Bus[k].Ireal+
561:     a.Bus[k].E*sin(ang)*a.Bus[k].Iimag;
562:     a.Bus[k].Qcal:=a.Bus[k].E*sin(ang)*a.Bus[k].Ireal-
563:     a.Bus[k].E*cos(ang)*a.Bus[k].Iimag;
564: end;
565: {-----dQ-----}
566: Qmaxmismatch[iter].mag:=0;
567: Rmaxmismatch[iter].mag:=0;
568: for i:=1 to nQ do
569: begin
570: k:=dB2[i].Bus;
571: dB2[i].mag:=(a.Bus[k].Qsp-a.Bus[k].Qcal)/a.Bus[k].E;
572:     if abs(dB2[i].mag) > abs(Qmaxmismatch[iter].mag) then
573:     begin
574:         Qmaxmismatch[iter].mag:=dB2[i].mag;
575:         Qmaxmismatch[iter].Bus:=k;
576:     end;
577: end;
578: {-----dQterm-----}
579:     for i:=nQ+1 to nQ+nQterm do
580:     begin
581:         k:=dB2[i].Bus;
582:         m:=a.BtonC(k);
583:
584:         with a.conv[m] do
585:         begin
586:             Pd:=Vd*Id;
587:             Qd:=a.Bus[k].E*k_1*tap*Id*sin(angle);
588:             dB2[i].mag:=(a.Bus[k].Qsp-a.Bus[k].Qcal-Qd)/a.Bus[k].E;
589:         end;
590:             if abs(dB2[i].mag) > abs(Qmaxmismatch[iter].mag) then
591:             begin
592:                 Qmaxmismatch[iter].mag:=dB2[i].mag;
593:                 Qmaxmismatch[iter].Bus:=k;
594:             end;
595:         end;
596:
597: {-----dR-----}
598: i:=nQ+nQterm+1;
599: for m:=1 to nconv do
600: begin

```

```

601:         k:=a.nDtoB(m);
602:         j:=a.conv[m].T_Bus;
603:         j:=a.BtonC(j);
604:         with a.conv[m] do
605:             begin
606:                 // R1
607:                 dB2[i].mag:=k_1*tap*a.Bus[k].E*cos(angle)-Vd;
608:                 if abs(dB2[i].mag) > abs(Rmaxmismatch[iter].mag) then
609:                     begin
610:                         Rmaxmismatch[iter].mag:=dB2[i].mag;
611:                         Rmaxmismatch[iter].Bus:=k;
612:                     end;
613:                 i:=i+1;
614:                 // R2
615:                 if typed='Rec' then dB2[i].mag:=k_1*tap*a.Bus[k].E*cosfir-Vd-k_2*Id*X
616:                     else dB2[i].mag:=k_1*tap*a.Bus[k].E*cosfir+k_2*Id*X-Vd;
617:                 if abs(dB2[i].mag) > abs(Rmaxmismatch[iter].mag) then
618:                     begin
619:                         Rmaxmismatch[iter].mag:=dB2[i].mag;
620:                         Rmaxmismatch[iter].Bus:=k;
621:                     end;
622:                 i:=i+1;
623:                 // R3
624:                 dB2[i].mag:=-Vd-a.conv[j].Vd+Rdc*Id;
625:                 if abs(dB2[i].mag) > abs(Rmaxmismatch[iter].mag) then
626:                     begin
627:                         Rmaxmismatch[iter].mag:=dB2[i].mag;
628:                         Rmaxmismatch[iter].Bus:=k;
629:                     end;
630:                 i:=i+1;
631:                 // R4
632:                 dB2[i].mag:=cosfirsp-cosfir;
633:                 if abs(dB2[i].mag) > abs(Rmaxmismatch[iter].mag) then
634:                     begin
635:                         Rmaxmismatch[iter].mag:=dB2[i].mag;
636:                         Rmaxmismatch[iter].Bus:=k;
637:                     end;
638:                 i:=i+1;
639:                 // R5
640:                 if typed='Rec' then
641:                     begin
642:                         dB2[i].mag:=Pdsp-Vd*Id;
643:                         Pd:=Vd*Id;
644:                     end
645:                 else dB2[i].mag:=Vdsp-Vd;
646:                 if abs(dB2[i].mag) > abs(Rmaxmismatch[iter].mag) then
647:                     begin
648:                         Rmaxmismatch[iter].mag:=dB2[i].mag;
649:                         Rmaxmismatch[iter].Bus:=k;
650:                     end;
651:                 i:=i+1;
652:                 end;
653:                 end;
654:
655: // mismatch
656: if abs(Qmaxmismatch[iter].mag) > abs(Rmaxmismatch[iter].mag) then
657:     dB2max:=Qmaxmismatch[iter].mag
658: else
659:     dB2max:=Rmaxmismatch[iter].mag;
660:
661: if abs(dB1max) > abs(dB2max) then
662:     begin
663:         maxmismatch:=abs(dB1max);
664:         nowmax:=1;
665:     end
666: else
667:     begin
668:         maxmismatch:=abs(dB2max);
669:         nowmax:=2;
670:     end;
671:
672: {-----jacobian-----}
673:
674: {-----B'-----}
675: for i:=1 to nQ+nQterm do

```

```

676: for j:=1 to nQ+nQterm do
677: begin
678: k:=dB2[i].Bus;
679: m:=dX2[j].Bus;
680: jac2[i,j]:=-a.Y.B[k,m];
681: end;
682:
683:
684: {-----add B'' dc-----}
685: for i:=nQ+1 to nQ+nQterm do
686: for j:=nQ+1 to nQ+nQterm do
687: begin
688: k:=dB2[i].Bus;
689: m:=dX2[j].Bus;
690: if k=m then
691: begin
692: kl:=a.BtonC(k);
693: with a.conv[kl] do
694: begin
695: jac2[k,k]:=jac2[k,k]+k_1*tap*Id*sin(angle)/a.Bus[k].E;
696: end;
697: end;
698: end;
699: {-----AA''-----}
700: for i:=nQ+1 to nQ+nQterm do
701: for j:=nQ+nQterm+1 to nQ+nQterm+nconv*5 do
702: begin
703: k:=dB2[i].Bus;
704: kl:=a.BtonC(k);
705: h:=dX2[j].Bus;
706: jac2[i,j]:=0;
707: if k=h then
708: with a.conv[kl] do
709: begin
710: if dX2[j].typed='dagdc' then
711: jac2[i,j]:=k_1*tap*Id*cos(angle)
712: else if dX2[j].typed='da' then
713: jac2[i,j]:=k_1*Id*sin(angle)
714: else if dX2[j].typed='dId' then
715: jac2[i,j]:=k_1*tap*sin(angle);
716: end;
717: end;
718: {-----BB''-----}
719: for kl:=1 to nconv do
720: for i:=nQ+nQterm+(kl-1)*5+1 to nQ+nQterm+kl*5 do
721: for j:=nQ+1 to nQ+nQterm do
722: begin
723: k:=a.nCtoB(kl);
724: h:=dX2[j].Bus;
725: jac2[i,j]:=0;
726: if k=h then
727: with a.conv[kl] do
728: begin
729: // R1
730: if dB2[i].typed='R1' then jac2[i,j]:=-k_1*tap*cos(angle);
731: // R2
732: if dB2[i].typed='R2' then jac2[i,j]:=-k_1*tap*cosfir;
733: end;
734: end;
735: {-----A-----}
736: for kl:=1 to nconv do
737: for i:=nQ+nQterm+(kl-1)*5+1 to nQ+nQterm+kl*5 do
738: for j:=nQ+nQterm+(kl-1)*5+1 to nQ+nQterm+kl*5 do
739: begin
740: k:=a.nCtoB(kl);
741: m:=a.conv[kl].T_Bus;
742: jac2[i,j]:=0;
743: with a.conv[kl] do begin
744: {R1}if dB2[i].typed='R1' then
745: if dX2[j].typed='dagdc' then
746: jac2[i,j]:=k_1*tap*a.Bus[k].E*sin(angle)
747: else if dX2[j].typed='da' then
748: jac2[i,j]:=-k_1*a.Bus[k].E*cos(angle)
749: else if dX2[j].typed='dVd' then
750: jac2[i,j]:=1;

```

```

751:
752: {R2}if dB2[i].typed='R2' then
753:     if typed='Rec' then
754:         begin
755:             if dX2[j].typed='da' then
756:                 jac2[i,j]:=-k_1*a.Bus[k].E*cosfir
757:             else if dX2[j].typed='dId' then
758:                 jac2[i,j]:=k_2*X
759:             else if dX2[j].typed='dcosfir' then
760:                 jac2[i,j]:=-k_1*tap*a.Bus[k].E
761:             else if dX2[j].typed='dVd' then
762:                 jac2[i,j]:=1;
763:         end
764:     else
765:         begin
766:             if dX2[j].typed='da' then
767:                 jac2[i,j]:=-k_1*a.Bus[k].E*cosfir
768:             else if dX2[j].typed='dId' then
769:                 jac2[i,j]:=-k_2*X
770:             else if dX2[j].typed='dcosfir' then
771:                 jac2[i,j]:=-k_1*tap*a.Bus[k].E
772:             else if dX2[j].typed='dVd' then
773:                 jac2[i,j]:=1;
774:         end;
775:
776: {R3}if dB2[i].typed='R3' then
777:     begin
778:         if (dX2[j].typed='dId') then
779:             jac2[i,j]:=-Rdc
780:         else if dX2[j].typed='dVd' then
781:             jac2[i,j]:=1;
782:         for h:=nQ+nQterm+1 to ndB2 do
783:             if (dX2[h].typed='dVd') and (dX2[h].Bus=m) then
784:                 jac2[i,h]:=1;
785:         end;
786:
787: {R4}if dB2[i].typed='R4' then
788:     if dX2[j].typed='dcosfir' then
789:         jac2[i,j]:=1;
790:
791: {R5}if dB2[i].typed='R5' then
792:     if typed='Rec' then
793:         begin
794:             if dX2[j].typed='dId' then
795:                 jac2[i,j]:=Vd
796:             else if dX2[j].typed='dVd' then
797:                 jac2[i,j]:=Id;
798:         end
799:     else
800:         if dX2[j].typed='dVd' then
801:             jac2[i,j]:=1;
802:     end;
803: end;
804:
805: for i:=1 to nQ do
806:     for j:=nQ+nQterm+1 to nQ+nQterm+nconv*5 do
807:         jac2[i,j]:=0;
808:
809:     for i:=nQ+nQterm+1 to nQ+nQterm+nconv*5 do
810:         for j:=1 to nQ do
811:             jac2[i,j]:=0;
812:
813:         end;
814:
815: procedure Tequa_acdc.solveACDC1(a:Tcomplete_data);
816: var i,j,k:integer;
817:     value:real;
818: begin
819:     {triangulated}
820:     for i:=1 to ndB1 do
821:         begin
822:             for j:=1 to ndB1 do
823:                 if i=j then
824:                     begin
825:                         value:=jac1[i,i];

```

```

826:         dB1[i].mag:=dB1[i].mag/value;
827:         for k:=1 to ndB1 do
828:             jacl[i,k]:=jacl[i,k]/value;
829:         end
830:         else if j<i then
831:             begin
832:                 value:=jacl[i,j];
833:                 dB1[i].mag:=dB1[i].mag-dB1[j].mag*value;
834:                 for k:=1 to ndB1 do
835:                     jacl[i,k]:=jacl[i,k]-jacl[j,k]*value;
836:                 end;
837:             end;
838:         end;
839:         {back substituted}
840:         for i:= ndB1 downto 1 do
841:             begin
842:                 dX1[i].mag:=dB1[i].mag;
843:                 if i< ndB1 then
844:                     for j:=i+1 to ndB1 do
845:                         dX1[i].mag:=dX1[i].mag-dX1[j].mag*jacl[i,j];
846:                     end;
847:                 {update data}
848:                 {dP}
849:                 for i:=1 to nP do
850:                     begin
851:                         j:=dX1[i].Bus;
852:                         a.Bus[j].angle:=a.Bus[j].angle+dX1[i].mag;
853:                     end;
854:                 {dPterm}
855:                 for i:=nP+1 to nP+nPterm do
856:                     begin
857:                         j:=dX1[i].Bus;
858:                         a.Bus[j].angle:=a.Bus[j].angle+dX1[i].mag;
859:                     end;
860:                 {dR}
861:
862:
863:                 for i:=nP+nPterm+1 to ndB1 do
864:                     begin
865:                         j:=dB1[i].Bus;
866:                         k:=a.BtonC(j);
867:                         if dX1[i].typed='dagdc' then
868:                             begin
869:                                 a.conv[k].angle:=a.conv[k].angle+dX1[i].mag;
870:                             end
871:                         else if dX1[i].typed='da' then
872:                             begin
873:                                 a.conv[k].tap:=a.conv[k].tap+dX1[i].mag;
874:                                 //if a.conv[k].tap > 1 then a.conv[k].tap:=a.conv[k].tap-2*dX1[i].mag;
875:                             end
876:                         else if dX1[i].typed='dId' then
877:                             begin
878:                                 a.conv[k].Id:=a.conv[k].Id+dX1[i].mag;
879:                             end
880:                         else if dX1[i].typed='dcofir' then
881:                             begin
882:                                 a.conv[k].cofir:=a.conv[k].cofir+dX1[i].mag;
883:                             end
884:                         else if dX1[i].typed='dVd' then
885:                             begin
886:                                 a.conv[k].Vd:=a.conv[k].Vd+dX1[i].mag;
887:                             end;
888:                         end;
889:
890:                 end;
891:
892:
893:
894: procedure Tequa_acdc.solveACDC2(a:Tcomplete_data);
895: var i,j, k:integer;
896:     value:real;
897: begin
898:
899: {triangulated}
900:     for i:=1 to ndB2 do

```

```

901:   for j:=1 to ndB2 do
902:     if i=j then
903:       begin
904:         value:=jac2[i,i];
905:         dB2[i].mag:=dB2[i].mag/value;
906:         for k:=1 to ndB2 do
907:           jac2[i,k]:=jac2[i,k]/value;
908:         end
909:       else if j<i then
910:         begin
911:           value:=jac2[i,j];
912:           dB2[i].mag:=dB2[i].mag-dB2[j].mag*value;
913:           for k:=1 to ndB2 do
914:             jac2[i,k]:=jac2[i,k]-jac2[j,k]*value;
915:           end;
916:         {back substituted}
917:         for i:= ndB2 downto 1 do
918:           begin
919:             dX2[i].mag:=dB2[i].mag;
920:             if i< ndB2 then
921:               for j:=i+1 to ndB2 do
922:                 dX2[i].mag:=dX2[i].mag-dX2[j].mag*jac2[i,j];
923:             end;
924:           {update data}
925:           {dQ}
926:           for i:=1 to nQ do
927:             begin
928:               j:=dX2[i].Bus;
929:               a.Bus[j].E:=a.Bus[j].E+dX2[i].mag;
930:             end;
931:           {dQterm}
932:           for i:=nQ+1 to nQ+nQterm do
933:             begin
934:               j:=dX2[i].Bus;
935:               a.Bus[j].E:=a.Bus[j].E+dX2[i].mag;
936:             end;
937:           {dR}
938:           for i:=nQ+nQterm+1 to ndB2 do
939:             begin
940:               j:=dX2[i].Bus;
941:               k:=a.BtonC(j);
942:               if dX2[i].typed='dagdc' then
943:                 begin
944:                   a.conv[k].angle:=a.conv[k].angle+dX2[i].mag
945:                 end
946:               else if dX2[i].typed='da' then
947:                 begin
948:                   a.conv[k].tap:=a.conv[k].tap+dX2[i].mag;
949:                   // if a.conv[k].tap > 1 then a.conv[k].tap:=a.conv[k].tap-2*dX2[i].mag;
950:                 end
951:               else if dX2[i].typed='dId' then
952:                 begin
953:                   a.conv[k].Id:=a.conv[k].Id+dX2[i].mag;
954:                 end
955:               else if dX2[i].typed='dcosfir' then
956:                 begin
957:                   a.conv[k].cosfir:=a.conv[k].cosfir+dX2[i].mag;
958:                 end
959:               else if dX2[i].typed='dVd' then
960:                 begin
961:                   a.conv[k].Vd:=a.conv[k].Vd+dX2[i].mag;
962:                 end;
963:             end;
964:           end;
965:         end;
966:
967:
968:
969: end.

```

```

1: unit output;
2:
3: interface
4: uses SysUtils, classes, grids, data, complete_data;
5: const dpr=57.29577951;
6: type
7:   Tload_flow1 = class
8:     F_bus, line, T_bus: integer;
9:     name: string[15];
10:    V_mag, V_ag, GenMW, GenMVAR, LoadMW, LoadMVAR, flowMW,
11:    flowMVAR, flowMVA, line_charg, Gii, Gij, Bii, Bij, Gji, Gjj, Bji, Bjj,
12:    R_serie, X_serie, G_serie, B_serie, Bshunt: real;
13:    constructor create;
14:    destructor destroy; override;
15:  end;
16:   Tloss_flow1 = class
17:     F_bus, line, T_bus: integer;
18:     name: string[15];
19:     V_mag, V_ag, GenMW, GenMVAR, LoadMW, LoadMVAR, lossMW,
20:     lossMVAR, lossMVA, line_charg, F_MVAR_line_charg, T_MVAR_line_charg,
21:     MVAR_line_charg, MVAR_charg: real;
22:     constructor create;
23:     destructor destroy; override;
24:  end;
25:
26:   Tloss_flow = class
27:     flow : array[1..2*maxline] of Tloss_flow1;
28:     nloss: integer;
29:     total_P, total_Q, total_charging: real;
30:     procedure getdata(a:Tdata);
31:     procedure writeon(a:Tstringgrid);
32:     constructor create;
33:     destructor destroy; override;
34:  end;
35:
36:   Tload_flow = class
37:     flow : array[1..2*maxline] of Tload_flow1;
38:     nline: integer;
39:     total_MW_load, total_MVAR_load, total_MW_Gen, total_MVAR_Gen: real;
40:     procedure getdata(a:Tdata);
41:     procedure getoutput(a:Tcomplete_data; b:Tloss_flow);
42:     {call after getdata}
43:     procedure exporttofile(b:Tloss_flow);
44:     {call after getoutput}
45:     procedure writeon(a:Tstringgrid);
46:     constructor create;
47:     destructor destroy; override;
48:  end;
49:   Tcomplex = record
50:     r : real;
51:     i : real;
52:  end;
53:
54: var load_flow : Tload_flow;
55:     loss_flow : Tloss_flow;
56:
57: implementation
58: uses main, export_files;
59:
60: function rts(numreal:real):string;
61: begin
62: str(numreal:1:3, result);
63: end;
64:
65: constructor Tload_flow1.create;
66: begin
67: inherited create;
68:   F_bus:=0; T_bus:=0;
69:   name:='';
70:   V_mag:=0; V_ag:=0; GenMW:=0; GenMVAR:=0; LoadMW:=0;
71:   LoadMVAR:=0; flowMW:=0;
72:   flowMVAR:=0; flowMVA:=0;
73: end;
74: destructor Tload_flow1.destroy;
75: begin

```

```

76:         F_bus:=0; T_bus:=0;
77:         name:='';
78:         V_mag:=0;V_ag:=0;GenMW:=0;GenMVAR:=0;LoadMW:=0;
79:         LoadMVAR:=0;flowMW:=0;
80:         flowMVAR:=0;flowMVA:=0;
81:         inherited destroy;
82:     end;
83: constructor Tloss_flow1.create;
84: begin
85:     inherited create;
86:         F_bus:=0; T_bus:=0;
87:         name:='';
88:         V_mag:=0;V_ag:=0;GenMW:=0;GenMVAR:=0;LoadMW:=0;
89:         LoadMVAR:=0;lossMW:=0;
90:         lossMVAR:=0;lossMVA:=0;
91:         line_charg:=0;
92:     end;
93: destructor Tloss_flow1.destroy;
94: begin
95:         F_bus:=0; T_bus:=0;
96:         name:='';
97:         V_mag:=0;V_ag:=0;GenMW:=0;GenMVAR:=0;LoadMW:=0;
98:         LoadMVAR:=0;lossMW:=0;
99:         lossMVAR:=0;lossMVA:=0;
100:        line_charg:=0;
101:        inherited destroy;
102:    end;
103:
104: constructor Tload_flow.create;
105: var i:integer;
106: begin
107:     inherited create;
108:     for i:=1 to 2*maxline do
109:         flow[i]:=Tload_flow1.create;
110:     end;
111: destructor Tload_flow.destroy;
112: var i:integer;
113: begin
114:     for i:=1 to 2*maxline do
115:         flow[i].destroy;
116:     inherited destroy;
117: end;
118:
119: constructor Tloss_flow.create;
120: var i:integer;
121: begin
122:     inherited create;
123:     for i:=1 to 2*maxline do
124:         flow[i]:=Tloss_flow1.create;
125:     end;
126: destructor Tloss_flow.destroy;
127: var i:integer;
128: begin
129:     for i:=1 to 2*maxline do
130:         flow[i].destroy;
131:     inherited destroy;
132: end;
133:
134:
135: {-----end of constructor destructor-----}
136: procedure cplus(var c:Tcomplex; a,b:Tcomplex);
137: begin
138:     c.r:=a.r+b.r;
139:     c.i:=a.i+b.i;
140: end;
141: {-----}
142: procedure csub(var c:Tcomplex; a,b:Tcomplex);
143: begin
144:     c.r := a.r - b.r;
145:     c.i := a.i - b.i;
146: end;
147: {-----}
148: procedure conj(var c:Tcomplex; a:Tcomplex);
149: begin
150:     c.r := a.r;

```

```

151:   c.i := -a.i;
152: end;
153: {-----}
154: procedure cmult(var c:Tcomplex; a,b:Tcomplex);
155: begin
156:   c.r := a.r * b.r - a.i * b.i;
157:   c.i := a.r * b.i + a.i * b.r;
158: end;
159: {-----}
160: procedure cmag( var c:real; a:Tcomplex );
161: begin
162:   c := sqrt( a.r * a.r + a.i * a.i );
163: end;
164: {-----}
165:
166:
167: procedure Tload_flow.getdata(a:Tdata);
168: var i,j,k:integer;
169:   gline,bline,z,tr_real,tr_imag,t:real;
170: begin
171: k:=1;
172: { build the object that correspond to load flow table }
173: for i:=1 to a.nbus do
174: begin
175:   for j:=1 to a.nbranch do
176:   begin
177:     if a.Branch[j].F_bus=i then
178:     begin
179:       flow[k].line:=j;
180:       flow[k].F_bus:=a.Branch[j].F_bus;
181:       flow[k].T_bus:=a.Branch[j].T_bus;
182:       flow[k].name:=a.Bus[i].name;
183:       flow[k].line_charg:=a.Branch[j].line_charg;
184:       flow[k].R_serie:=a.Branch[j].resistance;
185:       flow[k].X_serie:=a.Branch[j].reactance;
186:       z:=a.branch[j].resistance*a.branch[j].resistance+a.branch[j].reactanc
187:       flow[k].G_serie:=a.branch[j].resistance/z;
188:       flow[k].B_serie:=-a.branch[j].reactance/z;
189:       flow[k].Bshunt:=a.Bus[i].Bshunt;
190:       gline:=flow[k].G_serie;
191:       bline:=flow[k].B_serie;
192:
193:       if (a.branch[j].Tr_ratio=0.0) and (a.branch[j].Tr_ag=0.0) then
194:       begin
195:         flow[k].Gii:=gline;
196:         flow[k].Gij:=-gline;
197:         flow[k].Bii:=bline;
198:         flow[k].Bij:=-bline;
199:         flow[k].Gji:=-gline;
200:         flow[k].Bji:=-bline;
201:         flow[k].Gjj:=gline;
202:         flow[k].Bjj:=bline;
203:       end
204:       else
205:       begin
206:         // transformer is correct side accoding to *.cdf
207:         t:=1/a.Branch[j].Tr_ratio;
208:         tr_real:=t*cos(a.branch[j].Tr_ag);
209:         tr_imag:=t*sin(a.branch[j].Tr_ag);
210:         flow[k].Gii:=gline*t*t;
211:         flow[k].Gij:=(gline*tr_real + tr_imag*bline);
212:         flow[k].Bij:=(tr_real*bline - tr_imag*gline);
213:         flow[k].Bii:=(bline*t*t);
214:         flow[k].Gji:=(gline*tr_real - tr_imag*bline);
215:         flow[k].Bji:=(tr_real*bline + tr_imag*gline);
216:         flow[k].Gjj:=gline;
217:         flow[k].Bjj:=bline;
218:       end;
219:       k:=k+1;
220:     end;
221:
222:     if a.Branch[j].T_bus=i then
223:     begin
224:       flow[k].line:=j;
225:       flow[k].F_bus:=a.Branch[j].T_bus;

```

```

226:         flow[k].T_bus:=a.Branch[j].F_bus;
227:         flow[k].name:=a.Bus[i].name;
228:         flow[k].line_charg:=a.Branch[j].line_charg;
229:         flow[k].R_serie:=a.Branch[j].resistance;
230:         flow[k].X_serie:=a.Branch[j].reactance;
231:         z:=a.branch[j].resistance*a.branch[j].resistance+a.branch[j].reactan
232:         flow[k].G_serie:=a.branch[j].resistance/z;
233:         flow[k].B_serie:=-a.branch[j].reactance/z;
234:         flow[k].Bshunt:=a.Bus[i].Bshunt;
235:         gline:=flow[k].G_serie;
236:         bline:=flow[k].B_serie;
237:         if (a.branch[j].Tr_ratio=0.0) and (a.branch[j].Tr_ag=0.0) then
238:         begin
239:             flow[k].Gii:=gline;
240:             flow[k].Gij:=-gline;
241:             flow[k].Bii:=bline;
242:             flow[k].Bij:=-bline;
243:                 flow[k].Gji:=-gline;
244:                 flow[k].Bji:=-bline;
245:                 flow[k].Gjj:=gline;
246:                 flow[k].Bjj:=bline;
247:         end
248:         else
249:         begin
250:             // see the transformer from another side
251:             t:=1/a.Branch[j].Tr_ratio;
252:             tr_real:=t*cos(a.branch[j].Tr_ag);
253:             tr_imag:=t*sin(a.branch[j].Tr_ag);
254:             flow[k].Gjj:=gline*t*t;
255:             flow[k].Gji:=(gline*tr_real + tr_imag*bline);
256:             flow[k].Bji:=(tr_real*bline - tr_imag*gline);
257:             flow[k].Bjj:=bline*t*t;
258:             flow[k].Gij:=(gline*tr_real - tr_imag*bline);
259:             flow[k].Bij:=(tr_real*bline + tr_imag*gline);
260:             flow[k].Gii:=gline;
261:             flow[k].Bii:=bline;
262:         end;
263:             k:=k+1;
264:         end;
265:     end;
266: end;
267: nline:=k-1;
268: end;
269:
270: procedure Tload_flow.writeon(a:Tstringgrid);
271: var i:integer;
272: begin
273:     a.rowcount:=nline+2;
274:     for i:=1 to nline do
275:         begin
276:             if (flow[i-1].F_bus<>flow[i].F_bus) then
277:                 begin
278:                     a.cells[0,i]:=' '+inttostr(flow[i].F_bus);
279:                     a.cells[1,i]:=flow[i].name;
280:                     a.cells[2,i]:=rts(flow[i].V_mag);
281:                     a.cells[3,i]:=rts(flow[i].V_ag);
282:                     a.cells[4,i]:=rts(flow[i].GenMW);
283:                     a.cells[5,i]:=rts(flow[i].GenMvar);
284:                     a.cells[6,i]:=rts(flow[i].loadMW);
285:                     a.cells[7,i]:=rts(flow[i].loadMVAR);
286:                 end;
287:                 a.cells[8,i]:=inttostr(flow[i].T_bus);
288:                 a.cells[9,i]:=rts(100*flow[i].flowMW);
289:                 a.cells[10,i]:=rts(100*flow[i].flowMVAR);
290:                 a.cells[11,i]:=rts(100*flow[i].flowMVA);
291:             end;
292:             a.cells[1,nline+1]:='Total';
293:             a.cells[4,nline+1]:=rts(total_MW_gen);
294:             a.cells[5,nline+1]:=rts(total_MVAR_gen);
295:             a.cells[6,nline+1]:=rts(total_MW_load);
296:             a.cells[7,nline+1]:=rts(total_MVAR_load);
297:         end;
298:     end;
299: procedure Tloss_flow.getdata(a:Tdata);
300: var i,j,k:integer;

```

```

301: begin
302: k:=1;
303: for i:=1 to a.nbus do
304: begin
305:     for j:=1 to a.nbranch do
306:         begin
307:             if a.Branch[j].F_bus=i then
308:                 begin
309:                     flow[k].F_bus:=a.Branch[j].F_bus;
310:                     flow[k].T_bus:=a.Branch[j].T_bus;
311:                     flow[k].name:=a.Bus[i].name;
312:                     k:=k+1;
313:                 end;
314:             if a.Branch[j].T_bus=i then
315:                 begin
316:                     flow[k].F_bus:=a.Branch[j].T_bus;
317:                     flow[k].T_bus:=a.Branch[j].F_bus;
318:                     flow[k].name:=a.Bus[i].name;
319:                     k:=k+1;
320:                 end;
321:             end;
322:         end;
323:     nloss:=k-1;
324: end;
325:
326: procedure Tloss_flow.writeon(a:Tstringgrid);
327: var i:integer;
328: begin
329: a.rowcount:=nloss+4;
330:   for i:=1 to nloss do
331:     begin
332:       if (flow[i-1].F_bus<>flow[i].F_bus) then
333:         begin
334:           a.cells[0,i]:=inttostr(flow[i].F_bus);
335:           a.cells[1,i]:=flow[i].name;
336:           a.cells[2,i]:=rts(flow[i].V_mag);
337:           a.cells[3,i]:=rts(flow[i].V_ag);
338:           a.cells[4,i]:=rts(flow[i].GenMW);
339:           a.cells[5,i]:=rts(flow[i].GenMvar);
340:           a.cells[6,i]:=rts(flow[i].loadMW);
341:           a.cells[7,i]:=rts(flow[i].loadMVAR);
342:           a.cells[13,i]:=rts(flow[i].MVAR_charg);
343:         end;
344:         a.cells[8,i]:=inttostr(flow[i].T_bus);
345:         a.cells[9,i]:=rts(100*flow[i].lossMW);
346:         a.cells[10,i]:=rts(100*flow[i].lossMVAR);
347:         a.cells[11,i]:=rts(100*flow[i].lossMVA);
348:         a.cells[12,i]:=rts(100*flow[i].MVAR_line_charg);
349:       end;
350: a.cells[1,nloss+1]:='Total';
351: a.cells[4,nloss+2]:=rts(100*total_P);
352: a.cells[5,nloss+2]:=rts(100*total_Q);
353: a.cells[6,nloss+2]:=rts(100*total_charging);
354:   a.cells[4,nloss+1]:='total P loss';
355:   a.cells[5,nloss+1]:='total Q loss';
356:   a.cells[6,nloss+1]:='total_charging';
357: end;
358:
359: procedure Tload_flow.getoutput(a:Tcomplete_data;b:Tloss_flow);
360: var i,j,jc:integer;
361:   line_currentF,line_currentT,charging,charging_current,
362:   flow_,Ec_f,Ec_t,temp_y,Yii,Yij,Yji,Yjj,c1,c2:Tcomplex;
363:
364:   Emag_fbus,Emag_tbus,line_current_mag:real;
365: label endproc;
366: begin
367:
368: b.total_P:=0; b.total_Q:=0;
369:
370: {-----total MW MVAR load gen-----}
371: total_MW_load:=0; total_MVAR_load:=0;
372: total_MW_gen:=0; total_MVAR_gen:=0;
373: for i:=1 to a.nBus do
374: begin
375: total_MW_load:=total_MW_load+a.Bus[i].Pload;

```

```

376: total_MVAR_load:=total_MVAR_load+a.Bus[i].Qload;
377:   if (a.Bus[i].typed='PV') or (a.Bus[i].typed='SL') then
378:     begin
379:       total_MW_gen:=total_MW_gen+a.Bus[i].Pgen;
380:       total_MVAR_gen:=total_MVAR_gen+a.Bus[i].Qgen;
381:     end;
382: end;
383: {-----}
384:
385: for i:=1 to nline do
386: begin
387: with flow[i] do begin
388:   {-----common data-----}
389:   V_mag:=a.Bus[F_bus].E;
390:   V_ag:=a.Bus[F_bus].angle*dpr;
391:   GenMw:=a.Bus[F_bus].Pgen;
392:   GenMvar:=a.Bus[F_bus].Qgen;
393:   LoadMW:=a.Bus[F_bus].Pload;
394:   LoadMvar:=a.Bus[F_bus].Qload;
395:
396:   b.flow[i].V_mag:=a.Bus[F_bus].E;
397:   b.flow[i].V_ag:=a.Bus[F_bus].angle*dpr;
398:   b.flow[i].GenMw:=a.Bus[F_bus].Pgen;
399:   b.flow[i].GenMvar:=a.Bus[F_bus].Qgen;
400:   b.flow[i].LoadMW:=a.Bus[F_bus].Pload;
401:   b.flow[i].LoadMvar:=a.Bus[F_bus].Qload;
402:
403: {find line current , emag fbus}
404: Ec_f.r:=a.Bus[F_bus].E*cos(a.Bus[F_bus].angle);
405: Ec_f.i:=a.Bus[F_bus].E*sin(a.Bus[F_bus].angle);
406: Ec_t.r:=a.Bus[T_bus].E*cos(a.Bus[T_bus].angle);
407: Ec_t.i:=a.Bus[T_bus].E*sin(a.Bus[T_bus].angle);
408: temp_y.r:=-a.Y.G[F_bus,T_bus];
409: temp_y.i:=-a.Y.B[F_bus,T_bus];
410:
411:   Yii.r:=Gii;   Yii.i:=Bii;   Yij.r:=Gij;   Yij.i:=Bij;
412:   Yji.r:=Gji;   Yji.i:=Bji;   Yjj.r:=Gjj;   Yjj.i:=Bjj;
413:
414: cmult(c1,Ec_f,Yii);
415: cmult(c2,Ec_t,Yij);
416: cplus(line_currentF,c1,c2);
417: cmag(emag_fbus,Ec_f);
418:
419: cmult(c1,Ec_f,Yji);
420: cmult(c2,Ec_t,Yjj);
421: cplus(line_currentT,c1,c2);
422: cmag(emag_tbus,Ec_t);
423:
424: {-----loss flow in each line and MVAR charging-----}
425: conj(line_currentF,line_currentF);
426: cmult(flow_,line_currentF,Ec_f);
427: flowMW:=flow_.r;
428: flowMVAR:=flow_.i;
429: conj(line_currentT,line_currentT);
430: cmult(flow_,line_currentT,Ec_t);
431:
432: b.flow[i].lossMW:=flowMW+flow_.r;
433: b.flow[i].lossMVAR:=flowMVAR+flow_.i;
434: b.flow[i].lossMVA:=sqrt(b.flow[i].lossMW*b.flow[i].lossMVAR);
435: b.flow[i].F_MVAR_line_charg:=emag_fbus*emag_fbus*line_charg;
436: b.flow[i].T_MVAR_line_charg:=emag_tbus*emag_tbus*line_charg;
437: b.flow[i].MVAR_line_charg:=b.flow[i].F_MVAR_line_charg+b.flow[i].T_MVAR_line
438: b.flow[i].MVAR_charg:=emag_fbus*emag_fbus*Bshunt;
439:
440: conj(line_currentF,line_currentF);
441: conj(line_currentT,line_currentT);
442: {add MVAR charging from HVDC link}
443: j:=flow[i].F_bus;
444: if (a.Bus[j].haveconv) then
445: begin
446: jc:=a.BtonC(j);
447: b.flow[i].MVAR_charg:=b.flow[i].MVAR_charg+
448: emag_fbus*emag_fbus*a.conv[jc].Bf;
449: end;
450: {-----load flow-----}

```

```

451:   charging.r:=0;
452:   charging.i:=line_charg;
453:   cmult(charging_current, Ec_f, charging);
454:   cplus(line_currentF, charging_current, line_currentF);
455:   conj(line_currentF, line_currentF);
456:   cmult(flow_, line_currentF, Ec_f);
457:
458:   flowMW:=flow_.r;
459:   flowMVAR:=flow_.i;
460:   flowMVA:=sqrt(flowMW*flowMW+flowMVAR*flowMVAR);
461:
462:   cmult(charging_current, Ec_T, charging);
463:   cplus(line_currentT, charging_current, line_currentT);
464:   conj(line_currentT, line_currentT);
465:   cmult(flow_, line_currentT, Ec_t);
466:
467: endproc:
468:   {-----total P Q loss from each line -----}
469:   if (F_bus<T_bus) then
470:   begin
471:   cmag(line_current_mag, line_currentF);
472:   b.total_P:=b.total_P+b.flow[i].lossMW;
473:   b.total_Q:=b.total_Q+b.flow[i].lossMVA;
474:   end;
475:   end;
476:   end;
477:
478:   { total Mvar charging }
479:   b.total_charging:=0;
480:   for i:=1 to a.nBus do
481:   b.total_charging:=b.total_charging+a.Bus[i].E*a.Bus[i].E*a.Bus[i].total_Bshu
482: end;
483:
484:
485: // export to file ----- //////////
486:
487: procedure Tload_flow.exporttofile(b:Tloss_flow);
488: var i,j,k,m:integer;
489:
490: begin
491:   for i:=1 to datat.nBranch do
492:   begin
493:   export_file.branch[i].F_bus:=datat.branch[i].F_bus;
494:   export_file.branch[i].T_bus:=datat.branch[i].T_bus;
495:   j:=export_file.branch[i].F_bus;
496:   k:=export_file.branch[i].T_bus;
497:   for m:=1 to nline do
498:   begin
499:   if flow[m].F_bus=j then
500:   if flow[m].T_bus=k then
501:   begin
502:   export_file.branch[i].F_flowMW:=flow[m].flowMW;
503:   export_file.branch[i].F_flowMVAR:=flow[m].flowMVAR;
504:   export_file.branch[i].F_chargMVAR:=b.flow[m].F_MVAR_line_charg;
505:   export_file.branch[i].lossMW:=b.flow[m].lossMW;
506:   export_file.branch[i].lossMVAR:=b.flow[m].lossMVAR;
507:   export_file.bus[j].MWcompensat:=0;
508:   export_file.bus[j].MVARcompensat:=b.flow[m].MVAR_charg;
509:   end;
510:
511:   if flow[m].F_bus=k then
512:   if flow[m].T_bus=j then
513:   begin
514:   export_file.branch[i].T_flowMW:=flow[m].flowMW;
515:   export_file.branch[i].T_flowMVAR:=flow[m].flowMVAR;
516:   export_file.branch[i].T_chargMVAR:=b.flow[m].F_MVAR_line_charg;
517:   export_file.branch[i].lossMW:=b.flow[m].lossMW;
518:   export_file.branch[i].lossMVAR:=b.flow[m].lossMVAR;
519:   end;
520:   end;
521:   end;
522: end;
523:
524: end.

```

## กิตติกรรมประกาศ

ขอขอบพระคุณอย่างสูงสำหรับ รองศาสตราจารย์ ศิริวัฒน์ โปธิเวชกุล อาจารย์ที่ปรึกษา ตลอดจนอาจารย์ทุกท่านที่ได้ให้ความรู้ คำแนะนำที่เป็นประโยชน์เป็นแนวทางในการทำงานครั้งนี้ จนทำให้ปริญญาบัตรนี้สำเร็จลุล่วงได้ด้วยดี

ขอขอบคุณพี่สืบพงศ์ สุวรรณภู, พี่พัฒนา อินทนิ และพี่ๆปริญญาโททุกๆ ท่าน, เพื่อนๆ ที่ห้องโปรเจก, เพื่อนๆ ร่วมรุ่น และน้องๆ ตลอดจนทุกท่านที่ได้ให้ความอนุเคราะห์ ซึ่งถ้าปราศจากบุคคลเหล่านี้แล้ว ปริญญาบัตรฉบับนี้คงไม่สามารถสำเร็จลุล่วงได้ด้วยดี

สุดท้ายนี้ขอกราบขอบพระคุณคุณพ่อ และคุณแม่ ที่เป็นผู้สนับสนุนและให้กำลังใจเสมอมา

## หนังสืออ้างอิง

- [1] ชำนาญ ห่อเกียรติ, “การวิเคราะห์ระบบไฟฟ้ากำลัง”, สำนักพิมพ์มหาวิทยาลัยเกษตรศาสตร์
- [2] ประจตุ บุญชัยภิกษิทธิ์, วิทยานิพนธ์เรื่อง “การวิเคราะห์โหลดฟลว์ของระบบไฟฟ้ากำลัง ซึ่งประกอบด้วยระบบสายส่งกระแสตรงแรงดันสูง”, ภาควิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2528
- [3] J. Arrillaga, C.P. Arnold, B’J’ Harker, “Computer Modelling of Electrical Power System”, John Wiley & Sons, Inc., 1983
- [4] J. Arrillaga, H. Sato, “Improve Load-flow Techniques for Integrated A.C.-D.C. Systems”, PROC. IEE, Vol. 116, No. 4, 1969.
- [5] K. Prechanon, “HVDC Transmission Line Modeling and Analysis”, Asian Institute of Technology Bangkok, Thailand, 1998