การประสานสัมพันธ์แบบเหมาะที่สุดของรีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทางในระบบไฟฟ้ากำลัง

นายสมบูรณ์ทรัพย์ รอดพร

มหาวิทยาลัยเทคโนโลยีสุรนารี

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมไฟฟ้า
มหาวิทยาลัยเทคโนโลยีสุรนารี
ปีการศึกษา 2557

OPTIMAL COORDINATION OF OVER-CURRENT AND DISTANCE RELAYS IN POWER SYSTEM

Somboonsup Rodporn

A Thesis Submitted in Partial Fulfillment of the Requirements for the

Degree of Master of Engineering in Electrical Engineering

Suranaree University of Technology

Academic Year 2014

การประสานสัมพันธ์แบบเหมาะที่สุดของรีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทางในระบบไฟฟ้ากำลัง

มหาวิทยาลัยเทคโนโลยีสุรนารี อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

คณะกรรมการสอบวิทยานิพนธ์

_____

(รศ. ดร.กิตติ อัตถกิจมงคล)
ประธานกรรมการ

_____

(รศ. ดร.ธนัดชัย กุลวรวานิชพงษ์)
กรรมการ (อาจารย์ที่ปรึกษาวิทยานิพนธ์)

_____

(ผศ. ดร.อนันท์ อุ่นศิวิไลย์)
กรรมการ

_____

(ผศ. ดร.บุญเรือง มะรังศรี)
กรรมการ

_____

(ผศ. ดร.เผด็จ เผ่าละออ)
กรรมการ

_____ _____

(ศ. ดร.ชูกิจ ลิมปิจำนงค์)
รองอธิการบดีฝ่ายวิชาการและนวัตกรรม

(รศ. ร.อ. ดร.กนต์ธร ชำนิประศาสน์)
คณบดีสำนักวิชาวิศวกรรมศาสตร์

มหาวิทยาลัยเทคโนโลยีสุรนารี

สมบูรณ์ทรัพย์ รอดพร : การประสานสัมพันธ์แบบเหมาะที่สุดของรีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทางในระบบไฟฟ้ากำลัง (OPTIMAL COORDINATION OF OVER-CURRENT AND DISTANCE RELAYS IN POWER SYSTEM) อาจารย์ที่ปรึกษา : รองศาสตราจารย์ ดร.ธนัดชัย กุลวรวานิชพงษ์, 195 หน้า.

ระบบไฟฟ้ากำลังประกอบด้วย ระบบผลิต ระบบส่งจ่าย และระบบจำหน่าย ในสภาวะปกติระบบจะมีการไหลของกำลังไฟฟ้าจ่ายไปถึงโหลด ทำให้การจ่ายโหลดสามารถถูกจำกัดอยู่ที่ค่าพิกัดที่กำหนดได้ เมื่อเกิดการลัดวงจรขึ้น จะมีกระแสไหลในสายส่ง และอุปกรณ์ต่าง ๆ เป็นปริมาณสูง รีเลย์ป้องกัน (Protective relays) เป็นอุปกรณ์ที่มีหน้าที่ตรวจจับกระแสลัดวงจรในระบบและสั่งงานไปยังชุดเซอร์กิตเบรกเกอร์ (Circuit Breaker, CB) ให้ตัดวงจรส่วนที่เกิดความผิดพร่องนั้นออก เพื่อให้ระบบยังคงทำงานต่อไปได้ ในงานวิจัยนี้ จะทำการศึกษาการประสานความสัมพันธ์แบบเหมาะที่สุดของรีเลย์ป้องกันกระแสเกิน (Over Current Relay) และรีเลย์ระยะทาง (Distance Relay) ในระบบไฟฟ้ากำลัง ปัญหานี้จะอยู่ในรูปของสมการที่ไม่เชิงเส้น และมีเงื่อนไขจำนวนมาก ตัวแปรตัดสินใจของสมการประกอบด้วย ตัวคูณเวลา (Namely Time Dial Setting, *TDS*) และค่ากระแสทำงาน (Plug setting, *PS*) ของรีเลย์ป้องกันกระแสเกิน สมการฟังก์ชันวัตถุประสงค์ของปัญหาประกอบด้วยพจน์ 3 พจน์ พจน์แรกคือ ผลรวมเวลาการทำงานของรีเลย์ป้องกันกระแสเกิน พจน์ที่สองคือ ผลต่างเวลาของรีเลย์ป้องกันกระแสเกินที่ทำหน้าที่ป้องกันหลักและสำรอง และพจน์สุดท้ายหาได้จากผลต่างเวลาของรีเลย์ระยะทางที่ทำหน้าที่ป้องกันหลักและรีเลย์ป้องกันกระแสเกินที่ทำหน้าที่ป้องกันสำรอง และงานวิจัยจะทำการศึกษาระเบียบวิธีการวิวัฒนาการผลต่าง (Differential Evolution, DE) เพื่อนำมาใช้แก้หาผลเฉลยของปัญหาดังกล่าว DE เป็นวิธีการหาค่าเหมาะสมแบบสโตคาสติก มีพื้นฐานการหาผลเฉลยโดยวิธีสุ่มเลือก ซึ่งพัฒนามาจาก Genetic Algorithm (GA) แต่มีโครงสร้างที่ไม่ซับซ้อน สามารถประยุกต์ใช้หาผลเฉลยของปัญหาที่มีตัวแปรจำนวนมากได้รวดเร็ว และมีประสิทธิภาพ ระบบทดสอบมาตรฐานที่ใช้ในการศึกษาประกอบด้วย IEEE 9 bus, IEEE 14 bus และ IEEE 30 bus โดยเปรียบเทียบผลเฉลยค่าเหมาะที่สุดที่ได้กับวิธีการหาผลเฉลยอื่น

สาขาวิชา วิศวกรรมไฟฟ้า ลายมือชื่อนักศึกษา_____

ปีการศึกษา 2557 ลายมือชื่ออาจารย์ที่ปรึกษา_____

ลายมือชื่ออาจารย์ที่ปรึกษาร่วม_____

SOMBOONSUP RODPORN : OPTIMAL COORDINATION OF OVER-CURRENT AND DISTANCE RELAYS IN POWER SYSTEM. THESIS ADVISOR : ASSOC. PROF. THANATCHAI KULWORAWANICHPONG, Ph.D., 195 PP.

DIFFERENTIAL EVOLUTION / OPTIMAL COORDIANTION OVER-CURRENT RELAY / OPTIMAL COMBINED OVER-CURRENT AND DISTANCE RELAYS

In electrical power system during normal stage, the power was transferred from an generator to load. When the fault occurs, the relay can detect the fault currents and order the circuit breaker to cut out the fault from the system. However, the other parts still work normally. The objective of this thesis was to study the optimal coordination of over-current, (OC) and distance, (DIS) relays. This problem was set as the non-linear equation under various conditions. The important factors were a namely time dial setting, (TDS) and Plug setting, (PS) of the over-current relays. The objective function equation composed of the summation time of OC relays, the time different between the main OC and backup OC relays and the time different between main DIS and backup OC relays. The Differential Evolution (DE) was studied and used to solve the problem in this work. DE was a random method developing from Genetic Algorithm (GA). DE is easier than GA leading to a fast and

efficient solving method. Finally, the optimal coordination of OC and DIS relays in power system was studied on the IEEE 9 bus, IEEE 14 bus and IEEE 30 bus.

มหาวิทยาลัยเทคโนโลยีสุรนารี

School of Electrical Engineering Student's Signature_____

Academic Year 2014 Advisor's Signature_____

Co - Advisor's Signature_____

กิตติกรรมประกาศ

วิทยานิพนธ์นี้สำเร็จลุล่วงด้วยดี เนื่องจากได้รับความช่วยเหลืออย่างดียิ่ง ทั้งด้านวิชาการและด้านการดำเนินงานวิจัย จากบุคคลและกลุ่มบุคคลต่างๆ ได้แก่

รองศาสตราจารย์ ดร.ธนัดชัย กุลวรวานิชพงษ์ อาจารย์ที่ปรึกษาวิทยานิพนธ์และผู้ช่วยศาสตราจารย์ ดร.อนันท์ อุ่นศิวิไลย์ อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม ที่ได้ให้คำปรึกษา แนะนำ และแนะแนวทางอันเป็นประโยชน์ยิ่งต่องานวิจัย รวมถึงได้ช่วยตรวจทาน และแก้ไขรายงานวิทยานิพนธ์เล่มนี้จนทำให้มีความสมบูรณ์ยิ่งขึ้น รวมทั้งเป็นกำลังใจ และเป็นแบบอย่างที่ดีในการดำเนินชีวิตหลาย ๆ ด้านให้กับผู้วิจัยเสมอมา

อาจารย์ประจำสาขาวิชาวิศวกรรมไฟฟ้า มหาวิทยาลัยเทคโนโลยีสุรนารีทุกท่าน ที่ได้กรุณาให้คำปรึกษา แนะนำ และความรู้ทางด้านวิชาการอย่างดียิ่งมาโดยตลอด

ขอขอบคุณนางสาวชนากานต์ ชลสุข เพื่อนที่ช่วยตรวจทานคำผิดและรูปแบบการจัดพิมพ์เล่มวิทยานิพนธ์จนเสร็จสมบูรณ์ เรียบร้อยสวยงาม

ขอบคุณ พี่ ๆ เพื่อน ๆ น้อง ๆ บัณฑิตศึกษาทุกท่าน รวมถึงมิตรสหายทั้งในอดีตและปัจจุบันที่คอยถามไถ่และให้กำลังใจในการทำวิจัยมาโดยตลอด

สุดท้ายนี้ ผู้วิจัยขอขอบคุณอาจารย์ผู้สอนทุกท่านที่ประสิทธิ์ประสาทความรู้ทางด้านต่าง ๆ ทั้งในอดีตและปัจจุบัน และขอกราบขอบพระคุณ บิดา มารดา รวมถึงญาติพี่น้องของผู้วิจัยทุกท่านที่ได้ให้ความรัก ความอบอุ่น ความห่วงใย การอบรมเลี้ยงดู และให้การสนับสนุนทางด้านการศึกษาอย่างดียิ่งมาโดยตลอด รวมทั้งเป็นกำลังใจที่ยิ่งใหญ่ในยามที่ผู้วิจัยท้อและทุกข์ใจ ช่วยให้มีพลังเข้มแข็งพร้อมเผชิญกับปัญหาอุปสรรคต่าง ๆ จนทำให้ผู้วิจัยประสบความสำเร็จในชีวิตเรื่อยมา

สมบูรณ์ทรัพย์ รอดพร

สารบัญ

หน้า

บทคัดย่อ (ภาษาไทย) ..... ก
บทคัดย่อ (ภาษาอังกฤษ) ..... ข
กิตติกรรมประกาศ ..... ง
สารบัญ ..... จ
สารบัญตาราง ..... ซ
สารบัญรูป ..... ญ
คำอธิบายสัญลักษณ์และคำย่อ ..... ฏ

บทที่

1 บทนำ 1
1.1 ความสำคัญและที่มาของปัญหา 1
1.2 วัตถุประสงค์ของงานวิจัย 2
1.3 ข้อตกลงเบื้องต้น 2
1.4 ขอบเขตงานวิจัย 2
1.5 ประโยชน์ที่คาดว่าจะได้รับ 4
1.6 การจัดรูปเล่มวิทยานิพนธ์ 4
2 ปริทัศน์วรรณกรรมและทฤษฎีที่เกี่ยวข้อง 4
2.1 บทนำ 4
2.2 ปริทัศน์วรรณกรรม 4
2.3 รีเลย์ป้องกัน 7
2.3.1 รีเลย์ระยะทาง 7
2.3.2 รีเลย์ป้องกันกระแสเกิน 9
2.3.3 การปรับตั้งและการประสานสัมพันธ์รีเลย์ 17
1. รีเลย์ระยะทาง 17
2. รีเลย์ป้องกันกระแสเกิน 26

สารบัญ (ต่อ)

หน้า

2.4 การหาค่าเหมาะที่สุด 34
2.4.1 วิธีเชิงพันธุกรรม (GA) 34
2.4.2 วิธีวิวัฒนาการผลต่าง (DE) 37
2.4.3 วิธีคล้ายนิวตัน (BFGS) 40
2.5 สรุป 41
3 ปัญหาการจัดความสัมพันธ์และการหาผลเฉลยของปัญหา 42
3.1 บทนำ 42
3.2 ปัญหาการจัดความสัมพันธ์ 42
3.2.1 รีเลย์ป้องกันกระแสเกินกับรีเลย์ป้องกันกระแสเกิน 42
3.2.2 จัดความสัมพันธ์รีเลย์ป้องกันกระแสเกินกับรีเลย์ระยะทาง 44
3.3 การแก้ปัญหาการประสานสัมพันธ์รีเลย์ 47
3.3.1 รีเลย์ป้องกันกระแสเกินกับรีเลย์ป้องกันกระแสเกิน 47
3.3.2 จัดความสัมพันธ์รีเลย์ป้องกันกระแสเกินกับรีเลย์ระยะทาง 51
3.4 สรุป 52
4 ผลการทดสอบ 53
4.1 บทนำ 53
4.2 ค่าปรับตั้งรีเลย์ป้องกันระยะทาง 53
4.2.1ระบบทดสอบมาตรฐาน WSCC 9 บัส 53
4.2.2 ระบบทดสอบมาตรฐาน IEEE 14 บัส 56
4.2.3 ระบบทดสอบมาตรฐาน IEEE 30 บัส 60
4.3 ค่าเหมาะที่สุดการประสานสัมพันธ์รีเลย์ป้องกระแสเกิน 67
4.3.1ระบบทดสอบมาตรฐาน WSCC 9 บัส 67
4.3.2 ระบบทดสอบมาตรฐาน IEEE 14 บัส 70
4.3.3 ระบบทดสอบมาตรฐาน IEEE 30 บัส 71
4.4 ค่าเหมาะที่สุดการประสานสัมพันธ์รีเลย์ป้องกระแสเกิน 73

สารบัญ (ต่อ)

หน้า

4.4.1ระบบทดสอบมาตรฐาน WSCC 9 บัส 73
4.4.2 ระบบทดสอบมาตรฐาน IEEE 14 บัส 75
4.4.3 ระบบทดสอบมาตรฐาน IEEE 30 บัส 76
4.5 ค่าผลต่างเวลาการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทาง 78
4.6 สรุป 90
5 บทสรุปและข้อเสนอแนะ 91
5.1 บทนำ 91
5.2 สรุป 91
5.3 ข้อเสนอแนะ 92
รายการอ้างอิง 93
ภาคผนวก
ภาคผนวก ก. ระบบทดสอบ 95
ภาคผนวก ข. ผลการจำลองกระแสลัดวงจรจากโปรแกรม Power world Simulator 106
ภาคผนวก ค. MATLAB CODE 127
ภาคผนวก ง. บทความทางวิชาการที่ได้รับการตีพิมพ์เผยแพร่ในระหว่างศึกษา 180
ประวัติผู้เขียน 195

สารบัญตาราง

ตารางที่ **หน้า**

3.1 ค่ากระแสโหลดที่ไหลผ่านรีเลย์....50
3.2 ค่ากระแสลัดวงจร....50
3.3 ค่ากระแสลัดวงจรของเมื่อพิจารณาการทำงานรีเลย์หลักและสำรอง....51
4.1 ผลการจัดโซนป้องกันรีเลย์ระยะทาง ระบบ 9 บัส....54
4.2 ค่าปรับตั้งรีเลย์ระยะทาง ระบบ 9 บัส....55
4.3 ผลการจัดโซนป้องกันรีเลย์ระยะทาง ระบบ 14 บัส....57
4.4 ค่าปรับตั้งรีเลย์ระยะทาง ระบบ 14 บัส....58
4.5 ผลการจัดโซนป้องกันรีเลย์ระยะทาง ระบบ 30 บัส....61
4.6 ค่าปรับตั้งรีเลย์ระยะทาง ระบบ 30 บัส....63
4.7 ค่าการค้นหาคำตอบของ DE ระบบ 9 บัส....67
4.8 ค่าการค้นหาคำตอบของ GA ระบบ 9 บัส....67
4.9 ค่าการค้นหาคำตอบของ BFGS ระบบ 9 บัส....67
4.10 ผลการจัดความสัมพันธ์รีเลย์ป้องกันกระแสเกิน ระบบ 9 บัส....68
4.11 ค่าการค้นหาคำตอบของ DE ระบบ 14 บัส....70
4.12 ผลการจัดความสัมพันธ์รีเลย์ป้องกันกระแสเกิน ระบบ 14 บัส....70
4.13 ค่าการค้นหาคำตอบของ DE ระบบ 30 บัส....71
4.14 ผลการจัดความสัมพันธ์รีเลย์ป้องกันกระแสเกิน ระบบ 30 บัส....71
4.15 ค่าการค้นหาคำตอบของ DE ในระบบ 9 บัส....73
4.16 ค่าการค้นหาคำตอบของ GA ในระบบ 9 บัส....74
4.17 ค่าการค้นหาคำตอบของ BFGS ในระบบ 9 บัส....74
4.18 ผลการจัดความสัมพันธ์รีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทาง ระบบ 9 บัส....74
4.19 ค่าการค้นหาคำตอบของ DE ในระบบ 14 บัส....75
4.20 ผลการจัดความสัมพันธ์รีเลย์ระยะทางและรีเลย์ป้องกันกระแสเกิน ในระบบ 14 บัส....75
4.21 ค่าการค้นหาคำตอบของ DE ในระบบ 30บัส....76

สารบัญตาราง (ต่อ)

ตารางที่ **หน้า**

4.22 ผลการจัดความสัมพันธ์รีเลย์ระยะทางและรีเลย์ป้องกันกระแสเกิน ในระบบ 30 บัส.....77
4.23 ผลต่างเวลาการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและระยะทาง (9 บัส) 79
4.24 ผลต่างเวลาการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและระยะทาง (14บัส) 79
4.25 ผลต่างเวลาการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและระยะทาง (30บัส) 89
4.26 จำนวนคู่ประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทาง (9 บัส) 89
4.27 จำนวนคู่ประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทาง (14 บัส) 90

มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญรูป

รูปที่ **หน้า**

2.1 การจ่ายโหลดทางเดียว....7
2.2 การหาขนาดความต้านทานลัดวงจร....8
2.3 แผนภาพอิมพีแดนซ์รีเลย์....8
2.4 แผนภาพรีเลย์โมห์....8
2.5 อิมพีแดนซ์ของรีเลย์รูปแบบต่าง....9
2.6 คุณลักษณะการทำงานของรีเลย์กระแสเกิน....9
2.7 เส้นโค้งเวลาและกระแสของรีเลย์....11
2.8 ตำแหน่งการเรียกชื่อของรีเลย์เมื่อเกิดความผิดพร่อง....11
2.9 เส้นโค้งเวลาและกระแสของรีเลย์ป้องกันแบบมาตรฐาน....13
2.10 มุมเฟสของกระแสและแรงดัน....14
2.11 ระบบไฟฟ้ากำลังแบบใช้สายส่ง 2 เส้นขนานกัน....15
2.12 ระยะป้องกันของรีเลย์ระยะทาง....17
2.13 ตัวอย่างการปรับตั้งค่ารีเลย์....18
2.14 ตัวอย่างที่ 2 การปรับตั้งค่ารีเลย์....19
2.15 ระบบไฟฟ้ากำลังที่มีแหล่งจ่ายหลายชุด....21
2.16 ตัวอย่างระบบไฟฟ้ากำลังที่มีแหล่งจ่ายหลายชุด....22
2.17 ตัวอย่างที่ 3 การปรับตั้งค่ารีเลย์....23
2.18 เส้นโค้งเวลาทำงานของรีเลย์....27
2.19 ระบบป้องกันตัวอย่างที่ 2.5....28
2.20 ลำดับความสัมพันธ์การทำงานของรีเลย์ป้องกันกระแสเกิน....29
2.21 ระบบป้องกันตัวอย่างสำหรับ ตัวอย่างที่ 2.6....30
2.22 วงจรประกอบการคำนวณกระแสลัดวงจร....30
2.23 เส้นโค้งการจัดลำดับของรีเลย์....33
2.24 แผนผังการทำงานของวิธีเชิงพันธุกรรม....36

สารบัญรูป (ต่อ)

รูปที่ **หน้า**

2.25 กระบวนการทำงานของวิวัฒนาการผลต่าง....38
2.26 กระบวนการ crossover ระหว่าง target vector
และ mutant vector ที่มีตัวแปรการตัดสินใจ (D) เท่ากับ 7....39
3.1 ความสัมพันธ์การทำงานระหว่างรีเลย์ป้องกันกระแสเกินกับรีเลย์ระยะทาง....44
3.2 ความสัมพันธ์การทำงานระหว่างรีเลย์ป้องกันกระแสเกินกับรีเลย์ระยะทาง....45
3.3 แผนผังการแก้ปัญหาค่าเหมาะที่สุด....47
3.4 การติดตั้งรีเลย์ในระบบทดสอบ WSCC 9 บัส....48
3.5 การจำลองระบบจ่ายโหลดในสภาวะปกติ (9 บัส)....49
3.6 การจำลองการลัดวงจรด้วย PowerWorld Simulator (9 บัส)....49
4.1 ระบบทดสอบมาตรฐาน WSCC 9 บัส....54
4.2 ระบบทดสอบมาตรฐาน IEEE 14 บัส....56
4.3 ระบบทดสอบมาตรฐาน IEEE 30 บัส....60
4.4 ผลการลู่เข้าหาคำตอบของวิธีการ DE ระบบทดสอบ 9 บัส....69
4.5 ผลการลู่เข้าหาคำตอบของวิธีการ GAระบบทดสอบ 9 บัส....69
4.6 ผลการลู่เข้าหาคำตอบของวิธีการ BFGS ระบบทดสอบ 9 บัส....69

คำอธิบายสัญลักษณ์และคำย่อ

M	=	Primary Relay
B	=	Backup Relay
$t_b(F)$	=	เวลาเริ่มทำงานของรีเลย์ป้องกันกระแสเกิน(รีเลย์สำรอง)
t_{Z2}	=	เวลาเริ่มทำงานโซน2 ของรีเลย์ป้องกันระยะทาง
CTI	=	ค่าเผื่อผลต่างเวลาของการจัดลำดับรีเลย์
PS	=	ค่าตัวคูณเผื่อกระแสโหลด
TDS	=	ค่าความชันเส้นโค้งเวลาของรีเลย์ป้องกันกระแสเกิน
N_{cl}	=	จำนวนของรีเลย์ที่อยู่ใกล้จุดลัดวงจร
N_{far}	=	จำนวนของรีเลย์ที่อยู่ไกลจุดลัดวงจร
$T^{i}_{pri_cl_in}$	=	เวลาเริ่มทำงานของรีเลย์ที่อยู่ใกล้จุดลัดวงจร
$T^{i}_{pri_far_bus}$	=	เวลาเริ่มทำงานของรีเลย์ที่อยู่ไกลจุดลัดวงจร
T_{backup}	=	เวลาการทำงานของรีเลย์ที่ทำหน้าที่เป็นรีเลย์สำรอง(backup)
$T_{primary}$	=	เวลาการทำงานของรีเลย์ที่ทำหน้าที่เป็นรีเลย์หลัก (primary)
$\alpha_{1,}\beta_{2,}\beta_{3}$	=	ตัวประกอบถ่วงน้ำหนัก
$\Delta t_{mb\|k_1\|}$	=	ผลต่างเวลาทำงานของรีเลย์ป้องกันกระแสเกิน
$\Delta t_{mbDISOC\|k_2\|}$	=	ผลต่างเวลาการทำงานรีเลย์ระยะทาง(หลัก)กับรีเลย์ป้องกันกระแสเกิน (สำรอง)
$t_{bOC\|k_2\|}$	=	เวลาทำงานของรีเลย์ป้องกันกระแสเกิน(สำรอง) ขณะเกิดลัดวงจรที่ ปลายสาย
$t_{mDIS\|k_2\|}$	=	เวลาการทำงานรีเลย์ระยะทาง(หลัก) ทำหน้าที่ป้องกันโซน2
DE	=	Differential Evolution , วิวัฒนาการผลต่าง
GA	=	Genetic Algorithm, วิธีเชิงพันธุกรรม
BFGS	=	Broyden–Fletcher–Goldfarb–Shanno,วิธีคล้ายนิวตัน
R	=	ความต้านทาน (โอห์ม)
X	=	รีแอกแตนซ์ (โอห์ม)

บทที่ 1
บทนำ

1.1 ความสำคัญของปัญหา

ในระบบไฟฟ้ากำลังประกอบด้วย ระบบผลิต ระบบส่งจ่าย และระบบจำหน่าย ในสภาวะปกติระบบจะมีการไหลของกำลังไฟฟ้าจ่ายไปถึงโหลด ทำให้การจ่ายโหลดสามารถถูกจำกัดอยู่ที่ค่าพิกัดที่กำหนดได้ ถ้าโหลดเพิ่มขึ้นอย่างต่อเนื่องจนเกินค่าพิกัดดังกล่าว อาจจะยอมให้เกิดการทำงานในสภาวะโหลดเกินได้ในช่วงเวลาสั้นๆ ค่าหนึ่ง ถ้าเกินเวลาที่กำหนดแล้วโหลดยังอยู่ในสภาวะโหลดเกินอยู่ อุปกรณ์ป้องกันต้องทำการปลดวงจรในส่วนนั้นออกเพื่อป้องกันไม่ให้เกิดความเสียหายกับอุปกรณ์อื่นๆ เรียกการป้องกันลักษณะนี้ว่า การป้องกันโหลดเกิน (Overload protection) ส่วนการป้องกันอีกลักษณะหนึ่งคือ การป้องการลัดวงจร (Short Circuit Protection) เมื่อเกิดการลัดวงจรขึ้นจะมีกระแสไหลในสายส่ง และอุปกรณ์ต่างๆ เป็นปริมาณสูง อุปกรณ์ป้องกันจะตรวจพบกระแสลัดวงจรนี้ และทำหน้าที่ตัดส่วนของระบบที่เกิดความผิดพร่องนี้ออกไป โดยส่วนที่ไม่เกิดความผิดพร่องยังคงทำงานได้ตามปกติ รีเลย์ป้องกัน (Protective relays) เป็นอุปกรณ์ที่มีความสำคัญของระบบป้องกัน มีหน้าที่ตรวจจับกระแสลัดวงจรในระบบและสั่งงานไปยังเซอร์กิตเบรกเกอร์ (CB) เพื่อตัดวงจรส่วนที่เกิดความผิดพร่องนั้นออกไป ในงานวิจัยนี้จะได้ทำการศึกษารีเลย์ป้องกัน 2 ชนิด คือ รีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทาง เนื่องจากรีเลย์ทั้งสองชนิดนี้มีความสำคัญในการป้องกันสายส่งในระบบส่งจ่ายกำลังไฟฟ้า และจะได้ทำการจัดความสัมพันธ์ระหว่างรีเลย์ทั้งสองชนิด เพื่อให้รีเลย์ทั้งสองชนิดทำงานร่วมกันอย่างมีประสิทธิภาพสูงสุด

1.2 วัตถุประสงค์การวิจัย

1.2.1 เพื่อศึกษาการประสานสัมพันธ์แบบเหมาะที่สุดของการป้องกันระบบไฟฟ้ากำลัง

- รีเลย์ระยะทาง
- รีเลย์ป้องกันกระแสเกิน
- รีเลย์ระยะทางและรีเลย์ป้องกันกระแสเกิน

1.2.2 เพื่อนำขั้นตอนวิธีวิวัฒนาการผลต่าง (Differential Evolution) มาประยุกต์ใช้กับปัญหาที่ทำการศึกษาได้

1.3 ข้อตกลงเบื้องต้น

1.3.1 ใช้โปรแกรม MATLAB ในการคำนวณ

1.3.2 ใช้โปรแกรม Power World Simulation ในการหาค่ากระแสลัดวงจรและกระแสโหลดของระบบทดสอบ

1.3.3 ไม่คิดผลของกำลังงานสูญเสียในสายส่ง

1.3.4 รีเลย์ที่ใช้เป็นดิจิตอลรีเลย์ (สามารถปรับค่าเป็นทศนิยมได้)

1.3.5 กำหนดให้ความต้านทานลัดวงจรมีค่าประมาณศูนย์

1.3.6 ใช้ชุดคำสั่งโปรแกรมวิวัฒนาการผลต่าง (Differential Evolution, DE) จากเวปไซต์ http://www.icsi.berkeley.edu/~storn/code.html

1.4 ขอบเขตของงานวิจัย

1.4.1 ศึกษาและวิเคราะห์การป้องกันด้วยรีเลย์ป้องกันกระแสเกิน (Over-Current Relay) ในระบบ WSCC 9 บัส IEEE 14 บัส และ IEEE 30 บัส

1.4.2 ศึกษาและวิเคราะห์การป้องกันด้วยรีเลย์ระยะทาง (Distance Relay) ในระบบ WSCC 9 บัส IEEE 14 บัส และ IEEE 30 บัส

1.4.3 ศึกษาและวิเคราะห์การประสานสัมพันธ์ของรีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทาง ในระบบ WSCC 9 บัส IEEE 14 บัส และ IEEE 30 บัส

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1.5.1 สามารถจัดความสัมพันธ์รีเลย์ป้องกันกระแสเกินกับรีเลย์ป้องกันระยะทางได้อย่างมีประสิทธิภาพได้ คือ

1. รีเลย์ทำงานได้เร็ว
2. รีเลย์ทำงานไม่ผิดพลาด

1.5.2 สามารถนำไปประยุกต์ใช้ระเบียบวิธีการค้นหาค่าเหมาะสุด (Differential Evolution, DE) กับระบบที่ซับซ้อนขึ้นหรือระบบที่ใช้งานจริงได้

1.5.3 ช่วยให้เข้าใจกระบวนการทำงานของขั้นตอนระเบียบวิธีการค้นหาค่าเหมาะสุด (Differential Evolution, DE) ชนิดใหม่ได้ดียิ่งขึ้น

1.6 การจัดรูปเล่มวิทยานิพนธ์

วิทยานิพนธ์เล่มนี้ประกอบด้วย 5 บท และ 4 ภาคผนวก ดังนี้

บทที่ 1 เป็นบทนำกล่าวถึงความสำคัญของปัญหา วัตถุประสงค์ ข้อตกลงเบื้องต้น ขอบเขตของงานวิจัย และประโยชน์ที่คาดว่าจะได้รับจากการวิจัย รวมทั้งแนะนำเนื้อหาเบื้องต้นของวิทยานิพนธ์ฉบับนี้ ส่วนบทอื่น ๆ ประกอบด้วย เนื้อหาดังต่อไปนี้

บทที่ 2 กล่าวถึงปริทัศน์วรรณกรรมและทฤษฎีที่เกี่ยวข้อง

บทที่ 3 กล่าวถึงปัญหาการจัดความสัมพันธ์และการหาผลเฉลยของปัญหา

บทที่ 4 กล่าวถึงผลการทดสอบ

บทที่ 5 กล่าวถึงบทสรุปและข้อเสนอแนะต่าง ๆ

ภาคผนวก ก. รายละเอียดของระบบทดสอบต่าง ๆ ที่ใช้ในวิทยานิพนธ์นี้

ภาคผนวก ข. ผลการจำลองกระแสลัดวงจรจากโปรแกรม Power World Simulator

ภาคผนวก ค. MATLAB Code

ภาคผนวก ง. บทความที่ได้รับการตีพิมพ์เผยแพร่ในขณะศึกษา

มหาวิทยาลัยเทคโนโลยีสุรนารี

บทที่ 2
ปริทัศน์วรรณกรรมและทฤษฎีที่เกี่ยวข้อง

2.1 บทนำ

เนื้อหาในบทนี้กล่าวถึงปริทัศน์วรรณกรรมโดยจะสรุปย่อเกี่ยวกับงานวิจัยของนักวิจัยต่าง ๆ ที่ทำการวิจัยเกี่ยวกับการประสานสัมพันธ์รีเลย์แบบเหมาะที่สุดที่ผ่านมา ซึ่งปริทัศน์วรรณกรรมนี้เป็นส่วนหนึ่งของงานวิจัยที่เคยปรากฏมาก่อนแล้วเท่านั้น ทั้งนี้เพื่อให้เกิดความเข้าใจพื้นฐานเกี่ยวกับทฤษฎีที่เกี่ยวข้องงานวิจัยนี้ทั้งหมด และจะกล่าวถึงทฤษฎีบทต่างๆ ที่เกี่ยวของกับงานวิจัยนี้ คือ การป้องกันระบบไฟฟ้าด้วยรีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทาง ระเบียบวิธีการที่จะใช้หาคำตอบของปัญหาการประสานสัมพันธ์รีเลย์

2.2 ปริทัศน์วรรณกรรม

จากการศึกษาปริทัศน์วรรณกรรมที่เกี่ยวข้อง สามารถสรุปทฤษฎี หลักการ และการดำเนินงานวิจัยต่าง ๆ ที่ใช้ในการศึกษาการป้องกันระบบส่งจ่ายกำลังไฟฟ้าด้วยด้วยรีเลย์ระยะทาง รีเลย์ป้องกันกระแสเกิน และการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทาง โดยใช้วิธีหาค่าเหมาะที่สุด เช่น วิธี GA วิธี Linear Programming วิธี PSO และ วิธี DE

2.2.1 ปัญหาการจัดความสัมพันธ์รีเลย์ระยะทาง

Ravikumar (2008) ได้ศึกษาวิธีการป้องกันสายส่งด้วยรีเลย์ระยะทางแบบใหม่ จากเดิมที่เป็นแบบออนไลน์เป็นการให้รีเลย์ประจำตำแหน่งนั้น ๆ อ่านค่าอิมพีแดนซ์ที่มีการเปลี่ยนแปลง ซึ่งวิธีนี้จะทำให้การป้องกันมีความรวดเร็วและความยืดหยุ่น และได้ทำการประสานสัมพันธ์ของรีเลย์ระยะทางด้วยเทคนิค Support Vector Machines (SVMs) โดยทำการทดสอบกับระบบทดสอบ 9 บัส

Sidhu (2004) ได้ศึกษาปรับปรุงการหาค่าปรับตั้งรีเลย์ระยะทางโซน 2 โดยเพิ่มการพิจารณาผลของเครื่องกำเนิดไฟฟ้าเข้าไปจากเดิมที่คิดเฉพาะผลของอิมพีแดนซ์ของสายส่งแต่ละโซน และหาคำตอบด้วยวิธี Adaptive Setting ทำการทดลองในระบบทดสอบ 43 บัส ผลการศึกษาพบว่าสมการการปรับตั้งโซน 2 ใหม่ให้ผลการปรับตั้งที่ดีกว่าสมการเดิม

2.2.2 ปัญหาการจัดความสัมพันธ์รีเลย์ป้องกันกระแสเกิน

Razavi (2008) ได้ทำการศึกษาการจัดความสัมพันธ์รีเลย์ป้องกันกระแสเกินโดยทำการปรับปรุงฟังก์ชั่นวัตถุประสงค์เดิมที่ให้ผลการประสานสัมพันธ์รีเลย์ที่เกิดความผิดพลาดในการจับคู่รีเลย์หลักและสำรอง ได้ทำการพัฒนาสมการใหม่โดยการเพิ่มการปรับโทษเมื่อผลต่างเวลาของการประสานมีค่าน้อยกว่าศูนย์ เข้าไปในสมการวัตถุประสงค์ ทำให้ได้สมการฟังก์ชั่นวัตถุประสงค์ใหม่หลังจากนั้น ทำการหาผลเฉลยของสมการด้วยวิธี GA ผลที่ได้สามารถประสานสัมพันธ์รีเลย์ได้สำเร็จ คือ สามารถจับคู่รีเลย์ได้ครบ

Bashir (2010) ทำการศึกษาการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินในระบบไฟฟ้าด้วยเทคนิค A New Hybrid Particle Swarm Optimization, PSO โดยมีฟังก์ชั่นการประสานสัมพันธ์รีเลย์เป็นผลรวมเวลาการทำงานของรีเลย์ของระบบทดสอบคูณกับค่าตัวคูณถ่วงน้ำหนัก หลังจากนั้นได้ทำการปรับปรุงการสมการลู่เข้าของ PSO แล้วจึงนำมาหาผลเฉลยของการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกิน เพื่อเปรียบเทียบกับผลเฉลยที่ได้เทียบกับผลที่ได้จากวิธี Linear Programming และ GA สรุปผลที่ได้คือ PSO สามารถให้ค่าฟังก์ชั่นต่ำสุดเหมือนอีกสองวิธี แต่มีการลู่เข้าที่เร็วกว่า

Barzegari (2011) ทำการศึกษาการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินในระบบไฟฟ้าด้วยวิธี Linear Programming เมื่อพิจารณาผลของเครื่องกำเนิดไฟฟ้าพลังงานลมขณะเปิดและปิดที่ตำแหน่งบัสต่าง ๆ บนระบบทดสอบ 8 บัส โดยใช้สมการวัตถุประสงค์ดังสมการที่ ซึ่งพบปัญหาการจับคู่ประสานสัมพันธ์บางกรณีไม่สามารถทำได้ จึงแก้ไขด้วยการเพิ่มค่า *CTI* จาก 0.25 เป็น 0.4 จึงสามารถทำการประสานสัมพันธ์ได้

Ralhan (2013) ได้ศึกษาการประสานสัมพันธ์รีเลย์ป้องกันระยะทางด้วยวิธี Linear Programming Intervals 3 แบบ คือ Interval Simplex Interval Two Phase Simplex และ Interval Revised Simplex เพื่อเปรียบเทียบผลเฉลยของระบบทดสอบ 3 บัส และ 6 บัส ผลคือ ทั้ง 3 วิธีมีค่าผลเฉลยที่ได้ไม่ต่างกันมาก และเวลาที่ใช้การหาคำเรียงจากน้อยไปมาก คือ Revised Simplex Revised Simplex และ Simple

2.2.3 ปัญหาการจัดความสัมพันธ์รีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทาง

Abyaneh (2008)ได้ทำการศึกษาการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทาง โดยใช้ฟังก์ชันวัตถุประสงค์ใหม่ดังสมการที่ และหาผลเฉลยด้วยระเบียบวิธีการ GA เพื่อทำการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินกับรีเลย์ระยะทางแบบเหมาะที่สุด ทำการ

ทดสอบกับระบบ 6 บัส ผลการทดสอบสามารถประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทางได้

Reza (2012) ได้ศึกษาการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทาง โดยใช้สมการเหมือน Abyaneh แต่ได้ปรับปรุงผลเฉลยจนได้ผลที่ดีกว่าและได้ทดสอบกับระบบทดสอบ 6 บัส และ 30 บัส ด้วยวิธี GA

2.2.4 วิธีหาค่าเหมาะสมที่สุดด้วยวิธีวิวัฒนาการผลต่าง (Differential Evolution,DE)

Chittaladakorn (2005) ในการศึกษานี้ได้ประยุกต์ใช้ DE ในการพัฒนาโปรแกรมเพื่อหาขนาดท่อที่เหมาะสมสำหรับการออกแบบระบบท่อจ่ายน้ำ โดยเชื่อมโยงกับ โปรแกรม EPANET ซึ่งเป็นแบบจำลองคณิตศาสตร์สำหรับวิเคราะห์ทางชลศาสตร์ของการไหลในระบบท่อโครงข่าย โดยใช้ toolkit ของตัวโปรแกรมและทดสอบโปรแกรมที่สร้างขึ้นกับระบบท่อประปาจังหวัดนครสวรรค์ เพื่อเปรียบเทียบผลที่ได้กับการออกแบบระบบท่อประปาด้วยวิธีดั้งเดิมการออกแบบระบบท่อจ่ายน้ำอย่างเหมาะสมมีฟังก์ชันวัตถุประสงค์ คือ หาขนาดท่อที่เหมาะสมโดยการใช้ค่าลงทุนและค่าดำเนินการต่ำที่สุดภายได้ข้อกำหนด (Penalty Cost) ดังสมการที่ จากการนำวิธี DE มาใช้แก้หาผลเฉลยพบว่าสามารถหาคำตอบได้รวดเร็วและแม่นยำเมื่อเทียบกับวิธีSimulated Annealing (SA)

Thangaraj (2010)ได้ศึกษาการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินชนิดมีทิศทาง โดยมีฟังก์ชันวัตถุประสงค์ที่ประกอบด้วยผลรวมเวลาทำงานของรีเลย์ที่อยู่ใกล้และไกลตำแหน่งลัดวงจร ดังสมการที่ และได้นำเอาวิธีการ DE มาใช้ฟังก์ชันวัตถุประสงค์เพื่อหาค่าต่ำสุดในการปรับตั้งค่ารีเลย์กระแสเกินแบบมีทิศทาง โดยได้มีการปรับปรุงในส่วนของการ Mutation $V_{i,g+1} = V_{r1,g+1},...,V_{D,i,g+1}$ โดยมีสมการ ดังนี้ ระบบที่นำมาหาผลเฉลยคือ 3 บัส 4 บัส และ 6 บัส ซึ่งสามารถทำให้กระบวนการมีการลู่เข้าหาค่าต่ำสุดได้เร็วขึ้นและมีประสิทธิภาพมากขึ้นด้วย

2.3 รีเลย์ป้องกัน

ในระบบไฟฟ้ากำลังประกอบด้วย ระบบผลิต ระบบส่งจ่าย และระบบจำหน่าย ระบบส่งจ่ายเป็นส่วนสำคัญที่มีความเสี่ยงเกิดความเสียหาย เนื่องจากมีระยะทางที่ยาว จึงจำเป็นต้องมีการออกแบบการป้องกัน เพราะเมื่อเกิดความผิดปกติที่ส่วนใดส่วนหนึ่งของระบบ อุปกรณ์ป้องกันจะตัดส่วนนั้นออกจากระบบเพื่อให้ระบบยังทำงานต่อไปได้ และเพื่อทำการตรวจซ่อมบำรุง ในหัวข้อนี้จะกล่าวถึงรีเลย์ระยะทาง รีเลย์ป้องกันกระแสเกิน และการประสานสัมพันธ์รีเลย์ เพื่อใช้สำหรับออกแบบการป้องกันระบบไฟฟ้ากำลัง

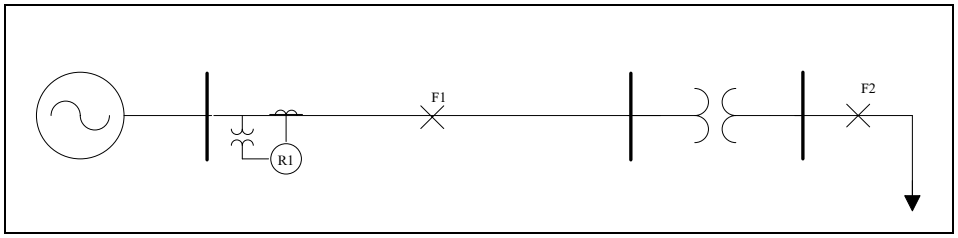
2.3.1 รีเลย์ระยะทาง

รีเลย์ระยะหรือรีเลย์อิมพีแดนซ์ทางเป็นรีเลย์ที่ได้รับการพัฒนาขึ้นเพื่อใช้ป้องกันสายส่งกำลังไฟฟ้าโดยเฉพาะ ในปัจจุบันรีเลย์ระยะทางใช้งานกันอย่างแพร่หลายสำหรับการป้องกันระบบส่งจ่ายกำลังไฟฟ้า รีเลย์ชนิดนี้อาศัยหลักการวัดค่าอิมพีแดนซ์สายส่งที่ถูกป้องกัน ในสภาวะปกติที่มีการจ่ายโหลดหรือสภาวะไร้โหลดใด ๆ อิมพีแดนซ์ที่รีเลย์วัดได้จะมีค่าสูง เนื่องจากเป็นค่าอิมพีแดนซ์ของสายส่งรวมกับอิมพีแดนซ์ของโหลด ในกรณีที่เกิดลัดวงจรลงดินโดยตรงหรือผ่านอิมพีแดนซ์ลัดวงจรค่าน้อย ๆ จะพบว่าอิมพีแดนซ์ที่รีเลย์ตรวจจับได้จะมีค่าต่ำกว่ามาก ดังนั้น การตั้งค่ารีเลย์ให้ทำงานที่ค่าอิมพีแดนซ์ปรับตั้งที่เหมาะสมทำให้การป้องกันระบบสายส่งง่ายและมีประสิทธิภาพ นอกจากนี้ ค่าอิมพีแดนซ์ที่วัดได้ในขณะเกิดลัดวงจรยังสะท้อนถึงตำแหน่งของการเกิดลัดวงจรในสายส่งอีกด้วย จากรูปจะพบว่าถ้าทำการวัดกระแสและแรงดัน ณ ตำแหน่งติดตั้งรีเลย์แล้วนำมาหาค่าอิมพีแดนซ์จะได้ว่า

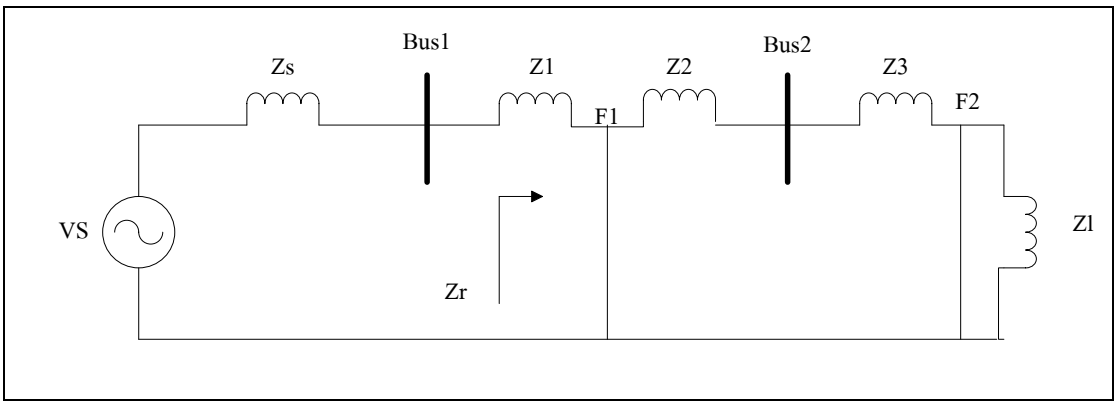
สภาวะปกติ $Z_R = Z_1 + Z_2 + Z_3 + Z_L$

ลัดวงจรที่ F2 $Z_{R=} \quad Z_1 + Z_2 + Z_3$

ลัดวงจรที่ F1 $Z_R = Z_1$



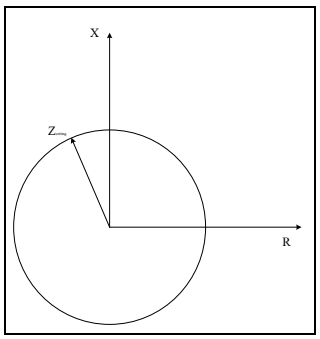
รูปที่ 2.1 การจ่ายโหลดทางเดียว



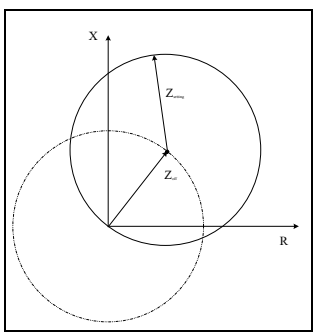
รูปที่ 2.2 การหาขนาดความต้านทานลัดวงจร

เมื่อเกิดความผิดพร่องขึ้นในระบบไฟฟ้ากำลัง การตรวจวัดอิมพีแดนซ์อาจจะดูเหมือนมีประสิทธิภาพ แต่การวัดค่าอิมพีแดนซ์นี้อาจจะก่อให้เกิดปัญหาบางประการขึ้นได้ ถ้าระบบสายส่งเป็นแบบป้อนหลายทาง ซึ่งการไหลของกำลังไฟฟ้าสามารถไหลได้ 2 ทิศทางแล้ว ค่าอิมพีแดนซ์ที่วัดได้อาจจะมีทิศทางอยู่ด้านหน้ารีเลย์หรือด้านหลังรีเลย์ก็ได้

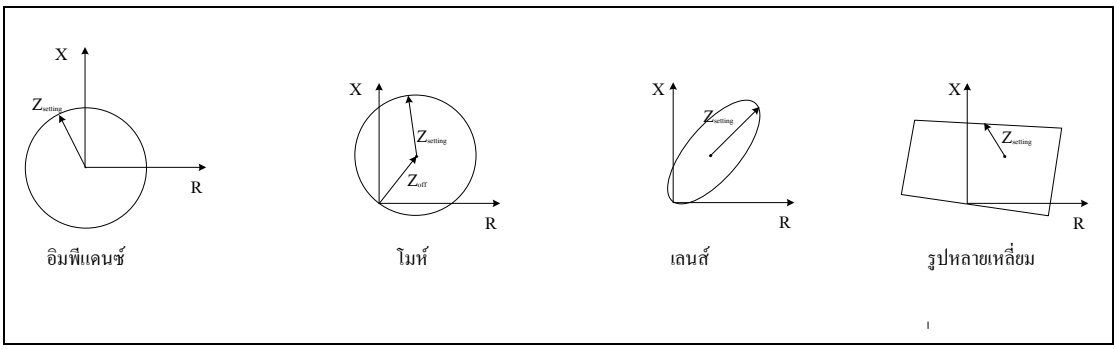
รีเลย์โมห์ เป็นรีเลย์ที่รวมคุณสมบัติของอิมพีแดนซ์และการจำแนกทิศทาง หลักการทำงานใช้การตั้งค่าออฟเซต ให้กับอิมพีแดนซ์รีเลย์ ทำให้วงกลมทำงานของอิมพีแดนซ์รีเลย์แบบปกติถูกยกให้เลื่อนไปอยู่เหนือแกน X=0 (รีแอกแตนซ์ =0) จากการเพิ่มค่าอิมพีแดนซ์ออฟเซตที่เหมาะสมเข้าไปในวงจรจะส่งผลให้รูปวงกลมทำงาน ถูกเลื่อนขึ้นไปอยู่เหนือแกน X = 0 อย่างไรก็ตาม การเลื่อนขึ้นของวงกลมทำงานต้องอยู่ภายใต้เงื่อนไขที่ว่า เส้นรอบวงของวงกลมทำงานต้องผ่านจุด (0,0) นอกจากนี้ รีเลย์ระยะทางอาจจะทำให้มีรูปร่างของขอบเขตทำงานที่หลากหลาย เช่น lens characteristic ,polygonal characteristic ,combined characteristic หรืออื่น ๆ



รูปที่ 2.3 แผนภาพอิมพีแดนซ์รีเลย์



รูปที่ 2.4 แผนภาพรีเลย์โมห์



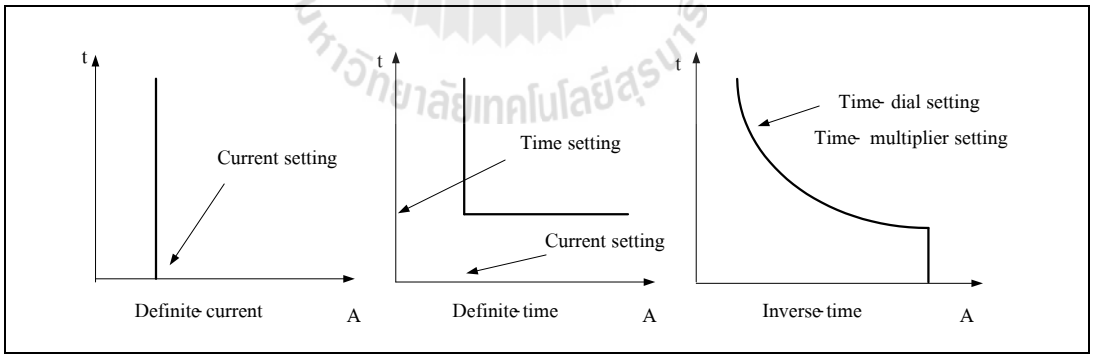
รูปที่ 2.5 อิมพีแดนซ์ของรีเลย์รูปแบบต่าง

2.3.2 รีเลย์ป้องกันกระแสเกิน

รีเลย์ป้องกันกระแสเกิน (Over-current relay: o/c relay) แบ่งออกตามคุณลักษณะการทำงานได้ 3 ชนิด ได้แก่

1. รีเลย์กระแสเกินชนิดจำกัดกระแส (Definite-current o/c relay)
2. รีเลย์กระแสเกินชนิดเวลาจำกัด (Definite-time o/c relay)
3. และรีเลย์กระแสเกินชนิดเวลาผกผัน (Inverse-time o/c relay)

รีเลย์กระแสเกินแต่ละชนิดมีกราฟคุณลักษณะของการทำงานที่ต่างกันดังรูปที่ 2.6



รูปที่ 2.6 คุณลักษณะการทำงานของรีเลย์กระแสเกิน

รีเลย์กระแสเกินชนิดจำกัดกระแสเป็นรีเลย์ที่ทำงานทันทีทันใด เมื่อมีกระแสไหลเกินค่าเริ่มต้นทำงานของรีเลย์ โดยปกติรีเลย์ชนิดนี้มักใช้งานร่วมกับรีเลย์กระแสเกินชนิดอื่น ๆ เช่น การทำงานร่วมกับรีเลย์กระแสเกินชนิดเวลาจำกัด เนื่องจากรีเลย์กระแสเกินชนิดเวลาจำกัดนี้จะมี

ช่วงเวลาทำงานของรีเลย์ที่แตกต่างกัน เพื่อลดปัญหาในเรื่องการทำงานพร้อมกันของรีเลย์ แต่การใช้งานรีเลย์ชนิดนี้มีข้อเสียคือ ไม่ว่าจะเป็นการเกิดโหลดเกินหรือเกิดลัดวงจร รีเลย์จะทำงานด้วยความเร็วเท่ากัน ซึ่งในความเป็นจริงแล้ว การเกิดลัดวงจรด้วยกระแสที่สูงกว่า รีเลย์ควรจะทำงานได้เร็วกว่า การเกิดโหลดเกินนั้นรีเลย์ต้องยอมให้มีการหน่วงเวลาที่ยาวนานกว่า เพื่อเหตุผลของการจัดลำดับความสัมพันธ์การทำงานของรีเลย์ ทำให้รีเลย์ชนิดเวลาผกผันถูกนำมาใช้งาน เนื่องจากรีเลย์ชนิดนี้มีช่วงเวลาการทำงานของรีเลย์เป็นสัดส่วนกลับกับค่าของกระแส กล่าวคือเมื่อเกิดความผิดปกติขึ้นในระบบที่ค่ากระแสที่เกินพิกัดน้อย รีเลย์จะมีการหน่วงเวลาในการทำงาน จะไม่ตัดในทันที ในขณะที่ถ้ากระแสมีค่าสูงเช่น เกิดความผิดพร่อง รีเลย์ต้องทำงานอย่างรวดเร็วโดยมีการหน่วงเวลาน้อยหรือทำงานตัดวงจรทันที

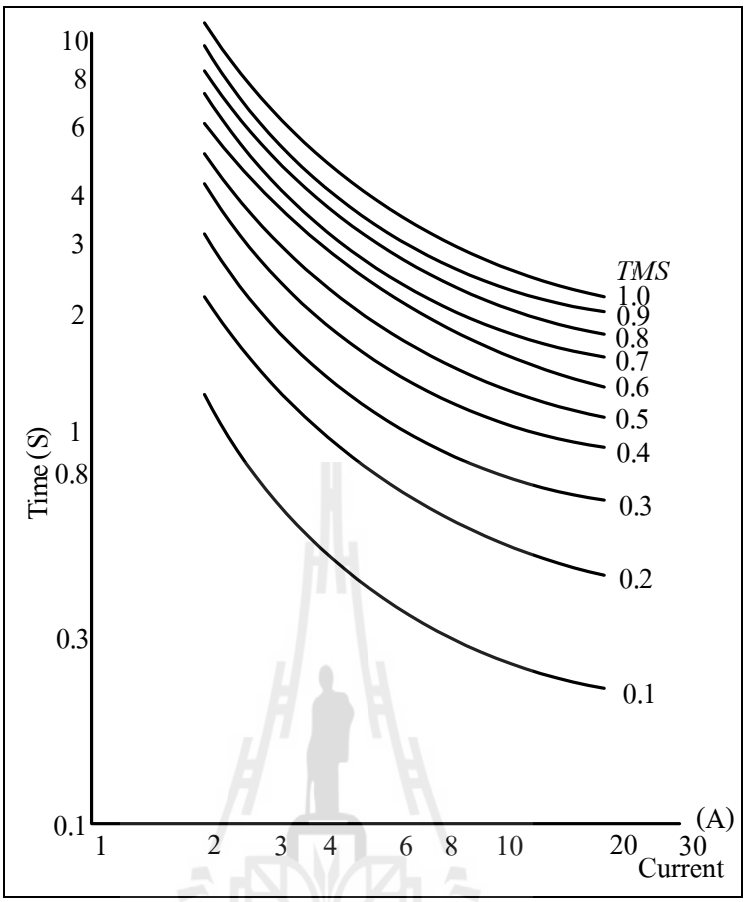
การทำงานของรีเลย์ จะพิจารณาจากเส้นโค้งเวลาและกระแสของรีเลย์ดังรูปที่ 3.2 โดยที่ค่ากระแสต่างกัน รีเลย์กระแสเกินจะทำงานด้วยเวลาที่ต่างกันด้วย การตั้งค่าของรีเลย์ทำได้ดังนี้

1. Pickup setting เป็นการปรับตั้งค่าให้รีเลย์เริ่มต้นทำงานที่ค่ากระแสค่าหนึ่ง ซึ่งรีเลย์จะทำงานเมื่อรีเลย์ตรวจพบกระแสที่มีค่าเท่ากับค่านี้หรือมากกว่า

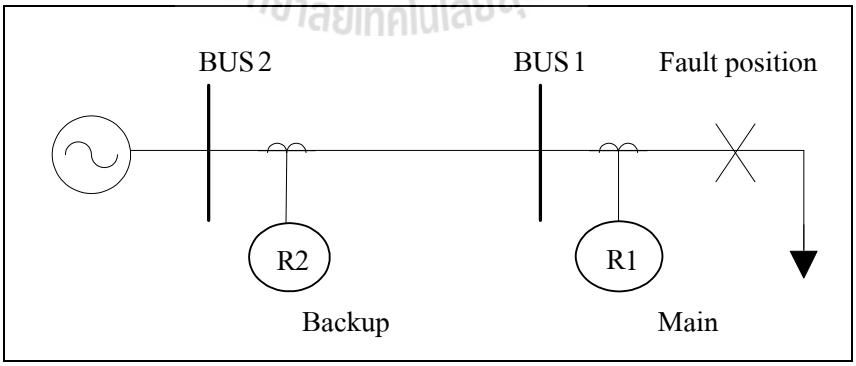
2. Current tap setting (CTS) เป็นค่าแท็บของขดลวดด้านอินพุต เมื่อระบุค่ากระแสทำงาน (Pickup current) ได้แล้ว ต้องทำการเลือก CTS ให้สอดคล้องเพื่อการป้องกันที่สมบูรณ์

3. Time-delay setting (TDS) เป็นการตั้งค่าการหน่วงเวลาในการทำงานของรีเลย์ หรือกล่าวอีกนัยหนึ่งได้ว่า TDS คือการเลือกเส้นโค้งเวลาการทำงานของรีเลย์นั่นเอง

การหน่วงเวลาของรีเลย์แต่ละตัว เป็นปัจจัยที่สำคัญอีกประการหนึ่งสำหรับการจัดลำดับความสัมพันธ์การทำงานของรีเลย์ป้องกันกระแสเกิน การหน่วงเวลาที่จะกล่าวถึงนี้จะพิจารณาสำหรับรีเลย์ตัวที่อยู่ติดกันโดยจะเรียกรีเลย์ตัวที่อยู่ใกล้ความผิดพร่องมากกว่าว่า Main ส่วนรีเลย์ตัวที่อยู่ถัดไปเรียกว่า Backup ดังรูปที่ 2.8 โดยการจัดลำดับความสัมพันธ์ของรีเลย์ 2 ตัวนี้ ให้บวกค่าเวลาส่วนเผื่อของการจัดลำดับความสัมพันธ์ (Coordination time interval: CTI) ประมาณ 0.3 – 0.5 วินาที



รูปที่ 2.7 เส้นโค้งเวลาและกระแสของรีเลย์



รูปที่ 2.8 ตำแหน่งการเรียกชื่อของรีเลย์เมื่อเกิดความผิดพร่อง

ความไวในการทำงานของรีเลย์ยังถูกแบ่งออกได้เป็นหลายแบบตามรูปร่างของเส้นโค้งเวลาการทำงานของรีเลย์ ได้แก่

1. แบบเวลาผกผันมาตรฐาน (Standard inverse time) พบได้ในระบบจำหน่าย และระบบส่งจ่ายทุกระดับแรงดัน ไม่ว่าจะเป็นรีเลย์หลักหรือรีเลย์สำรอง ปกติจะใช้ในกรณีที่ไม่มีการจัดลำดับความสัมพันธ์การทำงานของรีเลย์กับอุปกรณ์ป้องกันชนิดอื่น เช่น ฟิวส์ เป็นต้น มีสมการกำกับเส้นโค้งการทำงานดังสมการที่ (2.1)

$$t = \frac{0.14 \times TDS}{PSM^{0.02} - 1} \tag{2.1}$$

2. แบบเวลาผกผันยาวนาน (Long time inverse) รีเลย์ชนิดนี้มักใช้เพื่อป้องกันตัวต้านทานสำหรับต่อลงดินของสายนิวทรอล หรือใช้สำหรับป้องกันโหลดเกินให้กับมอเตอร์หรือเครื่องกำเนิดไฟฟ้า มีสมการกำกับเส้นโค้งการทำงานดังสมการที่ (2.2)

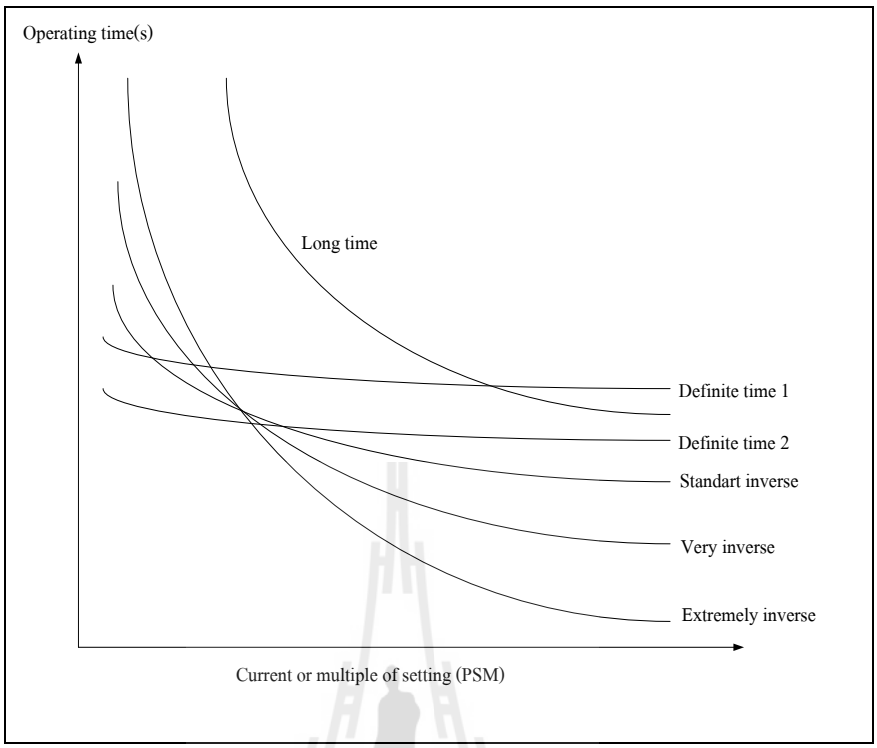
$$t = \frac{120 \times TDS}{PSM - 1} \tag{2.2}$$

3. แบบเวลาผกผันยาวนานมาก (Very inverse time) การใช้รีเลย์กระแสเกินแบบเวลาผกผันมาตรฐานมีผลกระทบต่อการป้องกันในกรณีที่เกิดการลัดวงจรใกล้ ๆกับแหล่งจ่าย จะทำให้เวลาในการกำจัดความผิดพร่องโดยเฉลี่ยช้ากว่าเวลาการกำจัดความผิดพร่อง ณ ตำแหน่งอื่น ๆ การนำรีเลย์กระแสเกินแบบเวลาผกผันยาวนานมากมาใช้จะช่วยแก้ปัญหานี้ได้ มีสมการกำกับเส้นโค้งการทำงานดังสมการที่ (2.3)

$$t = \frac{13.5 \times TDS}{PSM - 1} \tag{2.3}$$

4. แบบเวลาผกผันยาวนานที่สุด (Extremely inverse time) เหมาะสำหรับนำมาใช้จัดลำดับความสัมพันธ์การทำงานของรีเลย์กับฟิวส์ หรือใช้ป้องกันวงจรที่มีกระแสสูงขณะเปิดวงจร เช่น สายป้อน วงจรเครื่องทำความร้อน ปั๊ม เป็นต้น มีสมการกำกับเส้นโค้งการทำงานดังสมการที่ (2.4)

$$t = \frac{80 \times TDS}{PSM^{2} - 1} \tag{2.4}$$



รูปที่ 2.9 เส้นโค้งเวลาและกระแสของรีเลย์ป้องกันแบบมาตรฐาน

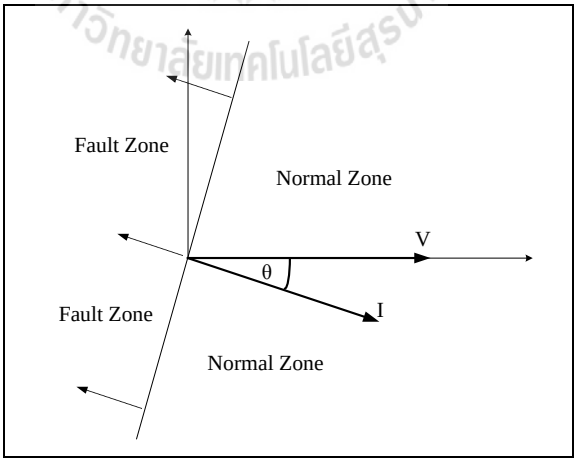
รีเลย์ที่ใช้ป้องกันระบบไฟฟ้าแบ่งตามลำดับการทำงานได้ 2 ประเภท คือ

1. รีเลย์หลัก (Primary Relay) เป็นรีเลย์ที่ใช้ป้องกัน โดยปกติการป้องกันจะแบ่งเขตป้องกันไว้เฉพาะเมื่อเกิดฟอลต์ขึ้นภายในเขตป้องกัน รีเลย์หลักจะสั่งให้เซอร์กิตเบรกเกอร์ทุกตัวในเขตป้องกันนั้นเปิดวงจร (Trip) เขตป้องกันของรีเลย์หลักจะจัดแบ่งให้คาบเกี่ยวกัน (Overlap) เพื่อป้องกันมิให้มีจุดบอดขึ้นในระบบป้องกัน รีเลย์หลักที่ทำงานถูกต้องจะสั่งให้เซอร์กิตเบรกเกอร์ทำงานน้อยที่สุดโดยจะสั่งให้เปิดวงจรเฉพาะส่วนที่เกิดฟอลต์ขึ้นเท่านั้น

2. รีเลย์สำรอง (Backup Relay) จะใช้ป้องกันแทนรีเลย์หลัก กรณีที่รีเลย์หลักไม่ทำงาน ซึ่งอาจเกิดขึ้นเนื่องจากไม่ได้จ่ายแหล่งจ่ายไฟฟ้าให้กับวงจรสั่งเปิดวงจรหรือกลไกของเซอร์กิตเบรกเกอร์ขัดข้อง เป็นต้น เวลาการทำงานของรีเลย์สำรองจะช้ากว่ารีเลย์หลัก ส่วนตำแหน่งที่ติดตั้งของรีเลย์สำรองนี้จะอยู่ที่สถานีไฟฟ้าย่อยต่างๆ หรืออยู่ไกลจากรีเลย์หลัก ถ้าเป็นไปได้มักจะให้รีเลย์สำรองนี้อยู่คนละตำแหน่งกับรีเลย์หลัก คืออยู่ต่างสถานีไฟฟ้าย่อยกัน และการสั่งเปิดวงจรเบรกเกอร์ด้วยรีเลย์สำรองนี้ อาจจะทำให้ส่วนที่ไม่เกี่ยวข้องกับจุดที่เกิดฟอลต์ถูกสั่งให้เปิดวงจรเบรกเกอร์ไปด้วย นอกจากนี้หน้าที่ของรีเลย์สำรองอีกอย่างหนึ่งก็คือ ทำหน้าที่เป็นรีเลย์หลักในกรณีที่มีการซ่อมแซมหรือบำรุงรักษารีเลย์หลัก

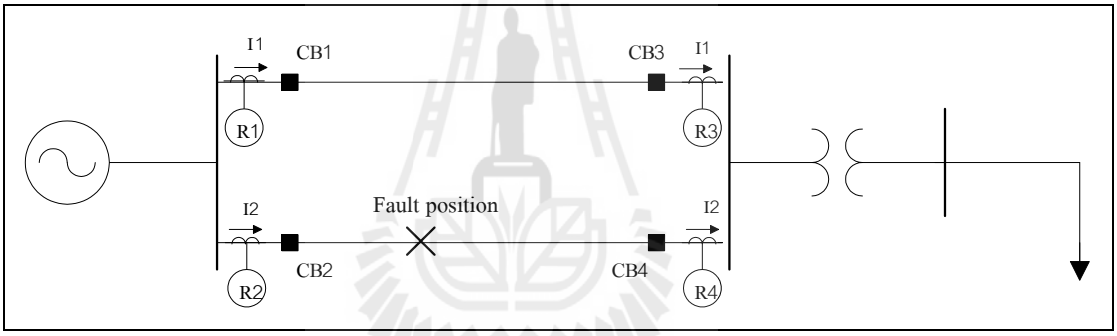
รีเลย์ป้องกันกระแสเกินแบบมีทิศทาง รีเลย์ป้องกันกระแสเกินแบบมีทิศทางเป็นรีเลย์กระแสเกินที่มีตัวเปรียบเทียบเฟสหรือทิศทางการไหลของกำลังไฟฟ้า จึงเรียกว่ารีเลย์ป้องกันกระแสเกินแบบมีทิศทาง (Directional Over-Current Relay) นิยมใช้กับระบบไฟฟ้ากำลังที่มีแหล่งจ่ายมากกว่า 1 แหล่งจ่ายหรือในระบบไฟฟ้ากำลังที่มีวงจรลูปหรือโครงข่ายจำนวนมาก เมื่อเกิดความผิดพร่องขึ้นในระบบไฟฟ้า กระแสลัดวงจรจะไหลผ่านรีเลย์ได้สองทิศทาง จึงจำเป็นต้องมีปริมาณอ้างอิงเพื่อใช้กำหนดทิศทางของกระแสลัดวงจรที่เกิดขึ้น โดยการตรวจสอบนี้ต้องใช้รีเลย์ที่วัดปริมาณ 2 ปริมาณ ได้แก่ กระแสที่ตรวจจับและสัญญาณอ้างอิง โดยปกติจะใช้แรงดันบัส ณ ตำแหน่งติดตั้งรีเลย์เป็นสัญญาณอ้างอิง เมื่อมุมเฟสของสัญญาณที่นำมาเปรียบเทียบกันซึ่งปกติได้แก่ กระแสและแรงดัน มีความต่างเฟสกันเกินกว่าที่กำหนด หมายความว่า การไหลของกำลังไฟฟ้าเกิดการกลับทิศทาง

พิจารณารูปที่ 2.10 การตรวจสอบและพิจารณาทิศทางของกระแสในขณะใด ๆ เทียบกับเฟสอ้างอิง โดยในสภาวะการจ่ายโหลดปกตินั้น กระแสจะล้าหลังแรงดันเล็กน้อยไม่เกิน 90 องศา เมื่อเกิดการลัดวงจรจะส่งผลให้มุมเฟสของกระแสลัดวงจรมีค่าประมาณ 180 องศา ดังนั้นการตรวจสอบความต่างเฟสนี้ เมื่อเทียบกับเฟสอ้างอิงเดิมในสภาวะปกติ มุมเฟสของกระแสกับแรงดันจะต่างกันไม่เกิน 90 องศา แต่เมื่อเกิดความผิดพร่องขึ้นในระบบ มุมเฟสจะต่างกันมากกว่า 90 องศา ทำให้นำมาใช้พิจารณาและวินิจฉัยเพื่อสั่งการไปยังเซอร์กิตเบรกเกอร์ให้ตัดวงจรได้ นอกเหนือจากการใช้ขนาดกระแสมาพิจารณาเพียงอย่างเดียว



รูปที่ 2.10 มุมเฟสของกระแสและแรงดัน

จากระบบไฟฟ้ากำลังดังรูปที่ 2.11 จะพบว่า การใช้รีเลย์ป้องกันกระแสเกินดังรูป เมื่อเกิดความผิดพร่องขึ้นระหว่างสายส่งจะทำให้มีกระแสลัดวงจรไหลผ่าน R1 R2 R3 และR4 นั่นคือ รีเลย์ทุกตัวเห็นความผิดพร่อง และจะสั่งเซอร์กิตเบรกเกอร์ตัดวงจร ทำให้สายส่งทั้ง 2 ชุดถูกตัดขาดออก ส่งผลให้โหลดถูกขัดจังหวะการจ่ายไฟ แต่เมื่อมีการติดตั้งรีเลย์ป้องกันกระแสเกินแบบมีทิศทางจะทำให้รีเลย์ R2 ทำงานเนื่องจากมีค่ากระแสลัดวงจรสูงที่สุด ดังนั้น CB2 จะเปิดวงจร ในขณะเดียวกัน R4 จะตรวจพบความต่างเฟสที่เกิน 90 องศา และขนาดกระแสที่สูงเกินกำหนด ดังนั้น CB4 จะถูกเปิดวงจรออก ในขณะที่ R1 และ R3 มีการตรวจสอบทิศทางจะไม่พบความผิดปกติใด ๆ เพียงแต่มีกระแสค่าสูงชั่วขณะ แต่จะยังไม่ทำงาน เมื่อ R2 และ R4 ทำงาน กระแสที่ผ่าน R1 และ R3 จะเข้าสู่สภาวะปกติ ทำให้สายส่งเส้นที่ไม่เกิดความผิดพร่องยังคงทำงานจ่ายโหลดได้ตามปกติ



รูปที่ 2.11 ระบบไฟฟ้ากำลังแบบใช้สายส่ง 2 เส้นขนานกัน

การปรับตั้งรีเลย์ป้องกันกระแสเกินแบบมีทิศทาง

ทำได้โดยการตั้งค่า Pickup setting และค่า TDS ที่เหมาะสมทั้งสองทิศทาง

1. Pickup setting ให้พิจารณาค่าโหลดสูงสุดที่เป็นไปได้จากการจ่ายโหลดทั้งสองทิศทาง และให้ใช้ตัวปรับคูณที่เหมาะสม เช่น 1.25, 1.3, 1.4 หรือ 1.5 เป็นต้น

2. TDS ให้พิจารณาตามหลักการของรีเลย์กระแสเกินทุกประการ โดยให้บวกค่า time grading margin ประมาณ 0.3 – 0.5 วินาทีสำหรับ upstream relay

3. Instantaneous setting ให้ใช้ได้ตามความเหมาะสม

ช่วงความเผื่อ (Grading Margin)

การจัดลำดับเวลาการทำงานให้สัมพันธ์กันรีเลย์ที่อยู่ใกล้ฟอลต์ที่สุด จะถูกจัดเป็นการป้องกันหลัก (Primary Relay) จะต้องทำงานก่อนรีเลย์ทุก ๆ ตัว โดยรีเลย์ตัวถัดไปจะจัดเป็นการป้องกันสำรอง (Backup Relay) และจะต้องหน่วงเวลาไว้ปริมาณหนึ่ง การหน่วงเวลาของรีเลย์สำรองจะเรียกว่าช่วงความเผื่อ (Grading Margin) ที่จะต้องพิจารณาให้เหมาะสมเนื่องจากหากมีค่ามากเกินไป ความเสียหายต่อระบบและอุปกรณ์จะมาก แต่ถ้ามีค่าน้อยเกินไปการทำงานของรีเลย์สำรองอาจทำงานก่อนรีเลย์ป้องกันหลักซึ่งจะตัดวงจรของระบบไฟฟ้าออกมากเกินความจำเป็น ช่วงความเผื่อจะขึ้นอยู่กับปัจจัยต่างๆ ต่อไปนี้

1. เวลาในการตัดวงจรของเซอร์กิตเบรกเกอร์เมื่อมีไฟเลี้ยงวงจรทริป จะเกิดแรงทำให้หน้าสัมผัสของเซอร์กิตเบรกเกอร์เคลื่อนที่ออกและเกิดอาร์กระหว่างหน้าสัมผัสทั้งสอง เนื่องจากการเคลื่อนที่ทางกลจะต้องใช้เวลา โดยเวลาทั้งหมดในการตัดวงจรของเซอร์กิตเบรกเกอร์จะขึ้นอยู่กับชนิดของเซอร์กิตเบรกเกอร์ด้วย โดยมีค่าประมาณ 5 รูปคลื่น(Cycle) หรือประมาณ 0.1 วินาที

2. เวลาที่รีเลย์ทำงานก่อน (Overshoot time) หลังจากที่รีเลย์ถูกตัดไฟออกแล้ว รีเลย์ยังคงทำงานต่ออีกเล็กน้อยจนกระทั่งพลังงานที่เก็บไว้ของรีเลย์หมดไป ตัวอย่างเช่น Induction Disc Element จะเก็บไว้ในรูปพลังงานจลน์ ส่วนสแตติครีเลย์จะมีพลังงานที่เก็บไว้ในตัวเก็บประจุ โดยทั่วไปเวลาที่รีเลย์ทำงานก่อน (Overshoot time) จะมีค่าเท่ากับ 0.05 วินาทีสำหรับรีเลย์แบบดิจิตอลซึ่งเป็นรีเลย์สมัยใหม่จะมีค่าน้อยมากจึงอาจตัดทิ้งได้

3. ค่าความผิดพลาดของอุปกรณ์ที่ใช้วัดเช่น รีเลย์ หม้อแปลงวัดกระแส จะมีความผิดพลาดและค่าลักษณะทางเวลาของรีเลย์มีค่าผิดพลาดทั้งทางด้านลบและทางด้านบวก โดยที่ค่าความผิดพลาดของหม้อแปลงวัดกระแสเกิดจากลักษณะทางแม่เหล็ก(Magnetizing Characteristic) โดยค่าความผิดพลาดของหม้อแปลงวัดกระแสจะไม่มีผลต่อรีเลย์แบบทำงานที่เวลาตายตัว

4. ค่าเวลาเผื่อไว้เพื่อความปลอดภัย(Safety Margin) ปกติแล้วเวลา 100 มิลลิวินาทีจะถูกรวมเข้าไปในขั้นตอนสุดท้ายของการคำนวณเพื่อให้แน่ใจว่าการจัดลำดับเวลาการทำงานให้สัมพันธ์กันได้ถูกต้อง โดยค่าที่นิยมเลือกใช้กัน ดังนี้

เวลาในการตัดวงจรของเซอร์กิตเบรกเกอร์ (Circuit Breaker)	0.10 วินาที
เวลาที่รีเลย์ทำงานก่อน(Overshoot time)	0.05 วินาที
ค่าความผิดพลาดของอุปกรณ์ทั้งหมด	0.10 วินาที
ค่าเวลาเผื่อไว้เพื่อความปลอดภัย(Safety Margin)	0.10 วินาที
ผลรวมทั้งหมด	0.35 วินาที

โดยทั่วไปค่าช่วงความเผื่อ(Grading Margin) ที่นิยมใช้กับคือ 0.3 - 0.5 วินาที ซึ่งเพียงพอสำหรับการตรวจจับได้อย่างถูกต้อง

2.3.3 การปรับตั้งและการประสานสัมพันธ์รีเลย์

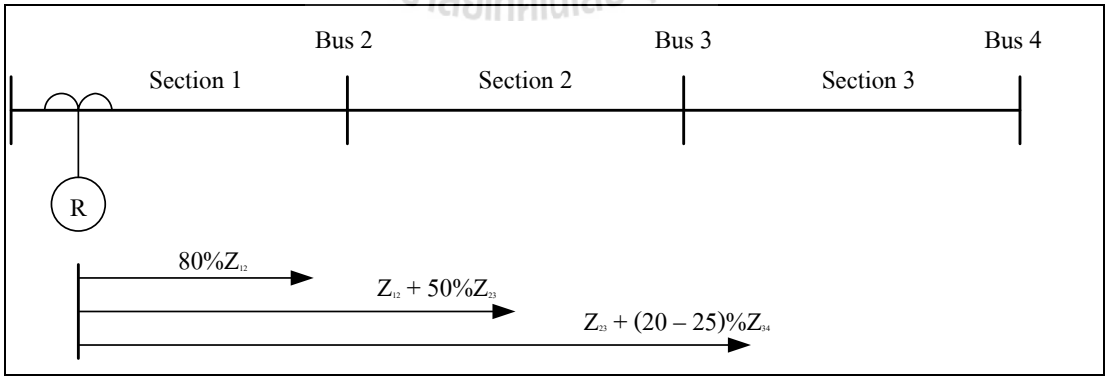
การปรับตั้งค่ารีเลย์เพื่อการป้องกันนั้น แบ่งออกเป็น 2 แบบ คือ การทำงานเพียงตัวเดียวและการทำงานแบบประสานสัมพันธ์ ดังนั้นการเราจึงต้องพิจารณาถึงการทำงานรวมกันของรีเลย์เพื่อให้การป้องกันมีความปลอดภัยสูงสุด โดยในส่วนนี้จะกล่าวถึงการป้องกันและการประสานสัมพันธ์ของรีเลย์ระยะทาง รีเลย์ป้องกันกระแสเกิน การประสานสมพันธ์รีเลย์ป้องกันกระแสเกิน และการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและรีเลย์รีเลย์ระยะทาง

1. **รีเลย์ระยะทาง**

การปรับตั้งค่ารีเลย์ระยะทางสำหรับการป้องกันจะใช้การตั้งค่าจากค่าอิมพีแดนซ์ลำดับบวกของสายส่ง โดยค่าอิมพีแดนซ์ที่ปรับตั้งที่ตัวรีเลย์จะผ่านการแปลงของ CT และ VT มาแล้ว นั่นคือ

$$Z_{relay} = \frac{V_{relay}}{I_{relay}} = \frac{V_{line}/VTR}{I_{line}/CTR} = Z_{line} \times \frac{CVT}{VTR} \tag{2.5}$$

การป้องกันระบบไฟฟ้ากำลังที่เป็นแบบการป้อนทางเดียวนั้น จะมีการแบ่งโซนป้องกันออกเป็นชั้นๆ แต่ละโซนจะมีเวลาทำงานที่แตกต่างกัน โดยแบ่งระยะป้องกันตามส่วนของสายส่งที่รีเลย์ป้องกันอยู่ดังรูป



รูปที่ 2.12 ระยะป้องกันของรีเลย์ระยะทาง

การปรับตั้งรีเลย์ตามโซนนี้จะเรียกว่าการปรับตั้งระยะเขตคุ้มกัน(setting the reach) มีหลักการปรับตั้งดังต่อไปนี้

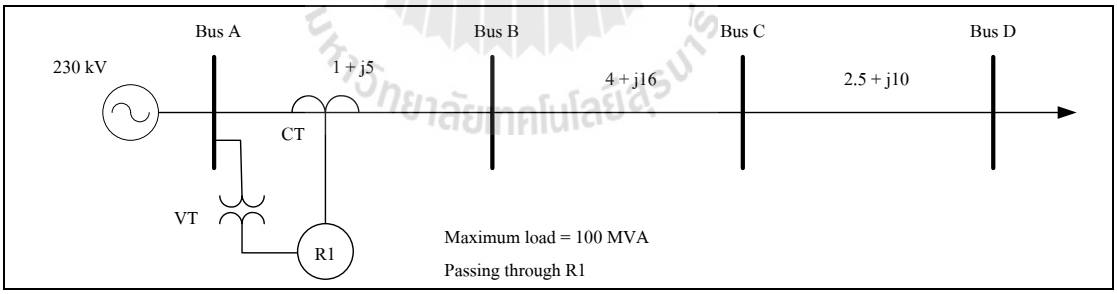
โซน 1 ครอบคลุมค่าอิมพีแดนซ์ 80% - 85% ของสายป้อนในช่วงที่ 1

โซน 2 ครอบคลุมค่าอิมพีแดนซ์ของสายป้อนช่วงที่ 1 บวกกับ 20% - 50% ของอิมพีแดนซ์สายป้อนช่วงถัดไป

โซน 3 ครอบคลุมค่าอิมพีแดนซ์ของสายป้อนช่วงที่ 1 และ ช่วงที่ 2 บวกกับ 20% - 25% ของอิมพีแดนซ์สายป้อนในช่วงถัดไป

โดยปกติการปรับตั้งเวลาการทำงานของโซนที่ 1 นี้ จะถูกปรับตั้งโดยบริษัทผู้ผลิตให้ทำงานแบบทันที(instantaneous tripping) สำหรับโซนถัดไป จะต้องเกิดการหน่วงเวลา โดยโซนที่ 2 ให้หน่วงเวลาประมาณ 0.25 – 0.4 วินาที โซนที่ 3 ให้ทำงานประมาณ 0.6 – 1.0 วินาที นอกจากนี้แล้ว อาจมีการปรับตั้งโซนที่ 4 และ โซนที่ 5 ได้ โดยปรับตั้งระยะเขตคุ้มกันไว้ที่ค่าอิมพีแดนซ์รวมทั้ง 3 ช่วง บวกกับ 20% ของอิมพีแดนซ์สายป้อนในช่วงที่ 4 และการปรับตั้งโซนที่ 5 ให้ปรับตั้งระยะเขตคุ้มกันไว้ที่ค่าอิมพีแดนซ์รวมทั้ง 4 ช่วง บวกกับ 20% ของอิมพีแดนซ์สายป้อนในช่วงที่ 5 สำหรับการหน่วงเวลาในแต่ละโซนให้ทำการบวกเวลาการหน่วง 0.3 – 0.4 วินาที สำหรับโซนถัดไป

ตัวอย่างที่ 1 ให้ปรับตั้งรีเลย์ระยะทางสำหรับป้องกันสายป้อนต่อไปนี้



รูปที่ 2.13 ตัวอย่างการปรับตั้งค่ารีเลย์

ขั้นตอนแรก: เลือก CTR และ VTR สำหรับรีเลย์

ที่ค่าโหลดสูงสุดมีกระแส 251 A ดังนั้นใช้ CTR = 250:5

สำหรับแรงดันใช้ VTR = 230-kV : 120-V

นั่นคือ CTR/VTR = 0.0261

ขั้นตอนที่ 2: เลือกการปรับตั้งสำหรับโซน 1 โดยปรับตั้งที่ 80% ของ Z_{AB}นั่นคือ

$$Zone\,1:\ Z_{setting} = 0.8(1 + j5) = 0.8 + j4\,\Omega = 4.08\angle 78.7^\circ\,\Omega\ (actual\ impedance)$$

$$Z_{setting} = 0.0261 \times 4.08\angle 78.7^\circ\Omega = 0.1065\angle 78.7^\circ\,\Omega$$

ทำงานแบบทันทีทันใด (instantaneous)

ขั้นตอนที่ 3: เลือกการปรับตั้งสำหรับโซน 2 โดยปรับตั้งที่ Z_{AB} + 50% ของ Z_{BC}นั่นคือ

$$Zone\,2:\ Z_{setting} = (1 + j5) + 0.5(4 + j16) = 13.3\angle 77.0^\circ\,\Omega\ (actual\ impedance)$$

$$Z_{setting} = 0.0261 \times 13.3\angle 77.0^\circ\Omega = 0.3471\angle 77.0^\circ\,\Omega$$

ทำงานโดยหน่วงเวลา 0.3 s

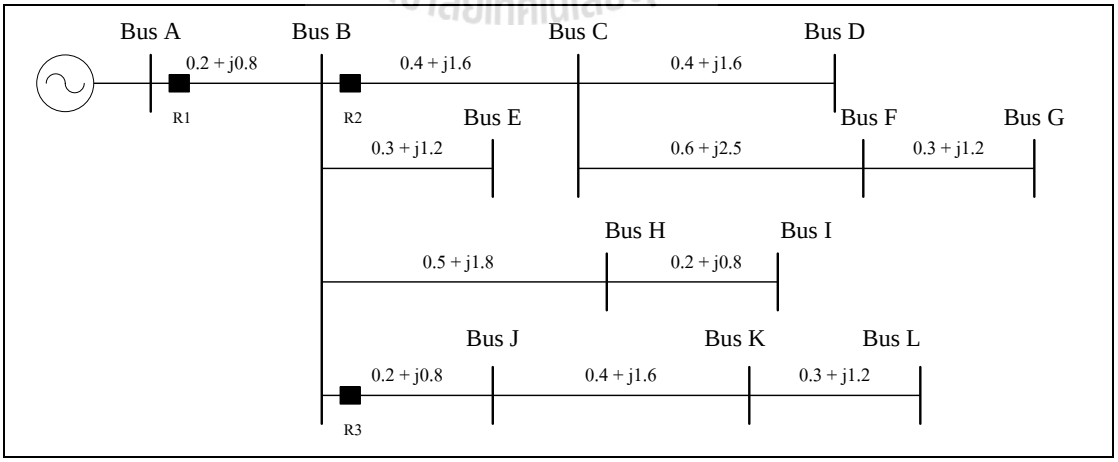
ขั้นตอนที่ 4 : เลือกการปรับตั้งสำหรับโซน 3 โดยปรับตั้งที่ Z_{AC} + 20% ของ Z_{CD}นั่นคือ

$$Zone\,3:\ Z_{setting} = (5 + j21) + 0.2(2.5 + j10) = 23.65\angle 76.6^\circ\,\Omega\ (actual\ impedance)$$

$$Z_{setting} = 0.0261 \times 23.65\angle 76.6^\circ\Omega = 0.6173\angle 76.6^\circ\,\Omega$$

ทำงานโดยหน่วงเวลา 0.8 s

ตัวอย่างที่ 2 จงออกแบบระบบป้องกันโดยใช้รีเลย์ระยะทางที่ตำแหน่งรีเลย์ R1 R2 และ R3



รูปที่ 2.14 ตัวอย่างที่ 2 การปรับตั้งค่ารีเลย์

โดยกำหนดให้ CTR ที่ละ R_3มีค่าเป็น 600:5 250:5 และ 150:5 ตามลำดับ

VTR ใช้ 69-kV:120-V

@R_1 CTR/VTR = 0.2087

@R_2 CTR/VTR = 0.0807

@R_3 CTR/VTR = 0.0522

<u>พิจารณารีเลย์ R_1</u>

Zone 1 : $Z_{setting} = 0.2087 \times 0.8 \times (0.2 + j0.8) = 0.1377 \angle 75.96° \Omega$

<u>สำหรับโซน 2</u>จะพบว่า บัส B มีสายป้อนเชื่อมอยู่ถึง 4 เส้น ได้แก่ B-C, B-E, B-H, B-J แต่ละเส้นมีค่าอิมพีแดนซ์แตกต่างกัน สำหรับการออกแบบเพื่อป้องกันระยะคุ้มกันโซน 2 เกิด over reach ดังนั้น ให้ปรับตั้งโซนที่ 2 ด้วยค่าอิมพีแดนซ์ของสายที่มีค่าน้อยที่สุด นั่นคือ Z_{B-J}จะได้ว่า

Zone 2 : $Z_{setting} = 0.2087[(0.2 + j0.8) + 0.5(0.2 + j0.8)] = 0.2581 \angle 75.96° \Omega$

<u>สำหรับโซน 3</u>จะพบว่า สายส่งในช่วงถัดไปต่อเชื่อมจากบัส C, E, H, J จะมีรีเลย์ติดตั้งอยู่ ดังนั้นเพื่อป้องกันการเกิด under reach การออกแบบให้ปรับตั้งโซนที่ 3 ให้คิดจากช่วงของสายที่มีค่าอิมพีแดนซ์ของสายมากที่สุด นั่นคือ Z_{B-H}จะได้ว่า

Zone 3 : $Z_{setting} = 0.2087[(0.7 + j2.6) + 0.2(0.2 + j0.8)] = 0.9061 \angle 75.30° \Omega$

<u>พิจารณารีเลย์ R_2 :</u>

Zone 1 : $Z_{setting} = 0.0870 \times 0.8(0.4 + j1.6) = 0.1148 \angle 75.96° \Omega$

Zone 2 : $Z_{setting} = 0.0870 [(0.4 + j1.6) + 0.5(0.4 + j1.6) = 0.2152 \angle 75.96° \Omega$ (CD)

Zone 3 : $Z_{setting} = 0.0870 [(1.0 + j4.1) + 0.2(0.3 + j1.2) = 0.3887 \angle 76.27° \Omega$(CF)

<u>พิจารณารีเลย์ R_3 :</u>

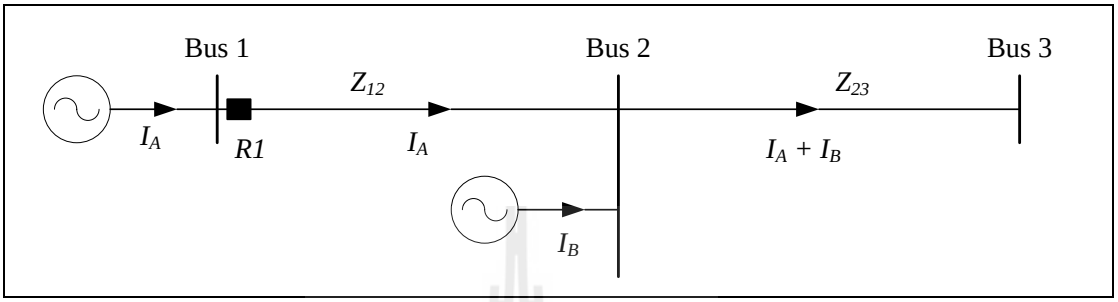
Zone 1 : $Z_{setting} = 0.0522 \times 0.8(0.2 + j0.8) = 0.0344 \angle 75.96° \Omega$

Zone 2 : $Z_{setting} = 0.0522 [(0.2 + j0.8) + 0.5(0.4 + j1.6) = 0.0861 \angle 75.96° \Omega$

Zone 3 : $Z_{setting} = 0.0522 [(0.6 + j2.4) + 0.2(0.3 + j1.2) = 0.1420 \angle 75.96° \Omega$

ผลของแหล่งจ่ายหลายแหล่งต่อการปรับตั้งรีเลย์ระยะทาง

ระบบไฟฟ้ากำลังที่ติดตั้งรีเลย์ระยะทางเป็นรีเลย์ป้องกันอาจจะมองเห็นค่าอิมพีแดนซ์ได้คลาดเคลื่อนถ้ามีแหล่งจ่ายต่อเชื่อมเข้ากับสายส่งที่อยู่หลังตำแหน่งติดตั้งรีเลย์



รูปที่ 2.15 ระบบไฟฟ้ากำลังที่มีแหล่งจ่ายหลายชุด

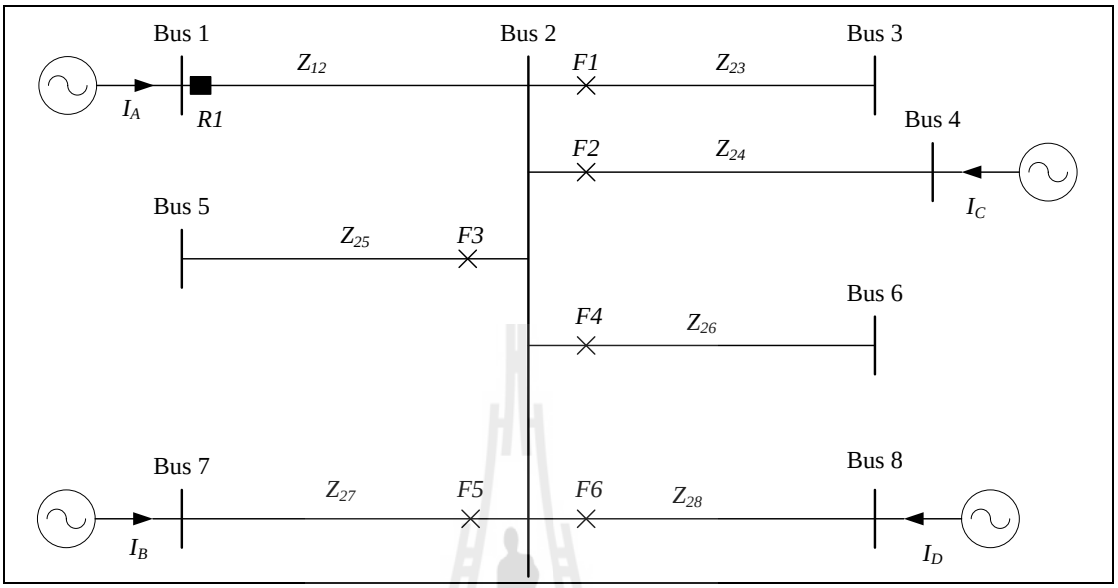
จากรูป ถ้ากำหนดให้เกิดการลัดวงจรที่บัส 3 จะพบว่า เครื่องกำเนิดไฟฟ้าทั้ง 2 ตัวในระบบจะช่วยกันจ่ายกระแสลัดวงจรมายังตำแหน่งเกิดลัดวงจร ส่งผลให้อิมพีแดนซ์ที่รีเลย์ตรวจพบมีค่าไม่เท่ากับ $Z_{12} + Z_{23}$ โดยจะตรวจพบอิมพีแดนซ์เท่ากับ

$$V_1 = Z_{12}I_A + Z_{23}(I_A + I_B)$$
$$Z_{R1} = \frac{V_1}{I_A} = Z_{12} + \left(1 + \frac{I_B}{I_A}\right)Z_{23} = Z_{12} + (1 + K)Z_{23}$$

จะเรียกสัมประสิทธิ์ K ว่า infeed constant ซึ่งคำนวณได้จากอัตราส่วนดังต่อไปนี้

$$K = \frac{\text{total infeed current}}{\text{relay current}} \tag{2.6}$$

พิจารณาคำนวณ infeed constant ได้ดังนี้



รูปที่ 2.16 ตัวอย่างระบบไฟฟ้ากำลังที่มีแหล่งจ่ายหลายชุด

จะได้ค่า infeed constant ที่ตำแหน่งเกิดลัดวงจรต่าง ๆ เมื่อพิจารณาโดยใช้รีเลย์ R1 เป็นจุดอ้างอิง

$$K_{F1} = \frac{I_B + I_C + I_D}{I_A}; \quad K_{F2} = \frac{I_B + I_D}{I_A}; \quad K_{F3} = \frac{I_B + I_C + I_D}{I_A}$$

$$K_{F4} = \frac{I_B + I_C + I_D}{I_A}; \quad K_{F5} = \frac{I_C + I_D}{I_A}; \quad K_{F6} = \frac{I_B + I_C}{I_A}$$

การปรับตั้งรีเลย์ระยะทางสำหรับระบบไฟฟ้ากำลังที่มีแหล่งจ่ายหลายชุด

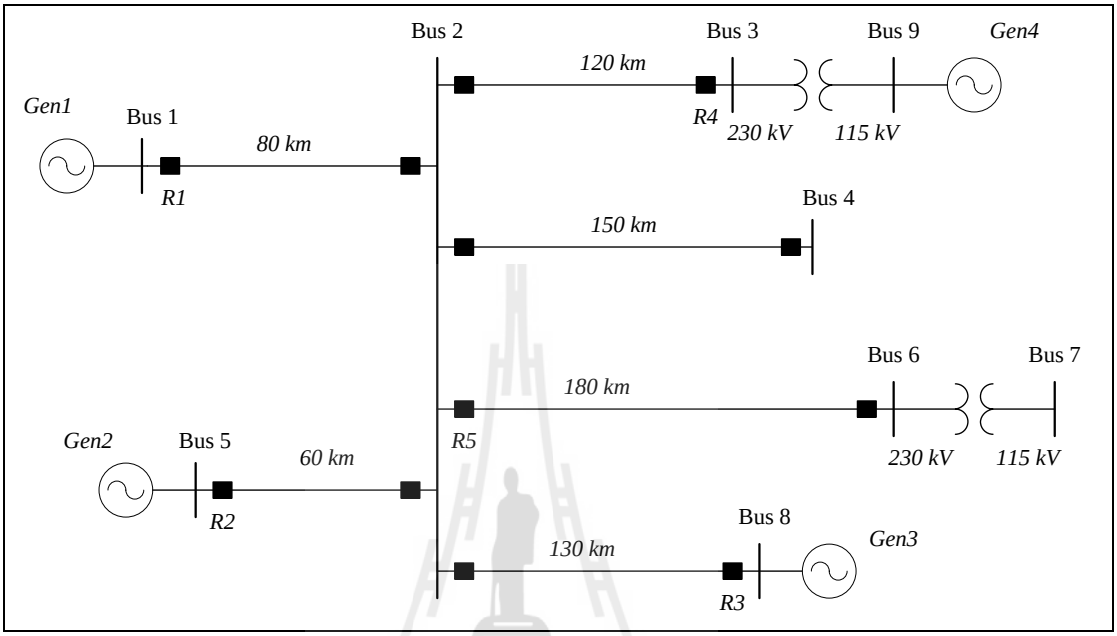
การปรับตั้งจะคล้ายกับรีเลย์ระยะทางแบบปกติ โดยแบ่งการปรับตั้งเป็นโซนป้องกันย่อย โดยปกติ จะมี 3 โซน ดังนั้น ในที่นี้จะพิจารณาเพียง 3 โซนเท่านั้น การพิจารณาการปรับตั้งในแต่ละโซนจะใช้ค่า infeed constant ในการคำนวณดังนี้

โซนที่ 1 : $Z_{setting} = 0.8 \times Z_{section,1}$

โซนที่ 2 : $Z_{setting} = Z_{section,1} + 0.5(1 + K_{12})Z_{section,2}$

โซนที่ 3 : $Z_{setting} = Z_{section,1} + (1 + K_{12})Z_{section,2} + 0.25(1 + K_{23})Z_{section,3}$

ตัวอย่างที่ 3 จงปรับตั้งค่ารีเลย์ระยะทาง R_1 สำหรับระบบไฟฟ้ากำลังที่มีค่าฐานกำลังไฟฟ้า 100 MVA ต่อไปนี้



รูปที่ 2.17 ตัวอย่างที่ 3 การปรับตั้งค่ารีเลย์

จากการคำนวณการไหลของกำลังไฟฟ้าและการคำนวณกระแสลัดวงจรได้กระแสลัดวงจรที่เครื่องกำเนิดไฟฟ้าทั้ง 4 ตัว จ่ายมายังจุดเกิดลัดวงจรแบบ 3 เฟส ที่บัส 2 (แสดงผลเฉพาะกระแสเฟส a)

Gen1 : $752.73 \angle -127.35°$ A

Gen2 : $227.34 \angle -34.58°$ A

Gen3 : $224.53 \angle -29.02°$ A

Gen4 : $617.33 \angle -5.95°$ A

กำหนดให้ใช้ CT ratio เท่ากับ 100:5 สำหรับรีเลย์ R_1; จะได้ CTR/VTR = 0.0104

สายส่งมีอิมพีแดนซ์ $Z_{line} = 0.2 + j0.6\,\Omega/km$ เท่ากันทุกสาย

หม้อแปลงมีอิมพีแดนซ์ทางด้านแรงสูง $Z_{tf} = j25\,\Omega$

$Zone\,1:\quad Z_{section,1} = (0.2 + j0.6)80 = 16 + j48\,\Omega$ (line 1 – 2)

$$Z_{setting} = 0.8(16 + j48) \times 0.0104 = 0.1331 + j0.3994\,\Omega = 0.4210\angle 71.57^\circ\,\Omega$$

$Zone\,2:\quad Z_{section,2} = (0.2 + j0.6)60 = 12 + j36\,\Omega$ (line 5 – 2)

เพื่อป้องกัน over-reaching ดังนั้น ใช้การปรับตั้งโซน 2 จากสายที่มีอิมพีแดนซ์ต่ำที่สุด คำนวณ infeed constant ในช่วงนี้

$$K_{Z2} = (I_{G3} + I_{G4}) / I_{G1} = 1.1008\angle 115.3^\circ$$
$$Z_{setting} = [(16 + j48) + 0.5(1 + 1.1008\angle 115.3^\circ)(12 + j36)] \times 0.0104$$
$$= 0.0131 + j0.6604\,\Omega$$
$$= 0.6605\angle 88.86^\circ\,\Omega$$

$Zone\,3:\quad Z_{section,3} = (0.2 + j0.6)180 = 36 + j108\,\Omega$ (line 6 – 2)

เพื่อป้องกัน under-reaching ใช้การปรับตั้งโซน 3 จากสายที่มีอิมพีแดนซ์มากที่สุด คำนวณ infeed constant ในช่วงนี้

$$K_{Z3} = (I_{G2} + I_{G3} + I_{G4}) / I_{G1} = 1.3846\angle 110.5^\circ$$
$$Z_{setting} = \begin{bmatrix} (16 + j48) + (1 + 1.1008\angle 115.3^\circ)(12 + j36) \\ + 0.25(1 + 1.3846\angle 110.5^\circ)(36 + j108) \end{bmatrix} \times 0.0104$$
$$= 0.0131 + j0.6604\,\Omega$$
$$= 0.6605\angle 88.86^\circ\,\Omega$$

มหาวิทยาลัยเทคโนโลยีสุรนารี

พิจารณาค่าปรับตั้ง รีเลย์ R_2 R_3 R_4 และ R_5

รีเลย์ R_2

กำหนดให้ใช้ CT ratio เท่ากับ 100:5 ; จะได้ CTR/VTR = 0.0104

$Zone\,1:\quad Z_{section,1} = (0.2 + j0.6)60 = 12 + j36\,\Omega$ (line 5 – 2)

$$Z_{setting} = 0.8(12 + j36) \times 0.0104 = 0.3168\angle 71.57^\circ\,\Omega$$

$Zone\,2:\quad Z_{section,2} = (0.2 + j0.6)80 = 16 + j48\,\Omega$ (line 2 – 1)

$$K_{Z2} = (I_{G3} + I_{G4}) / I_{G2} = 3.8575\angle -29.43^\circ$$
$$Z_{setting} = [(12 + j36) + 0.5(1 + K_{Z2})(16 + j48)] \times 0.0104$$
$$= 1.5912\angle 85.12^\circ\,\Omega$$

$Zone\,3:\quad Z_{section,3} = j25\,\Omega$ (line 6 – 7)

$$K_{Z3} = (I_{G1} + I_{G3} + I_{G4}) / I_{G2} = 3.7324\angle 30.52^\circ$$
$$Z_{setting} = [(12 + j36) + (1 + K_{Z3})(36 + j108) + 0.25(1 + K_{Z3})(j25)] \times 0.0104$$
$$= 6.0393\ \angle 49.35^\circ\,\Omega$$

รีเลย์ $\mathbf{R_3}$

กำหนดให้ใช้ CT ratio เท่ากับ 100:5 ; จะได้ CTR/VTR = 0.0104

$Zone\,1:\quad Z_{section,1} = (0.2 + j0.6)130 = 26 + j78\,\Omega$ (line 8 – 2)

$$Z_{setting} = 0.8(26 + j78)\times 0.0104 = 0.6864\angle 71.57^\circ\,\Omega$$

$Zone\,2:\quad Z_{section,2} = (0.2 + j0.6)60 = 12 + j36\,\Omega$ (line 2 – 5)

$$K_{Z2} = (I_{G1} + I_{G4})/I_{G3} = 3.032\angle -47.62^\circ$$
$$Z_{setting} = [(26 + j78) + 0.5(1 + K_{Z2})(12 + j36)]\times 0.0104$$
$$= 1.5264\angle 54.68^\circ\,\Omega$$

$Zone\,3:\quad Z_{section,3} = j25\,\Omega$ (line 6 – 7)

$$K_{Z3} = (I_{G1} + I_{G2} + I_{G4})/I_{G3} = 3.8468\angle -37.52^\circ$$
$$Z_{setting} = [(26 + j78) + (1 + K_{Z3})(36 + j108) + 0.25(1 + K_{Z3})(j25)]\times 0.0104$$
$$= 6.6123\ \angle 46.09^\circ\,\Omega$$

รีเลย์ $\mathbf{R_4}$

กำหนดให้ใช้ CT ratio เท่ากับ 100:5 ; จะได้ CTR/VTR = 0.0104

$Zone\,1:\quad Z_{section,1} = (0.2 + j0.6)120 = 24 + j72\,\Omega$ (line 3 – 2)

$$Z_{setting} = 0.8(24 + j72)\times 0.0104 = 0.6336\angle 71.57^\circ\,\Omega$$

$Zone\,2:\quad Z_{section,2} = (0.2 + j0.6)60 = 12 + j36\,\Omega$ (line 2 – 6)

$$K_{Z2} = (I_{G1} + I_{G3})/I_{G4} = 1.2209\angle -104.25^\circ$$
$$Z_{setting} = [(24 + j72) + 0.5(1 + K_{Z2})(12 + j36)]\times 0.0104$$
$$= 0.9595\angle 57.43^\circ\,\Omega$$

$Zone\,3:\quad Z_{section,3} = j25\,\Omega$ (line 6 – 7)

$$K_{Z3} = (I_{G1} + I_{G2} + I_{G3})/I_{G4} = 1.3615\angle -89.08^\circ$$
$$Z_{setting} = [(24 + j72) + (1 + K_{Z3})(36 + j108) + 0.25(1 + K_{Z3})(j25)]\times 0.0104$$
$$= 2.6872\ \angle 32.86^\circ\,\Omega$$

รีเลย์ $\mathbf{R_5}$

กำหนดให้ใช้ CT ratio เท่ากับ 100:5 ; จะได้ CTR/VTR = 0.0104

$Zone\,1:\quad Z_{section,1} = (0.2 + j0.6)180 = 36 + j108\,\Omega$ (line 2 – 6)

$$Z_{setting} = 0.8(36 + j108)\times 0.0104 = 0.9503\angle 71.57^\circ\,\Omega$$

$Zone\,2:\quad Z_{section,2} = j25\,\Omega$ (line 6 – 7)

$$K_{Z2} = 0$$
$$Z_{setting} = [(36 + j108) + 0.5(1 + K_{Z2})(j25)] \times 0.0104$$
$$= 1.3123 \angle 73.37° \ \Omega$$

$Zone\,3$: $Z_{section,3} = 0\,\Omega$ (no line)

$$Z_{setting} = 0$$

เมื่อปรับตั้งรีเลย์ระยะทางเรียนร้อยแล้ว ขั้นตอนต่อไปได้แก่การตรวจสอบจุดทำงาน ณ การจ่ายโหลดปกติ โดยทำการวาดกราฟอิมพีแดนซ์ไดอะแกรมเพื่อยืนยันว่าการออกแบบดังกล่าวต้องให้ค่าอิมพีแดนซ์โหลดอยู่นอกเขตป้องกัน ถ้าผลจากการออกแบบไม่เป็นไปตามข้อกำหนด ให้ทำการออกแบบใหม่อีกครั้งโดยการปรับตั้งค่ารีเลย์ให้เหมาะสม

2.รีเลย์ป้องกันกระแสเกิน

การปรับตั้งค่าหรือประสานสัมพันธ์รีเลย์ในระบบไฟฟ้ากำลังนั้นมีความซับซ้อนในการคำนวณค่าปรับตั้ง ดังนั้น จึงแสดงเฉพาะตัวอย่างระบบทดสอบอย่างง่าย ดังตัวอย่างต่อไปนี้ เพื่อใช้ศึกษาการปรับตั้งเพื่อให้มองภาพรวมได้ง่ายขึ้น

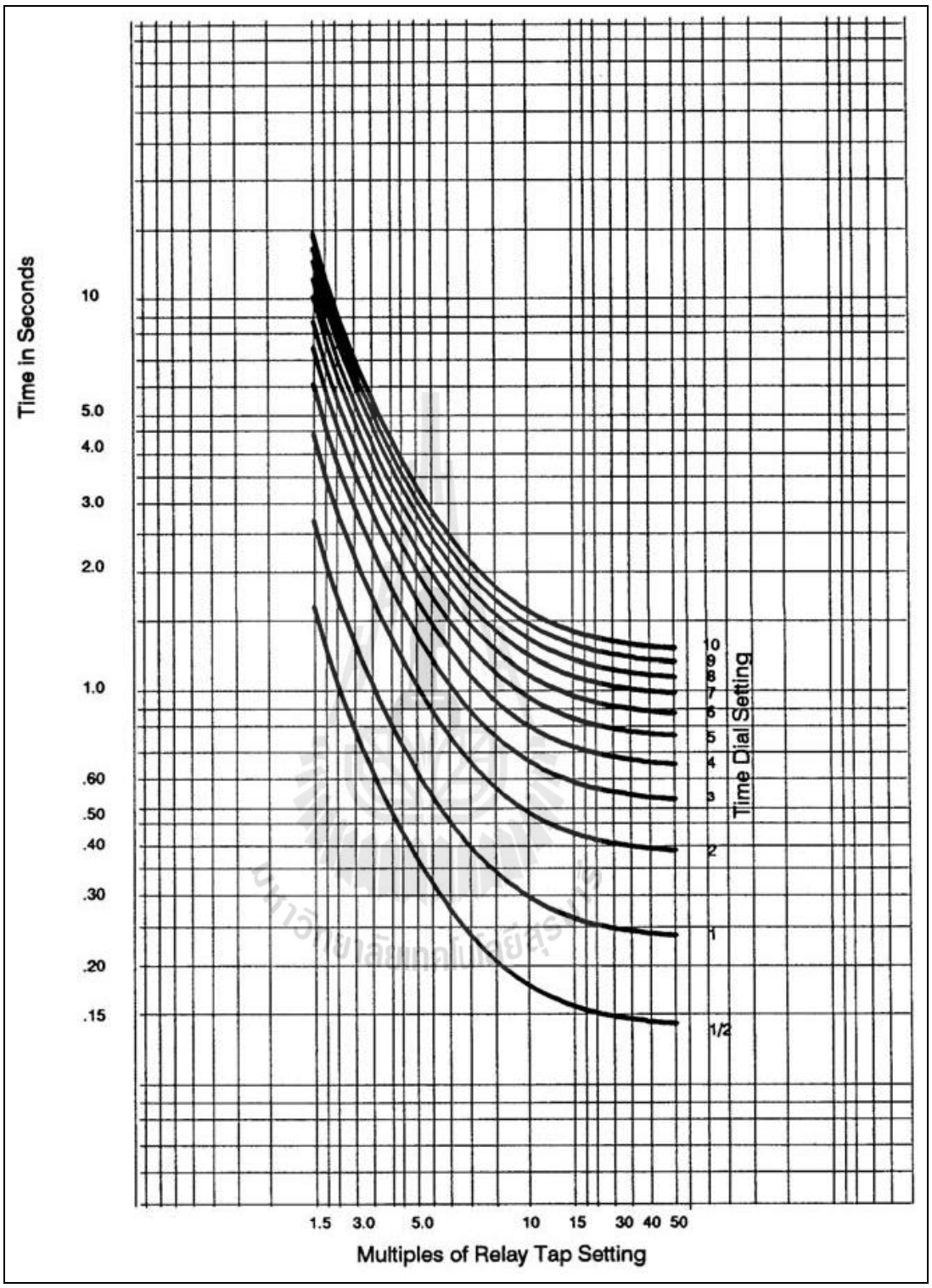
กำหนดให้ตั้งค่า pickup ไว้ที่ 5.0 A ให้อ่านเวลาที่รีเลย์ทำงานเมื่อกระแสที่วัดมีค่าเป็น 15.0 A เมื่อปรับตั้งค่า TDS เป็น ½ และ 4 ตามลำดับ ที่ค่ากระแส 15.0 A คิดเป็นกระแส 15.0/5.0 = 3 เท่าของค่า pickup

จากกราฟ (รูปที่ 2.18) ที่ค่า TDS = ½ อ่านค่าเวลาได้ 0.72 s.
จากกราฟ (รูปที่ 2.18) ที่ค่า TDS = 4 อ่านค่าเวลาได้ 3.5 s.

การเลือกค่า pickup คือ สิ่งที่สำคัญประการหนึ่ง คำถามที่สำคัญ คือ จะใช้กฎอะไรในการเลือกค่าดังกล่าว แต่จะเห็นได้ว่า รีเลย์กระแสเกินนี้ ต้องทำหน้าที่ยอมให้จ่ายโหลดในสภาวะปกติได้ นั่นคือการตั้งค่า pickup ต้องสูงกว่า maximum load ตัวปรับคูณที่นิยมใช้ คือ การยอมให้โหลดเกินได้ไม่เกิน 25% นั้นคือ $I_{pickup} \geq 1{,}25 \times I_{load,\max}$ อาจจะใช้ตัวปรับคูณมากกว่านี้ได้ เช่น 1.3 หรือ 1.5

อย่างไรก็ตาม กฎนี้ อาจมีข้อยกเว้นในกรณีที่ fault level มีค่าค่อนข้างต่ำ นั่นคือ รีเลย์ต้องป้องกันความผิดพร่องที่มีความรุนแรงน้อยที่สุดได้ ดังนั้น ค่า pickup จะต้องน้อยกว่า ค่า minimum fault เสมอ $I_{pickup} \leq I_{fault,\min}$ รวมเงื่อนไขเข้าด้วยกันจะได้

$$I_{pickup} = \min\left\{1.25 \times I_{\max,}, I_{fault,\min}\right\} \tag{2.7}$$

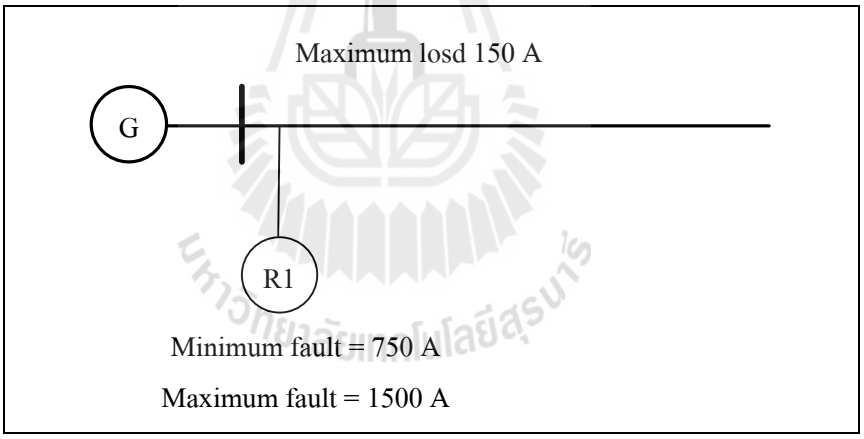


รูปที่ 2.18 เส้นโค้งเวลาทำงานของรีเลย์

การปรับตั้งค่ากระแสจำกัด (setting of instantaneous operation) มีความจำเป็นมาก นั่นคือ ตัวที่อยู่ใกล้จุดเกิดความผิดพร่องมากที่สุด ต้องทำหน้าที่ทันที ไม่มีกฎตายตัวสำหรับการปรับตั้งนี้ แต่มีข้อแนะนำดังนี้

- สำหรับสายส่งระหว่างสถานีไฟฟ้า ให้ปรับตั้งไว้ที่ค่าอย่างน้อย 125% ของ maximum fault ที่สถานีถัดไป
- สำหรับสายป้อนในระบบจำหน่ายทำได้ดังนี้ คือ ให้ปรับตั้งไว้ที่ 50 % ของ maximum fault ที่ตำแหน่งติดตั้งรีเลย์ หรือให้ตั้งค่าไว้ที่ประมาณ 6-10 เท่า ของค่ากระแสพิกัดสายป้อน
- อย่างไรก็ตาม ในหลาย ๆ กรณี อาจจะไม่จำเป็นต้องตั้ง Instantaneous tripping ก็ได้

ตัวอย่างที่ 2.5 สายป้อนวงจรย่อยในระบบจำหน่าย 22 kV วงจรหนึ่งดังรูปที่ 2.19 มีค่าโหลดสูงสุด กระแสลัดวงจรต่ำสุดและสูงสุดตามที่กำหนด ให้ปรับตั้งค่า pickup current และ instantaneous setting ให้เหมาะสม



รูปที่ 2.19 ระบบป้องกันตัวอย่างที่ 2.5

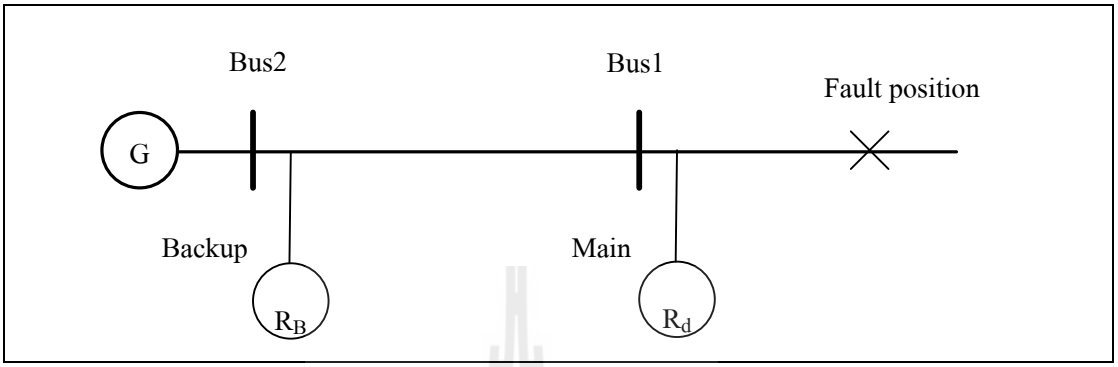
<u>วิธีทำ</u>

$$I_{pickup} = 1.25 \times 150 = 187.5\text{A} \quad < \text{minimum fault}$$

$$I_{inst} = 6 \times 150\text{A} = 900\text{A} \text{ หรือ } I_{inst} = 0.5 \times 1500\text{A} = 750\text{A}$$

การหน่วงเวลาของรีเลย์แต่ละตัว เป็นปัจจัยที่สำคัญอีกประการหนึ่งสำหรับการจัดลำดับความสัมพันธ์การทำงานของรีเลย์ป้องกันกระแสเกิน การหน่วงเวลาที่จะกล่าวถึง

นี้พิจารณาสำหรับรีเลย์ตัวที่อยู่ติดกันโดยจะเรียกรีเลย์ตัวที่อยู่ใกล้ความผิดพร่องมากกว่าว่า Main Relay ตัวที่อยู่ถัดไปเรียกว่า Backup Relay ดังรูป



รูปที่ 2.20 ลำดับความสัมพันธ์การทำงานของรีเลย์ป้องกันกระแสเกิน

การจัดลำดับความสัมพันธ์ของรีเลย์ 2 ตัวนี้ ให้บวกค่า time grading margin (t_{GM}) ประมาณ 0.3 – 0.5 นั่นคือ $t_B = t_M + t_{GM}$

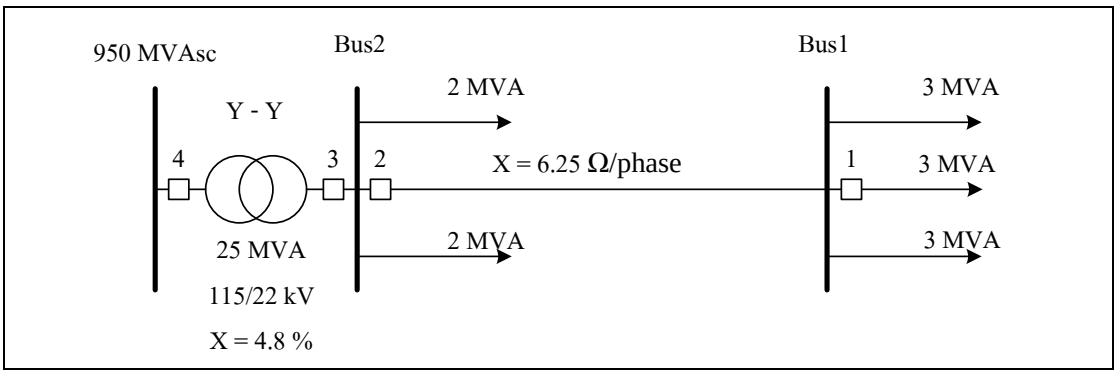
ตัวอย่างที่ 2.6 ระบบไฟฟ้ากำลังดังรูป ให้ออกแบบระบบรีเลย์ป้องกันให้เหมาะสม พร้อมทั้งเขียนเส้นโค้งการทำงานของรีเลย์แต่ละตัวในกราฟเดียวกันเพื่อตรวจสอบเวลาการทำงาน ถ้ากำหนดให้รีเลย์ที่ใช้มีคุณสมบัติดังสมการต่อไปนี้

$$t = \frac{TDS \times \beta}{\left(\frac{I_{actual}}{I_{pickup}}\right)^{\alpha} - 1} \qquad \text{เมื่อ } \alpha = 0.02,\ ß = 0.14$$

CT แบบ muti-rahio (600:5 MR)

50:5, 100:5, 150:5, 200:5, 250:5, 300:5, 400:5, 450:5, 500:5, 600:5

รีเลย์ปรับตั้งแบบทำงานทันทีตั้งค่าได้ในช่วง 6–144 A โดยมีความละเอียดของการปรับตั้งขั้นละ1 A รีเลย์ปรับตั้งแบบเวลาผกผันตั้งค่า CTS ได้ในช่วง 1–12 A มีความละเอียดของการปรับตั้งขั้นละ 1 A TDS ของรีเลย์ปรับได้ดังนี้ ½, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11

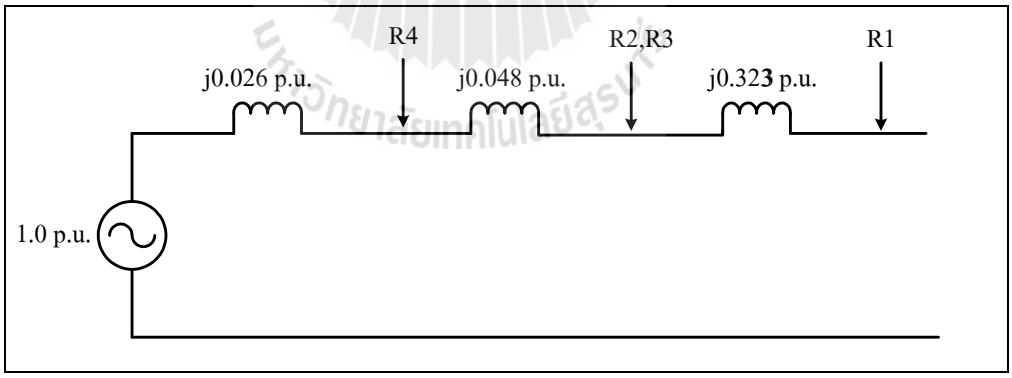


รูปที่ 2.21 ระบบป้องกันตัวอย่างสำหรับ ตัวอย่างที่ 2.6

<u>ขั้นตอนแรก</u> การคำนวณค่ากระแส ทั้งในสภาวะโหลดเต็มพิกัด และสภาวะที่เกิดความผิดพร่อง

พิจารณาโหลดเต็มพิกัด

- รีเลย์ตัวที่ 1 maximum load = 78.7 A (22 kV side)
- รีเลย์ตัวที่ 2 maximum load = 236.2 A (22 kV side)
- รีเลย์ตัวที่ 3 maximum load = 656.1 A (22 kV side)
- รีเลย์ตัวที่ 4 maximum load = 125.5 A (115 kV side)



รูปที่ 2.22 วงจรประกอบการคำนวณกระแสลัดวงจร

กระแสลัดวงจรสูงสุดที่รีเลย์แต่ละตัวมีค่าดังนี้

R1 :	2.52 p.u.	2.52×656.1=1653.4 A
R2, R3:	13.5 p.u.	13.5×656.1=8857.4 A

R4 : 38.5 p.u. 38.5×125.5 = 4831.8 A

เพื่อให้ง่ายในการศึกษา จะไม่คิด minimum fault (โดยตัวอย่างนี้ minimum fault มีค่าสูงกว่า maximum load มาก)

ขั้นตอนที่ 2 เลือกค่า CT ratio และ CTS

- รีเลย์ตัวที่ 1

  ป้องกันสายป้อนขนาด 3 MVA หรือที่กระแสโหลด 78.7 A คิดเผื่อโหลดเกิน 125% นั่นคือตั้งค่าที่ 1.25×78.7=98.4A ใช้ CT ratio 100:5 และตั้งค่า CTS ที่ 5.0 A

- รีเลย์ตัวที่ 2

  ป้องกันสายป้อนขนาด 9 MVA หรือที่กระแสโหลด 236.2 A คิดเผื่อโหลดเกิน 125% นั่นคือตั้งค่าที่ 1.25×236.2 = 295.25A ใช้ CT ratio 300:5 และตั้งค่า CTS ที่ 5.0 A

- รีเลย์ตัวที่ 3

  ป้องกันสายป้อนขนาด 25 MVA หรือที่กระแสโหลด 656.1 A คิดเผื่อโหลดเกิน 125% นั่นคือตั้งค่าที่ 1.25×656.1 = 820.1A ใช้ CT ratio 600:5 และตั้งค่า CTS ที่ 7.0 A

- รีเลย์ตัวที่ 4

  ป้องกันสายป้อนขนาด 25 MVA หรือที่กระแสโหลด 125.5 A คิดเผื่อโหลดเกิน 125% นั่นคือตั้งค่าที่ 1.25×125.5 = 156.9 A ใช้ CT ratio 150:5 และตั้งค่า CTS ที่ 6.0 A

ขั้นตอนที่ 3 จัดลำดับความสัมพันธ์ของรีเลย์ โดยการเลือก TDS ให้เหมาะสม กำหนด t_{GM} = 0.3 s.

- รีเลย์ตัวที่ 1 CT ratio = 100:5, CTS = 5.0 A

  เลือก TDS = ½ คำนวณเวลาการทำงานของกรณีโหลดเกิน และความผิดพร่องดังนี้

Case	I_{pri}(A)	I_{sec}(A)	I_{sec}/CTS	*Operating time (s)*
Overload	120	6.0	1.2	19.2
Maximum fault	1653.4	82.7	16.5	1.21

- รีเลย์ตัวที่ 2 CT ratio = 300:5, CTS = 5.0 A

เลือกเวลาการทำงานของกรณีโหลดเกิน และความผิดพร่อง โดยการบวกค่า t_{GM} ได้ดังนี้

Case	$I_{pri}(A)$	$I_{sec}(A)$	I_{sec}/CTS	*Operating time (s)*	*TDS*
Overload @R1	120	2	0.4	NOP	-
Maximum fault @ R1	1653.3	27.6	5.5	1.2 + 0.3 = 1.51	0.375
Overload @R2	320	5.3	1.06	60.03	½*
Maximum fault @ R2	8857.4	147.6	29.5	0.99	½*

*เลือกค่า TDS = ½

– รีเลย์ตัวที่ 3 CT ratio = 600:5, CTS = 7.0 A

เลือกเวลาการทำงานของกรณีโหลดเกิน และความผิดพร่อง โดยการบวกค่า t_{GM} ได้ดังนี้

Case	$I_{pri}(A)$	$I_{sec}(A)$	I_{sec}/CTS	*Operating time (s)*	*TDS*
Overload @R1	320	2.67	0.38	NOP	-
Maximum fault @ R1	8857.4	73.8	10.54	0.99 + 0.3 = 1.23	0.424
Overload @R2	900	7.5	1.07	50.7	½*
Maximum fault @ R2	8857.4	73.8	10.54	1.45	½*

*เลือกค่า TDS = 1/2

– รีเลย์ตัวที่ 4 CT ratio = 150:5, CTS = 6.0 A

เลือกเวลาการทำงานของกรณีโหลดเกิน และความผิดพร่อง โดยการบวกค่า t_{GM} ได้ดังนี้

Case	$I_{pri}(A)$	$I_{sec}(A)$	I_{sec}/CTS	*Operating time (s)*	*TDS*
Overload @R1	171.9*	5.73	0.96	NOP	-
Maximum fault @ R1	1691.8*	56.4	9.4	1.45 + 0.3 =1.75	0.573
Overload @R2	200	6.7	1.1	63.4	1**
Maximum fault @ R2	4831.8	161.1	26.8	2.06	1**

*overload @R3 มีค่ากระแส 900 A แต่ R4 มองเห็นกระ 171.9 A เท่านั้น ผ่านอตราส่วนหม้อแปลง

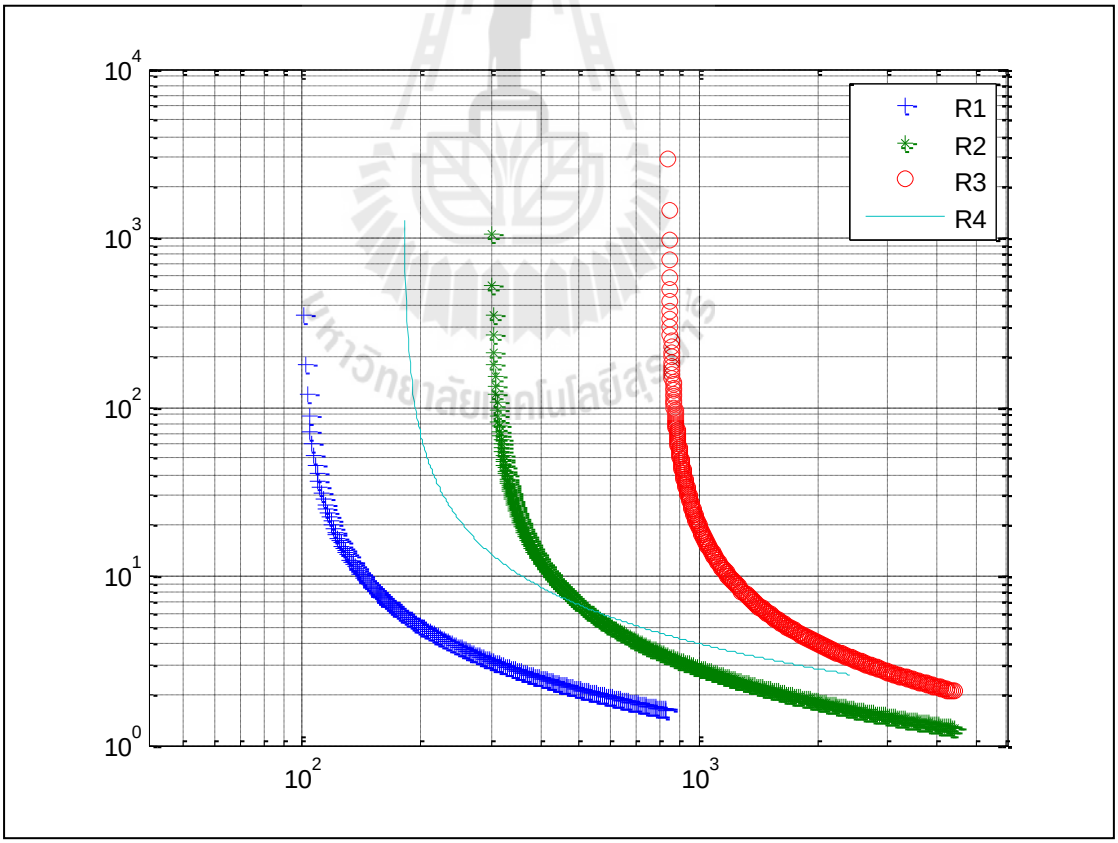
*maximum fault R4มีค่ากระแส 8857.4 A แต่ จะมองเห็นหระแส 1691.8

*สรุป TDS =1

สรุปการปรับตั้ง CT + รีเลย์ ณ ตำแหน่งต่าง ๆ ดังตารางต่อไปนี้

Case	*CT ratio*	*Pickup Setting*	*TDS*	*50 %;max Fault*	*Instant Setting*
R1		5.73	0.96	NOP	-
R2		56.4	9.4	1.45 + 0.3 =1.75	0.573
R3		6.7	1.1	63.4	1**
R4		161.1	26.8	2.06	1**

นำผลจากการปรับตั้งทั้งหมดตัวสร้าง 4 ตัวมากสร้างกราฟคุณลักษณะสมบัติเพื่อเทียบการทำงาน โดยการโอนย้ายกระแสไปทางด้าน 22 kV จะได้ดังรูปต่อไปนี้



รูปที่ 2.23 เส้นโค้งการจัดลำดับของรีเลย์

2.4 การหาค่าเหมาะที่สุด

2.4.1 วิธีเชิงพันธุกรรม (GA)

วิธีเชิงพันธุกรรมเป็นวิธีการซึ่งได้รับการพัฒนาโดย J.H. Holland และเป็นกระบวนวิธีหนึ่งสำหรับใช้ในการหาค่าเหมาะสมที่สุด ซึ่งมีข้อดีเมื่อเปรียบเทียบกับกระบวนวิธีการหาค่าเหมาะสมที่สุดแบบดั้งเดิม คือ สามารถแก้ปัญหาในปริภูมิที่ไม่ต่อเนื่อง (Discontinuous search space) ได้และไม่จำเป็นต้องทราบข้อมูลเกี่ยวกับการหาอนุพันธ์ของฟังก์ชันวัตถุประสงค์ของปัญหา แต่มีข้อเสียคือ ต้องอาศัยการวนซ้ำคำนวณหลาย ๆ รอบ จึงทำให้ต้องใช้เวลาในการคำนวณนาน

วิธีเชิงพันธุกรรมเป็นการเลียนแบบวิวัฒนาการในระดับยีน (ธนัดชัย กุลวรวานิชพงษ์, 2550) ทำได้โดยการสร้างกลุ่มประชากรโครโมโซมแทนผลเฉลย โดยการแปลงโครโมโซมให้อยู่ในระบบเลขฐานสอง จากนั้นประชากรในกลุ่มจะแข่งขันกันเพื่อความอยู่รอด โครโมโซมที่ถูกเลือกในแต่ละรุ่นการถ่ายทอด (Generation) เท่านั้นที่จะมีสิทธิ์สร้างลูกหลานในรุ่นถัดไป โดยมีหลักการดังนี้ เริ่มจากการสุ่มสร้างประชากรในกลุ่มขึ้น จากนั้นคำนวณค่าความแข็งแกร่ง (Fitness) ของประชากรแต่ละตัวจากฟังก์ชันวัตถุประสงค์ ซึ่งฟังก์ชันวัตถุประสงค์จะเปรียบเสมือนกับดัชนีที่ใช้บ่งชี้ถึงความสามารถในการดำรงชีวิตอยู่ภายใต้สิ่งแวดล้อม จากนั้นทำการเลือกสรรประชากรที่มีความแข็งแกร่ง เพื่อนำมาให้กำเนิดประชากรรุ่นถัดไป การให้กำเนิดประชากรรุ่นใหม่ในกลุ่มจะเกิดจากการแลกเปลี่ยนข้อมูลระหว่างรุ่นพ่อแม่ที่ได้รับการคัดสรรมาแล้ว โดยอาศัยสมมติฐานที่ว่าเมื่อนำข้อมูลจากรุ่นพ่อรุ่นแม่ที่มีความแข็งแกร่งมาผสมกันแล้วย่อมมีโอกาสที่จะได้รุ่นลูกที่มีความแข็งแกร่งยิ่งขึ้น ซึ่งคล้ายกับการแลกเปลี่ยนรหัสทางพันธุกรรมของยีน (Gene) ของสิ่งมีชีวิตในธรรมชาติ จากนั้นนำประชากรรุ่นใหม่ที่เกิดขึ้นไปแทนที่ประชากรเดิมบางตัวหรือทุกตัวในกลุ่ม ทำขั้นตอนเหล่านี้ซ้ำไปเรื่อย ๆ จนกระทั่งได้ผลคำตอบที่เป็นค่าเหมาะสมที่สุด ดังนั้นจึงต้องทำความเข้าใจเกี่ยวกับหลักการพื้นฐานก่อนดังนี้

1. การเข้ารหัสโครโมโซม (Chromosome encoding)

การเข้ารหัสโครโมโซมเป็นการแปลงเลขฐานสิบให้เป็นเลขฐานสอง ซึ่งใช้หลักการแปลงสัญญาณแอนะล็อกไปเป็นดิจิตอล โดยการกำหนดให้ตัวแปร x มีค่าพิสัยอยู่ในช่วง x_{min} ถึง x_{max} แล้วทำการแบ่งช่วงจากจุดขอบทั้งสองเป็น n จุดเท่า ๆ กัน ระยะห่างของข้อมูลจะถูกแบ่งออกเป็นส่วน ๆ และสองจุดที่อยู่ติดกันสามารถคำนวณได้จาก $\Delta x = \frac{x_{\max} - x_{\min}}{n-1}$ เนื่องจากใช้เลขฐานสอง จะได้ว่า $n = 2^m$ ดังนั้น

$\Delta x = \frac{x_{\max} - x_{\min}}{2^m - 1}$ เมื่อ m = จำนวนบิตของชุดข้อมูล

2. การดำเนินการทางสายพันธุ์ (Genetic Operator)

การดำเนินการทางสายพันธุ์เป็นการนำเอาสตริงโครโมโซมตั้งแต่ 1 ตัวขึ้นไปมากระทำกัน ซึ่งสามารถนำมากระทำกันได้ 2 วิธีดังนี้

2.1 ครอสโอเวอร์ (Cross Over)

การครอสโอเวอร์เป็นการนำเอาชิ้นส่วนโครโมโซมมาแลกเปลี่ยนกันดังนี้

Chrom 1 = 101001101101

Chrom 2 = 001111100011

ซึ่งการแลกเปลี่ยนชิ้นส่วนโครโมโซมด้วยวิธีการครอสโอเวอร์มีหลายรูปแบบ แต่ในที่นี้จะนำเสนอเฉพาะการครอสโอเวอร์จุดเดียวอย่างง่าย โดยมีขั้นตอนดังนี้

1. ทำการสุ่มตำแหน่งที่เป็นจุดแบ่งของโครโมโซม

$$Chrom1 = \underset{a_1}{10100} - \underbrace{1101101}_{a_2}$$

$$Chrom2 = \underset{b_1}{00111} - \underbrace{1100011}_{b_2}$$

2. ทำการแลกชิ้นส่วน *Chrom 1* กับ *Chrom 2* จะได้ว่า

$$Chrom3 = \underset{a_1}{10100} - \underbrace{1100011}_{b_2} = 101001100011$$

$$Chrom4 = \underset{b_2}{00111} - \underbrace{1101101}_{a_2} = 001111101101$$

2.2 การผ่าเหล่า (Mutation)

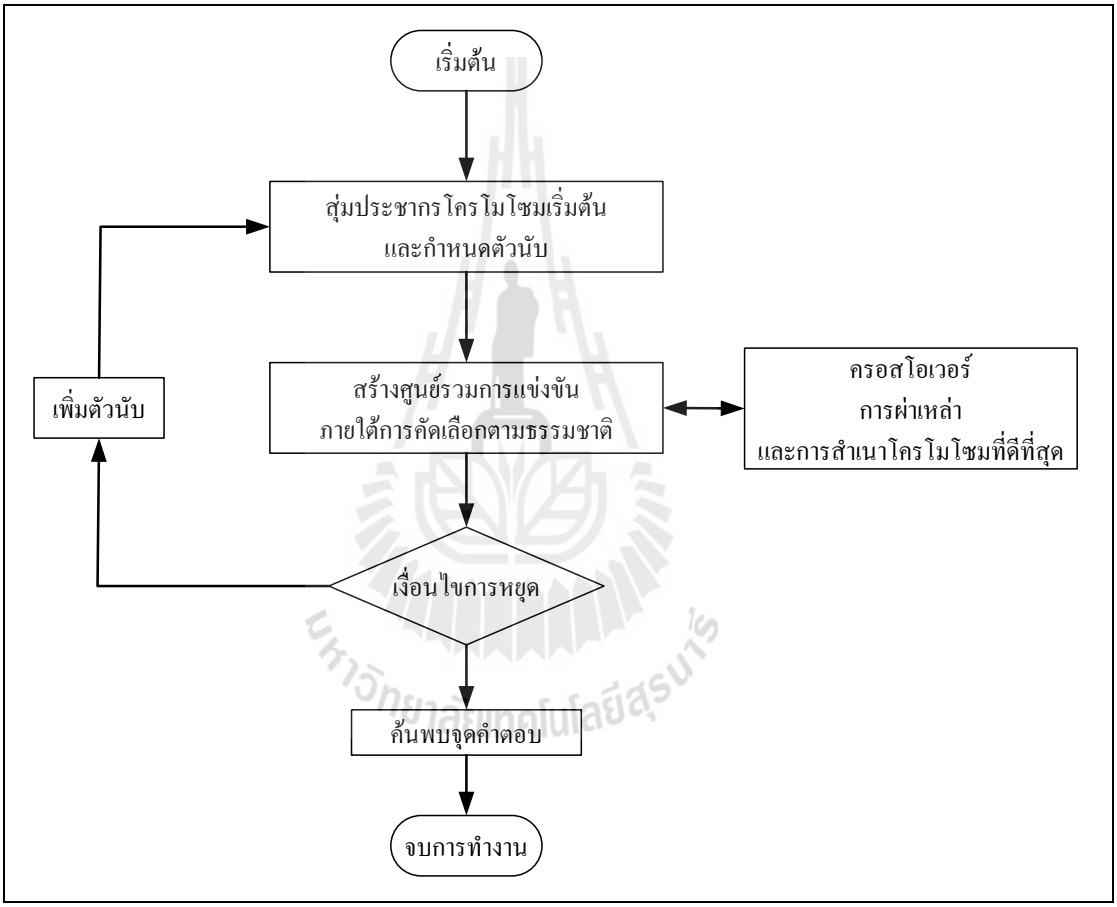
ขั้นตอนนี้จะเปรียบเหมือนการผ่าเหล่าของสิ่งมีชีวิตที่เกิดขึ้นในธรรมชาติ ซึ่งกระบวนการนี้จะใช้กับประชากรที่เกิดขึ้นใหม่เท่านั้น โดยทำการปรับเปลี่ยนข้อมูลของบิตสตริงตำแหน่งใด ๆ หรือหลายตำแหน่งเพื่อให้ข้อมูลมีค่าเปลี่ยนไปจากเดิม ทำให้ได้สายพันธุ์ที่มีรหัสเป็นเลขฐานสองแตกต่างไปจากเดิม ดังนี้

$Chrom1 = 101001101101$

ในการปรับเปลี่ยนข้อมูลของบิตสตริง ทำได้โดยการสุ่มตำแหน่งบิตที่จะเกิดการผ่าเหล่า เช่น *Chrom1* กำหนดให้เกิดการผ่าเหล่าที่ตำแหน่งบิตที่ 7 นั่นก็คือ รหัสเลขฐานสองที่ตำแหน่งดังกล่าวต้องถูกเปลี่ยนแปลงไปจากเดิมดังนี้

*Chrom*1＝1010011̲01101

*Chrom*2＝1010010̲01101



รูปที่ 2.24 แผนผังการทำงานของวิธีเชิงพันธุกรรม

3. การคัดเลือกสายพันธุ์ (Genetic Selection)

การเลือกสรร (Selection) เป็นขั้นตอนสำหรับเลือกประชากรเพื่อใช้ในการให้กำเนิดประชากรรุ่นถัดไป โดยในขั้นตอนแรกจะต้องหาค่าความแข็งแกร่ง (Fitness) ของประชากรแต่ละตัวในกลุ่มจากฟังก์ชันวัตถุประสงค์ จากนั้นจึงเลือกประชากรเพื่อนำมาใช้ในการกำเนิดรุ่นถัดไป

โดยการเลือกสรรแต่ละวิธีจะอยู่ในหลักการที่ว่าประชากรที่มีความแข็งแกร่งมากกว่าย่อมมีโอกาสได้รับการคัดเลือกเพื่อนำไปให้กำเนิดรุ่นถัดไปมากกว่า โดยมีหลักการคัดเลือกสายพันธุ์ คือใช้หลักความน่าจะเป็นมาเป็นตัวช่วย ซึ่งโครโมโซมแต่ละชุดจะมีโอกาสที่จะอยู่รอดสร้างรุ่นถ่ายทอดได้ไม่เท่ากัน โดยจะใช้การวัดค่าความเหมาะสมของการอยู่รอดด้วยค่าความฟิต (fitness value) การคัดเลือกสายพันธุ์ทำได้หลายวิธี เช่น แผนการวงล้อรูเล็ต (roulette-wheel scheme) หรือ แผนภาพทัวร์นาเมนต์ (tournament scheme) เป็นต้น ซึ่งวิธีเชิงพันธุกรรมมีขั้นตอนดังแผนภาพรูปที่ 2.24

2.4.2 วิธีวิวัฒนาการผลต่าง

วิวัฒนาการผลต่างเป็นวิธีการหาค่าเหมาะสมแบบสโตคาสติก มีพื้นฐานจากการหาคำตอบของปัญหา โดยวิธีสุ่มเลือก พัฒนามาจาก GA แต่มีโครงสร้างที่ไม่ซับซ้อน และสามารถประยุกต์ใช้หาคำตอบของปัญหาต่างๆ ได้รวดเร็ว และมีประสิทธิภาพ ขั้นตอนระเบียบวิธีวิวัฒนาการผลต่างแสดงใน รูปที่ 2.25

1. Initialization

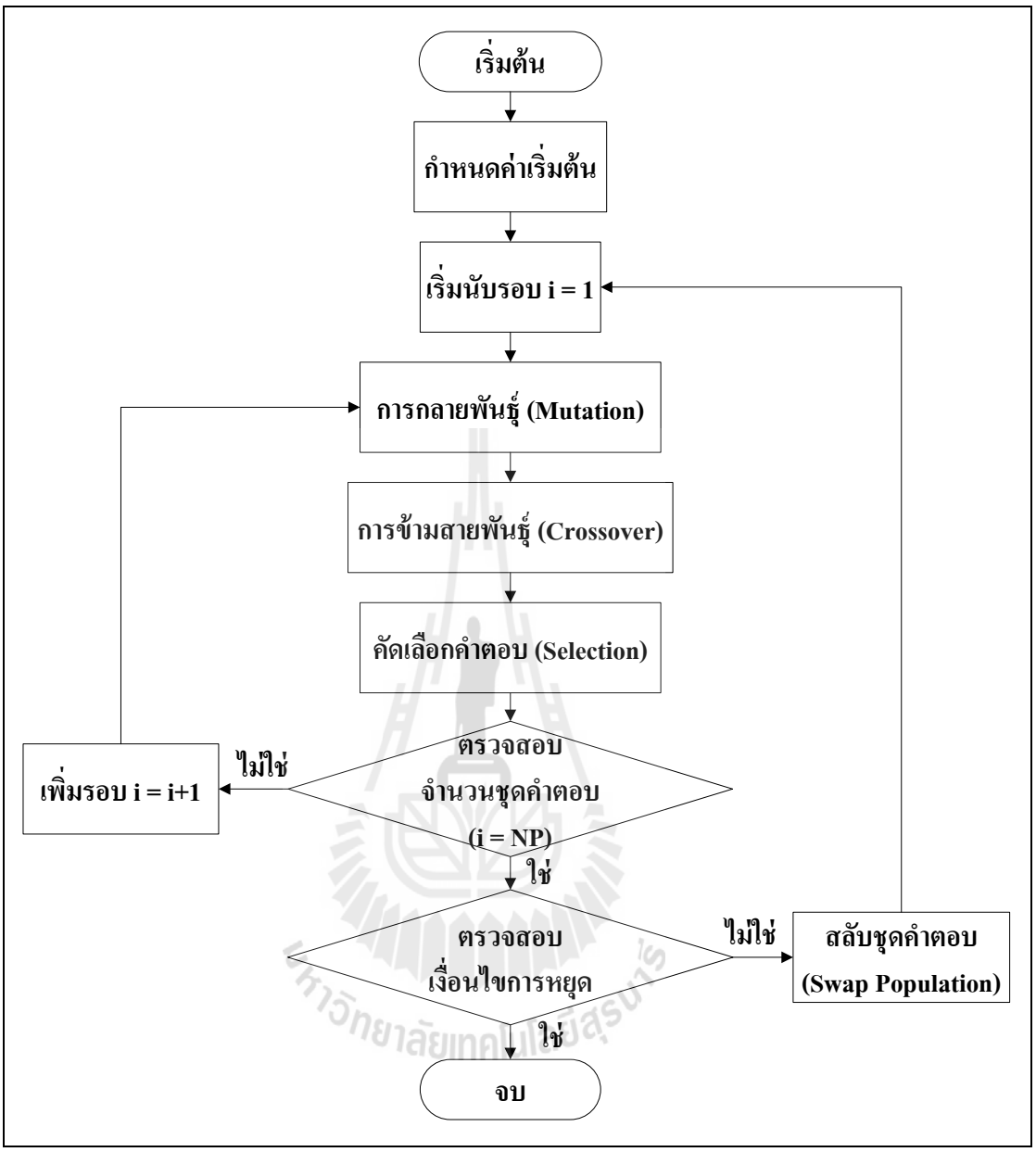
สร้างชุดคำตอบของตัวแปรตัดสินใจ(decision variables. X_i) จำนวน NP ชุด หนึ่งชุดประกอบด้วยตัวแปรตัดสินใจ D ตัว แล้วคำนวณหาค่าฟังก์ชันวัตถุประสงค์ (Objective function value) ของแต่ละชุดคำตอบ

2. Mutation

สร้างเวกเตอร์สัมผัส (target vector. $X_{i,G}$) จำนวน NP ชุด (ชุดละ D ตัว) สุ่มเลือก 3 เวกเตอร์ ($X_{r1,G}$, $X_{r2,G}$, $X_{r3,G}$) ที่ไม่ซ้ำกันกับเวกเตอร์สัมผัสจากชุดคำตอบที่สร้างไว้ เพื่อทำการกลายพันธุ์ (Mutation. $V_{i,G+1}$) โดยใช้ความสัมพันธ์ในสมการที่ 4-1

$$V_{i,G+1} = X_{r1,G} + F(X_{r2,G} - X_{r3,G}) \tag{2.8}$$

เมื่อ F คือ weighing factor มีค่าระหว่าง 0 ถึง 2



รูปที่ 2.25 กระบวนการทำงานของวิวัฒนาการผลต่าง

3. Crossover

เป็นกระบวนการข้ามสายพันธุ์ (crossover) เพื่อให้คำตอบมีความหลากหลายขึ้น สร้างได้จากความสัมพันธ์ตามสมการที่ 4-2

$$U_{ji,G+1} = (U^{1}_{i,G+1}, U^{2}_{i,G+1}, U^{3}_{i,G+1}, \ldots, U^{D}_{i,G+1}) \quad (2.9)$$

$$U_{ji,G+1} = \begin{cases} V_{ji,G+1} & if\ (randb(j) \le CR\ or\ \ j = rnbr(i)) \\ X_{ji,G} & if\ (randb(j) > CR\ or\ \ j \ne rnbr(i)) \end{cases} \quad (2.10)$$

เมื่อ $U_{ji,G+1}$ = เวกเตอร์ทดสอบ(Trial vector)

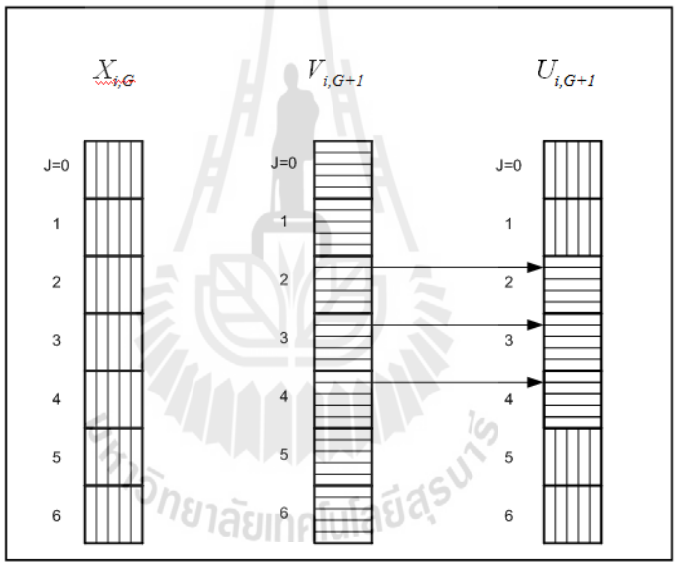
$V_{ji,G+1}$ = Mutant vector

$V_{ji,G}$ = Target vector

randb(j) = การสุ่มตัวจำนวนจริงที่มีค่าระหว่าง 0 ถึง 1 ครั้งที่ j

CR = Crossover Constant มีค่าเป็นเลขจำนวนจริงระหว่าง 0 ถึง 1

rnbr(i) = ค่า Index จากการสุ่มเลือก มีค่าเป็นเลขจำนวนเต็มระหว่าง 0,1,,D-1j = 0,1,…,D-1



รูปที่ 2.26 กระบวนการ crossover ระหว่าง target vector และ mutant vector ที่มีตัวแปรการตัดสินใจ (D) เท่ากับ 7

4. Selection

การคัดสรรคำตอบ ซึ่ง vector ที่ให้คำตอบที่ดีกว่าจะถูกเก็บไว้ต่อไป วิธีการคือเปรียบเทียบ function value ของ trial vector ($U_{i,G+1}$) กับ target vector ($X_{i,G}$) ในกรณีที่ trial vector ให้ค่า function value ที่ดีกว่า มันก็จะแทนที่ target vector ใน generation ต่อไป จากนั้น ทำซ้ำขั้นตอนที่ 2 ถึง 4 จนครบทุกชุดคำตอบ แล้วนำคำตอบชุดใหม่แทนที่คำตอบชุดเก่า ทำซ้ำกระบวนการจนพบเงื่อนไขการหยุด

การทดสอบเพื่อหาพารามิเตอร์ที่เหมาะสมสำหรับวิธี DE

พารามิเตอร์สำคัญสำหรับวิธี DE ประกอบด้วย Max Generation, NP, F และ CR

1. การทดสอบเพื่อหาค่า Max Generation ทำได้โดยการพิจารณาความสัมพันธ์ค่าสมการวัตถุประสงค์ที่คำนวณได้ในแต่ละ Generation กับ Generation ต่างๆ โดยสังเกตการลู่เข้าหาคำตอบซึ่งกราฟความสัมพันธ์ ดังกล่าวจะอยู่ในแนวราบ นั้นคือ ค่าสมการวัตถุประสงค์ที่คำนวณได้จะไม่เปลี่ยนแปลง ในขณะที่จำนวน Generation มากขึ้น

2. การทดสอบค่า NP, F และ CR เพื่อให้ได้ค่าที่เหมาะสมกับฟังก์ชันวัตถุประสงค์สามารถทำได้โดยแปรค่าพารามิเตอร์ตัวใดตัวหนึ่งโดยพารามิเตอร์ตัวที่เหลือคงที่ ในการศึกษานี้ได้ทำการแปรค่า NP ในช่วง 200-500 , ค่า F ในช่วง 0.4 - 0.95 , และค่า CR ในช่วง 0.4 - 0.95

2.4.3 วิธีคล้ายนิวตัน (BFGS)

วิธีการหาทิศทางการลู่เข้าสู่ค่าเหมาะที่สุดมีผลต่อความเร็วในการลู่เข้าสู่คำตอบของปัญหาอย่างมาก วิธีที่ใช้กันอยู่ในปัจจุบันมีหลายวิธี โดยแต่ละวิธีจะมีข้อดีและข้อเสียแตกต่างกันขึ้นอยู่กับลักษณะของปัญหาที่ใช้งาน ในที่นี้จะขอกล่าวถึงเพียงวิธีคล้ายนิวตัน (Quasi-Newton methods)เท่านั้น (พัฒนะ พงศ์จริยา, 2545) วิธีในกลุ่มของวิธีคล้ายนิวตันมีมากมายหลายวิธี แต่ทุกวิธีมีจุดประสงค์เพื่อประมาณค่า $\nabla^2 f(x_k)$ ด้วยเมตริกซ์ B_k ทิศทางการลู่เข้าสู่คำตอบ หาได้จากสมการ
ที่ (2.11)

$$B_k\vec{p}_k = -\nabla f(x_k) \text{ หรือ } \vec{p}_k = -H_k\nabla f(x_k) \tag{2.11}$$

การประมาณค่าของ $\nabla^2 f(x_k)$ เริ่มต้นจาก

$$f''(x_k) \approx \frac{f'(x_k) - f'(x_{k-1})}{x_k - x_{k-1}} \tag{2.12}$$

การประมาณให้เป็นเมตริกซ์ B_k นั้นจะต้องสอดคล้องกับเงื่อนไขซีแคนต์ (Secant condition) ซึ่งคือ

$$B_k(x_k - x_{k-1}) = \nabla f(x_k) - \nabla f(x_{k-1}) \tag{2.13}$$

เนื่องจากเงื่อนไขนี้ไม่เพียงพอที่จะทำให้ B_k มีเพียงแค่รูปแบบเดียวได้ จึงมีวิธีการหาค่า B_k หลายวิธีในที่นี้จะขอยกตัวอย่างการประมาณหาค่า B_k โดยใช้วิธี BFGS (Broyden-Fletcher-Goldfarb-Shanno)

$$B_{k+1} = B_k + \frac{\Delta g_k \Delta g_k^T}{\Delta g_k^T \Delta x_k} - \frac{B_k \Delta x_k \Delta x_k^T B_k}{\Delta x_k^T B_k \Delta x_k} \tag{2.14}$$

การประมาณหาค่า H_k โดยใช้วิธี DFP (Davidon-Fletcher-Powell)

$$H_{k+1} = H_k + \frac{\Delta x_k \Delta x_k^T}{\Delta x_k^T \Delta g_k} - \frac{H_k \Delta g_k \Delta g_k^T H_k}{\Delta g_k^T H_k \Delta g_k} \tag{2.15}$$

โดยที่ $g_k = \nabla f(x_k)$

$\Delta g_k = g_{k+1} - g_k$

$\Delta x_k = \Delta x_{k+1} - x_k$

$B_k^{-1} = H_k$

วิธีนี้มีข้อดีคือไม่ต้องหาค่า $\nabla^2 f(x_k)$ หรือ $\left[\nabla^2 f(x_k)\right]^{-1}$ ซึ่งใช้เวลานาน โดยวิธีคล้ายนิวตันนี้เป็นวิธีของ fminunc.m ซึ่งเป็นชุดคำสั่งสำเร็จรูปในโปรแกรมแมทแลปที่ใช้ในการแก้ปัญหาค่าเหมาะที่สุดแบบไม่มีเงื่อนไขบังคับ

มหาวิทยาลัยเทคโนโลยีสุรนารี

2.5 สรุป

บทที่ 2 ได้กล่าวถึงปริทัศน์วรรณกรรมที่เกี่ยวข้องกับงานวิจัยด้านการป้องกันระบบไฟฟ้ากำลังด้วยรีเลย์ระยะทางและรีเลย์ป้องกันกระแสเกินโดยใช้วิธีชาญฉลาดในการหาผลเฉลยซึ่งเป็นที่มาของงานวิจัยนี้และได้นำเสนอทฤษฎีเกี่ยวกับรีเลย์ระยะทางและรีเลย์ป้องกันกระแสเกินในการการปรับตั้งค่าการทำงานที่เหมาะสมของรีเลย์แต่ละชนิด และในบทถัดไปจะกล่าวถึงการแก้ปัญหาการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทาง ด้วยวิธีชาญฉลาด คือ DE GA และ BFGS

บทที่ 3
ปัญหาการจัดความสัมพันธ์และการหาผลเฉลยของปัญหา

3.1 บทนำ

การปัญหาการจัดความสัมพันธ์รีเลย์ในระบบที่มีขนาดเล็กและไม่ซับซ้อน สามารถคำนวณได้โดยไม่ยุ่งยาก แต่ถ้าระบบไฟฟ้ามีขนาดใหญ่ขึ้น จำนวนรีเลย์ป้องกันย่อมมีจำนวนมากขึ้นด้วย การจัดความสัมพันธ์จึงมีความซับซ้อนยิ่งขึ้น จนบางครั้งไม่อาจคำนวณได้ด้วย มนุษย์จึงจำเป็นต้องสร้างแบบจำลองสมการเพื่อใช้คำตอบของค่าปรับตั้งรีเลย์ ในบทนี้จะได้กล่าวถึงสมการแบบจำลองของรีเลย์ป้องกันกระแสเกิน การหาค่าปรับตั้งรีเลย์ระยะทางโดยนำโปรแกรม MATLAB มาช่วยในการคำนวณ และสมการการจัดความสัมพันธ์รีเลย์ระยะทางและรีเลย์ป้องกันกระแสเกิน สุดท้ายจะได้นำเอาสมการที่ได้จากการจำลองการจัดความสัมพันธ์ของรีเลย์แต่ละชนิดมาหาค่าเหมาะที่สุดด้วย

3.2 ปัญหาการจัดความสัมพันธ์

3.2.1 รีเลย์ป้องกันกระแสเกินกับรีเลย์ป้องกันกระแสเกิน

สมการทางคณิตศาสตร์ของปัญหานี้จะใช้รูปแบบของผลเวลาการทำงาน (time)ของรีเลย์ป้องกันกระแสเกินแบบมีทิศทาง (Direction Over Current Relays. DOCR) ประกอบด้วยค่าตัวแปรที่สำคัญคือ ตัวคูณเวลา (Time dial setting. TDS) ค่ากระแสทำงาน (Plug setting. PS) และกระแสลัดวงจร (Fault current. I_{fault}) ดังนั้นสมการเวลารีเลย์สามารถเขียนได้ดังนี้

$$T = \frac{\alpha * TDS}{(\frac{I}{PS * CT_{pri_rating}})^{\beta} - \gamma} \tag{3.1}$$

จากสมการจะมีเพียง TDS และ PS เท่านั้นที่ไม่ทราบค่า ซึ่งจะใช้ตัดสินใจในการปรับตั้ง (setting) สัญลักษณ์ที่อยู่ในสมการ α, β และ γ คือ ค่าคุณสมบัติของรีเลย์ชนิดนี้ มีค่าดังนี้ $\alpha = 0.14$, $\beta = 0.02$ และ $\gamma = 1$ [IEEE std (1997)] $CT_{\text{pri_rating}}$ คือ กระแสโหลดที่ไหลผ่านหม้อแปลง

กระแส (Current Transformer) ที่ต่อกับรีเลย์ โดยสามารถปรับอัตราส่วนนี้จากระดับกระแสในระบบปกติ จึงเป็นตัวแปรที่ทราบค่า ในการหาค่าต่ำสุดของการจัดลำดับการทำงานของรีเลย์จะใช้เทคนิคการหาค่าต่ำสุดของฟังก์ชันวัตถุประสงค์ สมการวัตถุประสงค์นี้จะอยู่ในรูปของผลรวมของเวลาการทำงานของรีเลย์หลัก (Primary) ซึ่งประกอบด้วย เวลาของรีเลย์ตัวที่อยู่ใกล้ขณะเกิดลัดวงจร (Close_in) และเวลาของรีเลย์ตัวที่อยู่ไกลขณะเกิดลัดวงจร สามารถเขียนสมการวัตถุประสงค์ได้ดังนี้

$$O_{bj} = \sum_{i=1}^{N_{cl}} T^{i}_{pri_cl_in} + \sum_{j=1}^{N_{far}} T^{j}_{pri_far_bus} \tag{3.2}$$

เมื่อ N_{cl} คือ จำนวนของรีเลย์ที่อยู่ใกล้จุดลัดวงจร

N_{far} คือ จำนวนของรีเลย์ที่อยู่ไกลจุดลัดวงจร

$T^{i}_{pri_cl_in}$ คือ เวลาเริ่มทำงานของรีเลย์ที่อยู่ใกล้จุดลัดวงจร

$T^{j}_{pri_far_bus}$ คือ เวลาเริ่มทำงานของรีเลย์ที่อยู่ไกลจุดลัดวงจร

เงื่อนไขของตัวแปร

1. $TDS^{i}_{min} \leq TDS^{i} \leq TDS^{i}_{max}$

   เมื่อ $TDS^{i}_{min} = 0.05$ $TDS^{i}_{max} = 1.1$

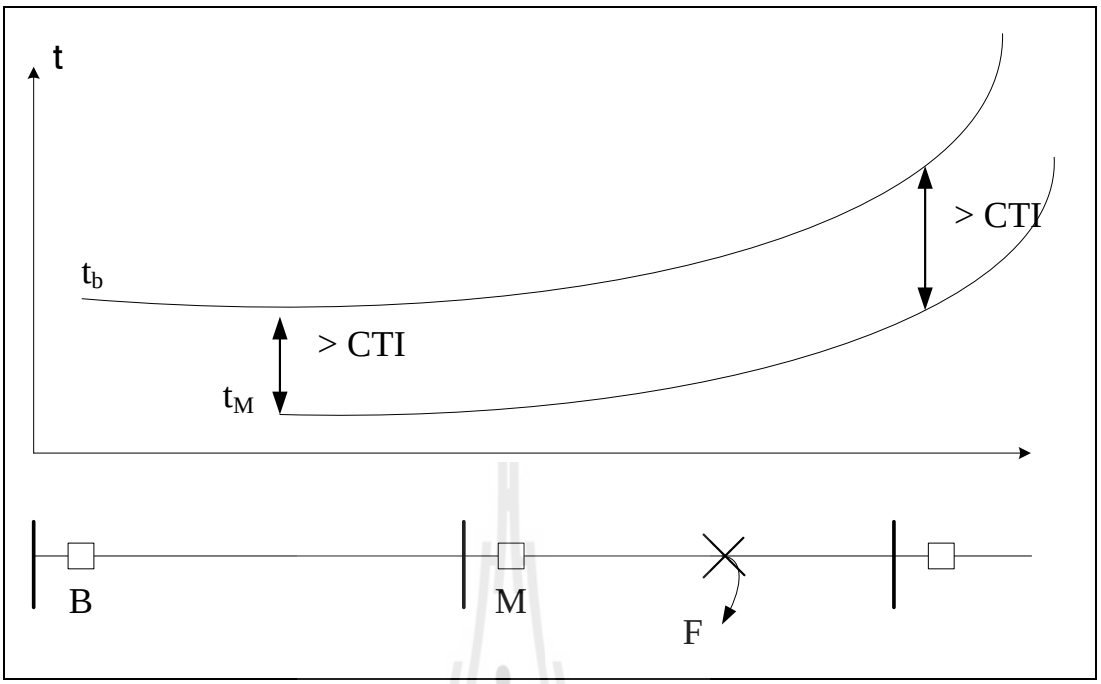
2. $PS^{i}_{min} \leq PS^{i} \leq PS^{i}_{max}$

   เมื่อ $PS^{i}_{min} = 1.25$ $PS^{i}_{max} = 1.50$

3. $T_{primary}$ มีค่าระหว่าง 0.05 ถึง 1.00

4. $T_{backup} - T_{primary} - CTI \geq 0$

เมื่อ T_{backup} คือ เวลาการทำงานของรีเลย์ที่ทำหน้าที่เป็นรีเลย์สำรอง (backup) $T_{primary}$ คือ เวลาการทำงานของรีเลย์ที่ทำหน้าที่เป็นรีเลย์หลัก (primary) และ CTI คือ เวลาการจัดความสัมพันธ์ของรีเลย์ ($CTI = 0.3$)



รูปที่ 3.1 ความสัมพันธ์การทำงานระหว่างรีเลย์ป้องกันกระแสเกินกับรีเลย์ระยะทาง

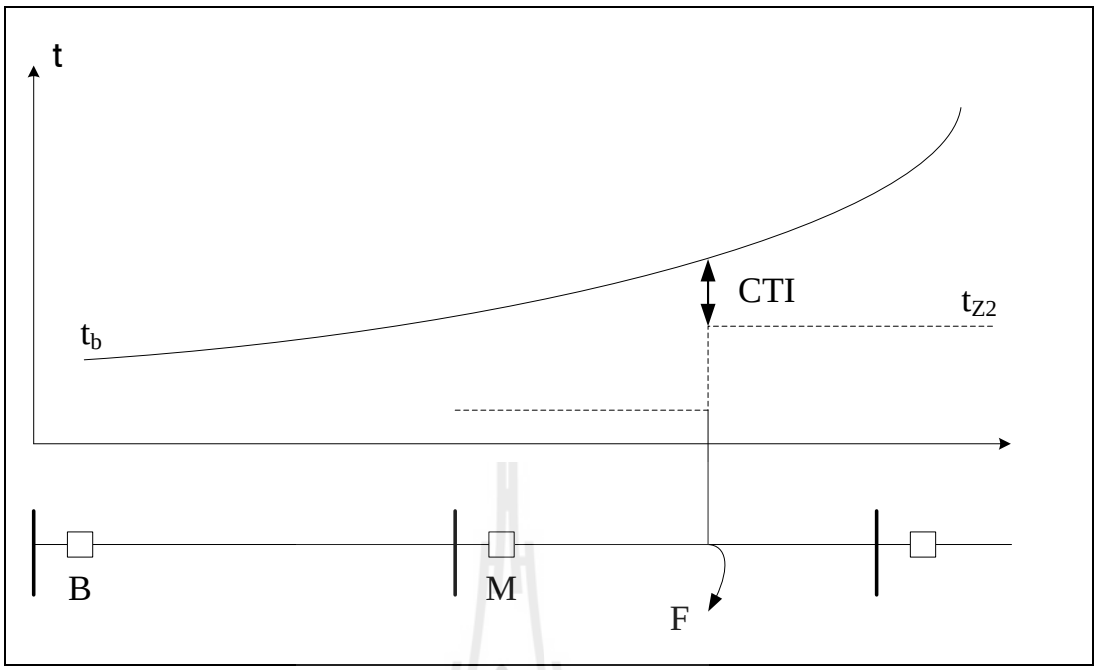
3.2.2 จัดความสัมพันธ์รีเลย์ป้องกันกระแสเกินกับรีเลย์ระยะทาง

จากรูปที่ 3.2 เส้นโค้ง t_{Z2} คือ เส้นโค้งการทำงานของรีเลย์ระยะทาง แสดงความสัมพันธ์ระหว่างค่าอิมพีแดนซ์กับเวลา และเส้นโค้ง t_b คือ เส้นโค้งการทำงานของรีเลย์ป้องกันกระแสเกินแสดงความสัมพันธ์ระหว่างกระแสกับเวลา เป็นการป้องกันระบบไฟฟ้าที่มีรีเลย์ระยะทางทำหน้าที่เป็นรีเลย์ป้องกันหลักและรีเลย์ป้องกันกระแสเกินทำหน้าที่เป็นรีเลย์สำรอง เมื่อเกิดลัดวงจรที่ตำแหน่ง F รีเลย์ระยะทาง (M) ตรวจพบกระแสลัดวงจรและทำหน้าที่ตัดวงจรส่วนที่ลัดวงจรออก แต่ถ้ารีเลย์ ระยะทาง (M) ไม่ทำงานหรือทำงานล้าช้ากว่าที่กำหนดไว้รีเลย์ป้องกันกระแสเกิน (B) จะต้องทำหน้าที่แทนรีเลย์ระยะทาง (M) แต่ต้องไม่ทำงานก่อน ดังสมการที่ 3.3

$$t_b(F) - t_{Z2} > CTI \tag{3.3}$$

t_b(F) คือ เวลาการทำงานของรีเลย์สำรอง

t_{Z2} คือ เวลาการทำงานของรีเลย์หลัก



รูปที่ 3.2 ความสัมพันธ์การทำงานระหว่างรีเลย์ป้องกันกระแสเกินกับรีเลย์ระยะทาง

การจัดความสัมพันธ์ระหว่างรีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทางทำได้โดยการหาค่าเหมาะสุดตามสมการฟังก์ชันวัตถุประสงค์

$$O.F = \alpha \times \sum_{i=1}^{N} (t_i)^2 + \beta_2 \times \sum_{k_1}^{P_1} (\Delta t_{mb|k_1|} - \left|\Delta t_{mb|k_1|}\right|)^2 + \beta_3 \times \sum_{k_2}^{P_2} (\Delta t_{mbDISOC|k_2|} - \left|\Delta t_{mbDISOC|k_2|}\right|)^2 \quad (3.4)$$

เมื่อ $\alpha_1, \beta_2, \beta_3$ คือ ตัวประกอบถ่วงน้ำหนัก

N คือ จำนวนรีเลย์ป้องกันกระแสเกิน

P_1 คือ จำนวนรีเลย์ป้องกันกระแสเกินหลักและสำรอง

P_2 คือ จำนวนรีเลย์ระยะทาง รีเลย์กระแสเกินที่ทำหน้าที่หลักและสำรอง

$\Delta t_{mb|k_1|}$ คือ ผลต่างเวลาทำงานของรีเลย์ป้องกันกระแสเกิน

$\Delta t_{mbDISOC|k_2|}$ คือ ผลต่างเวลาการทำงานรีเลย์ระยะทาง(หลัก)กับรีเลย์ป้องกันกระแสเกิน (สำรอง)

$$\Delta t_{mbDISOC|k_2|} = t_{bOC|k_2|} - t_{mDIS|k_2|} - CTI$$

$t_{bOC|k_2|}$ คือ เวลาทำงานของรีเลย์ป้องกันกระแสเกิน(สำรอง) ขณะเกิดลัดวงจรที่ปลายสาย

$t_{mDIS|k_2|}$ คือ เวลาการทำงานรีเลย์ระยะทาง(หลัก) ทำหน้าที่ป้องกันโซน2

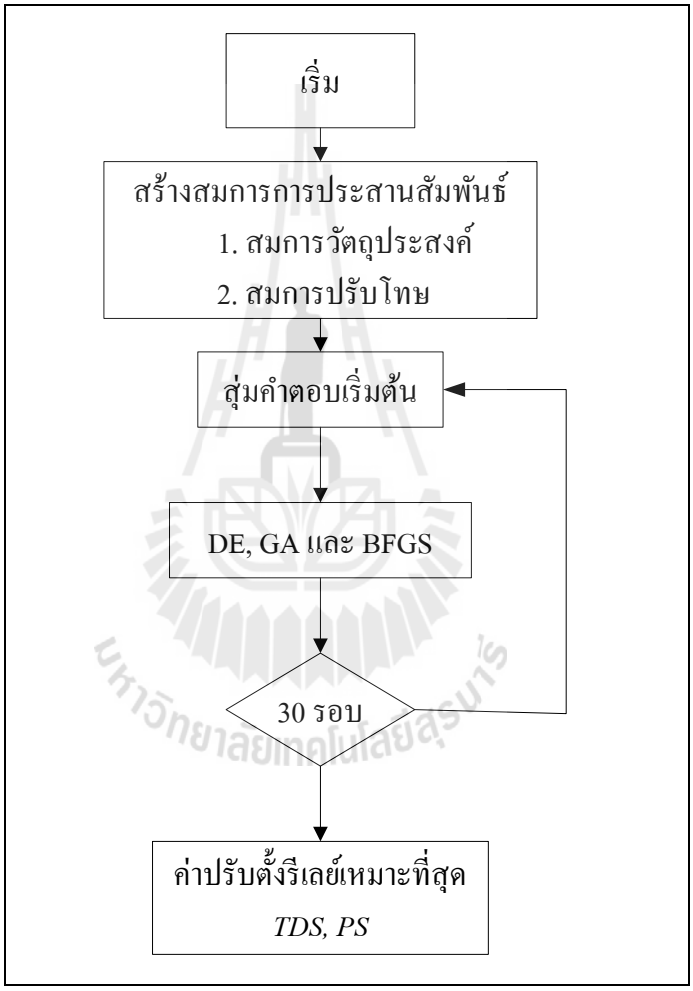
CTI คือ ค่าคงที่ความสำสัมพันธ์รีเลย์ *(CTI = 0.3)*

ฟังก์ชันวัตถุประสงค์ประกอบด้วยผลบวก 3 พจน์ 3 พจน์ พจน์แรกคือ ผลรวมเวลาการทำงานของรีเลย์ป้องกันกระแสเกิน พจน์ที่สองคือ ผลต่างเวลาของรีเลย์ป้องกันกระแสเกินที่ทำหน้าที่ป้องกันหลักและสำรอง และพจน์สุดท้าย หาได้จากผลต่างเวลาของรีเลย์ระยะทางที่ทำหน้าที่หลักและรีเลย์ป้องกันกระแสเกินที่ทำหน้าที่สำรอง

มหาวิทยาลัยเทคโนโลยีสุรนารี

3.3 การแก้ปัญหาการประสานสัมพันธ์รีเลย์

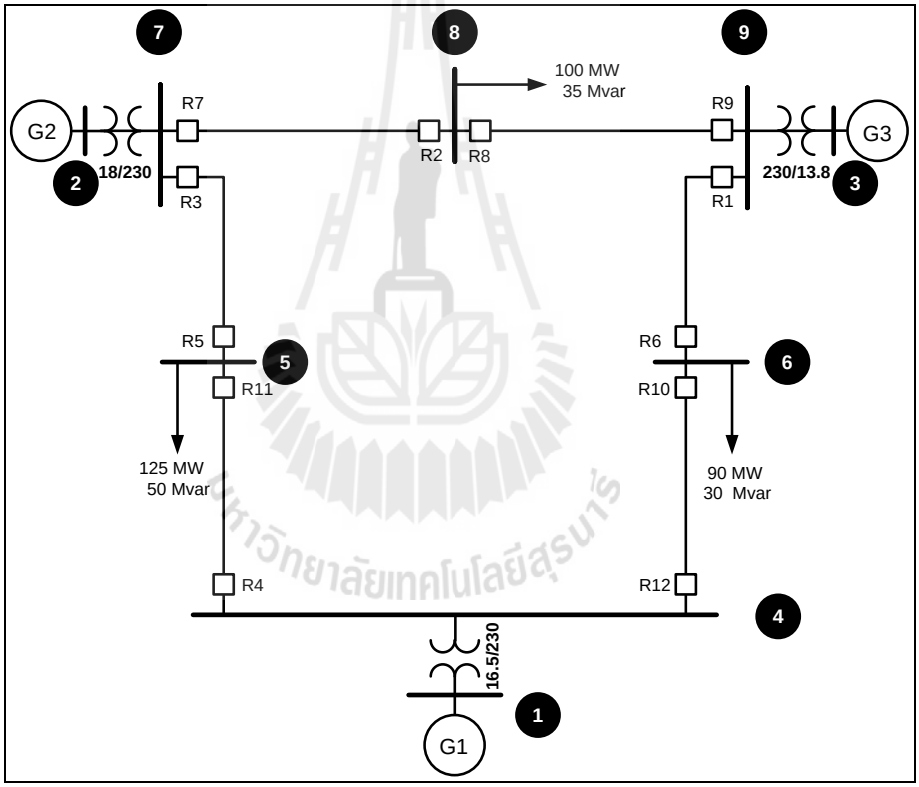
การแก้ปัญหาการประสานสัมพันธ์รีเลย์ทั้งสองแบบ คือ รีเลย์ป้องกันกระแสเกินกับรีเลย์ป้องกันกระแสเกิน และรีเลย์ป้องกันกระแสเกินกับรีเลย์ระยะทางนั้น เริ่มจากสร้างสมการของแต่ละปัญหาแล้วนำมาแก้ด้วยวิธี DE GA และ BFGS ทำซ้ำทั้งหมด 30 รอบ หลังจากนั้นเลือกค่าเหมาะที่สุดจากค่าวัตถุประสงค์ต่ำสุด มีขั้นตอนการแก้ปัญหาดัง รูปที่ 3.3



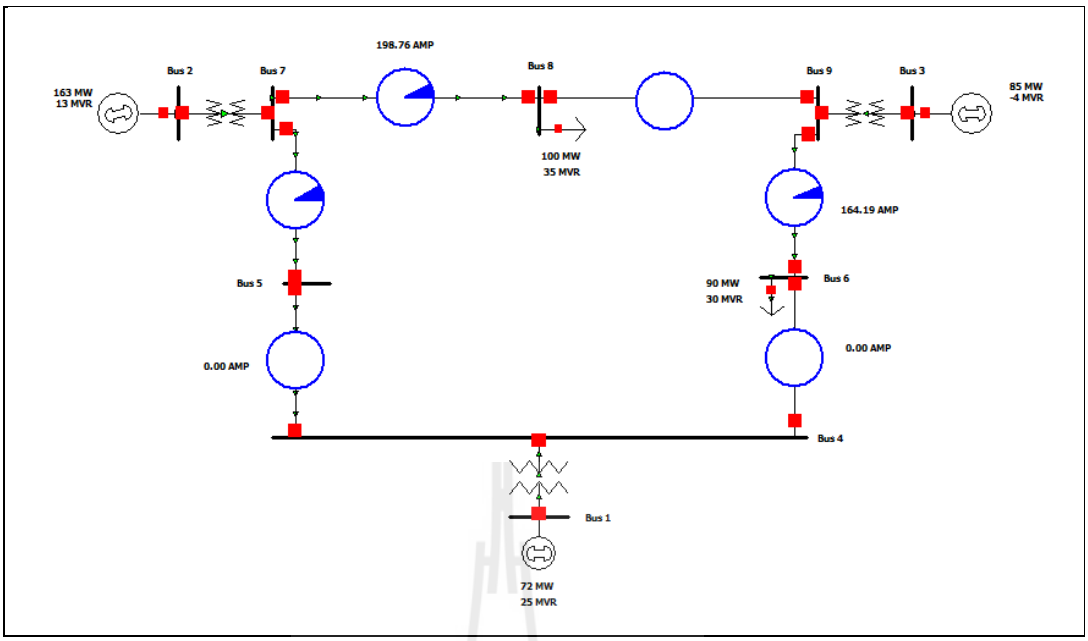
รูปที่ 3.3 แผนผังการแก้ปัญหาค่าเหมาะที่สุด

3.3.1 รีเลย์ป้องกันกระแสเกินกับรีเลย์ป้องกันกระแสเกิน

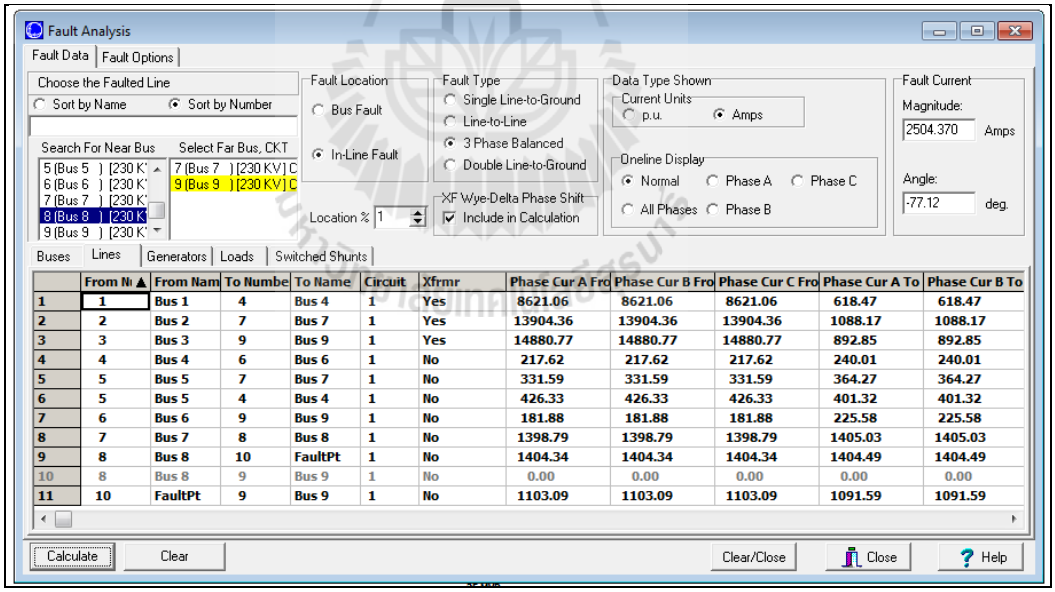
1. สมการวัตถุประสงค์ จากสมการที่ 3.2 จะหาค่า $T^{i}_{pri_cl_in}$ และ $T^{i}_{pri_far_bus}$ ได้จากสมการรีเลย์ป้องกันชนิด เวลาผกผันมาตรฐานดังสมการที่ 3.1 โดยค่า CT หรือค่ากระแสโหลดได้จากการจำลองการไหลโดยใช้โปรแกรม PowerWorld Simulator และค่า I_f หรือกระแสลัดวงจรได้จากการจำลองการลัดวงจรชนิด สมมาตร มีค่ากระแสลัดวงจรลงดินประมาณศูนย์ ซึ่งจะยกตัวอย่างการการจำลองในระบบทดสอบมาตรฐาน WSCC 9 บัส ซึ่งได้ทำการติดตั้งรีเลย์บนสายส่ง 6 เส้น ใช้รีเลย์ทั้งหมด12 ตัว สามารถสรุปค่ากระแสโหลดในสภาวะปกติและกระแสลัดวงจรที่ตำแหน่งรีเลย์แต่ละตัวมองเห็นได้ดังตารางที่ 3.1 ตารางที่ 3.2 และตารางที่ 3.3



รูปที่ 3.4 การติดตั้งรีเลย์ในระบบทดสอบ WSCC 9 บัส



รูปที่ 3.5 การจำลองระบบจ่ายโหลดในสภาวะปกติ (9 บัส)



	From N	From Nam	To Numbe	To Name	Circuit	Xfrmr	Phase Cur A Fro	Phase Cur B Fro	Phase Cur C Fro	Phase Cur A To	Phase Cur B To
1	1	Bus 1	4	Bus 4	1	Yes	8621.06	8621.06	8621.06	618.47	618.47
2	2	Bus 2	7	Bus 7	1	Yes	13904.36	13904.36	13904.36	1088.17	1088.17
3	3	Bus 3	9	Bus 9	1	Yes	14880.77	14880.77	14880.77	892.85	892.85
4	4	Bus 4	6	Bus 6	1	No	217.62	217.62	217.62	240.01	240.01
5	5	Bus 5	7	Bus 7	1	No	331.59	331.59	331.59	364.27	364.27
6	5	Bus 5	4	Bus 4	1	No	426.33	426.33	426.33	401.32	401.32
7	6	Bus 6	9	Bus 9	1	No	181.88	181.88	181.88	225.58	225.58
8	7	Bus 7	8	Bus 8	1	No	1398.79	1398.79	1398.79	1405.03	1405.03
9	8	Bus 8	10	FaultPt	1	No	1404.34	1404.34	1404.34	1404.49	1404.49
10	8	Bus 8	9	Bus 9	1	No	0.00	0.00	0.00	0.00	0.00
11	10	FaultPt	9	Bus 9	1	No	1103.09	1103.09	1103.09	1091.59	1091.59

รูปที่ 3.6 การจำลองการลัดวงจรด้วย PowerWorld Simulator (9 บัส)

ตารางที่ 3.1 ค่ากระแสโหลดที่ไหลผ่านรีเลย์

รีเลย์	ค่ากระแสโหลด (A)
1	167.57
2	203.44
3	204.63
4	80.51
5	204.63
6	167.57
7	203.44
8	75.07
9	75.07
10	78.61
11	80.51
12	78.61

ตารางที่ 3.2 ค่ากระแสลัดวงจร

สายส่ง	ตำแหน่ง Fault	รีเลย์ใกล้		รีเลย์ไกล	
		รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร
4 - 5	4	4	647.03	11	2191.06
	5	11	1347.46	4	882.24
4 - 6	4	12	2252.28	10	572.76
	6	10	840.22	12	1193.94
5 – 7	5	5	1336.29	3	869.6
	7	3	2351.02	5	614.11
6 - 9	6	6	1176.57	1	823.5
	9	1	2358.36	6	550.22
7 - 8	7	7	2164.03	2	814.55
	8	2	1093.9	7	1418.8
8 – 9	8	8	1404.49	9	1091.59
	9	9	2106.65	8	823.22

ตารางที่ 3.3 ค่ากระแสลัดวงจรของเมื่อพิจารณาการทำงานรีเลย์หลักและสำรอง

สายส่ง	ตำแหน่ง Fault	รีเลย์สำรอง		รีเลย์หลัก	
		รีเลย์	ค่ากระแส	รีเลย์	ค่ากระแส
4 – 5	4	11	638	12	2252
	5	3	648	11	2191
4 – 6	4	10	566	4	647
	6	1	564	10	573
5 – 7	5	4	1329	5	1336
	7	2	217	3	870
6 – 9	6	12	1177	6	1177
	9	8	175	1	824
7 – 8	7	5	606	7	2164
	8	9	821	2	815
8 – 9	8	7	1405	8	1404
	9	6	226	9	1092

2. สมการปรับโทษ สมการปรับโทษประกอบด้วย ขอบของตัวแปร *TDS PS* ขอบเวลาเริ่มทำงานของรีเลย์หลัก (T_{pri}) และเงื่อนไขการประสานสัมพันธ์ $T_{backup} - T_{primary} - CTI \geq 0$ โดยเงื่อนไขการประสานคำนวณได้จากค่ากระแสลัดวงจรที่รีเลย์หลักและรีเลย์สำรองแต่ละตัวอ่านได้ดังตารางที่ 3.1

3.3.2 รีเลย์ป้องกันกระแสเกินรีเลย์ระยะทางกับรีเลย์ระยะทาง

1. สมการวัตถุประสงค์ จากสมการที่ 3.4 เป็นสมการวัตถุประสงค์ของการประสานสัมพันธ์รีเลย์ป้องกันกระแสกับรีเลย์เกินระยะทาง สมการประกอบด้วยพจน์ 3 พจน์ คือ

1.1 ผลรวมเวลาของรีเลย์ป้องกันกระแสเกินหาค่า $T^{i}_{pri_cl_in}$ และ $T^{i}_{pri_far_bus}$ ได้จากสมการรีเลย์ป้องกันชนิด แบบเวลาผกผันมาตรฐานดังสมการที่ 3.1 โดยค่าCT หรือค่ากระแสโหลดได้จากการจำลองการไหลโดยใช้โปรแกรม POWER WORLD และค่า I หรือกระแสลัดวงจรได้จากการจำลองการลัดวงจรชนิด สมมาตร มีค่ากระแสลัดวงจรลงดินประมาณศูนย์ เช่นเดียวกับ สมการวัตถุประสงค์ของการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินกับรีเลย์ป้องกันกระแสเกิน

1.2 ผลต่างเวลาของรีเลย์ป้องกันกระแสเกินที่ทำหน้าที่หลักและสำรอง หาค่า $\Delta t_{mb|k_1|}$ ได้จากจำลองกระแสลัดวงจรเช่นเดียวกับ เงื่อนไขประสานสัมพันธ์ของรีเลย์ป้องกันกระแสเกินก่อนหน้านี้

1.3 ผลต่างเวลาทำงานของรีเลย์ระยะทางและรีเลย์รีเลย์ป้องกันกระแสเกิน ($\Delta t_{mbDISOC|k_2|}$) หาค่าได้จากสมการ $\Delta t_{mbDISOC|k_2|} = t_{bOC|k_2|} - t_{mDIS|k_2|} - CTI$ และสามารถหาค่า $t_{bOC|k_2|}$ ได้จากสมการที่ 1 ส่วนค่า $t_{mDIS|k_2|}$ และ ค่า *CTI* เป็นค่าคงที่ดังนี้ $t_{mDIS|k_2|}$ เป็นค่าเริ่มทำงานโซนที่ 2 ของรีเลย์ระยะทาง งานวิจัยนี้กำหนดให้รีเลย์ทุกตัวมีค่าเท่ากับ 0.3 และCTI เป็นค่าการประสานสัมพันธ์รีเลย์มีค่าเท่ากับ 0.3

2. สมการปรับโทษ สมการปรับโทษประกอบด้วย ขอบของตัวแปร *TDS PS* ขอบเวลาเริ่มทำงานของรีเลย์หลัก (T_{pri})

3.4 สรุป

ในบทที่ 3 นี้ได้แสดงรูปแบบของปัญหาพร้อมทั้งขั้นตอนการสร้างสมการของปัญหาเพื่อนำไปหา 2 ปัญหา คือ ปัญหาการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกิน และปัญหาการประสานสัมพันธ์รีเลย์ระยะทางและรีเลย์ป้องกันกระแสเกิน ในบทต่อไปจะนำเสนอผลการหาคำตอบด้วยวิธีการ DE GA และ BFGS และผลการหาคำตอบในระบบที่ใหญ่ขึ้นคือ 14 บัส และ 30 บัสต่อไป

มหาวิทยาลัยเทคโนโลยีสุรนารี

บทที่ 4
ผลการทดสอบ

4.1 บทนำ

จากบทที่ 3 กล่าวถึงการสร้างสมการสำหรับแก้ปัญหาการจัดความสัมพันธ์รีเลย์แบบต่าง ๆ แล้ว ในบทนี้จะได้ทำการหาค่าเหมาะที่สุดของจัดการความสัมพันธ์รีเลย์ โดยทำการหาค่าการประสานสัมพันธ์รีเลย์ 3 แบบ คือ 1. ค่าเหมาะที่สุดของการจัดความสัมพันธ์รีเลย์ระยะทาง 2. ค่าการจัดความสัมพันธ์รีเลย์ป้องกันกระแสเกิน 3. ค่าเหมาะที่สุดของการจัดความสัมพันธ์รีเลย์ระยะทางและรีเลย์ป้องกันกระแสเกิน โดยใช้วิธี DE GA และ BFGSในระบบมาตรฐาน 3 ระบบ คือ WSCC 9 Bus IEEE 14 Bus และ IEEE 30Bus

4.2 ค่าปรับตั้งรีเลย์ป้องกันระยะทาง

4.2.1 ระบบทดสอบมาตรฐาน WSCC 9 บัส

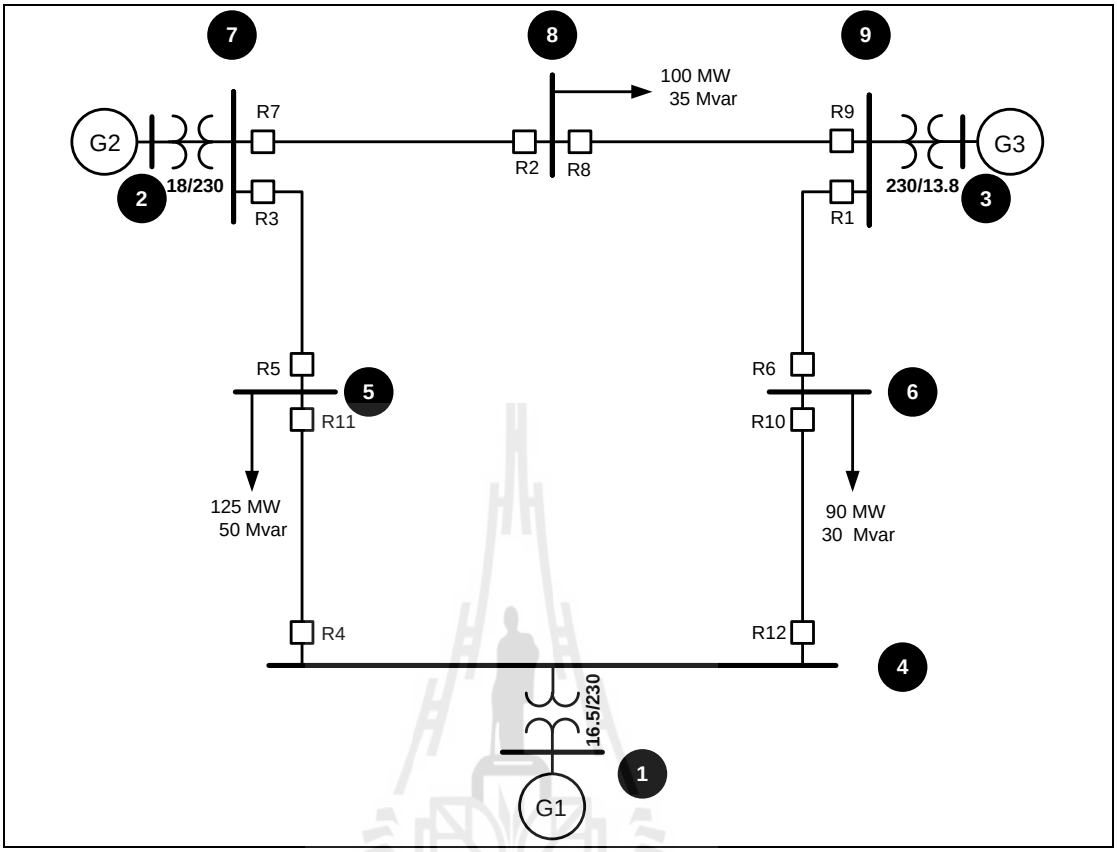
สำหรับระบบทดสอบ WSCC 9 บัส เป็นระบบชนิดโครงข่ายเชื่อมต่อกัน ประกอบด้วยสายส่ง 6 เส้น ทำการติดตั้งรีเลย์ทั้งหมด 12 ตัว บริเวณบัส การปรับค่าการทำงานแบ่งเป็น 3 โซน

โซนที่ 1 ปรับตั้งเท่ากับ 80% ของความต้านทานสายส่ง

โซนที่ 2 ปรับตั้ง เท่ากับ 100%(โซน 1) + 50% (โซน 2)

โซนที่ 3 ปรับตั้งเท่ากับ 100%(โซน 1) + 100%(โซน 2) + 20%(โซน 3)

จากที่ระบบทดสอบเป็นระบบโครงข่ายจึงจำเป็นต้องคิดค่าปรับตั้งโดยพิจารณาผลของหลายแหล่ง (Infeed constant) มีผลการคำนวณ Infeed บัส และผลการปรับตั้งค่ารีเลย์ระยะทางของระบบทดสอบ WSCC 9 บัส



รูปที่ 4.1 ระบบทดสอบมาตรฐาน WSCC 9 บัส

ใช้โค้ด MATLAB ช่วยในการหาโซนการป้องกัน (Zone1 Zone2 และ Zone3) และหาค่าปรับตั้งรีเลย์ระยะทางในระบบ ได้ผลดังตารางที่ 4.1 – 4.2

ตารางที่ 4.1 ผลการจัดโซนป้องกันรีเลย์ระยะทาง ระบบ 9 บัส

รีเลย์	โซนการป้องกันสายส่ง		
	โซน1	โซน2	โซน3
1	9 - 6	9 - 6 - 4	9 - 6 - 4 - 5
2	8 - 7	8 - 7 - 5	8 - 7 - 5 - 4
3	7 - 5	7 - 5 - 4	7 - 5 - 4 - 6
4	4 - 5	4 - 5 - 7	4 - 5 - 7 - 8

ตารางที่ 4.1 ผลการจัดโซนป้องกันรีเลย์ระยะทาง ระบบ 9 บัส (ต่อ)

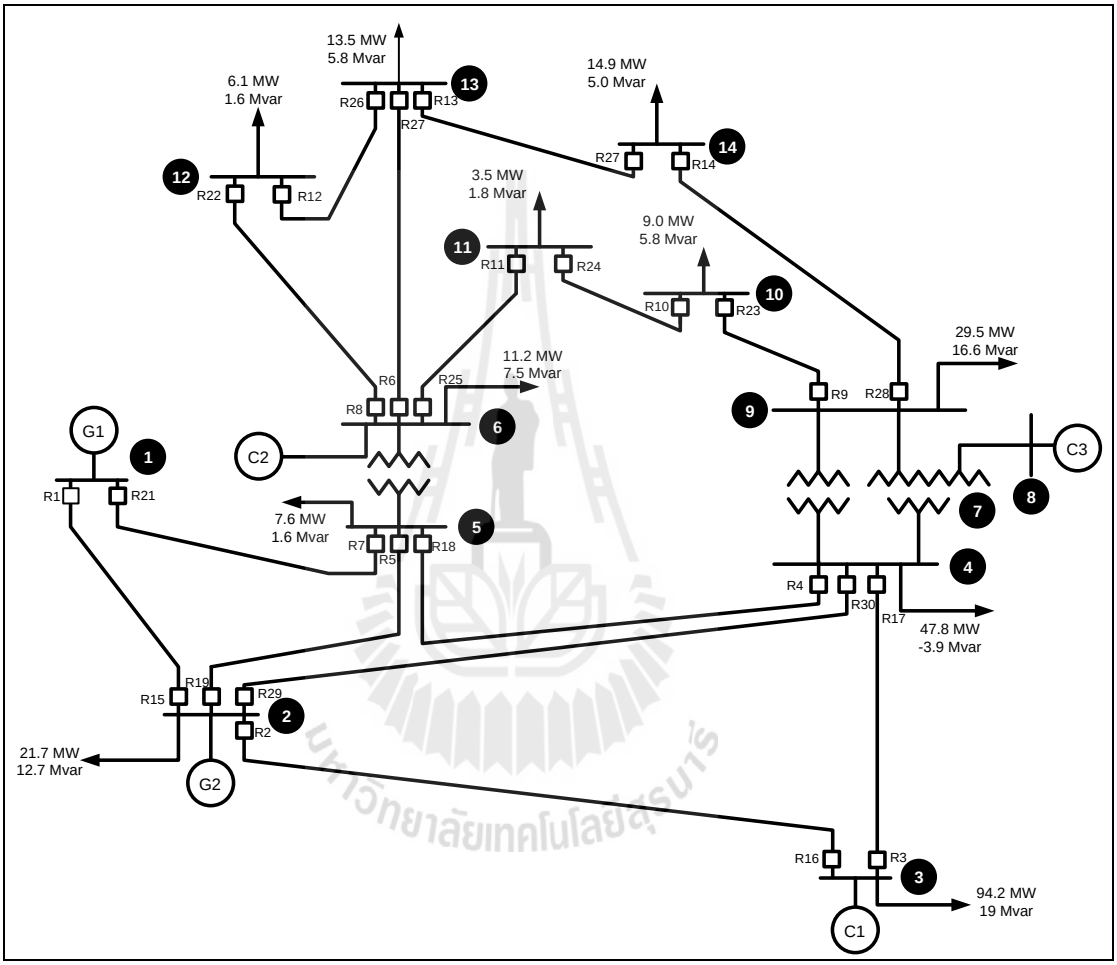
รีเลย์	โซนการป้องกันสายส่ง		
	โซน1	โซน2	โซน3
5	5 - 7	5 - 7 - 8	5 - 7 - 8 - 9
6	6 - 9	6 - 9 - 8	6 - 9 - 8 - 7
7	7 - 8	7 - 8 - 9	7 - 8 - 9 - 6
8	8 - 9	8 - 9 - 6	8 - 9 - 6 - 4
9	9 - 8	9 - 8 - 7	9 - 8 - 7 - 5
10	6 - 4	6 - 4 - 5	6 - 4 - 5 - 7
11	5 - 4	5 - 4 - 6	5 - 4 - 6 - 9
12	4 - 6	4 - 6 - 9	4 - 6 - 9 - 8

ตารางที่ 4.2 ค่าปรับตั้งรีเลย์ระยะทาง ระบบ 9 บัส

รีเลย์	โซน 1 (Ω)		โซน 2 (Ω)		โซน 3 (Ω)	
	R	jX	R	jX	R	jX
1	1.4869	77.35	2.3465	77.81	3.0134	78.31
2	0.486	81.61	1.4635	79.94	52.814	79.06
3	1.3703	78.76	2.0711	79.26	2.6735	79.6
4	0.5738	81.63	1.5731	80.07	2.5812	79.72
5	1.3703	78.76	2.0163	79.19	20.7907	81.83
6	1.4869	77.35	2.386	78.66	19.3848	82.51
7	0.486	81.61	1.137	82.38	2.1296	81.5
8	0.8473	83.27	1.9858	80.5	50.4868	77.72
9	0.8473	83.27	1.3628	82.9	2.0939	81.86
10	0.781	79.53	1.3347	80.1	18.4449	80.5
11	0.5738	81.63	1.2051	80.78	27.5118	78.9
12	0.781	79.53	1.9052	78.47	3.0983	78.54

4.2.2 ระบบทดสอบมาตรฐาน IEEE 14 บัส

ระบบนี้เป็นระบบทดสอบ ขนาด 14 บัส มีสายส่ง 15 เส้น จึงออกแบบระบบป้องกันโดยใช้รีเลย์ระยะทางทั้งหมด 30 ตัว ติดตั้งที่บริเวณใกล้บัส ทำการเลือกโซนป้องกันและค่าปรับ ตั้งด้วยโค้ดโปรแกรม MATLAB ได้ผลแสดงดังตารางที่ 4.3-4.4



รูปที่ 4.2 ระบบทดสอบมาตรฐาน IEEE 14 บัส

ตารางที่ 4.3 ผลการจัดโซนป้องกันรีเลย์ระยะทาง ระบบ 14 บัส

รีเลย์	โซนการป้องกัน		
	โซน1	โซน2	โซน3
1	1 - 2	1 - 2 - 5	1 - 2 - 3 - 4
2	2 - 3	2 - 3 - 4	2 - 3 - 4 - 5
3	3 - 4	3 - 4 - 5	3 - 4 - 2 - 1
4	4 - 5	4 - 5 - 2	4 - 5 - 1 - 2
5	5 - 2	5 - 2 - 1	5 - 2 - 3 - 4
6	6 - 13	6 - 13 - 12	6 - 13 - 14 - 9
7	5 - 1	5 - 1 - 2	5 - 1 - 2 - 5
8	6 - 12	6 - 12 - 13	6 - 12 - 13 - 6
9	9 - 10	9 - 10 - 11	9 - 10 - 11 - 6
10	10 - 11	10 - 11 - 6	10 - 11 - 6 - 13
11	11 - 6	11 - 6 - 13	11 - 6 - 12 - 13
12	12 - 13	12 - 13 - 6	12 - 13 - 14 - 9
13	13 - 14	13 - 14 - 9	13 - 14 - 9 - 10
14	14 - 9	14 - 9 - 10	14 - 9 - 10 - 11
15	2 - 1	2 - 1 - 5	2 - 1 - 5 - 4
16	3 - 2	3 - 2 - 1	3 - 2 - 4 - 5
17	4 - 3	4 - 3 - 2	4 - 3 - 2 - 1
18	5 - 4	5 - 4 - 3	5 - 4 - 2 - 1
19	2 - 5	2 - 5 - 4	2 - 5 - 1 - 2
20	13 - 6	13 - 6 - 11	13 - 6 - 12 - 13
21	1 - 5	1 - 5 - 4	1 - 5 - 2 - 1
22	12 - 6	12 - 6 - 13	12 - 6 - 11 - 10
23	10 - 9	10 - 9 - 14	10 - 9 - 14 - 13
24	11 - 10	11 - 10 - 9	11 - 10 - 9 - 14
25	6 - 11	6 - 11 - 10	6 - 11 - 10 - 9

มหาวิทยาลัยเทคโนโลยีสุรนารี

ตารางที่ 4.3 ผลการจัดโซนป้องกันรีเลย์ระยะทาง ระบบ 14 บัส (ต่อ)

รีเลย์	โซนการป้องกัน		
	โซน1	โซน2	โซน3
26	13 - 12	13 - 12 - 6	13 - 12 - 6 - 13
27	14 - 13	14 - 13 - 6	14 - 13 - 12 - 6
28	9 - 14	9 - 14 - 13	9 - 14 - 13 - 6
29	2 - 4	2 - 4 - 5	2 - 4 - 3 - 2
30	4 - 2	4 - 2 - 1	4 - 2 - 3 - 4

ตารางที่ 4.4 ค่าปรับตั้งรีเลย์ระยะทาง ระบบ 14 บัส

รีเลย์	โซน 1 (Ω)		โซน 2 (Ω)		โซน 3 (Ω)	
	R	jX	R	jX	R	jX
1	0.5198	71.86	9.0948	71.87	25.7177	76.3
2	1.6985	76.65	3.5887	73.35	5.2761	71.99
3	1.5334	68.6	2.3499	69.31	7.75	70.98
4	0.3688	72.41	2.6939	71.96	8.2869	74.76
5	1.5274	71.87	2.5569	71.87	45.9477	76.32
6	1.2196	63.08	5.0332	48.35	9.4548	63.8
7	1.9158	76.38	7.5375	73.3	16.9066	72.5
8	2.3692	64.34	4.4396	56.74	6.4567	53.76
9	0.7537	69.37	2.0314	68.03	3.6952	67.13
10	1.7435	66.87	3.3288	66.04	6.6191	64.81
11	1.84	64.47	6.5838	63.57	68.5455	64.1
12	2.487	42.14	4.0725	47.25	19.9701	60.58
13	3.2367	63.84	5.6045	64.11	28.1184	68.07
14	2.4941	64.82	45.0611	69.06	87.5578	69.19
15	0.5198	71.86	75.5376	76.34	150.6595	76.36
16	1.6985	76.65	2.7785	75.52	38.1623	72.03
17	1.5334	68.6	3.4415	72.18	5.2998	73.45

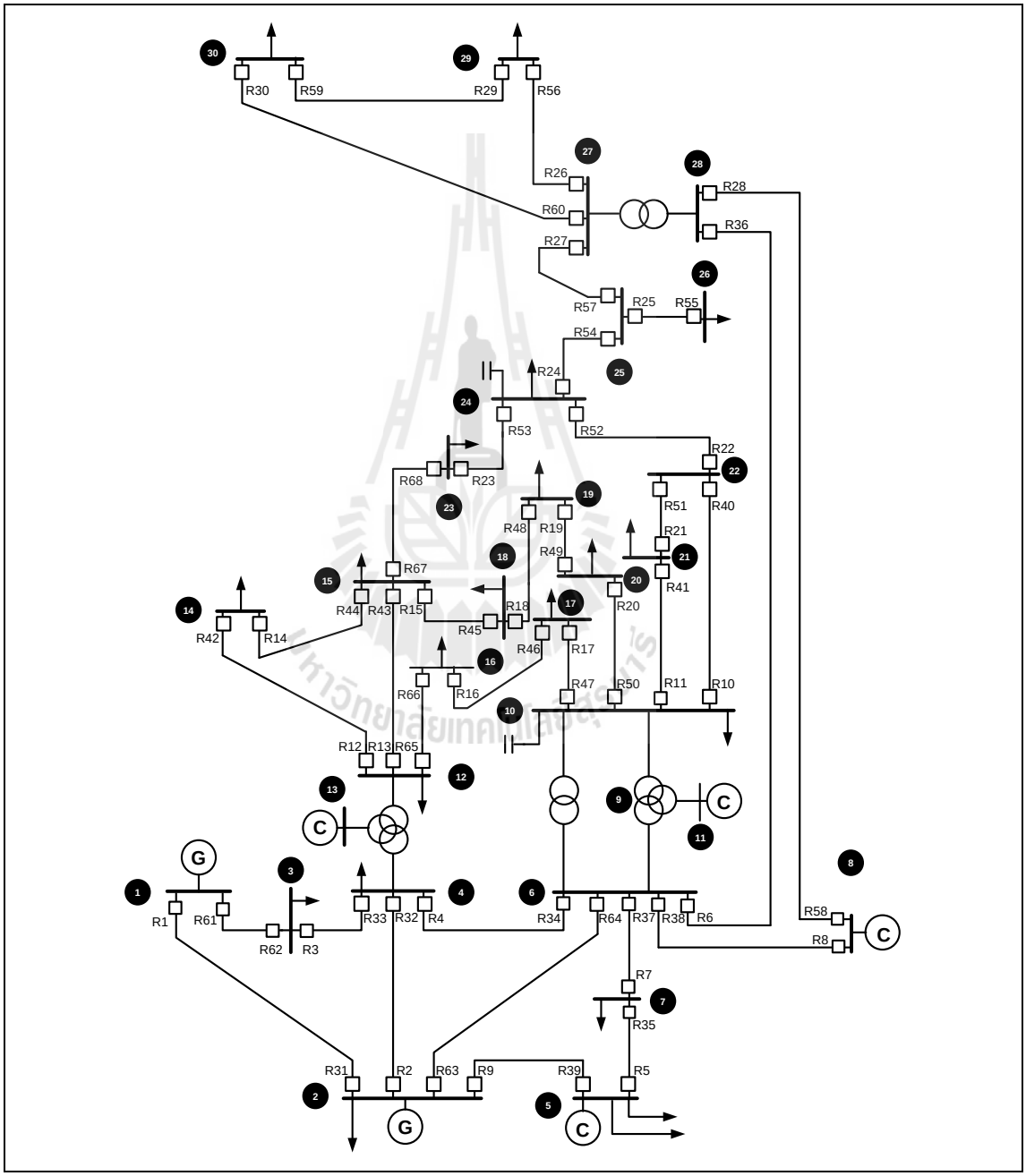
ตารางที่ 4.4 ค่าปรับตั้งรีเลย์ระยะทาง ระบบ 14 บัส (ต่อ)

รีเลย์	โซน 1 (Ω)		โซน 2 (Ω)		โซน 3 (Ω)	
	R	jX	R	jX	R	jX
18	0.3688	72.41	3.4614	69.11	4.5818	71.83
19	1.5274	71.87	2.3864	71.97	10.297	74.42
20	1.2196	63.08	5.3512	64.08	79.2519	64.1
21	1.9158	76.38	2.8589	75.74	7.5335	73.3
22	2.3692	64.34	8.5646	63.51	12.024	64.55
23	0.7537	69.37	220.2697	64.84	440.6116	64.83
24	1.7435	66.87	2.6501	67.31	112.7826	64.9
25	1.84	64.47	3.389	65.24	4.7135	65.83
26	2.487	42.14	4.5145	49.26	8.7289	56.2
27	3.2367	63.84	5.0017	63.7	19.871	47.27
28	2.4941	64.82	5.1403	64.44	7.6411	64.19
29	1.5498	71.76	2.3139	71.87	10.2629	69.8
30	1.5498	71.76	2.5905	71.79	46.3528	76.32

มหาวิทยาลัยเทคโนโลยีสุรนารี

4.2.3 ระบบทดสอบมาตรฐาน IEEE 30บัส

ระบบนี้เป็นระบบทดสอบ ขนาด 30 บัส มีสายส่ง 34 เส้น จึงออกแบบระบบโดยใช้รีเลย์ระยะทางทั้งหมด 68 ตัว ติดตั้ง ที่บริเวณใกล้บัส ทำการเลือกโซนป้องกัน และค่าปรับ ตั้งด้วยโค้ดโปรแกรมแมทแลปได้ผลแสดงดังตารางที่ 4.5 และ 4.6



รูปที่ 4.3 ระบบทดสอบมาตรฐาน IEEE 30 บัส

ตารางที่ 4.5 ผลการจัดโซนป้องกันรีเลย์ระยะทาง ระบบ 30 บัส

รีเลย์	โซนการป้องกัน		
	โซน 1	โซน 2	โซน 3
1	1 - 2	1 - 2 - 4	1 - 2 - 5 - 7
2	2 - 4	2 - 4 - 3	2 - 4 - 6 - 8
3	3 - 4	3 - 4 - 6	3 - 4 - 2 - 1
4	4 - 6	4 - 6 - 8	4 - 6 - 2 - 1
5	5 - 7	5 - 7 - 6	5 - 7 - 6 - 4
6	6 - 28	6 - 28 - 8	6 - 28 - 8 - 6
7	7 - 6	7 - 6 - 4	7 - 6 - 2 - 1
8	8 - 6	8 - 6 - 4	8 - 6 - 2 - 1
9	2 - 5	2 - 5 - 7	2 - 5 - 7 - 6
10	10 - 22	10 - 22 - 21	10 - 22 - 24 - 23
11	10 - 21	10 - 21 - 22	10 - 21 - 22 - 10
12	12 - 14	12 - 14 - 15	12 - 14 - 15 - 12
13	12 - 15	12 - 15 - 23	12 - 15 - 14 - 12
14	14 - 15	14 - 15 - 12	14 - 15 - 18 - 19
15	15 - 18	15 - 18 - 19	15 - 18 - 19 - 20
16	16 - 17	16 - 17 - 10	16 - 17 - 10 - 21
17	17 - 10	17 - 10 - 21	17 - 10 - 20 - 19
18	18 - 19	18 - 19 - 20	18 - 19 - 20 - 10
19	19 - 20	19 - 20 - 10	19 - 20 - 10 - 21
20	20 - 10	20 - 10 - 21	20 - 10 - 22 - 21
21	21 - 22	21 - 22 - 10	21 - 22 - 24 - 23
22	22 - 24	22 - 24 - 23	22 - 24 - 25 - 27
23	23 - 24	23 - 24 - 22	23 - 24 - 25 - 27
24	24 - 25	24 - 25 - 27	24 - 25 - 26 - 0
25	25 - 26	25 - 0 - 0	25 - 0 - 0 - 0

ตารางที่ 4.5 ผลการจัดโซนป้องกันรีเลย์ระยะทาง ระบบ 30 บัส (ต่อ)

รีเลย์	โซนการป้องกัน		
	โซน 1	โซน 2	โซน 3
26	27 - 29	27 - 29 - 30	27 - 29 - 30 - 27
27	27 - 25	27 - 25 - 24	27 - 25 - 26 - 0
28	28 - 8	28 - 8 - 6	28 - 8 - 6 - 4
29	29 - 30	29 - 30 - 27	29 - 30 - 27 - 25
30	30 - 27	30 - 27 - 25	30 - 27 - 29 - 30
31	2 - 1	2 - 1 - 3	2 - 1 - 3 - 4
32	4 - 2	4 - 2 - 1	4 - 2 - 5 - 7
33	4 - 3	4 - 3 - 1	4 - 3 - 1 - 2
34	6 - 4	6 - 4 - 3	6 - 4 - 2 - 1
35	7 - 5	7 - 5 - 2	7 - 5 - 2 - 1
36	28 - 6	28 - 6 - 4	28 - 6 - 2 - 1
37	6 - 7	6 - 7 - 5	6 - 7 - 5 - 2
38	6 - 8	6 - 8 - 28	6 - 8 - 28 - 6
39	5 - 2	5 - 2 - 1	5 - 2 - 6 - 4
40	22 - 10	22 - 10 - 21	22 - 10 - 20 - 19
41	21 - 10	21 - 10 - 17	21 - 10 - 20 - 19
42	14 - 12	14 - 12 - 15	14 - 12 - 16 - 17
43	15 - 12	15 - 12 - 16	15 - 12 - 14 - 15
44	15 - 14	15 - 14 - 12	15 - 14 - 12 - 15
45	18 - 15	18 - 15 - 12	18 - 15 - 14 - 12
46	17 - 16	17 - 16 - 12	17 - 16 - 12 - 15
47	10 - 17	10 - 17 - 16	10 - 17 - 16 - 12
48	19 - 18	19 - 18 - 15	19 - 18 - 15 - 12
49	20 - 19	20 - 19 - 18	20 - 19 - 18 - 15
50	10 - 20	10 - 20 - 19	10 - 20 - 19 - 18

ตารางที่ 4.5 ผลการจัดโซนป้องกันรีเลย์ระยะทาง ระบบ 30 บัส (ต่อ)

รีเลย์	โซนการป้องกัน		
	โซน 1	โซน 2	โซน 3
51	22 - 21	22 - 21 - 10	22 - 21 - 10 - 17
52	24 - 22	24 - 22 - 21	24 - 22 - 10 - 21
53	24 - 23	24 - 23 - 15	24 - 23 - 15 - 12
54	25 - 24	25 - 24 - 22	25 - 24 - 23 - 15
55	26 - 25	26 - 25 - 27	26 - 25 - 24 - 22
56	29 - 27	29 - 27 - 25	29 - 27 - 30 - 29
57	25 - 27	25 - 27 - 29	25 - 27 - 30 - 29
58	8 - 28	8 - 28 - 6	8 - 28 - 6 - 4
59	30 - 29	30 - 29 - 27	30 - 29 - 27 - 25
60	27 - 30	27 - 30 - 29	27 - 30 - 29 - 27
61	1 - 3	1 - 3 - 4	1 - 3 - 4 - 6
62	3 - 1	3 - 1 - 2	3 - 1 - 2 - 4
63	2 - 6	2 - 6 - 4	2 - 6 - 7 - 5
64	6 - 2	6 - 2 - 1	6 - 2 - 5 - 7
65	1 - 3	1 - 3 - 4	1 - 3 - 4 - 6
66	3 - 1	3 - 1 - 2	3 - 1 - 2 - 4
67	2 - 6	2 - 6 - 4	2 - 6 - 7 - 5
68	6 - 2	6 - 2 - 1	6 - 2 - 5 - 7

ตารางที่ 4.6 ค่าปรับตั้งรีเลย์ระยะทาง ระบบ 30 บัส

รีเลย์	โซน 1 (Ω)		โซน 2 (Ω)		โซน 3 (Ω)	
	R	jX	R	jX	R	jX
1	0.5061	71.54	1.5864	71.71	3.081	74.7
2	1.5261	71.83	2.117	71.73	2.4708	72.32
3	0.335	70.8	0.6433	71.9	2.4845	71.64

ตารางที่ 4.6 ค่าปรับตั้งรีเลย์ระยะทาง ระบบ 30 บัส (ต่อ)

รีเลย์	โซน 1 (Ω)		โซน 2 (Ω)		โซน 3 (Ω)	
	R	jX	R	jX	R	jX
4	0.3596	73.96	0.6774	73.99	2.5443	72.14
5	1.0417	68.37	1.7514	69.29	2.3131	70.04
6	0.5196	74.24	1.7442	73.06	2.953	72.84
7	0.7199	71.96	1.1245	72.36	2.995	71.81
8	0.3646	74.05	0.6805	74.02	2.5506	72.16
9	1.7016	76.61	2.7729	74.68	3.6457	73.39
10	1.3907	64.13	1.8756	64.1	4.7342	60.89
11	0.6894	65.08	0.999	64.91	1.5707	64.6
12	2.3705	64.31	4.4409	56.71	6.3336	53.55
13	1.2208	63.08	2.702	63.34	5.2838	51.1
14	2.4865	42.1	3.8303	46.19	5.9164	52.61
15	2.0321	63.85	3.2921	63.81	4.2425	63.77
16	1.6638	74.76	2.55	73.7	3.2344	72.44
17	0.7555	69.02	1.3745	67.79	3.5309	66.58
18	1.2032	63.68	1.9007	63.63	2.8944	64.07
19	0.6347	63.43	1.9877	64.9	3.3978	65.25
20	1.9117	65.87	2.8205	65.75	4.1961	65.12
21	0.2195	63.82	1.1436	64.05	3.2737	59.42
22	1.7761	57.28	3.7819	60.04	6.7904	59.44
23	2.5089	63.95	4.2406	62.21	7.7053	61.9
24	3.1667	60.2	5.1869	60.71	0	0
25	3.8174	56.2	0	0	0	0
26	3.9225	62.11	7.5789	62.11	12.0351	62.1
27	1.9667	62.36	4.4368	61.4	0	0
28	1.7519	72.36	2.4177	72.52	2.7579	72.7

ตารางที่ 4.6 ค่าปรับตั้งรีเลย์ระยะทาง ระบบ 30 บัส (ต่อ)

รีเลย์	โซน 1 (Ω)		โซน 2 (Ω)		โซน 3 (Ω)	
	R	jX	R	jX	R	jX
29	4.2813	62.11	8.9124	62.07	13.0877	62.07
30	5.6972	62.02	8.3506	62.07	13.3625	62.06
31	0.5061	71.54	1.5256	73.39	2.5236	73.74
32	1.5261	71.83	2.2239	71.79	4.355	73.91
33	0.335	70.8	1.3117	73.45	2.3632	73.8
34	0.3596	73.96	0.6587	72.96	2.515	72.19
35	1.0417	68.37	2.3596	72.07	3.5789	73.4
36	0.5196	74.24	0.8742	74.17	2.7441	72.34
37	0.7199	71.96	1.5502	70.46	2.7297	71.15
38	0.3646	74.05	1.5506	72.86	2.8079	72.74
39	1.7016	76.61	2.4422	75.95	4.1727	74.29
40	1.3907	64.13	2.1693	64.32	4.3258	65.06
41	0.6894	65.08	1.3333	66.47	3.4495	65.54
42	2.3705	64.31	3.726	64.06	5.7713	65.35
43	1.2208	63.08	2.6737	63.72	5.2184	60.72
44	2.4865	42.1	4.5146	49.23	6.3336	53.55
45	2.0321	63.85	3.303	63.67	6.2733	53.32
46	1.6638	74.76	3.216	71.14	4.7378	68.9
47	0.7555	69.02	1.9817	72.03	3.5897	71.63
48	1.2032	63.68	2.7741	63.76	4.4256	63.72
49	0.6347	63.43	1.5453	63.56	2.9324	63.65
50	1.9117	65.87	2.7859	65.53	3.5582	65.10
51	0.2195	63.82	0.7053	64.59	1.3717	65.51
52	1.7761	57.28	2.3564	57.66	4.1663	60.53
53	2.5089	63.95	4.312	63.87	5.8695	63.78

ตารางที่ 4.6 ค่าปรับตั้งรีเลย์ระยะทาง ระบบ 30 บัส (ต่อ)

รีเลย์	โซน 1 (Ω)		โซน 2 (Ω)		โซน 3 (Ω)	
	R	jX	R	jX	R	jX
54	3.1667	60.20	5.0673	59.56	7.6785	62.00
55	3.8174	56.2	5.9953	57.46	9.2799	57.97
56	3.9225	62.11	6.1322	62.16	13.3625	62.06
57	1.9667	62.36	4.9099	62.23	10.9177	62.11
58	1.7519	72.36	2.5145	72.6	2.9515	72.84
59	4.2813	62.11	7.8032	62.11	10.8693	62.12
60	5.6972	62.02	9.7973	62.04	13.6989	62.06
61	1.4297	74.7	1.9961	74.29	2.3176	73.96
62	1.4297	74.7	2.1031	74.22	2.8957	73.54
63	1.5496	71.76	2.1616	71.99	3.1618	71.47
64	1.5496	71.76	2.2533	71.73	4.3843	73.86
65	1.4297	74.7	1.9961	74.29	2.3176	73.96
66	1.4297	74.7	2.1031	74.22	2.8957	73.54
67	1.5496	71.76	2.1616	71.99	3.1618	71.47
68	1.5496	71.76	2.2533	71.73	4.3843	73.86

มหาวิทยาลัยเทคโนโลยีสุรนารี

4.3 ค่าเหมาะที่สุดการประสานสัมพันธ์รีเลย์ป้องกระแสเกิน

กรณีนี้เป็นการประสานสัมพันธ์ระหว่างรีเลย์ป้องกันกระแสเกิน โดยใช้รีเลย์ป้องกันกระแสเกินแบ่งหน้าที่เป็นรีเลย์หลักและรีเลย์สำรอง ใช้ฟังก์ชันวัตถุประสงค์ดังการที่ 3.2 และมีเงื่อนไขการประสานสัมพันธ์ดังสมการที่ 3.4 โดยมีค่าการประสานสัมพันธ์ $CTI = 0.3$

4.3.1 ระบบทดสอบมาตรฐาน WSCC 9 บัส

ทำการหาคำตอบของสมการการหาความสัมพันธ์รีเลย์ป้องกันกระแสเกิน โดยใช้วิธีการ DE และ วิธี BFGS หาคำตอบ โดยใช้ค่า พารามิเตอร์ดังตารางที่ 4.7 สำหรับ DE ตารางที่ 4.8 สำหรับ GA และ ตารางที่ 4.9 สำหรับ BFGS

ตารางที่ 4.7 ค่าการค้นหาคำตอบของ DE ระบบ 9 บัส

ค่าการค้นหาคำตอบ	ค่าที่ใช้
I_NP	300
I_itermax	3000
F_weight	0.5
F_CR	0.9
Model DE	standard

ตารางที่ 4.8 ค่าการค้นหาคำตอบของ GA ระบบ 9 บัส

ค่าการค้นหาคำตอบ	ค่าที่ใช้
PopulationSize	300
Generations	3000
CrossoverFraction	0.7

ตารางที่ 4.9 ค่าการค้นหาคำตอบของ BFGS ระบบ 9 บัส

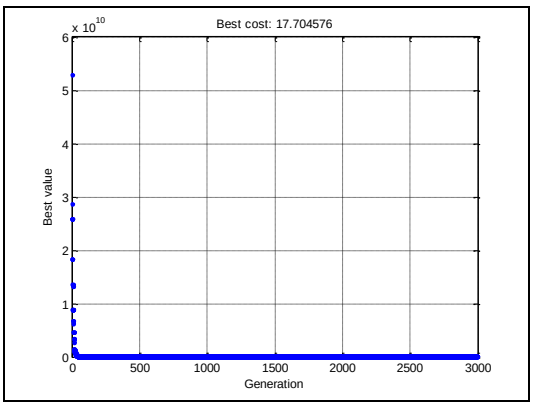
ค่าการค้นหาคำตอบ	ค่าที่ใช้
MaxFunEvals	1×10^{12}
MaxIter	1×10^{4}
TolX	1×10^{-12}
TolFun	1×10^{-12}

ทำการค้นหาคำตอบด้วยโค้ดโปรแกรมแมทแลป ทำการค้นหาคำตอบซ้ำ 30 รอบ เลือกคำในรอบที่มีค่าฟังค์ชันวัตถุประสงค์ต่ำสุด ได้ดังตารางที่ 4.10

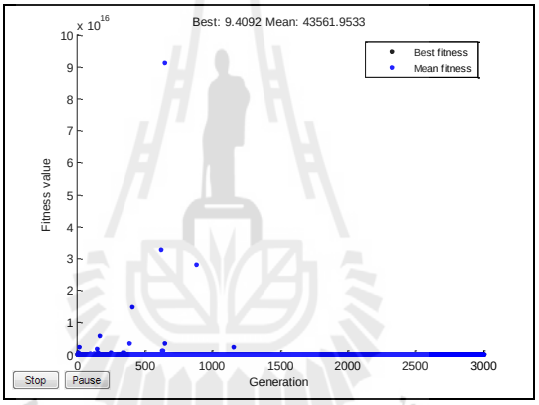
ตารางที่ 4.10 ผลการจัดความสัมพันธ์รีเลย์ป้องกันกระแสเกิน ระบบ 9 บัส

รีเลย์	DE		GA		BFGS	
	TDS	PS	TDS	PS	TDS	PS
1	0.1828	1.4973	0.0956	1.2508	0.1985	1.4027
2	0.0881	1.4999	0.0504	1.2508	0.1000	1.4147
3	0.0995	1.4897	0.0890	1.2502	0.1237	1.2817
4	0.1839	1.2535	0.1234	1.2503	0.1863	1.2500
5	0.0982	1.4969	0.0505	1.2503	0.1042	1.4703
6	0.0640	1.4991	0.0506	1.2501	0.0813	1.3185
7	0.2002	1.4936	0.1238	1.2505	0.2307	1.2637
8	0.1398	1.499	0.0501	1.2505	0.1651	1.2791
9	0.1523	1.4972	0.1206	1.2504	0.1650	1.4198
10	0.1554	1.4999	0.0503	1.2501	0.1656	1.4412
11	0.0790	1.4965	0.0505	1.2506	0.1033	1.2555
12	0.1280	1.4880	0.1213	1.2507	0.1538	1.2650
Min	10.0454		9.3361		16.3474	
Mean	10.0454		9.3506		23.135	
SD	0.51×10^{-6}		0.0205		7.0438	

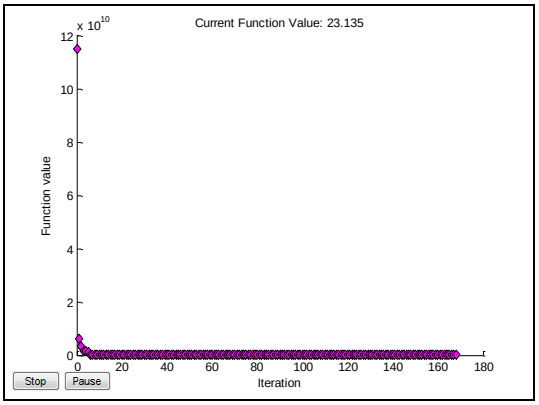
และมีผลการลู่เข้าหาคำตอบของวิธี DE ดังรูปที่ 4.4 วิธี GA ดังรูปที่ 4.5 และวิธี BFGS ดังรูปที่ 4.6 แต่ละรูปจะแสดงให้เห็นถึงการลดลงของฟังก์ชันวัตถุประสงค์ของแต่ละวิธี



รูปที่ 4.4 ผลการลู่เข้าหาคำตอบของวิธีการ DE ระบบทดสอบ 9 บัส



รูปที่ 4.5 ผลการลู่เข้าหาคำตอบของวิธีการ GAระบบทดสอบ 9 บัส



รูปที่ 4.6 ผลการลู่เข้าหาคำตอบของวิธีการ BFGS ระบบทดสอบ 9 บัส

4.3.2 ระบบทดสอบมาตรฐาน IEEE 14 บัส

ในระบบทดสอบนี้ มีจำนวนรีเลย์ 15 ตัว จึงทำให้มีตัวแปรตัดสินใจทั้งหมด 30 ตัว หาคำตอบด้วยวิธี DE โดยใช้ ค่าการค้นหาคำตอบ ดังตารางที่ 4.11

ตารางที่ 4.11 ค่าการค้นหาคำตอบของ DE ระบบ 14 บัส

ค่าการค้นหาคำตอบ	ค่าที่ใช้
I_NP	300
I_itermax	10000
F_weight	0.5
F_CR	0.9
Model DE	standard

ทำการค้นหาคำตอบด้วยโค้ดโปรแกรมแมทแลป ทำการค้นหาคำตอบซ้ำ 30 รอบ เลือกคำในรอบที่มีค่าฟังค์ชันวัตถุประสงค์ต่ำสุด ได้ดังตารางที่ 4.12

ตารางที่ 4.12 ผลการจัดความสัมพันธ์รีเลย์ป้องกันกระแสเกิน ระบบ 14 บัส

รีเลย์	TDS	PS	รีเลย์	TDS	PS
1	0.1295	1.2878	16	0.0501	1.2677
2	0.0522	1.3772	17	0.0634	1.439
3	0.0506	1.2539	18	0.0803	1.4250
4	0.0646	1.4617	19	0.0546	1.4995
5	0.0511	1.3727	20	0.0656	1.4934
6	0.1563	1.4760	21	0.0639	1.3714
7	0.0520	1.2766	22	0.0521	1.4762
8	0.1432	1.3142	23	0.0761	1.3649
9	0.1980	1.4907	24	0.0920	1.4935
10	0.1065	1.4211	25	0.1690	1.4868
11	0.0644	1.4704	26	0.1594	1.4184

ตารางที่ 4.12 ผลการจัดความสัมพันธ์รีเลย์ป้องกันกระแสเกิน ระบบ 14 บัส (ต่อ)

รีเลย์	TDS	PS	รีเลย์	TDS	PS
12	0.1059	1.4501	27	0.0512	1.3329
13	0.0749	1.4056	28	0.1256	1.3626
14	0.0511	1.3718	29	0.0566	1.2976
15	0.0748	1.4383	30	0.0505	1.348
Min				26.498	
Mean				26.5856	
SD				0.1239	

4.3.3 ระบบทดสอบมาตรฐาน IEEE 30 บัส

ในระบบทดสอบนี้ มีจำนวนรีเลย์ 68 ตัว จึงทำให้มี ตัวแปรตัดสินใจทั้งหมด 132 ตัว หาคำตอบด้วยวิธี DE โดยใช้ ค่าการค้นหาคำตอบ ดังตารางที่ 4.13

ตารางที่ 4.13 ค่าการค้นหาคำตอบของ DE ระบบ 30 บัส

ค่าการค้นหาคำตอบ	ค่าที่ใช้
I_NP	500
I_itermax	10000
F_weight	0.5
F_CR	0.9
Model DE	standard

ทำการค้นหาคำตอบด้วยโค้ดโปรแกรมแมทแลป ทำการค้นหาคำตอบซ้ำ 30 รอบ เลือกคำในรอบที่มีค่าฟังค์ชันวัตถุประสงค์ต่ำสุด ได้ดังตารางที่ 4.14

ตารางที่ 4.14 ผลการจัดความสัมพันธ์รีเลย์ป้องกันกระแสเกิน ระบบ 30 บัส

รีเลย์	TDS	PS	รีเลย์	TDS	PS
1	0.0771	1.4209	35	0.0509	1.2576
2	0.0785	1.2762	36	0.1590	1.4275

ตารางที่ 4.14 ผลการจัดความสัมพันธ์รีเลย์ป้องกันกระแสเกิน ระบบ 30 บัส (ต่อ)

รีเลย์	TDS	PS	รีเลย์	TDS	PS
3	0.0503	1.2874	37	0.1768	1.3484
4	0.1157	1.4076	38	0.2850	1.3417
5	0.0518	1.3017	39	0.0504	1.3972
6	0.1875	1.4999	40	0.2788	1.4979
7	0.0618	1.3536	41	0.2519	1.3929
8	0.0572	1.2960	42	0.0505	1.3397
9	0.0556	1.2713	43	0.0503	1.2712
10	0.2398	1.4191	44	0.1343	1.2730
11	0.6452	1.2529	45	0.0712	1.3622
12	0.1497	1.4079	46	0.1651	1.4666
13	0.3063	1.3585	47	0.4851	1.3244
14	0.1005	1.4782	48	0.1344	1.4891
15	0.2219	1.3505	49	0.2327	1.3195
16	0.2015	1.4740	50	0.5141	1.3641
17	0.2199	1.3411	51	0.6366	1.3405
18	0.1652	1.3465	52	0.222	1.3599
19	0.0604	1.3503	53	0.0668	1.3788
20	0.0559	1.4089	54	0.1074	1.3162
21	1.0978	1.4936	55	0.5277	1.4798
22	0.3037	1.3418	56	0.1658	1.3562
23	0.0794	1.4553	57	0.0921	1.351
24	0.1406	1.4096	58	0.5796	1.3485
25	0.0515	1.4756	59	0.0504	1.2523
26	0.0666	1.3827	60	0.068	1.4010
27	0.1902	1.3595	61	0.0831	1.3967
28	0.2546	1.3499	62	0.0691	1.3849

ตารางที่ 4.14 ผลการจัดความสัมพันธ์รีเลย์ป้องกันกระแสเกิน ระบบ 30 บัส (ต่อ)

รีเลย์	TDS	PS	รีเลย์	TDS	PS
29	0.0503	1.2965	63	0.0930	1.4497
30	0.0561	1.2872	64	0.0586	1.2805
31	0.1315	1.2981	65	0.2953	1.3998
32	0.1281	1.3814	66	0.0884	1.3089
33	0.0511	1.2654	67	0.1807	1.4510
34	0.1257	1.3832	68	0.0675	1.4419
Min			127.2084		
Mean			131.1516		
SD			5.5766		

4.4 ค่าเหมาะที่สุดการประสานสัมพันธ์รีเลย์ป้องกระแสเกินและรีเลย์ระยะทาง

กรณีนี้เป็นการประสานสัมพันธ์ระหว่างรีเลย์ระยะทางและรีเลย์ป้องกันกระแสเกิน โดยให้รีเลย์ระยะทางทำหน้าที่เป็นรีเลย์หลัก และให้รีเลย์ป้องกันกระแสเกินทำหน้าที่รีเลย์สำรอง

4.4.1 ระบบทดสอบมาตรฐาน WSCC 9 บัส

ในระบบทดสอบนี้ มีจำนวนรีเลย์ 12 ตัว จึงทำให้มี ตัวแปรตัดสินใจทั้งหมด 24 ตัว หาคำตอบด้วยวิธี DE วิธี GA และวิธี BFGSโดยใช้ ค่าการค้นหาคำตอบ ดังตารางที่ 4.15 สำหรับDE ตารางที่ 4.16 และ ตารางที่ 4.17

ตารางที่ 4.15 ค่าการค้นหาคำตอบของ DE ในระบบ 9 บัส

ค่าการค้นหาคำตอบ	ค่าที่ใช้
I_NP	300
I_itermax	3000
F_weight	0.5
F_CR	0.9
Model DE	standard

ทำการค้นหาคำตอบด้วยโค้ดโปรแกรมแมทแลป ทำการค้นหาคำตอบซ้ำ 30 รอบ เลือกคำในรอบที่มีค่าฟังค์ชันวัตถุประสงค์ต่ำสุด ได้ดังตารางที่ 4.16

ตารางที่ 4.16 ค่าการค้นหาคำตอบของ GA ในระบบ 9 บัส

ค่าการค้นหาคำตอบ	ค่าที่ใช้
PopulationSize	300
Generations	3000
CrossoverFraction	0.7

ตารางที่ 4.17 ค่าการค้นหาคำตอบของ BFGS ในระบบ 9 บัส

ค่าการค้นหาคำตอบ	ค่าที่ใช้
MaxFunEvals	1×10^{-12}
MaxIter	10000
TolX	1×10^{-12}
TolFun	1×10^{-12}

ตารางที่ 4.18 ผลการจัดความสัมพันธ์รีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทาง ระบบ 9 บัส

รีเลย์	DE		GA		BFGS	
	TDS	PS	TDS	PS	TDS	PS
1	0.1828	1.4973	0.2174	1.4127	0.098	1.2517
2	0.0881	1.4999	0.0957	1.3385	0.0501	1.2508
3	0.0995	1.4897	0.2088	1.358	0.1052	1.2502
4	0.1839	1.2535	0.1865	1.3229	0.1445	1.2501
5	0.0982	1.4969	0.05	1.5	0.0501	1.2501
6	0.064	1.4991	0.05	1.3964	0.05	1.2504
7	0.2002	1.4936	0.3265	1.2764	0.1489	1.2504
8	0.1398	1.499	0.2162	1.3054	0.0512	1.2501
9	0.1523	1.4972	0.2919	1.4538	0.1302	1.2896
10	0.1554	1.4999	0.0625	1.3233	0.0506	1.2501

ตารางที่ 4.18 ผลการจัดความสัมพันธ์รีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทางระบบ 9 บัส (ต่อ)

รีเลย์	DE		GA		BFGS	
	TDS	PS	TDS	PS	TDS	PS
11	0.079	1.4965	0.1943	1.3195	0.0504	1.2504
12	0.128	1.488	0.212	1.3968	0.1362	1.2504
Min	10.0454		20.3210		10.1596	
Mean	10.0454		23.8886		10.1788	
SD	1.61×10^{-6}		3.9152		0.0271	

4.4.2 ระบบทดสอบมาตรฐาน IEEE 14 บัส

ในระบบทดสอบนี้ มีจำนวนรีเลย์ 30 ตัว จึงทำให้มี ตัวแปรตัดสินใจทั้งหมด 60 ตัว หาคำตอบด้วยวิธี DE โดยใช้ ค่าการค้นหาคำตอบ ดังตารางที่ 4.19

ตารางที่ 4.19 ค่าการค้นหาคำตอบของ DE ในระบบ 14 บัส

ค่าการค้นหาคำตอบ	ค่าที่ใช้
I_NP	300
I_itermax	10000
F_weight	0.5
F_CR	0.9
Model DE	standard

ทำการค้นหาคำตอบด้วยโค้ดโปรแกรมแมทแลป ทำการค้นหาคำตอบซ้ำ 30 รอบ เลือกคำในรอบที่มีค่าฟังค์ชันวัตถุประสงค์ต่ำสุด ได้ดังตารางที่ 4.20

ตารางที่ 4.20 ผลการจัดความสัมพันธ์รีเลย์ระยะทางและรีเลย์ป้องกันกระแสเกิน ระบบ 14 บัส

รีเลย์	TDS	PS	รีเลย์	TDS	PS
1	0.1295	1.2878	16	0.0501	1.2677
2	0.0522	1.3772	17	0.0634	1.439
3	0.0506	1.2539	18	0.0803	1.425

ตารางที่ 4.20 ผลการจัดความสัมพันธ์รีเลย์ป้องกันกระแสเกินและ รีเลย์ระยะทางระบบ 14 บัส (ต่อ)

รีเลย์	TDS	PS	รีเลย์	TDS	PS
4	0.0646	1.4617	19	0.0546	1.4995
5	0.0511	1.3727	20	0.0656	1.4934
6	0.1563	1.476	21	0.0639	1.3714
7	0.052	1.2766	22	0.0521	1.4762
8	0.1432	1.3142	23	0.0761	1.3649
9	0.198	1.4907	24	0.092	1.4935
10	0.1065	1.4211	25	0.169	1.4868
11	0.0644	1.4704	26	0.1594	1.4184
12	0.1059	1.4501	27	0.0512	1.3329
13	0.0749	1.4056	28	0.1256	1.3626
14	0.0511	1.3718	29	0.0566	1.2976
15	0.0748	1.4383	30	0.0505	1.348
Min			26.498		
Mean			26.5856		
SD			0.1239		

4.4.3 ระบบทดสอบมาตรฐาน IEEE 30 บัส

ในระบบทดสอบนี้ มีจำนวนรีเลย์ 68 ตัว จึงทำให้มี ตัวแปรตัดสินใจทั้งหมด 136 ตัว หาคำตอบด้วยวิธี DE โดยใช้ ค่าการค้นหาคำตอบ ดังตารางที่ 4.21

ตารางที่ 4.21 ค่าการค้นหาคำตอบของ DE ในระบบ 30บัส

ค่าการค้นหาคำตอบ	ค่าที่ใช้
I_NP	500
I_itermax	20000
F_weight	0.5
F_CR	0.9
Model DE	standard

ทำการค้นหาคำตอบด้วยโค้ดโปรแกรมแมทแลป ทำการค้นหาคำตอบซ้ำ 30 รอบ เลือกคำในรอบที่มีค่าฟังค์ชันวัตถุประสงค์ต่ำสุด ได้ดังตารางที่ 4.22

ตารางที่ 4.22 ผลการจัดความสัมพันธ์รีเลย์ระยะทางและรีเลย์ป้องกันกระแสเกิน ในระบบ 30 บัส

รีเลย์	TDS	PS	รีเลย์	TDS	PS
1	0.0708	1.3821	35	0.0505	1.2527
2	0.0558	1.3512	36	0.1093	1.4363
3	0.0509	1.2606	37	0.1669	1.3977
4	0.1066	1.3062	38	0.1895	1.3520
5	0.0559	1.292	39	0.0550	1.2931
6	0.1449	1.4534	40	0.2488	1.4995
7	0.0533	1.3556	41	0.2246	1.3991
8	0.0547	1.3357	42	0.0501	1.4047
9	0.0525	1.2996	43	0.0517	1.2571
10	0.1992	1.4559	44	0.1356	1.2552
11	0.5728	1.2560	45	0.0680	1.3211
12	0.1389	1.3858	46	0.1081	1.4951
13	0.2957	1.3209	47	0.3404	1.4349
14	0.0979	1.4901	48	0.1198	1.4502
15	0.1580	1.4208	49	0.2010	1.2755
16	0.1880	1.4547	50	0.4553	1.3519
17	0.2056	1.3128	51	0.5714	1.3567
18	0.1285	1.3832	52	0.1626	1.4688
19	0.0618	1.3430	53	0.0592	1.3224
20	0.0549	1.3919	54	0.0793	1.375
21	0.9788	1.4731	55	0.5341	1.3928
22	0.2253	1.4110	56	0.0877	1.4589
23	0.0735	1.3913	57	0.0765	1.4064

ตารางที่ 4.22 ผลการจัดความสัมพันธ์รีเลย์ระยะทางและรีเลย์ป้องกันกระแสเกิน ระบบ 30 บัส(ต่อ)

รีเลย์	TDS	PS	รีเลย์	TDS	PS
24	0.1233	1.4545	58	0.4131	1.3601
25	0.0536	1.344	59	0.0515	1.2507
26	0.0523	1.320	60	0.0634	1.4554
27	0.1370	1.4003	61	0.0805	1.3217
28	0.2129	1.4719	62	0.0504	1.3885
29	0.0517	1.2775	63	0.0746	1.431
30	0.0574	1.3012	64	0.0511	1.2751
31	0.0866	1.4393	65	0.2361	1.3782
32	0.0562	1.4051	66	0.0570	1.4157
33	0.0505	1.2533	67	0.1514	1.4425
34	0.1106	1.4180	68	0.0562	1.4523
Min	110.6590				
Mean	112.2227				
SD	2.2114				

4.4 ค่าผลต่างเวลาการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทาง

เป็นค่าที่แสดงให้เห็นถึงเวลาการทำงานระหว่างรีเลย์หลักและรีเลย์สำรอง โดยมีเงื่อนไข คือ รีเลย์สำรองหรือรีเลย์ป้องกันกระแสเกินจะทำงานเมื่อรีเลย์หลักหรือรีเลย์ระยะทางโซน 2 เกิดความผิดพลาด โดยรีเลย์สำรองจะต้องมีค่าเวลาเริ่มทำงานมากกว่ารีเลย์หลักเพื่อป้องกัน การตัดวงจรของรีเลย์สำรองที่อาจทำงานเร็วกว่ารีเลย์หลัก ตามสมการที่ 3.3 ในการศึกนี้สนใจการประสานสัมพันธ์ของรีเลยรีเลย์ป้องกันกระแสเกินและระยะทาง ซึ่งมีผลต่างการประสานสัมพันธ์ดังตารางต่อไปนี้

ตารางที่ 4.23 ผลต่างเวลาการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและระยะทาง (9 บัส)

สายส่ง		DE		BFGS		GA	
		OC	DIS_OC	OC	DIS_OC	OC	DIS_OC
4	5	0.0916	0.0000	0.1982	0.1906	0.0942	0.0052
12	6	0.0705	0.0000	0.6024	1.0603	0.0777	0.0100
7	8	0.1078	0.0000	0.2459	0.1742	0.1022	0.0002
3	11	0.3722	0.2304	1.0952	0.9543	0.3632	0.2221
1	10	0.27	0.3797	1.6857	1.9805	0.2283	0.3440
9	2	0.1602	0.1572	0.6404	0.6809	0.14491	0.1593
4	5	0.3316	0.4278	0.4938	0.7550	0.3759	0.4810
12	6	0.4372	0.6053	1.1761	2.5602	0.4456	0.6202
7	8	0.3151	0.2905	0.5157	0.5449	0.2717	0.2555
3	11	0.0948	0.0000	0.6802	0.5867	0.0944	0.0004
1	10	0.0187	0.0000	1.0104	1.1180	0.0281	0.0123
9	2	0.0635	0.0000	0.4690	0.4375	0.0532	0.0016

ตารางที่ 4.24 ผลต่างเวลาการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและระยะทาง (14บัส)

รีเลย์		OC	DIS_OC
15	21	0.419541	0.246224
1	2	0.168265	0.083372
1	29	0.149425	0.081085
1	19	0.193537	0.080719
2	3	0.326922	0.340014
29	4	0.433473	0.611585
3	4	0.377193	0.555305
11	8	0.367484	0.478237
11	6	0.056736	0.456725

ตารางที่ 4.24 ผลต่างเวลาการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและระยะทาง (14บัส) (ต่อ)

รีเลย์		OC	DIS_OC
14	9	0.383578	0.80388
23	28	0.175081	0.321764
9	10	0.055215	0.279194
8	12	0.015017	0.25115
12	13	0.394373	0.532374
6	13	0.308622	0.446623
21	5	12.46491	10.12394
14	9	0.383578	0.80388
19	18	2.193587	2.565606
21	18	0.685445	1.057464
10	11	0.428592	1.133153
26	22	3.686152	2.35478
12	20	6.319387	1.605035
27	20	10.95605	6.241694
24	23	0.21887	0.813744
13	14	1.459253	2.467019
25	24	0.331718	1.048504
6	26	2.158708	2.718698
28	27	0.539723	1.272621
7	1	4.940462	5.298802
15	21	0.19782	0.700787
1	2	3.516712	4.123749
1	29	0.734264	1.604032

ตารางที่ 4.24 ผลต่างเวลาการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและระยะทาง (14บัส) (ต่อ)

รีเลย์		OC	DIS_OC
5	29	13.73079	14.60056
1	19	1.276342	1.758855
30	19	5.596003	6.078517
29	4	3.502926	3.92289
3	4	12.78497	13.20494
20	8	0.346802	0.885276
22	6	4.447797	5.181248
9	10	0.200747	0.708169
8	12	11.69846	12.49445
12	13	0.676839	1.236406
6	13	1.272884	1.832451
4	7	0.160167	0.144318
19	7	0.24608	0.230232
17	16	0.065804	0.281321
18	30	0.058829	0.089131
3	30	0.582525	0.612826
21	5	0.222136	0.243116
4	5	0.123338	0.144318
29	17	0.690745	0.657276
18	17	0.123222	0.089754
19	18	0.046333	0.20672
21	18	0.063257	0.223644
10	11	0.123711	0.215177

ตารางที่ 4.24 ผลต่างเวลาการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและระยะทาง (14บัส) (ต่อ)

รีเลย์		OC	DIS_OC
26	22	0.038787	0.269351
12	20	0.099136	0.52902
27	20	0.033984	0.463869
24	23	0.054802	0.425083
13	14	0.189968	0.271259
25	24	0.056935	0.419876
1	29	0.734264	1.604032
5	29	13.73079	14.60056
1	19	1.276342	1.758855
30	19	5.596003	6.078517
29	4	3.502926	3.92289
3	4	12.78497	13.20494
20	8	0.346802	0.885276
22	6	4.447797	5.181248
9	10	0.200747	0.708169
8	12	11.69846	12.49445
12	13	0.676839	1.236406
6	13	1.272884	1.832451
4	7	0.160167	0.144318
19	7	0.24608	0.230232
17	16	0.065804	0.281321
21	5	12.46491	10.12394
19	18	2.193587	2.565606

ตารางที่ 4.24 ผลต่างเวลาการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและระยะทาง (14บัส) (ต่อ)

รีเลย์		OC	DIS_OC
21	18	0.685445	1.057464
10	11	0.428592	1.133153
26	22	3.686152	2.35478
12	20	6.319387	1.605035
27	20	10.95605	6.241694
24	23	0.21887	0.813744
13	14	1.459253	2.467019
25	24	0.331718	1.048504
6	26	2.158708	2.718698
28	27	0.539723	1.272621
7	1	4.940462	5.298802
15	21	0.19782	0.700787
1	2	3.516712	4.123749

ตารางที่ 4.25 ผลต่างเวลาการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและระยะทาง (30บัส)

รีเลย์		OC	DIS_OC
1	2	1.443104	1.883983
1	9	0.652208	1.888856
1	63	0.072626	1.881463
2	4	2.070371	5.556798
2	33	1.709312	5.541416
3	4	0.778448	4.264875
4	37	1.605654	5.998559

ตารางที่ 4.25 ผลต่างเวลาการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและระยะทาง (30บัส) (ต่อ)

รีเลย์		OC	DIS_OC
4	6	3.053904	5.980888
4	64	1.802284	6.112337
4	64	1.802284	6.112337
4	6	58.07528	23.00069
5	7	2.005087	5.603669
6	28	3.505876	5.00261
9	5	7.044201	8.655281
10	22	0.579278	1.948099
10	51	0.336867	1.930918
10	22	7.063415	9.579195
11	21	0.170554	2.257764
12	14	0.464894	2.455093
12	14	24.39506	25.70167
13	15	1.294018	3.658698
13	67	2.855808	3.655954
13	44	77.96995	78.74306
13	44	1.639078	3.674609
13	15	2.200423	7.787139
13	67	5.029392	7.266118
14	15	3.211672	5.576352
14	67	4.768776	5.568922
14	43	2.534058	8.660048
14	43	5.13617	5.605121
14	15	3.876267	9.462983
14	67	6.274676	8.511402

ตารางที่ 4.25 ผลต่างเวลาการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและระยะทาง (30บัส) (ต่อ)

รีเลย์		OC	DIS_OC
15	18	2.070422	3.730257
15	18	3.645865	5.272901
16	17	0.202909	3.936889
16	17	1.154172	2.139537
17	50	0.758368	3.57443
17	11	2.096137	3.499953
17	10	2.839673	3.532944
17	64	113.1349	117.4449
17	34	89.15988	94.61803
17	6	193.7291	158.6545
17	11	5.631163	8.615514
17	10	9.687593	10.72826
17	6	193.7291	158.6545
18	19	1.14619	1.952045
18	19	2.019492	2.408405
19	20	0.208117	0.771964
21	22	0.490558	1.859379
21	40	13.92426	4.97658
21	22	0.158946	2.674726
22	24	0.644686	2.022655
22	53	4.858676	5.755279
22	53	0.558414	2.018364
22	24	3.368405	6.223924
23	24	1.615145	2.993114
23	52	1.529727	2.947515

ตารางที่ 4.25 ผลต่างเวลาการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและระยะทาง (30บัส) (ต่อ)

รีเลย์		OC	DIS_OC
24	25	1.482024	2.439434
24	57	0.688073	2.41598
24	25	11.98896	11.26878
24	57	2.613453	3.819524
27	25	0.553923	1.511334
27	29	11.18216	14.97749
27	54	1.038343	3.513751
27	54	0.636608	1.499511
27	25	7.915498	7.195323
27	29	12.79963	18.82874
28	8	0.646868	1.226595
29	30	15.32233	21.23651
31	61	2.83537	3.8427
34	33	18.89304	19.5858
34	33	2.551777	6.383881
37	35	78.89397	57.77074
37	35	5.242621	8.651813
38	58	1.159284	2.996008
41	10	0.209658	1.250323
42	13	0.631864	5.17352
43	12	8.621308	9.905564
43	65	5.361707	7.8069
43	12	5.301945	8.915752
44	42	5.223074	2.998548
44	42	0.48294	0.485264

ตารางที่ 4.25 ผลต่างเวลาการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและระยะทาง (30บัส) (ต่อ)

รีเลย์		OC	DIS_OC
45	67	2.225215	3.025362
45	44	0.996194	3.031725
46	66	1.407901	4.057935
46	66	0.923837	1.466388
47	46	3.293564	5.040912
47	46	0.803868	3.234384
48	45	1.185505	4.279185
48	45	1.037796	1.60409
49	48	1.016028	2.90307
49	48	0.093081	2.001205
50	49	3.113307	5.403174
50	49	0.398072	4.553998
51	41	13.44227	3.467466
51	41	0.140343	1.408526
52	40	18.66348	9.715802
52	51	2.768009	4.468822
52	40	1.603488	3.528447
52	51	1.883689	3.477741
53	68	0.030281	0.616966
54	52	28.24751	31.95634
54	53	672.2234	673.12
54	52	0.832667	2.250455
54	53	0.818156	2.278106
56	60	283.0819	260.8188
57	26	1.920167	2.38939

ตารางที่ 4.25 ผลต่างเวลาการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและระยะทาง (30บัส) (ต่อ)

รีเลย์		OC	DIS_OC
57	60	0.528091	2.413812
57	28	53.70597	55.19486
57	36	64.59871	60.61737
57	28	23.79409	25.29083
57	36	20.39128	22.37679
57	26	40.26482	18.40579
57	60	48.03099	25.76795
58	26	2.866885	3.336108
58	60	1.466344	3.352065
58	27	1.762065	3.543355
58	27	0.747036	3.3137
58	36	1.037361	3.022875
58	26	26.75612	4.8971
58	60	27.93904	5.675996
59	56	0.686325	5.681308
60	59	11.62411	14.03764
61	3	2.932923	5.02043
61	3	10.23643	10.83295
63	37	7.521177	11.91408
63	6	8.917039	11.84402
63	34	987.8467	994.4163
63	34	6.306285	11.76444
65	16	2.616068	3.981912
65	16	6.581646	7.572105
66	12	1.307006	2.591262

ตารางที่ 4.25 ผลต่างเวลาการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและระยะทาง (30บัส) (ต่อ)

รีเลย์		OC	DIS_OC
66	13	0.170027	2.529697
67	23	0.009271	1.087936
67	23	1.928316	2.682074
68	15	0.37954	2.744219
68	44	0.702838	2.738369

จากการหาค่าผลต่างเวลาทำงานของรีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทาง ของระบบมาตรฐาน 9 บัส 14 บัส และ30 บัส นำมาสรุปจำนวนคู่ของการประสานสัมพันธ์รีเลย์ในแต่ละระบบได้ดังตารางที่ 4.26-4.28 ซึ่งจะแสดงจำนวนคู่ของการประสานสัมพันธ์รีเลย์ในแต่ละระบบทดสอบ และแสดงผลการจับคู่ว่าสามารถจับคู่ได้จำนวนเท่าไหร่ในแต่ละระบบทดสอบ

ตารางที่ 4.26 จำนวนคู่ประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทาง (9 บัส)

ชนิดต่างเวลา	จำนวนการจับคู่ (12 คู่)	ค่าเฉลี่ย		
		DE	BFGS	GA
$\Delta t_{mb\|k_1\|} > 0$	12	0.1944	0.7344	0.1900
$\Delta t_{mb\|k_1\|} < 0$	0	-	-	-
$\Delta t_{mbDISOC\|k_2\|} > 0$	12	0.1742	0.9203	0.1760
$\Delta t_{mbDISOC\|k_2\|} < 0$	0	-	-	-

ตารางที่ 4.27 จำนวนคู่ประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทาง (14 บัส)

ชนิดต่างเวลา	จำนวนการจับคู่ (160 คู่)	ค่าเฉลี่ย
$\Delta t_{mb\|k_1\|} > 0$	63	1.8053
$\Delta t_{mb\|k_1\|} < 0$	97	-0.2245
$\Delta t_{mbDISOC\|k_2\|} > 0$	63	1.8927
$\Delta t_{mbDISOC\|k_2\|} < 0$	97	-2.5421

ตารางที่ 4.27 จำนวนคู่ประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทาง (30 บัส)

ชนิดต่างเวลา	จำนวนการจับคู่ (368 คู่)	ค่าเฉลี่ย
$\Delta t_{mb\|k_1\|} > 0$	137	25.44105
$\Delta t_{mb\|k_1\|} < 0$	231	-11.5301
$\Delta t_{mbDISOC\|k_2\|} > 0$	137	25.16451
$\Delta t_{mbDISOC\|k_2\|} < 0$	231	-21.0870

4.5 สรุป

ในบทนี้ได้ทำการหาผลเฉลยของการประสานสัมพันธ์รีเลย์ทั้ง 3 แบบ ยังได้คำนวณค่าผลต่างเวลาทำงานของรีเลย์หลักและรีเลย์สำรอง แบ่งออกเป็น 2 ชนิด คือ เปรียบเทียบผลต่างเวลาทำงานของการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและรีเลย์ป้องกันกระแสเกิน และเปรียบเทียบผลต่างเวลาระหว่างรีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทาง ซึ่งในบทถัดไปจะกล่าวถึงผลสรุปของผลเฉลยที่ได้จากการประสานสัมพันธ์รีเลย์ต่อไป

บทที่ 5

บทสรุปและข้อเสนอแนะ

5.1 บทนำ

จากบทที่แล้วได้ทำการหาค่าเหมาะที่สุดของการประสานสัมพันธ์รีเลย์แบบต่าง ๆ แล้ว ในบทนี้จะได้กล่าวถึง ข้อสรุปในการจัดความสัมพันธ์โดยใช้เงื่อนไขของการประสานสัมพันธ์ดังสมการที่ 3-3 มาตรวจสอบการประสานสัมพันธ์ แล้วเปรียบเทียบเวลาการทำงานระหว่างการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินกับรีเลย์ป้องกันกระแสเกิน และการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินกับรีเลย์ระยะทาง และสุดท้ายจะกล่าวถึงข้อเสนอแนะเพื่อเป็นแนวทางต่อไป

5.2 สรุป

จากบทที่ 4 ที่ได้จำลองค่ากระแสลัดวงจรของระบบทดสอบแล้วนำมาคำนวณผลเฉลยของปัญหาการประสานสัมพันธ์รีเลย์พบว่า ผลการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทาง และผลการแสดงค่าผลต่างเวลาของรีเลย์หลักและสำรอง พบว่า เงื่อนไขการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทางสามารถประสานสัมพันธ์ได้ถูกต้อง ดูได้จากค่าผลต่างเวลาทำงานระหว่างรีเลย์หลักและรีเลย์สำรองที่มีค่ามากกว่าค่าศูนย์ ดังแสดงในตารางที่ 4.26-4.28 จากตารางพบว่า สามารถประสานสัมพันธ์รีเลย์ได้ครบตามจำนวนคู่การประสานสัมพันธ์ในระบบ 9 บัส ส่วนในระบบ 14 บัสและ 30 บัสยังไม่สามารถจับคู่ประสานสัมพันธ์ได้ครบทุกคู่ สรุปว่า สามารถหาค่าเหมาะที่สุดของการประสานสัมพันธ์รีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทางในระบบไฟฟ้ากำลังมาตรฐานขนาด 9 บัสได้ถูกต้องครบถ้วน ส่วนระบบ 14 บัส และ 30 บัส ยังไม่สามารจับคู่ได้ถูกต้องทุกคู่ การหาผลเฉลยของปัญหาการหาค่าเหมาะสุดได้ทำการเลือกอัลกอริทึมหรือวิธีค้นหาคำตอบ โดยนำวิธี DE GAและ BFGS มาทดสอบหาผลเฉลยในระบบทดสอบ 9 บัส สรุปได้ว่า DE สามารคำตอบได้เร็วรองจาก BFGS ส่วนGAให้ค่าคำตอบที่ดีที่สุด แต่ใช้เวลานานเป็นสองเท่าของวิธี DE ซึ่งผลเฉลยจากวิธี DE และวิธี GA ได้ค่าที่ต่างกันเล็กน้อย จึงสรุปว่าวิธี DE เหมาะที่จะใช้แก้ปัญหาการประสานสัมพันธ์รีเลย์

5.3 ข้อเสนอแนะ

จากการศึกษาการป้องกันระบบไฟฟ้ากำลังด้วยรีเลย์ป้องกันกระแสเกิน พบปัญหาในการประสานสัมพันธ์ คือ ความไม่เหมาะสมของการจัดลำดับการป้องกัน เนื่องจากในระบบที่มีความซับซ้อน เช่น ระบบที่มีจำนวนสายส่งต่อบัสมาก จะเพิ่มความซับซ้อนในการจัดลำดับการป้องกันจึงทำให้มีปัญหาในประสานสัมพันธ์รีเลย์ ข้อเสนอแนะ คือ

- ควรเลือกจัดความสัมพันธ์เฉพาะกรณีที่สนใจหรือกรณีที่มีความสำคัญต่อระบบนั้น เพื่อให้ไม่เกิดปัญหาความไม่เหมาะสมในการประสานความสัมพันธ์รีเลย์
- เลือกการประสานสัมพันธ์จากทิศทางของกระแสลัดวงจร

มหาวิทยาลัยเทคโนโลยีสุรนารี

รายการอ้างอิง

เผด็จ เผ่าละออ (2550). การใช้โปรแกรมแมทแลบสำหรับวิศวกรไฟฟ้า. **เอกสารประกอบการสอนสาขาวิชาวิศวกรรมไฟฟ้า มหาวิทยาลัยเทคโนโลยีสุรนารี.**

ธนัดชัย กุลวรวานิชพงษ์ (2550). การวิเคราะห์ระบบไฟฟ้ากำลัง. **เอกสารประกอบการสอนสาขาวิชาวิศวกรรมไฟฟ้า มหาวิทยาลัยเทคโนโลยีสุรนารี.**

ธนัดชัย กุลวรวานิชพงษ์. (2550). เทคนิคการหาค่าเหมาะที่สุด. **เอกสารประกอบการสอนสาขาวิชาวิศวกรรมไฟฟ้า มหาวิทยาลัยเทคโนโลยีสุรนารี.**

ธนัดชัย กุลวรวานิชพงษ์. (2552). การป้องกันและรีเลย์ **เอกสารประกอบการสอนสาขาวิชาวิศวกรรมไฟฟ้า มหาวิทยาลัยเทคโนโลยีสุรนารี.**

ปริญญา สงวนสัตย์ (2553) .คู่มือ MATLAB ฉบับสมบูรณ์

Abyaneh H.A. , Kamangar S. , Razavi F. and Chabanloo R. M. (2008) A New Genetic Algorithm Method for Optimal Coordination of Overcurrent Relays in a Mixed Protection Scheme with Distance Relays, **Universities Power Engineering Conference, 2008. UPEC 2008. 43rd International**, page 1-5

Abdelaziz A. Y., Talaat H. E. A., Nosseir A. I., Hajjar A. A., (2002) An adaptive protection scheme for optimal coordination of overcurrent relays, **Electric Power Systems Research 61**, page1–9

Bashir M., Taghizadeh M., Sadeh J. and Mashhadi H. Rajabi, (2010) A New Hybride Particle Swarm Optimization for Optimal Coordination of Over Current Relay.,**International Conference on Power System Technology**

Chabanloo R. M., Abyaneh H. A., Sadat S., Kamangar H. and Razavi F., (2008) A New Gennetic Algorithm Method for Optimal Coordination of Overcurrent and Distance Relays Considering Various Characteristics for Overcurrent Relays., **2nd IEEE International Conterence on Power and Energy (PECon 0**8) December 1-3 2008 Johor Baharu, Malaysia

Chittaladakorn S., Prempree A., (2548) Optimal Design of water Distribution Piping system using differential evolution algorithm, **การประชุมวิชาการวิศวกรรมโยธาแห่งชาติครั้งที่ 10 , พฤษภาคม 2548**

Kamangar S. S. H., Abyaneh H. A., Razavi F. and Chabanloo R. M., (2010), Optimal Combined Overcurrent and Distance Relays Coordination Using a New Genetic Algorithm Method, **International Journal of Innovations in Energy Systems and Power**, Vol. 5 no. 1, April

Maleknia Mehdi, Karegar Hossein Kazemi, (2011) Optimal Coordination of Over-Current Relay with Distributed Generation Consideration., **The International Conference on Advanced Power System Automation and Protection**

Ralhan S. and Ray S., (2013) Directional Overcurrent Rrlays Coordination using Linear Programming Intervals : A Comparative Analusis., **Annual IEEE India Conference** (INDICON)

Ravikumar B., Thukaram D., and Khincha H. P., (2008) Knowledge Based Approach for transmission line Distance Relay Coordination , **National Power Systems Conference (NPSC)**

Razavi F., Abyaneh H.A., Al-Dabbagh M., Mohammadi R., Torkaman H., (2008) A new comprehensive genetic algorithm method for optimal overcurrent relays coordination , **Electric Power System Research**, Vol. 78, pp. 713-720

Shayesteh M. R. and Marvasti V., (2012) A New Approach for Optimal Distance Relays Coordination in the Meshed Networks., **International Conference on Electricla, Electronic and Communication Engineeing** (ICEEC'2012) September 8-9 Bangkok(Thailand)

Sidhu T. S., Baltazar D. S., Palomino R. M., and Sachdev M. S., (2004) A New Approach for Calculating Zone-2 Setting of Distance Relays and Its Use in an Adaptive Protection System., **IEEE Transactions On** Power Delivery Vol 19 no 1 January

Thangaraj R., Pant M., Deep K., (2010) Optimal coordination of over-current relays using modified differential evolution algorithms, **journal, Jan 16**

ภาคผนวก ก

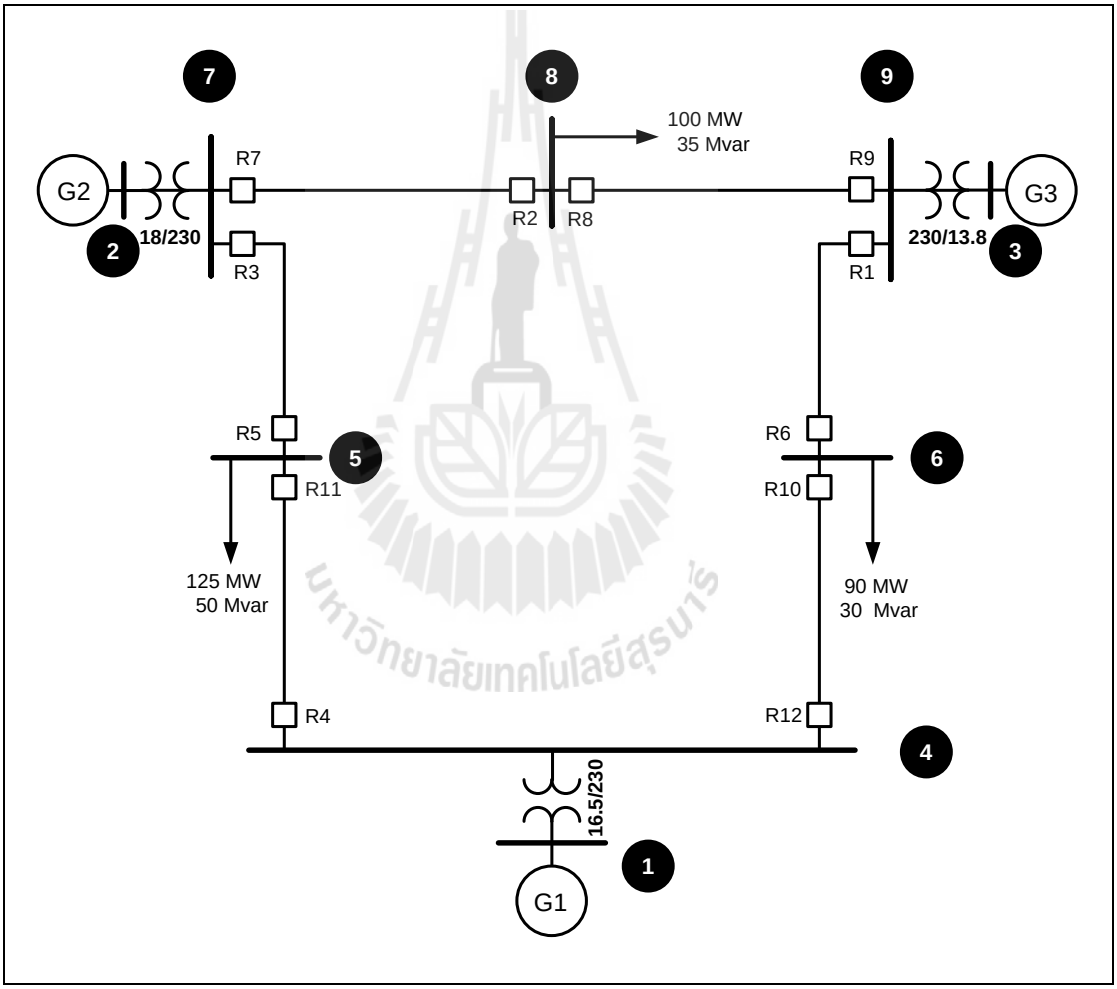
ระบบทดสอบ

มหาวิทยาลัยเทคโนโลยีสุรนารี

ระบบทดสอบ

ระบบทดสอบที่นำมาทดสอบในวิทยานิพนธ์นี้ ประกอบด้วยระบบทดสอบ 3 ระบบ คือ ระบบทดสอบIEEE 9 บัส 14 บัส และ 30 บัส รายละเอียดของระบบทดสอบแสดงได้ดังนี้

1. ระบบทดสอบมาตรฐาน WSCC 9 บัส



รูปที่ ก.1 ระบบทดสอบ WSCC 9 บัส

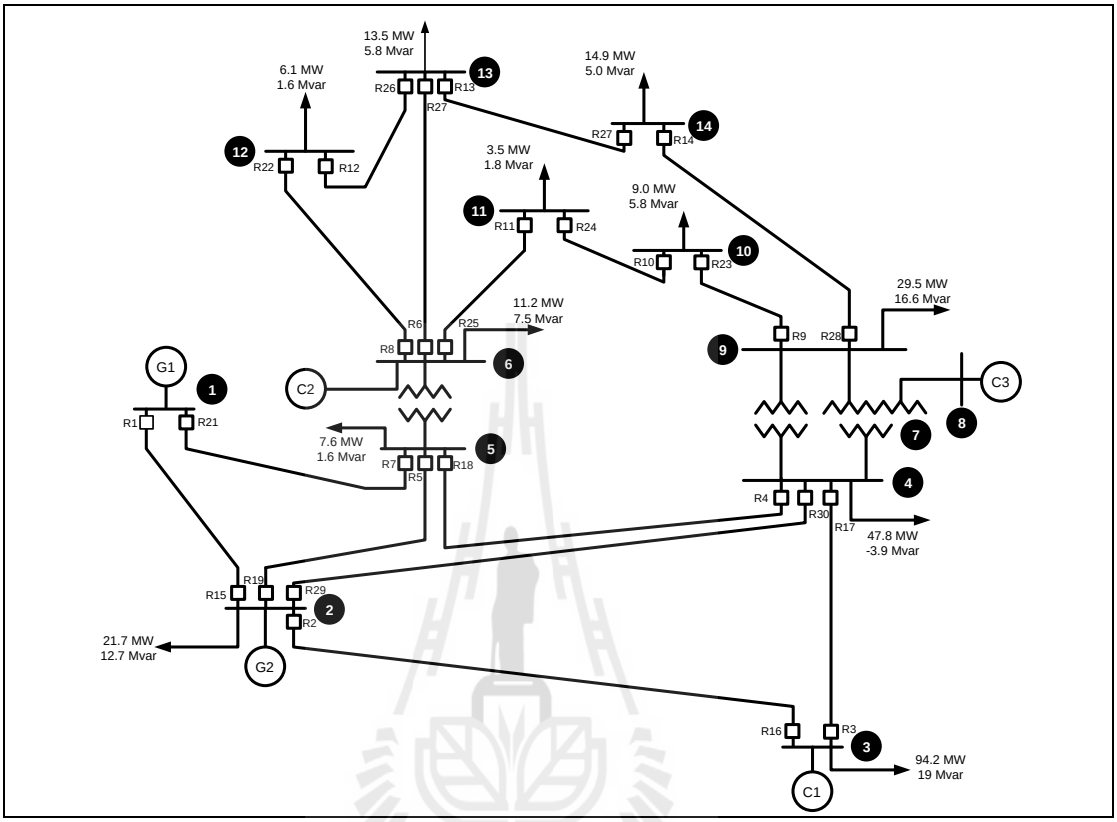
ตารางที่ ก.1 ข้อมูลสายส่งของระบบทดสอบ 9 บัส

สายส่ง	R (pu)	X (pu)	B/2	Tap
5 - 4	0.01	0.068	0.176	1
6 - 4	0.017	0.092	0.158	1
7 - 5	0.032	0.161	0.306	1
9 - 6	0.039	0.1738	0.358	1
7 - 8	0.0085	0.0576	0.149	1
8 - 9	0.0119	0.1008	0.209	1
4 - 1	0	0.0576	0	1
2 - 7	0	0.0625	0	1
9 - 3	0	0.0586	0	1

ตารางที่ ก.2 ข้อมูลบัสของระบบทดสอบ 9 บัส

บัส	ชนิด	แรงดัน (kV)	มุม (Deg)	เครื่องกำเนิด		โหลด	
				MW	Mvar	MW	Mvar
1	1	16.5	0	1	27.95	0	0
2	3	18.0	168.3126	163	4.903	0	0
3	3	13.8	70.9596	85	11.449	0	0
4	3	230	-150.002	0	0	0	0
5	3	230	-126.446	0	0	125	50
6	3	230	-100.318	0	0	90	30
7	3	230	153.103	0	0	0	0
8	3	230	307.579	0	0	100	35
9	3	230	202.304	0	0	0	0

2. ระบบทดสอบมาตรฐานIEEE 14 บัส



รูปที่ ก.2 ระบบทดสอบมาตรฐาน IEEE 14 บัส

ตารางที่ ก.3 ข้อมูลสายส่งของระบบทดสอบ 14 บัส

สายส่ง	R (pu)	X (pu)	B/2	Tap
1 - 2	0.01938	0.05917	0.0528	1
1 - 5	0.05403	0.22304	0.0492	1
2 - 3	0.04699	0.19797	0.0438	1
2 - 4	0.05811	0.17632	0.034	1
2 - 5	0.05695	0.17388	0.0346	1
3 - 4	0.06701	0.17103	0.0128	1
4 - 5	0.01335	0.04211	0	1

ตารางที่ ก.3 ข้อมูลสายส่งของระบบทดสอบ 14 บัส (ต่อ)

สายส่ง	R (pu)	X (pu)	B/2	Tap
4 - 7	0	0.20912	0	0.978
4 - 9	0	0.55618	0	0.969
5 - 6	0	0.25202	0	0.932
6 - 11	0.09498	0.1989	0	1
6 - 12	0.12291	0.25581	0	1
6 - 13	0.06615	0.13027	0	1
7 - 8	0	0.17615	0	1
7 - 9	0	0.11001	0	1
9 - 10	0.03181	0.0845	0	1
9 - 14	0.12711	0.27038	0	1
10 - 11	0.08205	0.19207	0	1
12 - 13	0.22092	0.19988	0	1
13 - 14	0.17093	0.34802	0	1

ตารางที่ ก.4 ข้อมูลบัสของระบบทดสอบ 14 บัส

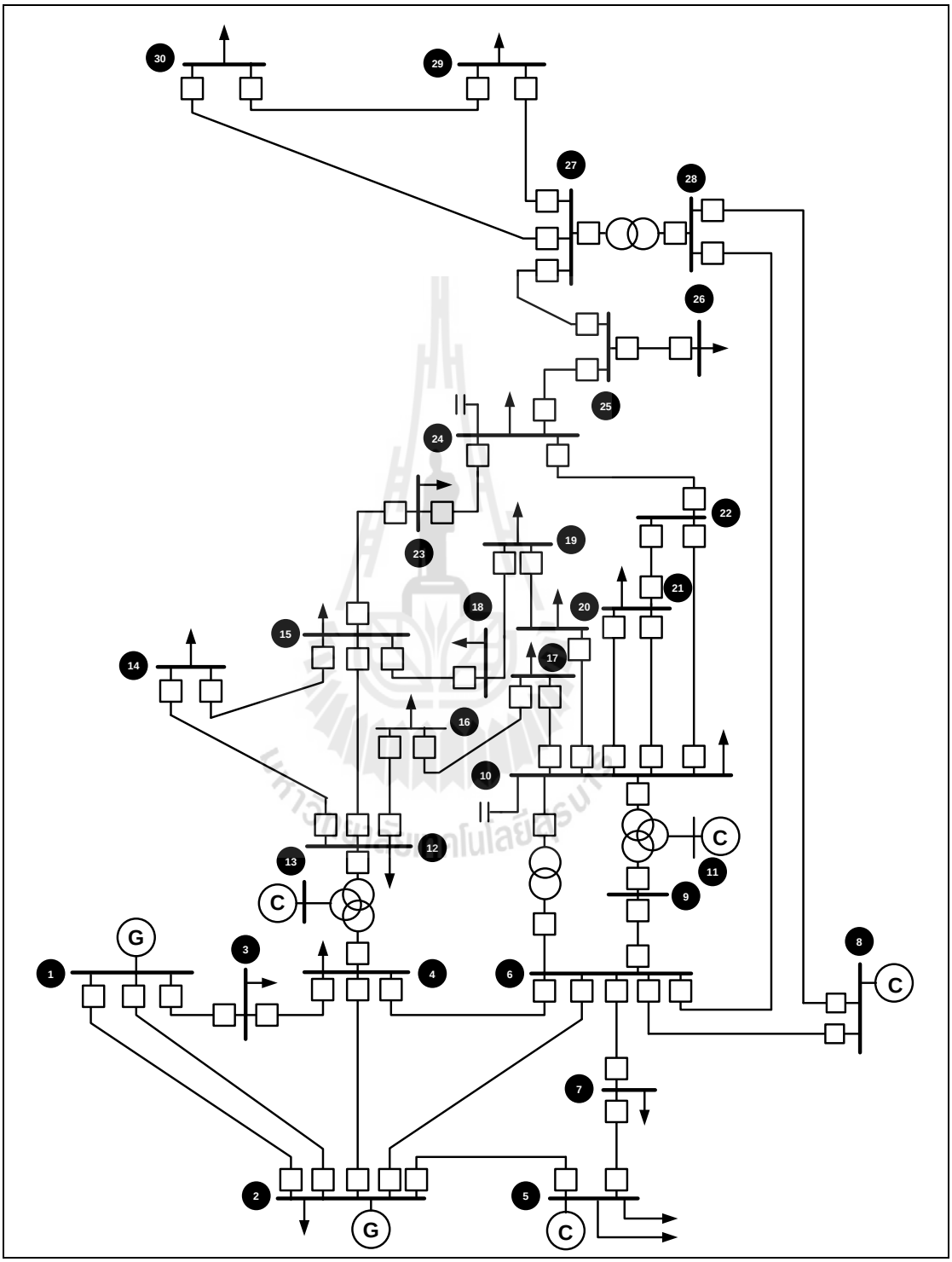
บัส	ชนิด	แรงดัน (kV)	มุม (Deg)	เครื่องกำเนิด		โหลด	
				MW	Mvar	MW	Mvar
1	1	132	0	232.4	-16.9	0	0
2	2	132	-4.98	40	42.4	21.7	12.7
3	2	132	-12.72	0	23.4	94.2	19
4	3	132	-10.33	0	0	47.8	-3.9
5	3	132	-8.78	0	0	7.6	1.6
6	2	33	-14.22	0	12.2	11.2	7.5
7	3	1	-13.37	0	0	0	0

ตารางที่ ก.4 ข้อมูลบัสของระบบทดสอบ 14 บัส (ต่อ)

บัส	ชนิด	แรงดัน (kV)	มุม (Deg)	เครื่องกำเนิด		โหลด	
				MW	Mvar	MW	Mvar
8	2	10	-13.36	0	17.4	0	0
9	3	33	-14.94	0	0	29.5	16.6
10	3	33	-15.1	0	0	9	5.8
11	3	33	-14.79	0	0	3.5	1.8
12	3	33	-15.07	0	0	6.1	1.6
13	3	33	-15.16	0	0	13.5	5.8
14	3	33	-16.04	0	0	14.9	5

มหาวิทยาลัยเทคโนโลยีสุรนารี

3. ระบบทดสอบมาตรฐาน IEEE 30 บัส



รูปที่ ก.3 ระบบมาตรฐาน IEEE 30 บัส

ตารางที่ ก.5 ข้อมูลสายส่งของระบบทดสอบ 30 บัส

สายส่ง	R (pu)	X (pu)	B/2	Tap
1 - 2	0.0192	0.0575	0.0528	1
1 - 3	0.0452	0.1652	0.0408	1
2 - 4	0.057	0.1737	0.0368	1
3 - 4	0.0132	0.0379	0.0084	1
2 - 5	0.0472	0.1983	0.0418	1
2 - 6	0.0581	0.1763	0.0374	1
4 - 6	0.0119	0.0414	0.009	1
5 - 7	0.046	0.116	0.0204	1
6 - 7	0.0267	0.082	0.017	1
6 - 8	0.012	0.042	0.009	1
6 - 9	0	0.208	0	0.978
6 - 10	0	0.556	0	0.969
9 - 11	0	0.208	0	1
9 - 10	0	0.11	0	1
4 - 12	0	0.256	0	0.932
12 - 13	0	0.14	0	1
12 - 14	0.1231	0.2559	0	1
12 - 15	0.0662	0.1304	0	1
12 - 16	0.0945	0.1987	0	1
14 - 15	0.221	0.1997	0	1
16 - 17	0.0524	0.1923	0	1
15 - 18	0.1073	0.2185	0	1
18 - 19	0.0639	0.1292	0	1

ตารางที่ ก.5 ข้อมูลสายส่งของระบบทดสอบ 30 บัส

สายส่ง	R (pu)	X (pu)	B/2	Tap
19 - 20	0.034	0.068	0	1
10 - 20	0.0936	0.209	0	1
10 - 17	0.0324	0.0845	0	1
10 - 21	0.0348	0.0749	0	1
10 - 22	0.0727	0.1499	0	1
21 - 22	0.0116	0.0236	0	1
15 - 23	0.1	0.202	0	1
22 - 24	0.115	0.179	0	1
23 - 24	0.132	0.27	0	1
24 - 25	0.1885	0.3292	0	1
25 - 26	0.2544	0.38	0	1
25 - 27	0.1093	0.2087	0	1
28 - 27	0	0.396	0	0.968
27 - 29	0.2198	0.4153	0	1
27 - 30	0.3202	0.6027	0	1
29 - 30	0.2399	0.4533	0	1
8 - 28	0.0636	0.2	0.0428	1
6 - 28	0.0169	0.0599	0.013	1

ตารางที่ ก.6 ข้อมูลบัสของระบบทดสอบ 30 บัส

บัส	ชนิด	แรงดัน (kV)	มุม (Deg)	เครื่องกำเนิด		โหลด	
				MW	Mvar	MW	Mvar
1	1	132	0	260.2	-16.1	0	0
2	2	132	-5.48	40	50	21.7	12.7
3	3	132	-7.96	0	0	2.4	1.2
4	3	132	-9.62	0	0	7.6	1.6
5	2	132	-14.37	0	37	94.2	19
6	3	132	-11.34	0	0	0	0
7	3	132	-13.12	0	0	22.8	10.9
8	2	132	-12.1	0	37.3	30	30
9	3	1	-14.38	0	0	0	0
10	3	33	-15.97	0	0	5.8	2
11	2	11	-14.39	0	16.2	0	0
12	3	33	-15.24	0	0	11.2	7.5
13	2	11	-15.24	0	10.6	0	0
14	3	33	-16.13	0	0	6.2	1.6
15	3	33	-16.22	0	0	8.2	2.5
16	3	33	-15.83	0	0	3.5	1.8
17	3	33	-16.14	0	0	9	5.8
18	3	33	-16.82	0	0	3.2	0.9
19	3	33	-17	0	0	9.5	3.4
20	3	33	-16.8	0	0	2.2	0.7
21	3	33	-16.42	0	0	17.5	11.2
22	3	33	-16.41	0	0	0	0
23	3	33	-16.61	0	0	3.2	1.6
24	3	33	-16.78	0	0	8.7	6.7
25	3	33	-16.35	0	0	0	0

ตารางที่ ก.6 ข้อมูลบัสของระบบทดสอบ 30 บัส (ต่อ)

บัส	ชนิด	แรงดัน (kV)	มุม (Deg)	เครื่องกำเนิด		โหลด	
				MW	Mvar	MW	Mvar
6	3	33	-16.77	0	0	3.5	2.3
27	3	33	-15.82	0	0	0	0
28	3	132	-11.97	0	0	0	0
29	3	33	-17.06	0	0	2.4	0.9
30	3	33	-17.94	0	0	10.6	1.9

มหาวิทยาลัยเทคโนโลยีสุรนารี

ภาคผนวก ข

ผลการจำลองกระแสลัดวงจรจากโปรแกรม Power World Simulator

มหาวิทยาลัยเทคโนโลยีสุรนารี

ในงานวิจัยนี้ ผู้จัดทำสนใจการประสานสัมพันธ์รีเลย์เหมาะที่สุดของรีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทางในระบบไฟฟ้ากำลัง ซึ่งจำเป็นต้องรู้ค่ากระแสลัดวงจรเพื่อนำไปใช้ในการคำนวณค่าเวลาการทำงานของรีเลย์ ซึ่งในวิทยานิพนธ์นี้ผู้จัดทำได้เลือกใช้โปรแกรมจำลองผลด้วย Power World Simulator เพื่อหาค่ากระแสลัดวงจร

ตารางที่ ข.1 ค่ากระแสลัดวงจรรีเลย์หลัก ระบบ 9 บัส

รีเลย์ปฐมภูมิ	กระแสลัดวงจร	รีเลย์ทุติยภูมิ	กระแสลัดวงจร
1	21.4834	6	10.647
2	10.9984	7	27.5024
3	39.5058	5	10.6438
4	39.608	11	11.2536
5	24.6094	3	15.9252
6	22.6776	1	11.2244
7	41.9646	2	8.4012
8	27.342	9	11.0674
9	15.8994	8	16.2818
10	11.1194	12	22.853
11	15.7584	4	24.8158
12	43.2644	10	7.5318

ตารางที่ ข.2 ค่ากระแสลัดวงจร รีเลย์หลักและสำรอง ระบบ 9 บัส

ปฐมภูมิ				ทุติยภูมิ			
รีเลย์สำรอง		รีเลย์สำหลัก		รีเลย์สำรอง		รีเลย์สำหลัก	
รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร
10	7.42	4	39.6	10	7.42	4	39.6
11	11.08	12	43.26	11	11.08	12	43.26
4	24.48	5	24.6	4	24.48	5	24.6
12	22.52	6	22.68	12	22.52	6	22.68
5	10.48	7	41.96	5	10.48	7	41.96
7	27.24	8	27.34	7	27.24	8	27.34
3	11.78	11	11.26	3	11.78	11	11.26
1	7.84	10	7.54	1	7.84	10	7.54
2	3.9	3	15.92	2	3.9	3	15.92
8	7.08	1	11.22	8	7.08	1	11.22
9	8.62	2	8.4	9	8.62	2	8.4
6	6.68	9	11.06	6	6.68	9	11.06
10	4.9333	4	24.82	10	4.9333	4	24.82
11	4.45	12	22.86	11	4.45	12	22.86
4	11.9	5	10.64	4	11.9	5	10.64
12	11.4	6	10.64	12	11.4	6	10.64
5	5.72	7	27.5	5	5.72	7	27.5
7	16.84	8	16.28	7	16.84	8	16.28
3	15.76	11	15.76	3	15.76	11	15.76
1	11.12	10	11.12	1	11.12	10	11.12
2	8.2	3	39.5	2	8.2	3	39.5
8	16.06	1	21.48	8	16.06	1	21.48
9	11	2	11	9	11	2	11
6	10.52	9	15.9	6	10.52	9	15.9

ตารางที่ ข.3 ค่ากระแสลัดวงจรรีเลย์หลัก ระบบ 14 บัส

รีเลย์ปฐมภูมิ	กระแสลัดวงจร	รีเลย์ทุติยภูมิ	กระแสลัดวงจร
1	12.5044	15	14.089
2	20.6131	16	5.1223
3	14.7889	17	7.0699
4	15.4258	18	8.9544
5	18.0602	19	5.6516
6	66.9011	20	5.9104
7	19.0622	21	5.4313
8	70.6817	22	3.8803
9	57.9282	23	13.3798
10	42.0023	24	18.6281
11	23.4066	25	31.8896
12	26.9213	26	19.6649
13	44.9022	27	11.8322
14	19.3407	28	27.6641
15	20.073	1	9.53
16	14.3476	2	7.8471
17	18.2237	3	5.839
18	13.5019	4	10.3778
19	21.3906	5	4.4567
20	20.737	6	36.5046
21	18.7356	7	5.5535
22	19.3177	8	27.3464
23	18.4642	9	42.4099
24	31.5411	10	23.6738
25	61.3182	11	11.504
26	47.4816	12	9.4048

ตารางที่ ข.3 ค่ากระแสลัดวงจรรีเลย์หลัก ระบบ 14 บัส (ต่อ)

รีเลย์ปฐมภูมิ	กระแสลัดวงจร	รีเลย์ทุติยภูมิ	กระแสลัดวงจร
27	27.2438	13	19.6052
28	62.3013	14	8.5321
29	21.1993	30	4.971
30	18.6752	29	5.6679

ตารางที่ ข.4 ค่ากระแสลัดวงจร รีเลย์หลักและสำรอง ระบบ 14 บัส

รีเลย์หลัก	กระแสลัดวงจร	รีเลย์สำรอง	กระแสลัดวงจร	รีเลย์หลัก	กระแสลัดวงจร	รีเลย์สำรอง	กระแสลัดวงจร
7	5.4286	1	12.5044	12	9.1271	20	20.737
15	13.8947	21	18.7356	27	11.6474	20	20.737
1	9.4582	2	20.6131	24	18.4654	23	18.4642
5	4.9393	2	20.6131	13	19.3477	14	19.3407
30	4.7471	2	20.6131	25	31.5454	24	31.5411
1	9.456	29	21.1993	27	11.5921	26	47.4816
16	5.6125	29	21.1993	6	35.9231	26	47.4816
5	4.8938	29	21.1993	28	27.2658	27	27.2438
1	9.4483	19	21.3906	7	4.7758	1	9.53
16	5.629	19	21.3906	15	7.7104	21	5.4313
30	4.7207	19	21.3906	1	7.7256	2	7.8471
2	7.6798	3	14.7889	5	4.03	2	7.8471
29	5.5199	4	15.4258	30	4.553	2	7.8471
23	4.2455	4	15.4258	1	7.719	29	5.6679
14	3.8283	4	15.4258	16	4.6028	29	5.6679
3	5.7429	4	15.4258	5	4.4151	29	5.6679
20	5.6244	25	61.3182	1	7.7059	19	5.6516
22	4.2733	25	61.3182	16	4.9155	19	5.6516
21	4.9941	25	61.3182	30	4.327	19	5.6516

ตารางที่ ข.4 ค่ากระแสลัดวงจร รีเลย์หลักและสำรอง ระบบ 14 บัส (ต่อ)

รีเลย์หลัก	กระแสลัดวงจร	รีเลย์สำรอง	กระแสลัดวงจร	รีเลย์หลัก	กระแสลัดวงจร	รีเลย์สำรอง	กระแสลัดวงจร
19	4.3559	25	61.3182	2	4.5877	3	5.839
4	5.3133	25	61.3182	29	4.4214	4	10.3778
20	5.3896	8	70.6817	23	1.645	4	10.3778
11	11.2237	8	70.6817	14	1.657	4	10.3778
4	5.3214	8	70.6817	3	5.0384	4	10.3778
19	4.3473	8	70.6817	19	4.2345	8	27.3464
21	4.9917	8	70.6817	21	4.94	8	27.3464
22	3.9453	6	66.9011	22	9.2867	6	36.5046
21	4.9946	6	66.9011	21	4.928	6	36.5046
19	4.3593	6	66.9011	19	4.4843	6	36.5046
4	5.3326	6	66.9011	4	4.6906	6	36.5046
11	11.2813	6	66.9011	11	5.4664	6	36.5046
18	4.838	9	57.9282	18	4.253	9	42.4099
29	4.5236	9	57.9282	29	4.9088	9	42.4099
3	4.6643	9	57.9282	3	4.7171	9	42.4099
14	8.3499	9	57.9282	14	5.2319	9	42.4099
23	13.1153	28	62.3013	23	4.6388	28	27.6641
18	4.8252	28	62.3013	18	4.777	28	27.6641
29	4.5129	28	62.3013	29	4.6033	28	27.6641
3	4.6428	28	62.3013	3	4.6107	28	27.6641
9	42.013	10	42.0023	9	24.1468	10	23.6738
8	26.9251	12	26.9213	8	9.566	12	9.4048
12	9.1201	13	44.9022	12	4.5404	13	19.6052
6	35.9168	13	44.9022	6	16.4908	13	19.6052

ตารางที่ ข.5 ค่ากระแสลัดวงจรรีเลย์หลัก ระบบ 30 บัส

รีเลย์ปฐมภูมิ	กระแสลัดวงจร	รีเลย์ทุติยภูมิ	กระแสลัดวงจร	รีเลย์ปฐมภูมิ	กระแสลัดวงจร	รีเลย์ทุติยภูมิ	กระแสลัดวงจร
1	1121.06	31	1387.76	35	1310.89	5	767.34
2	2063.9	32	413.8	36	578.35	6	1498.03
3	657.22	33	1581.33	37	2135.22	7	448.13
4	1182.98	34	1309.54	38	1991.05	8	645.04
5	1385.22	35	724.9	39	1368.54	9	771.46
6	2436.22	36	147.46	40	3809.24	10	1728.11
7	780.04	37	1323.94	41	2346.14	11	3219.29
8	981.74	38	1400.71	42	1888.11	12	2492.47
9	2030.74	39	511.89	43	2539.31	13	3064.82
10	5917.49	40	411.72	44	4778.56	14	792.59
11	5760.19	41	592.81	45	1737.69	15	2484.95
12	6445.3	42	241.57	46	3737.87	16	1696.15
13	5851.41	43	872.61	47	5144.38	17	1198.47
14	2453.6	44	1922.15	48	2391.66	18	1820.64
15	4678.9	45	890.05	49	2849.75	19	1531.45
16	2871.35	46	2147.37	50	5630.62	20	687.21
17	1680.9	47	3773	51	2951.66	21	2601.27
18	2463.69	48	1753.38	52	2353.13	22	2410.64
19	1811.89	49	2403.75	53	3395.2	23	1351.19
20	1519.72	50	2870.72	54	1822.19	24	1849.27
21	3192.92	51	2369.85	55	0.65	25	1845.92
22	4262.15	52	1283.41	56	610.65	26	1614.08
23	2566.39	53	1708.75	57	1832.23	27	1839.16

ตารางที่ ข.5 ค่ากระแสลัดวงจรรีเลย์หลัก ระบบ 30 บัส (ต่อ)

รีเลย์ปฐมภูมิ	กระแสลัดวงจร	รีเลย์ทุติยภูมิ	กระแสลัดวงจร	รีเลย์ปฐมภูมิ	กระแสลัดวงจร	รีเลย์ทุติยภูมิ	กระแสลัดวงจร
24	3702.99	54	1033.26	58	2074.72	28	249.65
25	3620	55	1861.87	59	1207.21	29	845.2
26	3925.71	56	10.7	60	3902.03	30	15.42
27	2699.24	57	1317.49	61	1745.7	62	562.54
28	1581.1	58	479.39	62	1560.51	61	664.8
29	1591.26	59	620.33	63	1992.71	64	504.96
30	832.17	60	1226.2	64	2154.05	63	465.06
31	2008.58	1	810.58	65	5732.17	66	972.64
32	1952.29	2	494.03	66	2121.98	65	2903.62
33	2067.85	3	478.52	67	4655.47	68	917.17
34	1907.49	4	741.7	68	1688.89	67	2596.55

ตารางที่ ข.6 ค่ากระแสลัดวงจร รีเลย์หลักและสำรอง ระบบ 30 บัส

ปฐมภูมิ				ทุติยภูมิ			
รีเลย์สำรอง		รีเลย์สำหลัก		รีเลย์สำรอง		รีเลย์สำหลัก	
รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร
31	15.2194	61	19.3967	32	4.9804	31	9.0064
1	8.9042	2	22.9322	64	5.4449	31	9.0064
39	5.5542	2	22.9322	39	5.5897	31	9.0064
64	5.3757	2	22.9322	32	4.7636	62	7.3867
61	7.2777	3	7.3024	2	5.33	33	5.3169
1	8.9198	9	22.5638	34	14.4563	33	5.3169
32	4.945	9	22.5638	66	3.9755	33	5.3169

ตารางที่ ข.6 ค่ากระแสลัดวงจร รีเลย์หลักและสำรอง ระบบ 30 บัส (ต่อ)

ปฐมภูมิ				ทุติยภูมิ			
รีเลย์สำรอง		รีเลย์สำหลัก		รีเลย์สำรอง		รีเลย์สำหลัก	
รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร
64	5.398	9	22.5638	43	3.302	33	5.3169
1	8.9151	63	22.1412	42	1.243	33	5.3169
32	4.9099	63	22.1412	35	7.8919	39	8.5718
39	5.5474	63	22.1412	4	8.0271	64	5.1673
3	5.2828	4	13.1442	20	1.935	64	5.1673
2	5.3264	4	13.1442	17	3.5047	64	5.1673
42	1.245	4	13.1442	41	3.595	64	5.1673
43	3.292	4	13.1442	40	2.3495	64	5.1673
66	3.912	4	13.1442	36	4.7	64	5.1673
9	8.4264	5	15.3913	8	7.0802	64	5.1673
63	5.6218	37	23.7247	63	5.6238	34	8.2411
4	8.1431	37	23.7247	7	5.5341	34	8.2411
20	1.9865	37	23.7247	8	7.1181	34	8.2411
17	3.5597	37	23.7247	36	4.7277	34	8.2411
41	3.5845	37	23.7247	40	2.345	34	8.2411
40	2.3425	37	23.7247	41	3.5885	34	8.2411
36	4.7247	37	23.7247	17	3.509	34	8.2411
8	7.107	37	23.7247	20	1.9475	34	8.2411
7	5.5369	6	27.0691	37	14.5582	35	8.7482
63	5.0141	6	27.0691	5	8.658	7	14.7104
4	8.1631	6	27.0691	28	5.162	8	16.6448
20	1.9965	6	27.0691	44	20.9893	42	27.6941
17	3.5817	6	27.0691	14	8.4977	43	34.0536

ตารางที่ ข.6 ค่ากระแสลัดวงจร รีเลย์หลักและสำรอง ระบบ 30 บัส (ต่อ)

ปฐมภูมิ				ทุติยภูมิ			
รีเลย์สำรอง		รีเลย์สำหลัก		รีเลย์สำรอง		รีเลย์สำหลัก	
รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร
41	3.652	6	27.0691	46	23.5904	66	32.2624
40	2.387	6	27.0691	13	33.4477	44	8.8066
36	0	6	27.0691	45	9.6863	44	8.8066
3	4.6926	12	71.6144	68	9.997	44	8.8066
2	4.3668	12	71.6144	47	41.5426	46	18.8461
34	6.5066	12	71.6144	48	19.3174	45	27.6106
43	9.1221	12	71.6144	49	26.5792	48	20.2293
66	10.5116	12	71.6144	50	31.6664	49	17.0161
42	4.0916	13	65.0157	19	16.8952	20	31.8969
3	4.696	13	65.0157	16	18.6788	17	41.9222
2	4.3812	13	65.0157	51	26.0687	41	35.7699
34	6.5228	13	65.0157	21	28.4391	40	19.2012
66	10.5569	13	65.0157	52	14.0642	40	19.2012
43	9.3308	65	63.6908	52	14.1369	51	28.903
42	4.2614	65	63.6908	10	18.7924	51	28.903
3	4.6944	65	63.6908	53	18.7692	68	28.8506
2	4.3753	65	63.6908	23	14.8097	52	26.7849
34	6.5087	65	63.6908	54	11.3463	52	26.7849
12	27.2673	14	27.2622	54	11.313	53	15.0132
65	31.9084	16	31.9039	22	26.4242	53	15.0132
68	9.9852	15	51.9878	27	20.2497	54	20.5474
14	8.5359	15	51.9878	30	0.265	27	14.6388
13	33.5889	15	51.9878	56	0.23	27	14.6388

ตารางที่ ข.6 ค่ากระแสลัดวงจร รีเลย์หลักและสำรอง ระบบ 30 บัส (ต่อ)

ปฐมภูมิ				ทุติยภูมิ			
รีเลย์สำรอง		รีเลย์สำหลัก		รีเลย์สำรอง		รีเลย์สำหลัก	
รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร
15	27.3761	18	27.3743	6	5.1586	27	14.6388
18	20.1348	19	20.1321	58	4.3845	27	14.6388
17	13.1072	50	62.5624	59	6.7858	56	17.9342
41	6.29	50	62.5624	29	9.2513	30	13.6244
40	4.6246	50	62.5624	60	13.4251	59	9.3911
36	4.8415	50	62.5624	6	16.3311	28	5.3266
8	5.1454	50	62.5624	57	5.2051	28	5.3266
7	4.4307	50	62.5624	30	0.76	28	5.3266
63	4.8926	50	62.5624	56	0.669	28	5.3266
4	4.263	50	62.5624	58	5.162	36	16.6448
40	4.6628	47	57.1598	56	0.633	36	16.6448
36	4.8475	47	57.1598	30	0.7195	36	16.6448
8	5.1576	47	57.1598	57	5.2374	36	16.6448
7	4.445	47	57.1598	3	4.6903	1	22.3176
63	4.8974	47	57.1598	31	7.788	61	17.339
4	4.2552	47	57.1598	1	6.8423	2	21.6921
20	7.4757	47	57.1598	39	4.38	2	21.6921
41	6.3418	47	57.1598	64	4.789	2	21.6921
17	13.211	11	64.0021	61	5.295	3	22.9761
20	7.4843	11	64.0021	1	6.8654	9	15.206
4	4.2502	11	64.0021	32	4.0408	9	15.206
2	4.4208	11	64.0021	64	4.4298	9	15.206
7	4.444	11	64.0021	1	6.7543	63	23.9339

ตารางที่ ข.6 ค่ากระแสลัดวงจร รีเลย์หลักและสำรอง ระบบ 30 บัส (ต่อ)

ปฐมภูมิ				ทุติยภูมิ			
รีเลย์สำรอง		รีเลย์สำหลัก		รีเลย์สำรอง		รีเลย์สำหลัก	
รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร
8	5.1558	11	64.0021	32	4.6495	63	23.9339
36	4.855	11	64.0021	39	4.4928	63	23.9339
40	4.4603	11	64.0021	3	4.6266	4	21.1943
36	4.853	10	65.7499	2	4.8005	4	21.1943
8	5.1479	10	65.7499	42	3.059	4	21.1943
7	4.4352	10	65.7499	43	4.8953	4	21.1943
63	4.8954	10	65.7499	66	4.7953	4	21.1943
4	4.2543	10	65.7499	9	4.7013	5	14.5654
20	7.4552	10	65.7499	63	4.5562	37	8.6671
17	13.167	10	65.7499	4	5.5604	37	8.6671
41	5.9551	10	65.7499	20	1.8835	37	8.6671
11	35.478	21	35.4769	17	2.6815	37	8.6671
14	8.542	67	51.7274	41	3.932	37	8.6671
13	33.6122	67	51.7274	40	2.57	37	8.6671
45	9.6857	67	51.7274	36	3.7677	37	8.6671
21	28.6428	22	47.3572	8	5.843	37	8.6671
10	18.7196	22	47.3572	7	4.4604	6	6.4261
67	28.5183	23	28.5154	63	4.7092	6	6.4261
23	14.7627	24	41.1443	4	6.3266	6	6.4261
22	26.4003	24	41.1443	20	1.51	6	6.4261
24	20.2358	25	40.2222	17	3.4423	6	6.4261
27	20.1637	25	40.2222	41	4.6573	6	6.4261
24	20.3619	57	20.3581	40	4.0583	6	6.4261

ตารางที่ ข.6 ค่ากระแสลัดวงจร รีเลย์หลักและสำรอง ระบบ 30 บัส (ต่อ)

ปฐมภูมิ				ทุติยภูมิ			
รีเลย์สำรอง		รีเลย์สำหลัก		รีเลย์สำรอง		รีเลย์สำหลัก	
รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร
30	0.582	26	43.619	36	0	6	6.4261
57	14.4257	26	43.619	3	4.601	12	20.979
6	5.1252	26	43.619	2	4.2684	12	20.979
58	4.3575	26	43.619	34	5.4534	12	20.979
57	14.3531	60	43.3559	43	11.0011	12	20.979
6	5.1	60	43.3559	66	4.8564	12	20.979
58	4.3385	60	43.3559	42	8.6481	13	28.2146
56	0.8375	60	43.3559	3	4.6223	13	28.2146
27	7.7806	29	17.6807	2	4.6344	13	28.2146
38	15.2008	58	23.0524	34	5.5309	13	28.2146
8	7.0629	6	27.0691	66	4.798	13	28.2146
7	5.5369	6	27.0691	43	1.824	65	23.5776
63	5.0141	6	27.0691	42	1.516	65	23.5776
4	8.1631	6	27.0691	3	4.528	65	23.5776
20	1.9965	6	27.0691	2	4.298	65	23.5776
17	3.5817	6	27.0691	34	4.8166	65	23.5776
41	3.652	6	27.0691	12	8.9589	14	53.0951
40	2.387	6	27.0691	65	18.9489	16	41.5319
32	4.7455	31	15.4196	68	4.4813	15	19.3077
64	4.4229	31	15.4196	14	5.4856	15	19.3077
39	4.8283	31	15.4196	13	19.7132	15	19.3077
32	4.605	62	6.2504	15	20.2988	18	26.574
2	4.853	33	17.5703	18	17.1181	19	31.6639

ตารางที่ ข.6 ค่ากระแสลัดวงจร รีเลย์หลักและสำรอง ระบบ 30 บัส (ต่อ)

ปฐมภูมิ				ทุติยภูมิ			
รีเลย์สำรอง		รีเลย์สำหลัก		รีเลย์สำรอง		รีเลย์สำหลัก	
รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร
34	11.5686	33	17.5703	17	5.3586	50	16.8858
66	2.9175	33	17.5703	41	3.9593	50	16.8858
43	3.381	33	17.5703	40	3.45	50	16.8858
42	2.141	33	17.5703	36	4.33	50	16.8858
35	3.133	39	5.6877	8	4.9419	50	16.8858
4	5.3245	64	5.6077	7	4.252	50	16.8858
20	2.339	64	5.6077	63	4.5618	50	16.8858
17	1.095	64	5.6077	4	4.4974	50	16.8858
41	4.262	64	5.6077	40	4.7782	47	18.6767
40	2.7855	64	5.6077	36	4.488	47	18.6767
36	3.502	64	5.6077	8	4.7908	47	18.6767
8	5.971	64	5.6077	7	4.793	47	18.6767
63	4.9038	34	14.5504	63	4.6814	47	18.6767
7	4.3953	34	14.5504	4	4.303	47	18.6767
8	6.4093	34	14.5504	20	4.7476	47	18.6767
36	3.6263	34	14.5504	41	4.5694	47	18.6767
40	2.87	34	14.5504	17	10.3566	11	26.0682
41	4.3915	34	14.5504	20	5.3588	11	26.0682
17	3.5953	34	14.5504	4	4.4703	11	26.0682
20	3.952	34	14.5504	2	4.2112	11	26.0682
37	8.2161	35	8.0544	7	4.8672	11	26.0682
5	5.0388	7	5.6016	8	4.7608	11	26.0682
28	4.7665	8	4.9153	44	4.7608	11	26.0682

ตารางที่ ข.6 ค่ากระแสลัดวงจร รีเลย์หลักและสำรอง ระบบ 30 บัส (ต่อ)

ปฐมภูมิ				ทุติยภูมิ			
รีเลย์สำรอง		รีเลย์สำหลัก		รีเลย์สำรอง		รีเลย์สำหลัก	
รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร
44	4.8896	42	4.8314	36	3.648	10	42.3249
14	4.1512	43	9.6957	8	4.7265	10	42.3249
46	10.8838	66	10.8071	7	4.8408	10	42.3249
13	11.3029	44	21.3572	63	4.7986	10	42.3249
45	5.1874	44	21.3572	4	4.493	10	42.3249
68	5.4614	44	21.3572	20	5.6179	10	42.3249
47	24.2596	46	23.8597	17	9.9409	10	42.3249
48	9.9467	45	9.8894	41	28.4391	10	42.3249
49	19.703	48	19.482	11	28.903	21	32.7962
50	26.7434	49	26.7083	14	5.0936	67	18.7654
19	7.6647	20	7.6357	13	20.5648	67	18.7654
16	13.4258	17	13.3163	45	4.7974	67	18.7654
51	6.5868	41	6.5868	21	16.1987	22	26.1459
21	5.9551	40	5.1465	10	10.5867	22	26.1459
52	10.5288	40	5.1465	67	15.1511	23	37.7244
52	12.9312	51	26.3317	23	7.6949	24	20.2466
10	13.5437	51	26.3317	22	13.7279	24	20.2466
53	10.2612	68	10.1908	24	10.3799	25	0.325
23	7.2112	52	14.2601	27	10.1889	25	0.325
54	7.4043	52	14.2601	24	14.785	57	29.9916
54	6.4862	53	18.9861	30	7.0793	26	6.785
22	13.1304	53	18.9861	57	7.8294	26	6.785
27	11.6689	54	11.4807	6	4.2822	26	6.785

ตารางที่ ข.6 ค่ากระแสลัดวงจร รีเลย์หลักและสำรอง ระบบ 30 บัส (ต่อ)

ปฐมภูมิ				ทุติยภูมิ			
รีเลย์สำรอง		รีเลย์สำหลัก		รีเลย์สำรอง		รีเลย์สำหลัก	
รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร	รีเลย์	กระแสลัดวงจร
30	1.7785	27	20.4351	58	3.518	26	6.785
56	1.565	27	20.4351	57	7.1226	60	9.2463
6	4.4583	27	20.4351	6	4.701	60	9.2463
58	4.373	27	20.4351	58	4.9085	60	9.2463
59	0.5405	56	0.535	56	9.3611	60	9.2463
29	0.815	30	0.771	27	7.1016	29	13.4134
60	7.1849	59	6.8926	38	5.017	58	17.5678
6	4.9193	28	4.993	8	4.3222	6	6.4261
57	4.9118	28	4.993	7	4.4604	6	6.4261
30	1.19	28	4.993	63	4.7092	6	6.4261
56	1.047	28	4.993	4	6.3266	6	6.4261
58	4.7665	36	4.9153	20	1.51	6	6.4261
56	0.412	36	4.9153	17	3.4423	6	6.4261
30	0.4685	36	4.9153	41	4.6573	6	6.4261
57	4.4526	36	4.9153	40	4.0583	6	6.4261

Infeed หาค่าInfeed คำนวณโดยใช้โค้ดโปรแกรมแมทแลบ โดยนำค่าที่ได้จากการจำลองการลัดวงจรแบบ 3 เฟส ด้วยโปรแกรม Power World Simulator มาคำนวณ สามารถคำนวณหาค่า Infeed ได้ดังตารางต่อไปนี้

ตารางที่ ข.7 ค่า infeed constant ของระบบ 9 บัส

รีเลย์	ค่า infeed constant			
	โซน 1	โซน 2	โซน 2, 3	โซน 3
1	0	0	0	4.4591
2	0	2.2946	2.2946	0
3	0	0	0	2.8833
4	0	0	0	1.9694
5	0	1.9694	1.9694	0
6	0	1.9147	1.9147	0
7	0	0	0	2.9034
8	0	2.9034	2.9034	0
9	0	0	0	2.2946
10	0	4.4591	4.4591	0
11	0	2.8833	2.8833	0
12	0	0	0	1.9147

ตารางที่ ข.8 ค่า infeed constant ของระบบ 14 บัส

รีเลย์	ค่า infeed constant			
	โซน 1	โซน 2	โซน 2, 3	โซน 3
1	0	7.8466	10.4640	0.5381
2	0	0.5381	0.5381	0.8826
3	0	0.8826	1.8437	1.0110
4	0	1.3391	1.1956	14.8469
5	0	0.9937	19.4009	0.5381
6	0	1.3030	0.7675	0
7	0	14.8469	14.8469	7.8466
8	0	0	0	0.3310
9	0	0	0	0

ตารางที่ ข.8 ค่า infeed constant ของระบบ 14 บัส

รีเลย์	ค่า infeed constant			
	โซน 1	โซน 2	โซน 2, 3	โซน 3
10	0	0	0	4.6204
11	0	4.6204	21.1261	0
12	0	0.3310	3.0214	0
13	0	0	0	88.0575
14	0	88.0575	88.0575	0
15	0	61.5467	61.5467	1.0182
16	0	1.0226	17.5102	0.6346
17	0	0.4442	0.4442	1.0226
18	0	2.1317	0.9586	1.011
19	0	1.0704	1.4311	14.8469
20	0	2.3278	25.0033	0
21	0	1.0182	1.5246	0.9937
22	0	6.3512	2.7036	0
23	0	139.7057	139.7057	0
24	0	0	0	139.7057
25	0	0.4442	0.4442	1.0226
26	0	2.1317	0.9586	1.011
27	0	1.0704	1.4311	14.8469
28	0	2.3278	25.0033	0
29	0	1.0182	1.5246	0.9937
30	0	6.3512	2.7036	0

มหาวิทยาลัยเทคโนโลยีสุรนารี

ตารางที่ ข.9 ค่า infeed constant ของระบบ 30 บัส

รีเลย์	ค่า infeed constant			
	โซน 1	โซน 2	โซน 2, 3	โซน 3
1	0	0.0023	0.0023	0.0008
2	0	0.0024	0.002	3.9797
3	0	0.0022	0.0025	0.0023
4	0	0.0252	0.0245	0.0023
5	0	0	0	0.0248
6	0	0.0001	0.0001	0
7	0	0.0565	0.0572	0.0023
8	0	3.9797	4.0271	0.0023
9	0	0.0005	0.0005	0
10	0	0.0006	0.0015	0.0007
11	0	0	0	0.0006
12	0	0	0	0.0006
13	0	0.0006	0.0006	0
14	0	0.0024	0.0051	0
15	0	0	0	0
16	0	0	0	0.0111
17	0	0.005	0.0049	0
18	0	0	0	0
19	0	0	0	0.012
20	0	0.0097	0.0099	0.0004
21	0	0.0004	0.0007	0.0007
22	0	0.0004	0.0005	0
23	0	0.0007	0.0019	0
24	0	0	0.001	0
25	0	0	0	0

มหาวิทยาลัยเทคโนโลยีสุรนารี

ตารางที่ ข.9 ค่า infeed constant ของระบบ 30 บัส (ต่อ)

รีเลย์	ค่า infeed constant			
	โซน 1	โซน 2	โซน 2, 3	โซน 3
26	0	0	0	0
27	0	0	0.001	0
28	0	0	0	0.0252
29	0	0	0	0.0005
30	0	0	0	0
31	0	0.0014	0.0014	0
32	0	0.0127	0.0171	0.0008
33	0	0	0	0.0014
34	0	0.0016	0.0016	0.0023
35	0	0.0008	0.0008	0.0023
36	0	0.962	0.9735	0.0023
37	0	0	0	0.0005
38	0	0	0	0.0001
39	0	0.0107	0.0147	0.0245
40	0	0.0189	0.0186	0
41	0	0.0111	0.012	0
42	0	0.0374	0.0368	0
43	0	0.0086	0.0095	0
44	0	0	0	0.0095
45	0	0.0018	0.0043	0
46	0	0	0	0.0086
47	0	0	0	0
48	0	0	0	0.0006
49	0	0	0	0
50	0	0	0	0

ตารางที่ ข.9 ค่า infeed constant ของระบบ 30 บัส (ต่อ)

รีเลย์	ค่า infeed constant			
	โซน 1	โซน 2	โซน 2, 3	โซน 3
51	0	0	0	0.005
52	0	0.0016	0.0024	0.0126
53	0	0	0	0.0006
54	0	0.0014	0.0026	0
55	0	0	0	0.0005
56	0	0	0	0
57	0	0.0005	0.0005	0
58	0	0.0003	0.0003	0.0252
59	0	0	0	0.0005
60	0	0	0	0
61	0	0	0	0.0016
62	0	0.0059	0.0059	0.0127
63	0	0.0351	0.0359	0
64	0	0.0131	0.0176	0.0008
65	0	0	0	0
66	0	0.0075	0.0081	0
67	0	0	0	0.0004
68	0	0.0018	0.0043	0

มหาวิทยาลัยเทคโนโลยีสุรนารี

ภาคผนวก ค

MATLAB CODE

มหาวิทยาลัยเทคโนโลยีสุรนารี

ในวิทยานิพนธ์นี้ได้ทำการวิเคราะห์การประสานสัมพันธ์แบบเหมาะที่สุดของรีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทางในระบบไฟฟ้ากำลัง ซึ่งจะนำเสนอโค้ดโปรแกรมโดยแบ่งเป็น 3 แบบ คือ แบบที่ 1 คือ การคำนวณค่าปรับตั้งรีเลย์ระยะทาง แบบที่ 2 การคำนวณค่าปรับตั้งของรีเลย์ป้องกันกระแสเกินและรีเลย์ระยะทาง และแบบที่ 2 การคำนวณค่าปรับตั้งของรีเลย์ป้องกันกระแสเกินโดยจะยกตัวอย่างโค้ดของโปรแกรมที่ใช้คำนวณค่าปรับตั้งในระบบทดสอบ 9 บัส

แบบที่ 1 โค้ดโปรแกรมการคำนวณค่าปรับตั้งรีเลย์ระยะทาง

1) infeed โดยจะแบ่งทั้งหมด 8 ฟังก์ชัน ฟังก์ชัน busdata linedata pointR Readgen Rundistance9bus Zone1 Zone2 และ Zone3 โดยฟังก์ชัน busdatas และ linedata เป็นฟังก์ชันแสดงข้อมูลของระบบทดสอบ 9 บัส ฟังก์ชัน pointR คือฟังก์ชันที่ทำหน้าที่บอกตำแหน่งที่ติดตั้งรีเลย์ ฟังก์ชัน Readgen เป็นฟังก์ชันที่ทำหน้าที่อ่านค่ากระแสลัดวงจรที่เครื่องกำเนิดไฟฟ้า ฟังก์ชัน Zone 1 Zone และ Zone3 ทำหน้าที่หาค่า infeed ของแต่ละโซน ส่วนฟังก์ชันสุดท้ายคือ ฟังก์ชัน Rundistance9bus ใช้ในการควบคุมตัวแปรที่ต้องการควบคุม เมื่อได้ค่า infeed ของแต่ละโซนแล้ว หลังจากนั้นจะนำค่า infeed ที่ได้จากโปรแกรมนั้นไปคำนวณต่อในโค้ดถัดไป เป็นการคำนวณค่าปรับตั้งรีเลย์ระยะทาง

2) การหาค่าปรับตั้งรีเลย์ระยะทาง มีทั้งหมด 6 ฟังก์ชัน ประกอบไปด้วย linedata pointR rundistane9bus Zone1 Zone2 และ Zone3 ซึ่งฟังก์ชัน linedata pointR ทำหน้าที่เหมือนดังที่กล่าวมาข้างต้น ส่วนฟังก์ชัน Zone1 Zone2 และ Zone3 นั้นใช้ทำหน้าที่ในการคำนวณหาค่าปรับตั้งรีเลย์ระยะทาง

มหาวิทยาลัยเทคโนโลยีสุรนารี

1) ฟังก์ชันที่ใช้ในการรันหาค่า infeed

ฟังก์ชัน busdata

```
function bus9 = busdatas()
bus9 = [
1   1   1.04    0   1   27.95   0   0
2   3   1.025   9.3507  163 4.903   0   0
3   3   1.025   5.142   85  -11.449 0   0
4   3   1.02531 -2.2174 0   0   0   0
5   3   0.99972 -3.6802 0   0   125 50
6   3   1.01225 -3.5666 0   0   90  30
7   3   1.02683 3.7961  0   0   0   0
8   3   1.01727     1.3373  0   0   100 35
9   3   1.03269     2.4448  0   0   0   0
];
end
```

ฟังก์ชัน linedata

```
function linedt = linedatas(num)
 line9 = [5   4   0.01    0.068
          6   4   0.017   0.092
          7   5   0.032   0.161
          9   6   0.039   0.1738
          7   8   0.0085  0.0576
          8   9   0.0119  0.1008
          4   1   0       0.0576
          2   7   0       0.0625
          9   3   0       0.0586];
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
% %     | From | To  |  R    |  X    |
Ex6_4   = [     1     2      16       48
                2     3      24       72
                2     4      30       90
                2     5      12       36
                2     6      36       108
                2     8      26       78
                3     9      0        25
                6     7      0        25
                                        ];
Ex6_3   = [
                1     2      0.2      0.8
                2     3      0.4      1.6
                2     5      0.3      1.2
                2     8      0.5      1.8
                2     10     0.2      0.8
                3     4      0.4      1.6
                3     6      0.6      2.5
                6     7      0.3      1.2
                8     9      0.2      0.8
                10    11     0.4      1.6
                11    12     0.3      1.2
                                          ];

Ex6_2   = [     1     2      1      5
                2     3      4      16
                3     4      2.5    10
                                     ];
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
if num==2
   linedt = Ex6_2;
elseif num==3
   linedt = Ex6_3;
elseif num==4
   linedt = Ex6_4;
elseif num==14
   linedt =line14;
   elseif num==9
   linedt =line9;
end
end
```

ฟังก์ชัน pointR

```
function r = pointR(num)
 %      | From | To  | relays | naer | far |
 bus9 = [
         4  5  4  4  5
         4  6  12 4  6
         5  7  5  5  7
         6  9  6  6  9
         7  8  7  7  8
         8  9  8  8  9
         4  5  11 5  4
         4  6  10 6  4
         5  7  3  7  5
         6  9  1  9  6
         7  8  2  8  7
         8  9  9  9  8
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
                    ];
Ex6_4  = [
        1    2      1      1      2
        2    3      4      3      2
        2    4      0      0      0
        2    5      2      5      2
        2    6      5      2      6
        2    8      3      8      2
        3    9      0      0      0
        6    7      0      0      0
                                  ];

Ex6_3  = [      1    2      1      1      2
                2    3      2      2      3
                2    5      0      0      0
                2    8      0      0      0
                2    10     3      2      10
                3    4      0      0      0
                3    6      0      0      0
                6    7      0      0      0
                8    9      0      0      0
                10   11     0      0      0
                11   12     0      0      0
                                          ];
    Ex6_2  = [1    2      1      1      2
                2    3      0      0      0
                3    4      0      0      0      ];
```

```
if num==2
   r = Ex6_2;
elseif num==3
   r = Ex6_3;
elseif num==4
   r = Ex6_4;
elseif num==9
   r = bus9;%74002
end
end
```

ฟังก์ชัน readGen

```
function [a b] = readGen(nb,s)
 for i=1:nb
   fid = fopen(s{i}, 'r');
   grades = textscan(fid', '%d %s %s %f %s %s %s %s %s %s %s %s', 'headerlines',3)';
   fclose(fid);
   c1=grades{1};
   c4=grades{4};
   for j=1:length(c1)
      a(i,j) = c1(j);
      b(i,j) = c4(j);
   end
end
 end
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

ฟังก์ชัน zone1

```
function [Zone1] = zone1(numR,num)
 relays = pointR(num);
relayn = relays(:,3);              %Number of Relays
i = 1;
k = 0;
while k == 0
   if relayn(i) == numR
       k=1;
       relaybf = relays(i,1);
       relaybt = relays(i,2);
       relaycl = relays(i,4);
       relayfar = relays(i,5);
   end
   i = i+1;
end
Zone1.busfrom      = relaybf;
Zone1.busto        = relaybt;
Zone1.installrelay = relaycl;
Zone1.uninstallrelay = relayfar;
end
```

ฟังก์ชัน Zone2

```
function [Zone2,If] = zone2(Zone1,num,aa,bb,cc,dd,ee)
 relaycl  = Zone1.installrelay;
relayfar = Zone1.uninstallrelay;
linedt   = linedatas(num);
n = length(linedt(:,1));
cc(1,:);
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
nf = length(cc(1,:));
k = 1;


If = [];
 x = 0;
rc = 0;
rf = 0;
zone2 =0;
for i=1:n
   if linedt(i,1)==relayfar && linedt(i,2)~=relaycl && linedt(i,3)~=0
      a1 = linedt(i,3);
      a2 = linedt(i,4)*1j;
      a = a1 + a2;
      x(k)=abs(a);
      rc(k)= relayfar;
      rf(k)=linedt(i,2);
      zone2(k)=a;
      k = k+1;
   end
   if linedt(i,2)==relayfar && linedt(i,1)~=relaycl && linedt(i,3)~=0
      a1 = linedt(i,3);
      a2 = linedt(i,4)*1j;
      a  = a1 + a2;
      x(k)=abs(a);
      rc(k)= relayfar;
      rf(k)=linedt(i,1);
      zone2(k)=a;
      k = k+1;
   end
end
```

```
line2M   = [x',rc',rf',zone2'];
[line2M,P]   = sortrows(line2M);
 nl      = length(line2M(:,1));
linemin  = [line2M(1,2),line2M(1,3),line2M(1,4)'];
linemax  = [line2M(nl,2),line2M(nl,3),line2M(nl,4)'];
 x_ = 0;
rc_ = 0;
rf_ = 0;
zone2_=0;
linein1 = 0;
l=1;
for i=1:n
%    [linedt(i,1) linedt(i,2) relayfar relaycl linemin(2)]
   if linedt(i,1)==relayfar && linedt(i,2)~=relaycl ...
         &&(linedt(i,1)~=linemin(2)&&linedt(i,2)~=linemin(2))
         %           &&linedt(i,1)~=linemin(2)
      a1_ = linedt(i,3);
      a2_ = linedt(i,4)*1j;
      a_ = a1_ + a2_;
      x_(l)=abs(a_);
      rc_(l)= relayfar;
      rf_(l)=linedt(i,2);
      zone2_(l)=a_;
      l = l+1;
%       disp('A')
   end
      if linedt(i,2)==relayfar && linedt(i,1)~=relaycl ...
         &&(linedt(i,1)~=linemin(2)&&linedt(i,2)~=linemin(2))
%              &&linedt(i,1)~=linemin(2)
```

```
        a1_ = linedt(i,3);
        a2_ = linedt(i,4)*1j;
        a_  = a1_ + a2_;
        x_(l)=abs(a_);
        rc_(l)= relayfar;
        rf_(l)=linedt(i,1);
        zone2_(l)=a_;
        l = l+1;
%         disp('B')
    end
end
linein1 = [x_',rc_',rf_',zone2_'];
nr      = length(linein1(:,1));
Ir = 0;
relayfar;
p   = relayfar-3;
for i=1:nf
    if linemin(1)==cc(p,i)&&linemin(2)==dd(p,i)
        Ir = ee(p,i);
    end
    if linemin(1)==dd(p,i)&&linemin(2)==cc(p,i)
        Ir = ee(p,i);
    end
end
 k=1;
if nr>=1 && linein1(1,1)~=0
    LIF = linein1;
    for i=1:length(linein1(:,1))
        for jj=1:nf
            if LIF(i,2)==cc(p,jj)&& LIF(i,3)==dd(p,jj)
```

```
            If(k)= ee(p,jj);
            k=k+1;
        end
        if LIF(i,2)==dd(p,jj)&&LIF(i,3)==cc(p,jj)
            If(k)= ee(p,jj);
            k=k+1;
        end
    end
  end
end

for i=1:length(aa(1,:))
    if p+3==aa(p,i)
        If(k) = bb(p,i);
        k=k+1;

    end
end
% k

Ifeed1 = (sum(If)/Ir);

Zone2.busfmin      = linemin(1);
Zone2.bustmin      = linemin(2);
Zone2.busfmax      = linemax(1);
Zone2.bustmax      = linemax(2);
Zone2.lineInfeed   = If;
Zone2.Irelay       = Ir;
Zone2.ind1         = Ifeed1;
Zone2.z1from       = relaycl;
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
Zone2.z1to          = relayfar;
end
```

ฟังก์ชัน Zone3

```
function [Zone3] = zone3(Zone2,num,aa,bb,cc,dd,ee)
linedt    = linedatas(num);
bus2maxcl = Zone2.busfmax;
bus2maxfar= Zone2.bustmax;
bus2mincl = Zone2.busfmin;
bus2minfar= Zone2.bustmin;
z1from    = Zone2.z1from;
z1to      = Zone2.z1to;
 % .....Infeed Zone2.....
nf = length(cc(1,:));
n = length(linedt(:,1));
If = [];
x_ = 0;
rc_ = 0;
rf_ = 0;
zone2_=0;
linein2 = 0;
l=1;
for i=1:n
    [ z1from z1to linedt(i,1) linedt(i,2) bus2maxcl bus2maxfar ];
   if linedt(i,1)==z1to && linedt(i,2)~=z1from ...
          &&(linedt(i,1)~=bus2maxfar&&linedt(i,2)~=bus2maxfar)
%            &&linedt(i,2)~=bus2maxfar
       a1_ = linedt(i,3);
       a2_ = linedt(i,4)*1j;
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
      a_ = a1_ + a2_;
      x_(l)=abs(a_);
      rc_(l)= bus2maxcl;
      rf_(l)=linedt(i,2);
      zone2_(l)=a_;
      l = l+1;
%       disp('AA')
   end
   if linedt(i,2)==z1to && linedt(i,1)~=z1from ...
       &&(linedt(i,1)~=bus2maxfar&&linedt(i,2)~=bus2maxfar)
%    &&linedt(i,1)~=bus2maxfar
      a1_ = linedt(i,3);
      a2_ = linedt(i,4)*1j;
      a_  = a1_ + a2_;
      x_(l)=abs(a_);
      rc_(l)= bus2maxcl;
      rf_(l)=linedt(i,1);
      zone2_(l)=a_;
      l = l+1;
%       disp('BB')
   end
end
linein2 = [x_',rc_',rf_',zone2_'];
nr = length(linein2(:,1));
Ir = [];
p   = bus2maxcl-3;
for i=1:nf
   if bus2maxcl==cc(p,i)&&bus2maxfar==dd(p,i)
      Ir = ee(p,i);
   end
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
   if bus2maxcl==dd(p,i)&&bus2maxfar==cc(p,i)
      Ir = ee(p,i);
   end
end
 k=1;
if nr>=1
   LIF = linein2;
   for i=1:length(linein2(:,1))
      for jj=1:nf
         if LIF(i,2)==cc(p,jj)&& LIF(i,3)==dd(p,jj)
            If(k)= ee(p,jj);
            k=k+1;
         end
         if LIF(i,2)==dd(p,jj)&&LIF(i,3)==cc(p,jj)
            If(k)= ee(p,jj);
            k=k+1;
         end
      end
   end
end
% If
for i=1:length(aa(1,:))
      if p+3==aa(p,i)
         If(k) = bb(p,i);
         k=k+1;

      end
end
% If2=If
% Ir2=Ir
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
Ifeed2 = (sum(If)/Ir);
%.............Z line Zone3.............................
n = length(linedt(:,1));
k = 1;
L=0;
x=0;
rc=0;
rf=0;
zone3=0;

for i=1:n
   if linedt(i,1)==bus2maxfar && linedt(i,2)~=bus2maxcl&&linedt(i,3)~=0
      a = linedt(i,3)+linedt(i,4)*1j;
      x(k)=abs(a);
      rc(k)= bus2maxfar;
      rf(k)=linedt(i,2);
      zone3(k)=a;
      k = k+1;
      L=1;
   end
   if linedt(i,2)==bus2maxfar && linedt(i,1)~=bus2maxcl&&linedt(i,3)~=0
      a = linedt(i,3)+linedt(i,4)*1j;
      x(k)=abs(a);
      rc(k)= bus2maxfar;
      rf(k)=linedt(i,1);
      zone3(k)=a;
      k = k+1;
      L=1;
   end
end
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
line3     = [x',rc',rf',zone3'];
line3     = sortrows(line3);
linemin   = [line3(1,2) line3(1,3) line3(1,4)'];
if L==0
   relayfar3  = 0;

end
% .............Infeed Zone3.....................
Ir = 0;
If = 0;
x_ = 0;
rc_ = 0;
rf_ = 0;
zone3_=0;
linein3 = 0;
l=1;
for i=1:n
%    [bus2maxfar bus2maxcl linedt(i,1) linedt(i,2) bus2mincl bus2minfar ]
   if linedt(i,1)==bus2maxfar && linedt(i,2)~=bus2maxcl ...
          &&(linedt(i,1)~=linemin(2)&&linedt(i,2)~=linemin(2))
      a1_ = linedt(i,3);
      a2_ = linedt(i,4)*1j;
      a_ = a1_ + a2_;
      x_(l)=abs(a_);
      rc_(l)= linemin(1);
      rf_(l)=linedt(i,2);
      zone3_(l)=a_;
      l = l+1;
%       disp('AA')
```

```
    end
        if linedt(i,2)==bus2maxfar && linedt(i,1)~=bus2maxcl ...
            &&(linedt(i,1)~=linemin(2)&&linedt(i,2)~=linemin(2))
        a1_ = linedt(i,3);
        a2_ = linedt(i,4)*1j;
        a_  = a1_ + a2_;
        x_(l)=abs(a_);
        rc_(l)= linemin(1);
        rf_(l)=linedt(i,1);
        zone3_(l)=a_;
        l = l+1;
%         disp('BB')
    end
end
linein3 = [x_',rc_',rf_',zone3_'];
nr = length(linein3(:,1));
Ir = [];
p   = linemin(1)-3;
for i=1:nf
    if linemin(1)==cc(p,i)&&linemin(2)==dd(p,i)
        Ir = ee(p,i);
    end
    if linemin(1)==dd(p,i)&&linemin(2)==cc(p,i)
        Ir = ee(p,i);
    end
end
 k=1;
if nr>=1
    LIF = linein3;
    for i=1:length(linein3(:,1))
```

```
      for jj=1:nf
         if LIF(i,2)==cc(p,jj)&& LIF(i,3)==dd(p,jj)
            If(k)= ee(p,jj);
            k=k+1;
         end
         if LIF(i,2)==dd(p,jj)&&LIF(i,3)==cc(p,jj)
            If(k)= ee(p,jj);
            k=k+1;
         end
      end
   end
end
% If3=If
% p
for i=1:length(aa(1,:))
      if p+3==aa(p,i)
         If(k) = bb(p,i);
         k=k+1;
%       disp('^_^')
      end
end
% If3=If
% Ir3=Ir
Ifeed3 = (sum(If)/Ir);
% .....................................................................

Zone3.line2from  = bus2maxcl;
Zone3.line2to    = bus2maxfar;
Zone3.busfrom    = bus2maxfar;
Zone3.busto      = line3(1,3);
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
Zone3.ind2      = Ifeed2;
Zone3.ind3      = Ifeed3;
 end
```

ฟังก์ชันที่ใช้ในการรันหาค่า infeed

```
%..................... Find Relay Setting.....................
 clc ;clear all;
%format long;
num = 9;
s ={'g4.txt'...   % File name Data Fault 14 bus
    'g5.txt'...
    'g6.txt'...
    'g7.txt'...
    'g8.txt'...
    'g9.txt'
    };
sl ={'4.txt'...   % File name Data Fault 14 bus
    '5.txt'...
    '6.txt'...
    '7.txt'...
    '8.txt'...
    '9.txt'
    };
[a b]= readGen(6,s);
[c d e] = readLine(6,sl);
numrelay = [1 2 3 4 5 6 7 8 9 10,...
          11 12
         ];                          % Number of Relay
N=length(numrelay);
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
for i=1:N
   disp('.....')
   disp(i)
   NR = numrelay (i);
   %*********Zone 1*********



   [z1] = zone1(NR,num);
   from1(i)  = z1.installrelay;
   to1(i)    = z1.uninstallrelay;
   %*********Zone 2**********
   [z2] = zone2(z1,num,a,b,c,d,e);
   from2(i)  = z2.busfmin;


   to2(i)    = z2.bustmin;
  infeed1(i) = z2.ind1;
   %*********Zone 3**********
   [z3] = zone3(z2,num,a,b,c,d,e);
   from23(i) = z3.line2from;
   to23(i)   = z3.line2to;
   from3(i)  = z3.busfrom;
   to3(i)    = z3.busto;
   infeed2(i) = z3.ind2;
   infeed3(i) = z3.ind3;
end
infeed0 = [0 0 0 0 0 0 0 0 0 0 ...
          0 0 ];


%       infeed1'
%       infeed2'
```

```
%       infeed3'
      infeed = [numrelay' infeed0' infeed1' infeed2' infeed3']
fprintf('.....\n');
fprintf('\n\n..... Line Protection .....\n\n');
fprintf('.....\n');

fprintf('| Relays|     Zone1    |     Zone2     |     Zone3      |\n');
fprintf('.....\n');
for i=1:N
   fprintf('|   %.0f   |%6.0f - %-6.0f|%4.0f - %2.0f - %-5.0f|%3.0f - %2.0f - %2.0f - %-3.0f|\n',...
      numrelay(i),from1(i),to1(i),from1(i),from2(i),to2(i),from1(i),from23(i),from3(i),to3(i));

end
fprintf('.....\n');
disp('^_^    *_*')
%format short;
```

2) ฟังก์ชันที่ใช้ในการหาค่าปรับตั้งรีเลย์ระยะทาง

ฟังก์ชัน linedatas และ PointR เป็นฟังก์ชันเดียวกับดังที่กล่าวมาข้างต้น

ฟังก์ชัน Zone 1

```
function [Zone1] = zone1(numR,CV,num)
relays = pointR(num);
relayn = relays(:,3);              %Number of Relays
i = 1;
k = 0;
while k == 0
   if relayn(i) == numR
      k=1;
```

```
        relaybf = relays(i,1);
        relaybt = relays(i,2);
        relaycl = relays(i,4);
        relayfar = relays(i,5);
    end
    i = i+1;
end
linedt = linedatas(num);
n = length(linedt(:,1));
for l=1:n
    for m=1:n
        if linedt(l,1)==relaybf && linedt(m,2)==relaybt && l==m
            zr = linedt(l,3);
            zx = linedt(l,4);
            rows = l;
        end
        if linedt(l,2)==relaybf && linedt(m,1)==relaybt && l==m
            zr = linedt(l,3);
            zx = linedt(l,4);
            rows = l;
        end
    end
end
zline = zr + zx*1i;
zsetting = 0.8*(zline)*CV;
zl = abs(zsetting);
zx = (180/pi)*angle(zsetting);
z1 = [zl zx];

Zone1.CV          = CV;
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
Zone1.numrelay     = numR;
Zone1.busfrom      = relaybf;
Zone1.busto        = relaybt;
Zone1.installrelay = relaycl;
Zone1.uninstallrelay = relayfar;
Zone1.row          = rows;
Zone1.zline        = zline;
Zone1.settingrec   = zsetting;
Zone1.settingpol   = z1;
Zone1.zone1full    = zline;
end
```

ฟังก์ชัน Zone2

```
function [Zone2] = zone2(Zone1,CV,K,num)
 zline1   = Zone1.zline;
relaycl  = Zone1.installrelay;
relayfar = Zone1.uninstallrelay;
linedt   = linedatas(num);
n = length(linedt(:,1));
k = 1;
for i=1:n
   if linedt(i,1)==relayfar && linedt(i,2)~=relaycl && linedt(i,3)~=0
      a = linedt(i,3)+linedt(i,4)*1j;
      x(k)=abs(a);
      rc(k)= relayfar;
      rf(k)=linedt(i,2);
      zone2(k)=a;
      k = k+1;
   end
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
   if linedt(i,2)==relayfar && linedt(i,1)~=relaycl && linedt(i,3)~=0
      a = linedt(i,3)+linedt(i,4)*1j;
      x(k)=abs(a);
      rc(k)= relayfar;
      rf(k)=linedt(i,1);
      zone2(k)=a;
      k = k+1;
   end
end

line2M   = [x',rc',rf',zone2'];
line2M = sortrows(line2M);
nl = length(line2M(:,1));
linemin = [line2M(1,2),line2M(1,3),line2M(1,4)'];
linemax = [line2M(nl,2),line2M(nl,3),line2M(nl,4)'];
K2 = K(2);

zsetting= (zline1+0.5*(1+K2)*linemin(3))*CV;

zl = abs(zsetting);
zx = (180/pi)*angle(zsetting);
z2 = [zl zx ];
Zone2.busfrom       = linemin(1);
Zone2.busto         = linemin(2);
Zone2.zlinemin      = linemin(3);
Zone2.busfmax       = linemax(1);
Zone2.bustmax       = linemax(2);
Zone2.zlinemax      = linemax(3);
Zone2.settingrec    = zsetting;
Zone2.settingpol    = z2;
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
Zone2.zone2full      = linemin(3);
Zone2.zone1full      = zline1;
Zone2.line2M         = line2M;
end
```

ฟังก์ชัน Zone 3

```
function [Zone3] = zone3(Zone2,CV,K,num)
 linedt    = linedatas(num);
zline1     = Zone2.zone1full;
bus2maxcl = Zone2.busfmax;
bus2maxfar= Zone2.bustmax;
zline2  = Zone2.zlinemax;
n = length(linedt(:,1));
 k = 1;
 L=0;
 x=0;
 rc=0;
 rf=0;
 zone3=0;

 for i=1:n
    if linedt(i,1)==bus2maxfar && linedt(i,2)~=bus2maxcl&&linedt(i,3)~=0
        a = linedt(i,3)+linedt(i,4)*1j;
        x(k)=abs(a);
        rc(k)= bus2maxfar;
        rf(k)=linedt(i,2);
        zone3(k)=a;
        k = k+1;
        L=1;
```

```
   end
   if linedt(i,2)==bus2maxfar && linedt(i,1)~=bus2maxcl&&linedt(i,3)~=0
      a = linedt(i,3)+linedt(i,4)*1j;
      x(k)=abs(a);
      rc(k)= bus2maxfar;
      rf(k)=linedt(i,1);
      zone3(k)=a;
      k = k+1;
      L=1;
   end
end

line3    = [x',rc',rf',zone3'];
line3    = sortrows(line3);

linemin  = [line3(1,2) line3(1,3) line3(1,4)'];
%  relaycl3  = linemin(1);
relayfar3 = linemin(2);
zline3   = linemin(3);
K2 = K(3);
K3 = K(4);
zsetting= (zline1+((1+K2)*zline2)+(0.25*(1+K3))*zline3)*CV;
zl = abs(zsetting);
zx = (180/pi)*angle(zsetting);
z3 = [zl zx];
if L==0
%    relaycl3   = 0;
   relayfar3  = 0;
   zsetting   = 0 + 0*1i;
   zl = abs(zsetting);
```

```
   zx = (180/pi)*angle(zsetting);
   z3 = [zl zx];
end

Zone3.line2from  = bus2maxcl;
Zone3.line2to    = bus2maxfar;
Zone3.busfrom    = bus2maxfar;
Zone3.busto      = relayfar3;
Zone3.settingrec = zsetting;
Zone3.settingpol = z3;
end
```

ฟังก์ชัน run

```
%..... Find Relay Setting.....
 clc ;clear all;
%global LL
num = 9;                                    % Ex. 6.4
% numrelay = [1 2 3 4 5];                   % Number of Relay
% CTR = [100/5 100/5 100/5 100/5  100/5];   % Current Transformer Relay
% VTR = [230/120  230/120  230/120  230/120  230/120]*1000;% Voltage Transformer Relay
numrelay =        [1 2 3 4 5 6 7 8 9 10,...
                   11 12
                                ];          % Number of Relay
CTR = [100/5 100/5 100/5 100/5 100/5 100/5 100/5 100/5 100/5 100/5,...
          100/5 100/5
           ];              % Current Transformer Relay
VTR = [230/120 230/120 230/120 230/120 230/120 230/120 230/120 230/120 230/120
          230/120,...
          230/120 230/120
```

```
            ];% Voltage Transformer Relay
CV  = CTR./VTR;
%     ..........................................................
%     | Zone1 |     Zone2       |              Zone3                |
KT = [
            0   0       0       4.4591
           0   2.2946  2.2946  0
           0   0       0       2.8833
           0   0       0       1.9694
           0   1.9694  1.9694  0
           0   1.9147  1.9147  0
           0   0       0       2.9034
           0   2.9034  2.9034  0
           0   0       0       2.2946
           0   4.4591  4.4591  0
           0   2.8833  2.8833  0
           0   0       0       1.9147
                                    ];
% KT = zeros(12,5);
N=length(numrelay);
for i=1:N
   K = KT(i,:);

   NR = numrelay (i);
   %*********Zone 1**********
   [z1] = zone1(NR,CV(i),num);
   Zone1x(i) = z1.settingpol(1);
   Zone1l(i) = z1.settingpol(2);
   from1(i)  = z1.installrelay;
   to1(i)    = z1.uninstallrelay;
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
   %*********Zone 2**********
   [z2] = zone2(z1,CV(i),K,num);
   Zone2x(i) = z2.settingpol(1);
   Zone2l(i) = z2.settingpol(2);
   from2(i)  = z2.busfrom;
   to2(i)    = z2.busto;
   %*********Zone 3**********
   [z3] = zone3(z2,CV(i),K,num);
   Zone3x(i) = z3.settingpol(1);
   Zone3l(i) = z3.settingpol(2);
   from23(i) = z3.line2from;
   to23(i)   = z3.line2to;
   from3(i)  = z3.busfrom;
   to3(i)    = z3.busto;
end

fprintf('.................Setting of Distanc Relays ..................\n\n');
fprintf('.....................................................\n');
fprintf('| Relays|      Zone1     |      Zone2     |      Zone3     |\n');
fprintf('|       |  Impedanc (pu) |  Impedanc (pu) |  Impedanc (pu) |\n');
fprintf('.....................................................\n');

for i=1:N
   fprintf('|   %-4.0f| %7.4f %-6.2f  | %7.4f %-6.2f  | %7.4f %6.2f
|\n',numrelay(i),Zone1x(i),Zone1l(i),Zone2x(i),Zone2l(i),Zone3x(i),Zone3l(i));
end
fprintf('.....................................................\n');

fprintf('.....................................................\n');
fprintf('\n\n..................... Line Protection .....................\n\n');
```

```
fprintf('.....\n');

fprintf('| Relays|    Zone1   |    Zone2    |    Zone3     |\n');
fprintf('.....\n');
for i=1:N
   fprintf('|   %.0f   |%6.0f - %-6.0f|%4.0f - %2.0f - %-5.0f|%3.0f - %2.0f - %2.0f - %-
3.0f|\n',numrelay(i),from1(i),to1(i),from1(i),from2(i),to2(i),from1(i),from23(i),from3(i),to3(i));

end
fprintf('.....\n');
```

แบบที่ 2 การคำนวณค่าปรับตั้งของรีเลย์ระยะทางและรีเลย์ป้องกันกระแสเกิน มีฟังก์ชันที่ใช้ในการรันทั้งหมด 5 ฟังก์ชันซึ่งประกอบไปด้วย ฟังก์ชัน deopt , fun , c_I, data_I และฟังก์ชัน run ซึ่งฟังก์ชัน deopt คือฟังก์ชันของวิธีการค้นหาคำตอบด้วยวิธีวิวัฒนาการผลต่าง ฟังก์ชัน fun คือฟังก์ชันวัตถุประสงค์ ฟังก์ชัน c_I เป็นฟังก์ชันของแท็ปหม้อแปลงใช้สำหรับปรับคูณค่ากระแส ฟังก์ชัน data_I คือฟังก์ชันรับค่ากระแสโหลดและกระแสลัดวงจร และฟังก์ชันสุดท้ายคือฟังก์ชัน run เอาไว้ควบคุมตัวแปรที่สนใจควบคุม

ฟังก์ชัน deopt

```
%-----This is just for notational convenience and to keep the code uncluttered.--------
I_NP        = S_struct.I_NP;
F_weight    = S_struct.F_weight;
F_CR        = S_struct.F_CR;
I_D         = S_struct.I_D;
FVr_minbound = S_struct.FVr_minbound;
FVr_maxbound = S_struct.FVr_maxbound;
I_bnd_constr = S_struct.I_bnd_constr;
I_itermax   = S_struct.I_itermax;
```

```
F_VTR       = S_struct.F_VTR;
I_strategy  = S_struct.I_strategy;
I_refresh   = S_struct.I_refresh;
I_plotting  = S_struct.I_plotting;
%-----Check input variables---------------------------------------------
if (I_NP < 5)
  I_NP=5;
  fprintf(1,' I_NP increased to minimal value 5\n');
end
if ((F_CR < 0) | (F_CR > 1))
  F_CR=0.5;
  fprintf(1,'F_CR should be from interval [0,1]; set to default value 0.5\n');
end
if (I_itermax <= 0)
  I_itermax = 200;
  fprintf(1,'I_itermax should be > 0; set to default value 200\n');
end
I_refresh = floor(I_refresh);
%-----Initialize population and some arrays-------------------------------
FM_pop = zeros(I_NP,I_D); %initialize FM_pop to gain speed
%----FM_pop is a matrix of size I_NPx(I_D+1). It will be initialized------
%----with random values between the min and max values of the-------------
%----parameters-----------------------------------------------------------
for k=1:I_NP
  FM_pop(k,:) = FVr_minbound + rand(1,I_D).*(FVr_maxbound - FVr_minbound);
end

FM_popold    = zeros(size(FM_pop));  % toggle population
FVr_bestmem  = zeros(1,I_D);         % best population member ever
FVr_bestmemit = zeros(1,I_D);        % best population member in iteration
```

```
I_nfeval      = 0;                % number of function evaluations

%------Evaluate the best member after initialization----------------------
I_best_index   = 1;               % start with first population member
S_val(1)       = feval(fname,FM_pop(I_best_index,:),S_struct);

S_bestval = S_val(1);             % best objective function value so far
I_nfeval  = I_nfeval + 1;

for k=2:I_NP                      % check the remaining members
 S_val(k)  = feval(fname,FM_pop(k,:),S_struct);
 I_nfeval  = I_nfeval + 1;
 if (left_win(S_val(k),S_bestval) == 1)
    I_best_index   = k;           % save its location
    S_bestval      = S_val(k);
 end
end
FVr_bestmemit = FM_pop(I_best_index,:); % best member of current iteration
S_bestvalit   = S_bestval;        % best value of current iteration

FVr_bestmem = FVr_bestmemit;      % best member ever
%------DE-Minimization-----------------------------------------------
%------FM_popold is the population which has to compete. It is--------
%------static through one iteration. FM_pop is the newly--------------
%------emerging population.-----------------------------------------
FM_pm1   = zeros(I_NP,I_D);   % initialize population matrix 1
FM_pm2   = zeros(I_NP,I_D);   % initialize population matrix 2
FM_pm3   = zeros(I_NP,I_D);   % initialize population matrix 3
FM_pm4   = zeros(I_NP,I_D);   % initialize population matrix 4
FM_pm5   = zeros(I_NP,I_D);   % initialize population matrix 5
```

```
FM_bm    = zeros(I_NP,I_D);   % initialize FVr_bestmember  matrix
FM_ui    = zeros(I_NP,I_D);   % intermediate population of perturbed vectors
FM_mui   = zeros(I_NP,I_D);   % mask for intermediate population
FM_mpo   = zeros(I_NP,I_D);   % mask for old population
FVr_rot  = (0:1:I_NP-1);              % rotating index array (size I_NP)
FVr_rotd = (0:1:I_D-1);       % rotating index array (size I_D)
FVr_rt   = zeros(I_NP);               % another rotating index array
FVr_rtd  = zeros(I_D);                % rotating index array for exponential crossover
FVr_a1   = zeros(I_NP);               % index array
FVr_a2   = zeros(I_NP);               % index array
FVr_a3   = zeros(I_NP);               % index array
FVr_a4   = zeros(I_NP);               % index array
FVr_a5   = zeros(I_NP);               % index array
FVr_ind  = zeros(4);

FM_meanv = ones(I_NP,I_D);

I_iter = 1;
while ((I_iter < I_itermax) & (S_bestval.FVr_oa(1) > F_VTR))
 FM_popold = FM_pop;                  % save the old population
 S_struct.FM_pop = FM_pop;
 S_struct.FVr_bestmem = FVr_bestmem;

 FVr_ind = randperm(4);               % index pointer array

 FVr_a1  = randperm(I_NP);                % shuffle locations of vectors
 FVr_rt  = rem(FVr_rot+FVr_ind(1),I_NP);     % rotate indices by ind(1) positions
 FVr_a2  = FVr_a1(FVr_rt+1);              % rotate vector locations
 FVr_rt  = rem(FVr_rot+FVr_ind(2),I_NP);
 FVr_a3  = FVr_a2(FVr_rt+1);
```

```
FVr_rt  = rem(FVr_rot+FVr_ind(3),I_NP);
FVr_a4  = FVr_a3(FVr_rt+1);
FVr_rt  = rem(FVr_rot+FVr_ind(4),I_NP);
FVr_a5  = FVr_a4(FVr_rt+1);
FM_pm1 = FM_popold(FVr_a1,:);             % shuffled population 1
FM_pm2 = FM_popold(FVr_a2,:);             % shuffled population 2
FM_pm3 = FM_popold(FVr_a3,:);             % shuffled population 3
FM_pm4 = FM_popold(FVr_a4,:);             % shuffled population 4
FM_pm5 = FM_popold(FVr_a5,:);             % shuffled population 5

for k=1:I_NP                        % population filled with the best member
  FM_bm(k,:) = FVr_bestmemit;             % of the last iteration
end
FM_mui = rand(I_NP,I_D) < F_CR;  % all random numbers < F_CR are 1, 0 otherwise
FM_mpo = FM_mui < 0.5;    % inverse mask to FM_mui
if (I_strategy == 1)                    % DE/rand/1
  FM_ui = FM_pm3 + F_weight*(FM_pm1 - FM_pm2);   % differential variation
  FM_ui = FM_popold.*FM_mpo + FM_ui.*FM_mui;     % crossover
  FM_origin = FM_pm3;
elseif (I_strategy == 2)                % DE/local-to-best/1
  FM_ui = FM_popold + F_weight*(FM_bm-FM_popold) + F_weight*(FM_pm1 - FM_pm2);
  FM_ui = FM_popold.*FM_mpo + FM_ui.*FM_mui;
  FM_origin = FM_popold;
elseif (I_strategy == 3)                % DE/best/1 with jitter
  FM_ui = FM_bm + (FM_pm1 - FM_pm2).*((1-0.9999)*rand(I_NP,I_D)+F_weight);
  FM_ui = FM_popold.*FM_mpo + FM_ui.*FM_mui;
  FM_origin = FM_bm;
elseif (I_strategy == 4)                % DE/rand/1 with per-vector-dither
   f1 = ((1-F_weight)*rand(I_NP,1)+F_weight);
   for k=1:I_D
```

```
    FM_pm5(:,k)=f1;
  end
  FM_ui = FM_pm3 + (FM_pm1 - FM_pm2).*FM_pm5;    % differential variation
  FM_origin = FM_pm3;
  FM_ui = FM_popold.*FM_mpo + FM_ui.*FM_mui;     % crossover
elseif (I_strategy == 5)                         % DE/rand/1 with per-vector-dither
  f1 = ((1-F_weight)*rand+F_weight);
  FM_ui = FM_pm3 + (FM_pm1 - FM_pm2)*f1;         % differential variation
  FM_origin = FM_pm3;
  FM_ui = FM_popold.*FM_mpo + FM_ui.*FM_mui;     % crossover
else                                             % either-or-algorithm
  if (rand < 0.5);                               % Pmu = 0.5
    FM_ui = FM_pm3 + F_weight*(FM_pm1 - FM_pm2);% differential variation
    FM_origin = FM_pm3;
  else                                           % use F-K-Rule: K = 0.5(F+1)
    FM_ui = FM_pm3 + 0.5*(F_weight+1.0)*(FM_pm1 + FM_pm2 - 2*FM_pm3);
  end
  FM_ui = FM_popold.*FM_mpo + FM_ui.*FM_mui;     % crossover
end
%-----Optional parent+child selection---------------------------------------
%-----Select which vectors are allowed to enter the new population------------
for k=1:I_NP
  %======Only use this if boundary constraints are needed==================
  if (I_bnd_constr == 1)
    for j=1:I_D %----boundary constraints via bounce back-------
      if (FM_ui(k,j) > FVr_maxbound(j))
        FM_ui(k,j) = FVr_maxbound(j) + rand*(FM_origin(k,j) - FVr_maxbound(j));
      end
      if (FM_ui(k,j) < FVr_minbound(j))
        FM_ui(k,j) = FVr_minbound(j) + rand*(FM_origin(k,j) - FVr_minbound(j));
```

```
        end
      end
    end
    %=====End boundary constraints=======================================
    S_tempval = feval(fname,FM_ui(k,:),S_struct);   % check cost of competitor
    I_nfeval  = I_nfeval + 1;
    if (left_win(S_tempval,S_val(k)) == 1)
      FM_pop(k,:) = FM_ui(k,:);                 % replace old vector with new one (for new
iteration)
      S_val(k)   = S_tempval;                   % save value in "cost array"
      %----we update S_bestval only in case of success to save time-----------
      if (left_win(S_tempval,S_bestval) == 1)
        S_bestval = S_tempval;                  % new best value
        FVr_bestmem = FM_ui(k,:);               % new best parameter vector ever
      end
    end
  end % for k = 1:NP
 FVr_bestmemit = FVr_bestmem;      % freeze the best member of this iteration for the coming
                                   % iteration. This is needed for some of the strategies.
%----Output section-----------------------------------------------------
 if (I_refresh > 0)
   if ((rem(I_iter,I_refresh) == 0) | I_iter == 1)


     if (I_plotting == 1)
       %semilogy(I_iter,ft,'.');
       plot(I_iter,S_bestval.FVr_oa(1),'.');
       xlabel ('Generation')
       ylabel ('Best value')
       title(sprintf('Best cost: %f',S_bestval.FVr_oa(1)));
%          title(sprintf('Best cost: %f',S_bestval.FVr_oa(1)));
```

```
        %axis([1 S_struct.I_itermax S_struct.F_VTR 1]);
        grid on;
        hold on;
        %PlotIt(FVr_temp,iter,S_struct);
        %PlotIt(FVr_bestmem,I_iter,S_struct);
    end
  end
 end

 I_iter = I_iter + 1;
end %---end while ((I_iter < I_itermax)  . . .
```

ฟังก์ชัน run

```
global rho Iload Ifa_m Ifa_mb Is_load  NR NL  data_TDS data_PS a1 b2 b3
 run('data_I')
[Is_load] = c_t(Iload)
[data_TDS] = c_t(Ifa_m)
[data_PS] = c_t(Ifa_mb)
NL = length(data_TDS(:,1));
rho = 1e9;
a1= 1;
b2 = 200;
b3 = 200;
NR=12
F_VTR = 1.0e-009;        % F_VTR  "Value To Reach" (stop when ofunc < F_VTR)
I_D   = 2*NR;            % I_D number of parameters of the objective function
FTac  = inf;

Nmax =1;
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
for i=1:NR
   FVr_minbound(i)   = 0.05;
   FVr_minbound(NR+i)= 1.10;
   FVr_maxbound(i)   = 1.25;
   FVr_maxbound(NR+i)= 1.50;
end
I_bnd_constr  = 1;            % 1: use bounds as bound constraints, 0: no bound constraints
I_NP          = 100;          % I_NP number of population members
I_itermax     = 500;          % I_itermax  maximum number of iterations (generations)
F_weight      = 0.5;          % F_weight DE-stepsize F_weight ex [0, 2]
F_CR          = 0.9;          % F_CR crossover probabililty constant ex [0, 1]
I_strategy    = 1;
I_refresh     = 1;
I_plotting    = 1;            % I_plotting Will use plotting if set to 1. Will skip plotting otherwise.
S_struct.I_NP        = I_NP;
S_struct.F_weight    = F_weight;
S_struct.F_CR        = F_CR;
S_struct.I_D         = I_D;
S_struct.FVr_minbound = FVr_minbound;
S_struct.FVr_maxbound = FVr_maxbound;
S_struct.I_bnd_constr = I_bnd_constr;
S_struct.I_itermax    = I_itermax;
S_struct.F_VTR        = F_VTR;
S_struct.I_strategy   = I_strategy;
S_struct.I_refresh    = I_refresh;
S_struct.I_plotting   = I_plotting;
fhandle = 'fun';
tic
for i=1:Nmax
```

```
  disp(i)
[FVr_x,S_y,I_nf] = deopt(fhandle,S_struct);
a(i)={[FVr_x(1:NR)' FVr_x(NR+1:2*NR)']};
b(i)=S_y.FVr_oa;
end
toc
b
[V,M] = min(b);
V
a{M}
```

ฟังก์ชัน c_T

```
function [Isec] = c_t(Ipri)
 % global Iload Ip Ct
 %Il = [Iload(:,1) Iload(:,2)*1.25]
[m,n] = size(Ipri); %m is row ; n is colume
 %No.Relays Transformer
%    CTratho     min    max
Ct = [    5/100       0     100
            5/150     101     150
            5/200     151     200
           5/250     201     250
     ];
%  disp('CTratho')
% fprintf('0.0500 is 5 : 100\n0.0333 is 5 : 150\n0.0250 is 5 : 200\n0.0200 is 5 : 250\n0.0167 is 5 :
300\n0.0143 is 5 : 350\n0.0125 is 5 : 400\n0.0111 is 5 : 450\n0.0100 is 5 : 500\n\n')
if n>3
   Ipri = [Ipri(:,1),Ipri(:,2);Ipri(:,3),Ipri(:,4)];
end
```

```
  for i=1:length(Ipri)
     for j=1:length(Ct)
        if Ipri(i,2)>Ct(j,2)||Ipri(i,2)==Ct(j,2) && Ipri(i,2)<Ct(j,3)||Ipri(i,2)==Ct(j,3)
           Ip(i) =  Ct(j,1);
        end
     end
  end
%    Ipri(:,2)
% Ip
  Isec = Ipri(:,2).*Ip';

  if n>3
     Isec = [Ipri((1:m),1),Isec(1:m),Ipri((m+1:2*m),1),Isec(m+1:2*m)];
  else
     Isec = [Ipri(:,1),Isec];
  end
%   disp('No.Relay    CTratho')
%   for i=1:length(Il)
%       fprintf(' %d       %f\n' ,Il(i,1),Ip(i))
%   end
return
```

ฟังก์ชัน data_I

```
function data_I()
global Iload Ifa_m Ifa_mb t_z2
I = [
          167.57
          203.44
          204.63
```

```
        80.51
        204.63
        167.57
        203.44
        75.07
        75.07
        78.61
        80.51
        78.61
                ];
N = [1:length(I)];
Iload = [N',I];
disp('TUN');
Ifa_m =   [
                 1 2358.36 6   550.22
                2   1093.9  7   1418.8
                3   2351.02 5   614.11
                4   647.03  11  2191.06
                5   1336.29 3   869.6
                6   1176.57 1   823.5
                7   2164.03 2   814.55
                8   1404.49 9   1091.59
                9   2106.65 8   823.22
                10  840.22  12  1193.94
                11  1347.46 4   882.24
                12  2252.28 10  572.76
                                        ];
      Ifa_mb = [
%       1%
                10 566 4   647
```

```
                11  638 12  2252
                4   1329    5   1336
                12  1177    6   1177
                5   606 7   2164
                7   1405    8   1404
                3   648 11  2191
                1   564 10  573
                2   217 3   870
                8   175 1   824
                9   821 2   815
                6   226 9   1092
                10  276 4   882
                11  244 12  1194
                4   669 5   614
                12  590 6   550
                5   369 7   1419
                7   858 8   823
                3   861 11  1347
                1   814 10  840
                2   797 3   2351
                8   803 1   2358
                9   1083    2   1094
                6   540 9   2107
                                ];
t_z2 = [
          1 0.3
          2 0.3
          3 0.3
          4 0.3
          5 0.3
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        6 0.3
        7 0.3
        8 0.3
        9 0.3
        10 0.3
        11 0.3
        12 0.3
                ];
    end
```

ฟังก์ชัน fun

```
clc;
close all;
clear all;
global rho Iload Ifa_m Ifa_mb Is_load  NR NL  data_TDS data_PS a1 b2 b3
 run('data_I')
[Is_load] = c_t(Iload)
[data_TDS] = c_t(Ifa_m)
[data_PS] = c_t(Ifa_mb)
NL = length(data_TDS(:,1));
rho = 1e9;
a1= 1;
b2 = 200;
b3 = 200;
NR=12
F_VTR = 1.0e-009;        % F_VTR  "Value To Reach" (stop when ofunc < F_VTR)
I_D   = 2*NR;            % I_D number of parameters of the objective function
FTac  = inf;
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
Nmax =1;


for i=1:NR
   FVr_minbound(i)   = 0.05;
   FVr_minbound(NR+i)= 1.10;
   FVr_maxbound(i)   = 1.25;
   FVr_maxbound(NR+i)= 1.50;
end
I_bnd_constr  = 1;            % 1: use bounds as bound constraints, 0: no bound constraints
I_NP          = 100;          % I_NP number of population members
I_itermax     = 500;          % I_itermax  maximum number of iterations (generations)
F_weight      = 0.5;          % F_weight DE-stepsize F_weight ex [0, 2]
F_CR          = 0.9;          % F_CR crossover probabililty constant ex [0, 1]
I_strategy    = 1;
I_refresh     = 1;
I_plotting    = 1;            % I_plotting Will use plotting if set to 1. Will skip plotting otherwise.
S_struct.I_NP         = I_NP;
S_struct.F_weight     = F_weight;
S_struct.F_CR         = F_CR;
S_struct.I_D          = I_D;
S_struct.FVr_minbound = FVr_minbound;
S_struct.FVr_maxbound = FVr_maxbound;
S_struct.I_bnd_constr = I_bnd_constr;
S_struct.I_itermax    = I_itermax;
S_struct.F_VTR        = F_VTR;
S_struct.I_strategy   = I_strategy;
S_struct.I_refresh    = I_refresh;
S_struct.I_plotting   = I_plotting;
fhandle = 'fun';
tic
```

```
for i=1:Nmax
  disp(i)
[FVr_x,S_y,I_nf] = deopt(fhandle,S_struct);
a(i)={[FVr_x(1:NR)' FVr_x(NR+1:2*NR)']};
b(i)=S_y.FVr_oa;
end
toc
b
[V,M] = min(b);
V
a{M}
```

แบบที่ 3 การคำนวณค่าปรับตั้งของรีเลย์ป้องกันกระแสเกิน

มีฟังก์ชันที่ใช้ในการรันทั้งหมด 5 ฟังก์ชันซึ่งประกอบไปด้วย ฟังก์ชัน deopt , fun , c_T, data_I และฟังก์ชัน run ซึ่งฟังก์ชัน deopt คือฟังก์ชันของวิธีการค้นหาคำตอบด้วยวิธีวิวัฒนาการผลต่างโดยที่ตัวโปรแกรมนั้นมีค่าเหมือนกันกับฟังก์ชัน deopt ที่ใช้กับแบบที่ 2 ดังที่กล่าวมาข้างต้น ฟังก์ชัน fun คือฟังก์ชันวัตถุประสงค์ ฟังก์ชัน c_T เป็นฟังก์ชันของแท็ปหม้อแปลงใช้สำหรับปรับคูณค่ากระแส ฟังก์ชัน data_I คือฟังก์ชันรับค่ากระแสโหลดและกระแสลัดวงจร และฟังก์ชันสุดท้ายคือฟังก์ชัน run เอาไว้ควบคุมตัวแปรที่สนใจควบคุม

ฟังก์ชัน c_T

```
function [Isec] = c_t(Ipri)
 % global Iload Ip Ct
 %Il = [Iload(:,1) Iload(:,2)*1.25]
[m,n] = size(Ipri); %m is row ; n is colume
 %No.Relays Transformer
%   CTratho    min   max
Ct = [   5/100      0    100
         5/150    101    150
```

```
          5/200     151     200
          5/250     201     250
                                        ];
%  disp('CTratho')
% fprintf('0.0500 is 5 : 100\n0.0333 is 5 : 150\n0.0250 is 5 : 200\n0.0200 is 5 : 250\n0.0167 is 5 :
300\n0.0143 is 5 : 350\n0.0125 is 5 : 400\n0.0111 is 5 : 450\n0.0100 is 5 : 500\n\n')
if n>3
   Ipri = [Ipri(:,1),Ipri(:,2);Ipri(:,3),Ipri(:,4)];
end


   for i=1:length(Ipri)
      for j=1:length(Ct)
         if Ipri(i,2)>Ct(j,2)||Ipri(i,2)==Ct(j,2) && Ipri(i,2)<Ct(j,3)||Ipri(i,2)==Ct(j,3)
            Ip(i) =  Ct(j,1);
         end
      end
   end
%     Ipri(:,2)
% Ip
   Isec = Ipri(:,2).*Ip';


   if n>3
      Isec = [Ipri((1:m),1),Isec(1:m),Ipri((m+1:2*m),1),Isec(m+1:2*m)];
   else
      Isec = [Ipri(:,1),Isec];
   end
return
```

ฟังก์ชัน data_I

```
function data_I()
global Iload Ifa_m Ifa_mb
I = [
        167.57
        203.44
        204.63
        80.51
        204.63
        167.57
        203.44
        75.07
        75.07
        78.61
        80.51
        78.61
                    ];
N = [1:length(I)];
Iload = [N',I];
disp('TUN');
Ifa_m =  [
                1 2358.36 6  550.22
                2  1093.9  7  1418.8
                3  2351.02 5  614.11
                4  647.03  11  2191.06
                5  1336.29 3  869.6
                6  1176.57 1  823.5
                7  2164.03 2  814.55
                8  1404.49 9  1091.59
                9  2106.65 8  823.22
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
                10  840.22  12  1193.94
                11  1347.46 4   882.24
                12  2252.28 10  572.76
                                                ];
Ifa_mb = [
%       1%
                10  566     4   647
                11  638     12  2252
                4   1329    5   1336
                12  1177    6   1177
                5   606     7   2164
                7   1405    8   1404
                3   648     11  2191
                1   564     10  573
                2   217     3   870
                8   175     1   824
                9   821     2   815
                6   226     9   1092

                10  276     4   882
                11  244     12  1194
                4   669     5   614
                12  590     6   550
                5   369     7   1419
                7   858     8   823
                3   861     11  1347
                1   814     10  840
                2   797     3   2351
                8   803     1   2358
                9   1083    2   1094
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
                6   540     9   2107

                                        ];
end
```

ฟังก์ชัน fun

```
function S_MSE = fun(FVr_temp, S_struct)
global rho Is_load Is_m Is_mb NR NL
%       TDS     PS    T(primary)
Amin = [ 0.05    1.25     0.05 ];
Amax = [ 1.10    1.50     2.00 ];
CTI = 0.3;
Tpri_cl   = 0;
Tpri_far  = 0;
Tbackup   = 0;
Tprimary  = 0;
TDS       = 0;
PS        = 0;
[m,n]=size(Is_mb);

for k=1:NR
    TDS(k)= FVr_temp(k);
    PS(k) = FVr_temp(NR+k);
 end
%.....................Function Objective.....................
for l=1:NR

   Tpri_cl(l)  = (0.14*TDS(l))/((((Is_m(l,2))/(PS(l)*Is_load(l,2)))^0.02)-1);
   Tpri_far(l)=0.14*TDS(Is_m(l,3))/(((Is_m(l,4)/(PS(Is_m(l,3))*Is_load(Is_m(l,3),2)))^0.02)-1);
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
    if Tpri_cl(l) <0
        Tpri_cl(l)=0.05;
    end
    if Tpri_far(l)<0
        Tpri_far(l)=0.05;
    end
end
ft = sum(Tpri_cl)+sum(Tpri_far);
%.....1.)TDS.....
fpen1 = 0;
for n=1:NR
    up=TDS(n)-Amax(1);
    down=Amin(1)-TDS(n);
    if up<=0
        up=0;
    end
    if down<=0
        down=0;
    end
    fpen1=fpen1+rho*(up)^2+rho*(down)^2;
end
%.....2.)PS.....
fpen2 = 0;
for n=1:NR
    up=PS(n)-Amax(2);
    down=Amin(2)-PS(n);
    if up<=0
        up=0;
    end
    if down<=0
```

```
    down=0;
  end
  fpen2=fpen2+rho*(up)^2+rho*(down)^2;
end
%.....................3.)Tpri.....................
fpen3 = 0;
for n=1:NL
  Tpri_cl(n);
  up=Tpri_cl(n)-Amax(3);
  down=Amin(3)-Tpri_cl(n);
  if up<=0
    up=0;
  end
  if down<=0
    down=0;
  end
  fpen3=fpen3+rho*(up)^2+rho*(down)^2;
end

fpen4 = 0;
for n=1:NL
  Tpri_far(n);
  up=Tpri_far(n)-Amax(3);
  down=Amin(3)-Tpri_far(n);
  if up<=0
    up=0;
  end
  if down<=0
    down=0;
  end
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
   fpen4=fpen4+rho*(up)^2+rho*(down)^2;
end
%.....................4.)CTI.....................
fpen5 = 0;
for l=1:m
   Tbackup  =
(0.14*TDS(Is_mb(l,1)))/((((Is_mb(l,2))/(PS(Is_mb(l,1))*Is_load(Is_mb(l,1),2)))^0.02)-1);
   Tprimary =
(0.14*TDS(Is_mb(l,3)))/((((Is_mb(l,4))/(PS(Is_mb(l,3))*Is_load(Is_mb(l,3),2)))^0.02)-1);
   dT = Tprimary + CTI - Tbackup;
   if dT<=0
      dT = 0;
   end
   if (Tbackup < 0) || (Tprimary < 0)
      dT = 0;
   end
   fpen5 = fpen5 + rho*(dT)^2;
end
%.....................................
FT = ft + fpen1 + fpen2 + fpen3 + fpen4 + fpen5;
%----strategy to put everything into a cost function-----------
S_MSE.I_nc      = 0;%no constraints
S_MSE.FVr_ca    = 0;%no constraint array
S_MSE.I_no      = 1;%number of objectives (costs)
S_MSE.FVr_oa(1) = FT;
return
```

ภาคผนวก ง

บทความทางวิชาการที่ได้รับการตีพิมพ์เผยแพร่ในระหว่างศึกษา

มหาวิทยาลัยเทคโนโลยีสุรนารี

รายชื่อบทความที่ได้รับการตีพิมพ์เผยแพร่ในระหว่างศึกษา

สมบูรณ์ทรัพย์ รอดพร ธนัดชัย กุลวรวานิชพงษ์ รัชดาพร อุ่นศิวิไลย์และอนันท์ อุ่นศิวิไลย์ (2554). **การจัดความสัมพันธ์รีเลย์ป้องกัน กระแสเกินเหมาะที่สุดโดยวิธีวิวัฒนาการผลต่าง**. การประชุมวิชาการเครือข่ายพลังงานแห่งประเทศไทยครั้งที่ 7 (ENETT2011) ณ โรงแรมภูเก็ต ออร์คิดรีสอร์ทแอนด์สปา จังหวัดภูเก็ต 3 - 5 พฤษภาคม 2554

สมบูรณ์ทรัพย์ รอดพร ชนากานต์ ชลสุข ธนัดชัย กุลวรวานิชพงษ์ อนันท์ อุ่นศิวิไลย์และรัชดาพร อุ่นศิวิไลย์. (2554). **การจัดความสัมพันธ์รีเลย์ป้องกันกระแสเกินเหมาะที่สุดโดยวิธีวิวัฒนาการผลต่างและจำลองผลด้วยโปรแกรม PSCAD/ EMTDC.** การประชุมวิชาการทางวิศวกรรมไฟฟ้าครั้งที่ 34 (EEcon2013) ณ โรงแรมแอมบาสซาเดอร์ ซิตี้ จอมเทียม พัทยา จังหวัดชลบุรี 26 - 28 ตุลาคม 2554

Somboonsup Rodporn Thanatchai Kulworawanichpong Dusit Uthitsunthorn Anant Oonsivilai Ratchadaporn Oonsivilai. (2553). **Optimal Coordination of Over - Current Relays Using Differential Evolution and Simulation Using PSCAD/EMTDC**. Electrical Engineering /Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), 2012 10th International Conference, Novotel Hua Hin-Chaam Beach Resort & Spa 16-18 May 2012

มหาวิทยาลัยเทคโนโลยีสุรนารี

การจัดความสัมพันธ์รีเลย์ป้องกันกระแสเกินเหมาะที่สุดโดยใช้การวิวัฒนาการผลต่าง

Optimal Coordination of Over-Current Relays Using Differential Evolution

สมบูรณ์ทรัพย์ รอดพร ธนัดชัย กุลวรวานิชพงษ์ อนันท์ อุ่นศิริไลย์ และ รัชฎาพร อุ่นศิริไลย์
สาขาวิชาวิศวกรรมไฟฟ้า สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี
111 ถ.มหาวิยาลัย ต.สุรนารี อ.เมือง จ.นครราชสีมา 30000 E-mail: somboon_tun@hotmail.com

บทคัดย่อ

บทความนี้ทำการศึกษาการหาค่าเหมาะที่สุดของการจัดความสัมพันธ์รีเลย์กระแสเกินแบบมีทิศทาง ปัญหานี้จะอยู่ในรูปของสมการที่ไม่เชิงเส้น และมีเงื่อนไขจำนวนมาก ตัวแปรตัดสินใจของสมการประกอบด้วย ตัวคูณเวลา (Namely time dial setting. TDS) และ ค่ากระแสทำงาน (Plug setting. PS) ของรีเลย์แต่ละตัว ฟังก์ชันวัตถุประสงค์สามารถหาได้จากผลรวมเวลาการทำงานของรีเลย์หลักต่ำสุด โดยนำ วิวัฒนาการผลต่าง (Differential Evolution. DE) มาใช้แก้หาผลเฉลย DE เป็นวิธีการหาค่าเหมาะสมแบบสโตคาสติก มีพื้นฐานจากการหาคำตอบของปัญหาโดยวิธีสุ่มเลือก พัฒนามาจาก Genetic Algorithm (GA) แต่มีโครงสร้างที่ไม่ซับซ้อน สามารถประยุกต์ใช้หาคำตอบของปัญหาที่มีตัวแปรที่ต้องการหาค่าความเหมาะสมหลายค่าได้รวดเร็ว และมีประสิทธิภาพ บทความนี้ได้นำเสนอการจำลองระบบทดสอบขนาด 6 บัส มาพิจารณาตัดสินใจปรับตั้งรีเลย์ในการจัดลำดับความสัมพันธ์

คำสำคัญ: วิวัฒนาการผลต่าง จีเนติกอัลกอริทึม รีเลย์ป้องกันกระแสเกินแบบมีทิศทาง

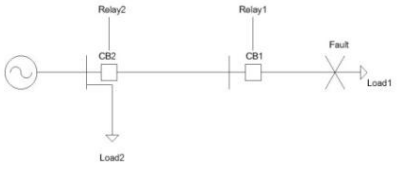
Abstract

In this paper, study of optimal coordination of directional over-current relays is proposed. The relay coordination problem is non-linear. It consists of two groups of control variables (Time Dial setting: TDS and Plug Setting:PS). The objective function used is the operation time of the main relay in all considered fault cases. The differential evolution (DE) is employed to solve for solutions of optimal relay coordination. It is a stochastic model base on finding the answer from the random method. This algorithm was developed from Genetic Algorithm (GA), but the structure is simpler. This paper employed a 6-bus test system as a test system. The results showed that the DE can find the best solution for optimal relay coordination problems.

Keywords: Differential Evolution, Coordination directional relay, Genetic Algorithm, Optimization

1. บทนำ

ในระบบไฟฟ้ากำลังประกอบด้วยระบบผลิต ระบบส่งจ่าย และระบบจำหน่าย ในสภาวะปกติระบบจะมีการไหลของกำลังไฟฟ้าจ่ายไปถึงโหลด ทำให้การจ่ายโหลดสามารถถูกจำกัดอยู่ที่ค่าพิกัดที่กำหนดได้ ถ้าโหลดเพิ่มขึ้นอย่างต่อเนื่องจนเกินค่าพิกัดดังกล่าว อาจจะยอมให้เกิดการทำงานในสภาวะโหลดเกินได้ในช่วงเวลาสั้นๆ ค่าหนึ่ง ถ้าเกินเวลาที่กำหนดแล้วโหลดยังอยู่ในสภาวะโหลดเกินอยู่ อุปกรณ์ป้องกันต้องทำการปลดวงจรในส่วนนั้นออกเพื่อป้องกันไม่ให้เกิดความเสียหายกับอุปกรณ์อื่นๆ เรียกการป้องกันลักษณะนี้ว่า การป้องกันโหลดเกิน (overload protection) ส่วนการป้องกันอีกลักษณะหนึ่งคือ การป้องการลัดวงจร(short circuit protection) เมื่อเกิดการลัดวงจรขึ้นจะมีกระแสไหลในสายส่งเส้นต่างๆ และอุปกรณ์ต่างๆ เป็นปริมาณสูง อุปกรณ์ป้องกันจะตรวจพบกระแสลัดวงจรนี้ และทำหน้าที่ตัดส่วนของระบบที่เกิดความผิดพร่องนี้ออกไป โดยส่วนที่ไม่เกิดความผิดพร่องยังคงทำงานได้ตามปกติ รีเลย์ป้องกัน (Protective relays)เป็นอุปกรณ์ที่มีความสำคัญของระบบป้องกัน[1] มีหน้าที่ตรวจจับกระแสลัดวงจรหรือกระแสเกินในระบบและสั่งงานไปยังเซอร์กิตเบรคเกอร์(CB) เพื่อตัดวงจรส่วนที่เกิดความผิดพร่องนั้นออกไปจากระบบ ดังรูปที่ 1 เมื่อเกิดลัดวงจรขึ้นที่รีเลย์1 และ รีเลย์ 2 ตรวจพบกระแสลัดวงจร ดังนั้นรีเลย์1 ต้องทำหน้าที่สั่งงานให้ CB1 ตัดวงจรในส่วนที่เกิดลัดวงจรของโหลด1 ออกจากระบบ เรียกรีเลย์นี้ว่า " รีเลย์หลัก(main) " ทำให้ระบบยังคงทำการจ่ายโหลด 2 ได้ต่อไป แต่ถ้ารีเลย์ 1 สั่งงานผิดพลาด หรือ CB1 ทำงานผิดพลาด ทำให้ไม่สามารถตัดส่วนที่เกิดลัดวงจรออกจากระบบ รีเลย์ 2 ต้องทำหน้าที่สั่งงานให้ CB2 ตัดส่วนวงจรส่วนที่ลัดวงจรออกจากระบบ เรียกรีเลย์นี้ว่า " รีเลย์สำรอง(backup)" ทำให้ระบบไม่สามารถจ่ายโหลดต่อไปได้ เนื่องจากจำเป็นจะต้องป้องกันไม่ให้เกิดความเสียหายแก่อุปกรณ์ไฟฟ้า ดังนั้นการออกแบบการทำงานของรีเลย์หลักกับรีเลย์สำรองจึงมีความสำคัญในระบบป้องกัน เพื่อที่จะทำการจ่ายโหลดได้อย่างต่อเนื่องและป้องกันความเสียหายแก่อุปกรณ์ต่างๆ



รูปที่ 1 การเกิดลัดวงจรในระบบ

ปัญหาการออกแบบการทำงานของรีเลย์ คือการจัดลำดับการทำงานของรีเลย์ เนื่องจากระบบอาจมีรีเลย์จำนวนมาก ซึ่งฟังก์ชันวัตถุประสงค์ของปัญหาคือ เวลาการทำงานของรีเลย์ทั้งระบบต่ำสุด ซึ่งต้องทำการปรับตั้งให้แต่ละตัวมีความไวในการทำงานตามขอบเขต ในงานวิจัยนี้ จะใช้ฟังก์ชันวัตถุประสงค์ในแนวทางของ[1] เมื่อสร้างฟังก์ชันวัตถุประสงค์แล้ว ในส่วนของการแก้ปัญหาที่ผ่านมามีการประยุกต์วิธีการหาค่าเหมาะสมมาใช้แก้ปัญหานี้มากมาย เช่น วิธี linear programming วิธี non-linear programming เป็นต้น ซึ่งเป็นวิธีการหาค่าเหมาะสมแบบดีเทอร์มินิสติก จากการศึกษาในอดีตพบว่า วิธีเหล่านี้มักจะให้ค่าเหมาะสมแบบเฉพาะที่ (local optimum) ภายหลังจึงได้มีการพัฒนาวิธีการหาค่าเหมาะสมแบบสโตคาสติกมาใช้ เช่น วิธี Genetic Algorithm (GA) วิธี Simulate annealing(SA) ซึ่งวิธีเหล่านี้สามารถหาคำตอบที่ใกล้เคียงค่าเหมาะสมที่แท้จริง (near global optimum)

วิวัฒนาการผลต่าง(Differential Evolution. DE)[3] เป็นวิธีการหาค่าเหมาะสมแบบสโตคาสติก มีพื้นฐานจากการหาคำตอบของปัญหาโดยวิธีสุ่มเลือก พัฒนามาจาก GA แต่มีโครงสร้างที่ไม่ซับซ้อน และสามารถประยุกต์ใช้หาคำตอบของปัญหาต่างๆ ได้รวดเร็ว และมีประสิทธิภาพ ในการศึกษานี้ได้ประยุกต์ใช้ DE ในการหาค่าเหมาะสมในการจัดลำดับการทำงานของรีเลย์กระแสเกิน เพื่อหาค่าเหมาะสมและนำไปปรับตั้งรีเลย์ป้องกันต่อไป

2.การสร้างสมการของปัญหา

2.1 สมการรูปทั่วไป

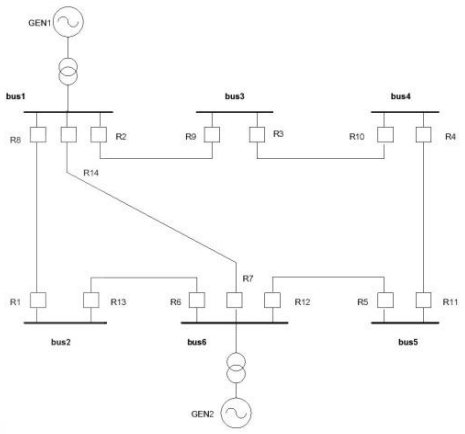
สมการทางคณิตศาสตร์ของปัญหานี้จะใช้รูปแบบของ [1] เวลาการทำงาน (T)ของรีเลย์ป้องกันกระแสเกินแบบมีทิศทาง (Direction Over Current Relays. DOCR ประกอบด้วย ตัวคูณเวลา (Time dial setting. TDS) ค่ากระแสทำงาน(Plug setting. PS) และกระแสลัดวงจร(Fault current. Ifault) ดังนั้นสมการเวลาของรีเลย์สามารถเขียนได้ดังนี้

$$T = \frac{\alpha * TDS}{(\frac{I}{PS * CT_{pri_rating}})^{\beta} - \gamma} \quad (1)$$

จากสมการจะมีเพียง TDS และ PS เท่านั้นที่ไม่ทราบค่า ซึ่งจะใช้ตัดสินใจในการปรับตั้ง (setting) สัญลักษณ์ที่อยู่ในสมการ α, β และ γ คือ ค่าคุณสมบัติของรีเลย์ชนิดนี้ มีค่าดังนี้ α = 0.14, β = 0.02 และ γ = 1 [IEEE std (1997)] CT_{pri_rating} คือ กระแสที่ไหลผ่านหม้อแปลงกระแส(Current Transformer)ที่ต่อกับรีเลย์ โดยสามารถปรับอัตราส่วนนี้จากระดับกระแสในระบบปกติ จึงเป็นตัวแปรที่ทราบค่า

2.2 ฟังก์ชันวัตถุประสงค์และเงื่อนไขของปัญหา

ในการหาค่าต่ำสุดของการจัดลำดับการทำงานของรีเลย์จะใช้เทคนิคการหาค่าต่ำสุดของฟังก์ชันวัตถุประสงค์ สมการวัตถุประสงค์นี้จะอยู่ในรูปของผลรวมของเวลาการทำงานของรีเลย์หลัก (Primary) ซึ่งประกอบด้วย เวลาของรีเลย์ตัวที่อยู่ใกล้ขณะเกิดลัดวงจร (Close-in) และเวลาของรีเลย์ตัวที่อยู่ไกลขณะเกิดลัดวงจร สามารถเขียนสมการวัตถุประสงค์ได้ดังสมการที่ (2)



รูปที่ 2 ระบบทดสอบ 6 บัส

Minimize

$$O_{bj} = \sum_{i=1}^{N_{cl}} T^{i}_{pri_cl_in} + \sum_{j=1}^{N_{far}} T^{j}_{pri_far_bus} \quad (2)$$

เมื่อ N_{cl} คือ จำนวนของรีเลย์ที่อยู่ใกล้จุดลัดวงจร

N_{far} คือ จำนวนของรีเลย์ที่อยู่ไกลจุดลัดวงจร

$T^{j}_{pri_cl_in}$ คือ เวลาเริ่มทำงานของรีเลย์ที่อยู่ใกล้จุดลัดวงจร

$T^{j}_{pri_far_bus}$ คือ เวลาเริ่มทำงานของรีเลย์ที่อยู่ไกลจุดลัดวงจร

เงื่อนไขของตัวแปร

1. $TDS^{i}_{min} \leq TDS^{i} \leq TDS^{i}_{max}$

   เมื่อ TDS^{i}_{min} = 0.05 TDS^{i}_{max} = 1.1

2. $PS^{i}_{min} \leq PS^{i} \leq PS^{i}_{max}$

   เมื่อ PS^{i}_{min} = 1.25 PS^{i}_{max} = 1.50

3. $T_{primary}$ มีค่าระหว่าง 0.05 ถึง 1.00

4. $T_{backup} - T_{primary} - CTI \geq 0$

เมื่อ T_{backup} คือ เวลาการทำงานของรีเลย์ที่ทำหน้าที่เป็นรีเลย์สำรอง(backup) $T_{primary}$ คือ เวลาการทำงานของรีเลย์ที่ทำหน้าที่เป็นรีเลย์หลัก(primary)และ CTI คือ เวลาการจัดความสัมพันธ์ของรีเลย์ (CTI = 0.3)

3. วิวัฒนาการผลต่าง (Differential evolution. DE)

ขั้นตอนการหาค่าเหมาะสมด้วย DE [2] มีดังนี้

3.1. Initialization

สร้างชุดคำตอบของตัวแปรตัดสินใจ(decision variables. X_i) จำนวน NP ชุด หนึ่งชุดประกอบด้วยตัวแปรตัดสินใจ D ตัว แล้วคำนวณหาค่าฟังก์ชันวัตถุประสงค์ของแต่ละชุดคำตอบ

3.2. Mutation

สร้างเวกเตอร์สัมผัส (target vector. $X_{i,G}$)จำนวน NP ชุด(ชุดละ D ตัว) สุ่มเลือก 3 เวกเตอร์($X_{r1,G}$, $X_{r2,G}$, $X_{r3,G}$)ที่ไม่ซ้ำกันกับเวกเตอร์สัมผัสจากชุดคำตอบที่สร้างไว้ เพื่อทำการกลายพันธุ์ (Mutation. $V_{i,G+1}$) โดยใช้ความสัมพันธ์ในสมการที่ 3 [3]

$$V_{i,G+1} = X_{r1,G} + F(X_{r2,G} - X_{r3,G}) \quad (3)$$

เมื่อ F คือ weighing factor มีค่าระหว่าง 0 ถึง 2

3.3. Crossover

เป็นกระบวนการข้ามสายพันธุ์(crossover) เพื่อให้คำตอบมีความหลากหลายขึ้น สร้างได้จากความสัมพันธ์ตามสมการที่ 5

$U_{ji,G+1} = (U_{1i,G+1}, U_{2i,G+1}, U_{3i,G+1},\ldots, U_{Di,G+1})$

$$U_{ji,G+1} = \begin{cases} V_{ji,G+1} & \text{if } (\text{randb}(j) \le CR \text{ or } j = \text{rnbr}(i) \\ X_{ji,G} & \text{if } (\text{randb}(j) > CR \text{ or } j \ne \text{rnbr}(i) \end{cases} \quad (4)$$

เมื่อ $U_{ji,G+1}$ = เวกเตอร์ทดสอบ(Trial vector)

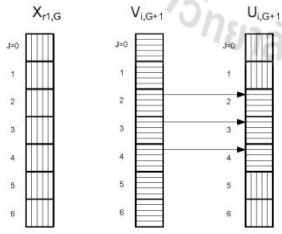
$X_{ji,G+1}$ = Mutant vector

$V_{ji,G}$ = Target vector

randb(j) = การสุ่มตัวจำนวนจริงที่มีค่าระหว่าง 0 ถึง 1 ครั้งที่ j

CR = Crossover Constant มีค่าเป็นเลขจำนวนจริง ระหว่าง 0 ถึง 1

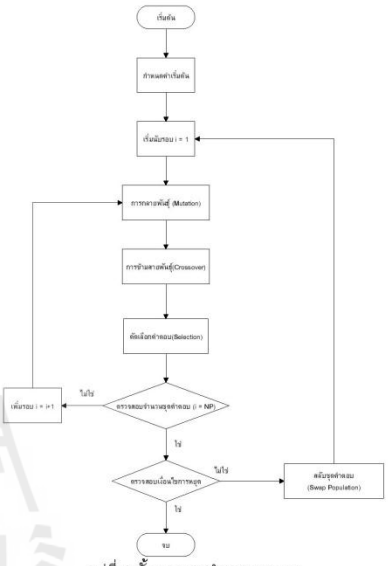
rnbr(i) = ค่า Index จากการสุ่มเลือก มีค่าเป็นเลขจำนวนเต็ม ระหว่าง 0,1,...,D-1j = 0,1,...,D-1



รูปที่ 3 กระบวนการ crossover ระหว่าง target vector และ mutant vector ที่มีตัวแปรการตัดสินใจ (D) เท่ากับ 7

3.4. Selection

การคัดสรรคำตอบ ซึ่ง vector ที่ให้คำตอบที่ดีกว่าจะถูกเก็บไว้ต่อไป วิธีการคือเปรียบเทียบค่าฟังก์ชันของ trial vector($U_{i,G+1}$) กับ target vector($V_{i,G+1}$) ในกรณีที่ trial vector ให้ค่า function value ที่ดีกว่า มันก็จะแทนที่ target vector ใน generation ต่อไป จากนั้นทำซ้ำขั้นตอนที่ 2 ถึง 4 จนครบทุกชุดคำตอบ แล้วนำคำตอบชุดใหม่แทนที่คำตอบชุดเก่า ทำซ้ำกระบวนการจนพบเงื่อนไขการหยุด



รูปที่ 4 ขั้นตอนการทำงานของ DE

4.หาค่าเหมาะที่สุด

ตารางที่1 กระแสลัดวงจรโดยพิจารณาตำแหน่งลัดวงจร

ใกล้จุดลัดวงจร			ไกลจุดลัดวงจร		
รีเลย์	a^I	b^I	รีเลย์	c^I	d^I
R1	67.06	3.32	R8	63.19	3.32
R2	67.02	2.76	R9	5.35	2.76
R3	166.72	2.73	R10	73.08	2.73
R4	111.71	2.61	R11	117.70	2.61
R5	45.05	3.39	R12	116.94	3.39
R6	496.48	2.61	R13	82.95	2.61
R7	84.65	2.90	R14	31.31	2.90
R8	124.04	3.32	R1	20.79	3.32
R9	18.05	2.76	R2	42.03	2.76
R10	116.72	2.73	R3	112.66	2.73
R11	174.03	2.61	R4	68.50	2.61
R12	178.81	3.39	R5	14.23	3.39
R13	249.06	2.61	R6	272.05	2.61
R14	84.65	2.90	R7	31.31	2.90

ตารางที่ 2 ลำดับการทำงานของรีเลย์หลักและรีเลย์สำรอง

รีเลย์สำรอง			รีเลย์หลัก		
รีเลย์	e^i	f^i	รีเลย์	g^i	h^i
R1	20.14	3.32	R2	67.02	2.76
R1	19.58	3.32	R14	84.65	2.90
R2	41.68	2.76	R3	166.72	2.73
R3	111.71	2.73	R4	111.71	2.61
R4	67.64	2.61	R5	45.05	3.39
R5	13.70	3.39	R6	496.48	2.61
R5	13.56	3.39	R7	84.65	2.90
R6	268.24	2.61	R1	67.06	3.32
R7	30.56	2.90	R2	67.02	2.76
R7	30.42	2.90	R8	124.04	3.32
R8	62.27	3.32	R13	249.06	2.61
R9	5.14	2.76	R8	124.04	3.32
R9	5.09	2.76	R14	84.65	2.90
R10	72.18	2.73	R9	18.05	2.76
R11	116.73	2.61	R10	116.72	2.73
R12	115.90	3.39	R11	174.03	2.61
R13	79.41	2.61	R7	84.65	2.90
R13	80.55	2.61	R12	178.81	3.39
R14	30.46	2.90	R6	496.48	2.61
R14	30.59	2.90	R12	178.81	3.39
R1	7.83	3.32	R2	42.03	2.76
R2	28.16	2.76	R3	112.66	2.73
R3	68.50	2.73	R4	68.50	2.61
R4	21.36	2.61	R5	14.23	3.39
R6	83.15	2.61	R1	20.79	3.32
R7	11.89	2.90	R2	42.03	2.76
R8	20.74	3.32	R13	82.95	2.61
R10	21.42	2.73	R9	5.35	2.76
R11	73.08	2.61	R10	73.08	2.73
R12	78.39	3.39	R11	117.70	2.61
R13	35.17	2.61	R12	116.94	3.39
R14	13.35	2.90	R12	116.94	3.39

จากรูปที่ 2 ระบบทดสอบ 6 บัส[3],[5] มีสายส่งทั้งหมด 7 เส้น ติดตั้งรีเลย์ป้องกันทั้งหมด 14 ตัว และจากสมการ 2 จะพบว่ามีการติดตั้งรีเลย์ทั้งหมด 14 ตัว N_{cl} = 14 และ N_{far} = 14 ทำให้เกิดตัวแปรในสมการวัตถุประสงค์ทั้งหมด 28 ตัว นำมาเขียนเป็นสมการเพื่อใช้หาค่าต่ำสุดได้ว่า

$$O_{bj} = \sum_{i=1}^{14} T^{i}_{pri_cl_in} + \sum_{j=1}^{14} T^{j}_{pri_far_bus}$$

$$T_{pri_cl_in} = \frac{0.14 * TDS}{(\frac{a}{PS * b})^{0.02} - 1}$$

$$T_{pri_far_bus} = \frac{0.14 * TDS}{(\frac{c}{PS * d})^{0.02} - 1}$$

ค่า a b c และ d คือค่ากระแสลัดวงจรและค่าอัตราส่วนของหม้อแปลง สามารถดูได้จากตารางที่ 1 นอกจากสมการวัตถุประสงค์แล้วยังต้องมีสมการของเงื่อนไขอีกด้วย ซึ่งสามารถเขียนได้ดังนี้

$$T_{backup} - T_{primary} - CTI \geq 0$$

$$T_{backup} = \frac{0.14 * TDS}{(\frac{e}{PS * f})^{0.02} - 1}$$

$$T_{primary} = \frac{0.14 * TDS}{(\frac{g}{PS * h})^{0.02} - 1}$$

ค่า e f g และ h คือค่ากระแสลัดวงจรและค่าอัตราส่วนของหม้อแปลง สามารถดูได้จากตารางที่ 2

5. ผลการแก้ปัญหา

เมื่อใช้วิธีวิวัฒนาการผลต่าง(Differential Evolution Algorithm) แก้ปัญหาฟังก์ชันวัตถุประสงค์ เพื่อหาค่าต่ำสุดในการทำงานของรีเลย์ทั้งระบบแล้ว จะได้ผลเฉลยที่เป็นค่าเหมาะสมในการจัดลำดับการงานของรีเลย์ คือ ค่าตัวคูณเวลา และค่ากระแสทำงาน แสดงในตารางที่ 3

ตารางที่ 3 จำนวนเท่าเวลาปรับตั้ง และ กระแสเริ่มต้นทำงาน

หมายเลขรีเลย์	ค่าตัวคูณเวลา	ค่าปรับตั้งกระแส
1	0.2143	1.4993
2	0.3423	1.5000
3	0.4141	1.4998
4	0.3100	1.5000
5	0.1489	1.4998
6	0.5801	1.4999
7	0.3077	1.4911
8	0.3761	1.4975
9	0.0500	1.3384
10	0.2874	1.4998
11	0.4096	1.5000
12	0.4371	1.4998
13	0.4695	1.4998
14	0.3072	1.4963
ค่าฟังก์ชันวัตถุประสงค์ = 22.48		

6. สรุป

การออกแบบรีเลย์ป้องกันให้ทำงานได้สอดคล้องและตามลำดับสามารถทำได้โดยสร้างฟังก์ชันวัตถุประสงค์ของปัญหา[1] แล้วนำวิธีการแก้ปัญหาแบบต่างๆ มาประยุกต์ใช้ ในงานวิจัยนี้ได้แสดงการแก้ปัญหาด้วยวิธี วิวัฒนาการผลต่าง(Differential Evolution Algorithms. DE) ซึ่งสามารถใช้แก้หาค่าที่เหมาะสมที่สุด เพื่อนำไปปรับตั้งค่าพารามิเตอร์การทำงานของรีเลย์ที่ถูกออกแบบไว้ให้ทำงานได้ตรงตามวัตถุประสงค์และใช้เวลาการทำงาน (ตัดวงจรส่วนที่ผิดพร่องออก)ที่น้อยที่สุด เพื่อให้เกิดความเสียหายแก่อุปกรณ์ไฟฟ้าและการสูญเสียกำลังไฟฟ้าน้อยที่สุด

เอกสารอ้างอิง

[1] R. Thangaraj, M. Pant, K. Deep "Optimal coordination of over-current relays using modified differential evolution algorithms" *journal*, Jan 16,2010

[2] S. Chittaladakorn, A. Prempree "Optimal Design of water Distribution Piping system using different evolution algorithm" *การประชุมวิชาการวิศวกรรมโยธาแห่งชาติครั้งที่ 10* , พฤษภาคม 2548

[3] F. Razavi, H. A. Abyaneh, M. Al-Dabbagh, R. Mohammadi, H. Torkaman "A new comprehensive genetic algorithm method for optimal overcurrent relays coorfination" *ScienceDirect Electrical Power Systems* 78(2008) 713-720

[4] ธนัตชัย กุลวรวานิชพงษ์ (กันยายน 2552) "การป้องกันและรีเลย์" สาขาวิศวกรรมไฟฟ้า สำนักวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

[5] R. M. Chabanloo, H. A. Abyaneh, S. S. H.Kamanger "A New Genetic Algorithm Method for Optimal Coordination of Overcurrent and Distance Relays Overcurrent Relays", *2nd IEEE International Conference on Power and Energy*, December 1-3, 2008

มหาวิทยาลัยเทคโนโลยีสุรนารี

การจัดความสัมพันธ์รีเลย์ป้องกันกระแสเกินเหมาะที่สุดโดยใช้การวิวัฒนาการผลต่าง และจำลองผลด้วยโปรแกรม PSCAD/EMTDC

Optimal Coordination of Over-Current Relays Using Differential Evolution And Simulation Using PSCAD/EMTDC

สมบูรณ์ทรัพย์ รอดพร[1] ชนากานต์ ชลสุข[1] ธนัดชัย อุลวรวานิชพงษ์[1] อนันท์ อุ่นศิวิไลย์[1] และรัชดาพร อุ่นศิวิไลย์[2]

[1] สาขาวิศวกรรมไฟฟ้า สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

[2]สาขาวิชาเทคโนโลยีอาหาร สำนักวิชาเทคโนโลยีการเกษตร มหาวิทยาลัยเทคโนโลยีสุรนารี

111 ถ.มหาวิยาลัย ต.สุรนารี อ.เมือง จ.นครราชสีมา 30000 โทรศัพท์ 044-223-000

บทคัดย่อ

บทความนี้ทำการศึกษาการหาค่าเหมาะที่สุดของการจัดความสัมพันธ์รีเลย์กระแสเกินแบบมีทิศทาง ปัญหานี้จะอยู่ในรูปของสมการที่ไม่เชิงเส้น และมีเงื่อนไขจำนวนมาก ตัวแปรตัดสินใจของสมการประกอบด้วย ตัวคูณเวลา (Time dial setting. TDS) และ ค่ากระแสทำงาน (Plug setting. PS) ของรีเลย์แต่ละตัว ฟังก์ชันวัตถุประสงค์สามารถหาได้จากผลรวมเวลาการทำงานของรีเลย์หลักต่ำสุด โดยนำวิธีวิวัฒนาการผลต่าง (Differential Evolution : DE) มาใช้แก้หาผลเฉลย วิธีวิวัฒนาการผลต่างเป็นวิธีการหาค่าเหมาะสมแบบสโตคาสติก มีพื้นฐานจากการหาคำตอบของปัญหาโดยวิธีสุ่มเลือก พัฒนามาจากจีเนติคอัลกอริทึม (Genetic Algorithm :GA) แต่มีโครงสร้างที่ไม่ซับซ้อน สามารถประยุกต์ใช้หาคำตอบของปัญหาที่มีตัวแปรที่ต้องการหาค่าความเหมาะสมหลายค่าได้รวดเร็ว และมีประสิทธิภาพ บทความนี้ได้นำเสนอการจำลองระบบทดสอบขนาด 6 บัส มาพิจารณาตัดสินใจปรับตั้งรีเลย์ในการจัดลำดับความสัมพันธ์ นำผลที่ได้นั้นไปจำลองด้วยโปรแกรม PSCAD/EMTDC เพื่อดูผลที่ได้จากการปรับตั้งรีเลย์

คำสำคัญ: วิวัฒนาการผลต่าง, จีเนติกอัลกอริทึม, รีเลย์ป้องกันกระแสเกินแบบมีทิศทาง

Abstract

In this paper, study of optimal coordination of directional over-current relays is proposed. The relay coordination problem is non-linear. It consists of two groups of control variables (Time Dial setting: TDS and Plug Setting :PS). The objective function used is the operation time of the main relay in all considered fault cases. The differential evolution (DE) is employed to solve for solutions of optimal relay coordination. It is a stochastic model base on finding the answer from the random method. This algorithm was developed from Genetic Algorithm (GA), but the structure is simpler. This paper employed a 6-bus test system as a test system. The results showed that the DE can find the best solution for optimal relay coordination problems. Then simulation the result by PSCAD/EMTDC for the effect of setting relay.

Keywords: Differential Evolution, Coordination directional relay, Genetic Algorithm, Optimization

1. บทนำ

ในการป้องกันการลัดวงจร (short circuit protection) เมื่อเกิดการลัดวงจรขึ้นจะมีกระแสไหลในสายส่ง และอุปกรณ์ต่างๆ เป็นปริมาณสูง อุปกรณ์ป้องกันจะตรวจพบกระแสลัดวงจรนี้ และทำหน้าที่ตัดส่วนของระบบที่เกิดความผิดพร่องนี้ออกไป โดยส่วนที่ไม่เกิดความผิดพร่องยังคงทำงานได้ตามปกติ โดยเฉพาะในระบบไฟฟ้าแบบลูปหรือแบบโครงข่ายจะมีความซับซ้อนในการจัดลำดับความสัมพันธ์ของรีเลย์ ดังนั้นจึงต้องแก้ปัญหาโดยหาค่าที่เหมาะสมมาใช้ในการปรับตั้งค่ารีเลย์ เพื่อให้มีความไวในการทำงานตามขอบเขต ซึ่งประกอบด้วยค่ากระแสทำงานและตัวคูณเวลา (Time Dial Setting) บทความนี้ใช้วิธีทางคณิตศาสตร์เพื่อหาค่าเวลาในการทำงานต่ำที่สุดในการจัดความสัมพันธ์ของรีเลย์โดยมีฟังก์ชันวัตถุประสงค์ของปัญหาคือ เวลาการทำงานของรีเลย์ทั้งระบบต่ำสุด ในบทความนี้จะใช้ฟังก์ชันวัตถุประสงค์ในแนวทางของ [1] ในส่วนของการแก้ปัญหาที่ผ่านมามีการประยุกต์วิธีการหาค่าเหมาะสมมาใช้แก้ปัญหานี้มากมาย เช่น วิธี linear programming วิธี non-linear programming เป็นต้น ซึ่งเป็นวิธีการหาค่าเหมาะสมแบบดีเทอร์มินีสติก จากการศึกษาในอดีตพบว่า วิธีเหล่านี้มักจะให้ค่าเหมาะสมแบบเฉพาะที่ (local optimum) ภายหลังจึงได้มีการพัฒนาวิธีการหาค่าเหมาะสมแบบสโตคาสติกมาใช้ เช่น วิธี Genetic Algorithm (GA)

วิธี Simulate annealing (SA) ซึ่งวิธีเหล่านี้สามารถหาคำตอบที่ใกล้เคียงค่าเหมาะสมที่แท้จริง (near global optimum)

วิวัฒนาการผลต่าง (Differential Evolution, DE) เป็นวิธีการหาค่าเหมาะสมแบบสโตคาสติก มีพื้นฐานจากการหาคำตอบของปัญหาโดยวิธีสุ่มเลือก พัฒนามาจากจีเนติคอัลกอริทึม (GA) แต่มีโครงสร้างที่ไม่ซับซ้อน และสามารถประยุกต์ใช้หาคำตอบของปัญหาต่าง ๆ ได้รวดเร็ว และมีประสิทธิภาพ ในการศึกษานี้ได้ประยุกต์ใช้วิธีวิวัฒนาการผลต่าง ในการหาค่าเหมาะสมการจัดลำดับการทำงานของรีเลย์กระแสเกิน

2. ฟังก์ชันวัตถุประสงค์และเงื่อนไขของปัญหา

ในการหาค่าต่ำสุดของการจัดลำดับการทำงานของรีเลย์จะใช้เทคนิคการหาค่าต่ำสุดของฟังก์ชันวัตถุประสงค์ สมการวัตถุประสงค์นี้จะอยู่ในรูปของผลรวมของเวลาการทำงานของรีเลย์หลัก (Primary) ซึ่งประกอบด้วย เวลาของรีเลย์ตัวที่อยู่ใกล้ขณะเกิดลัดวงจร (Close-in) และเวลาของรีเลย์ตัวที่อยู่ไกลขณะเกิดลัดวงจร สามารถเขียนสมการวัตถุประสงค์ได้ดังสมการที่ (1)

Minimize

$$O_{bj} = \sum_{i=1}^{N_{cl}} T^{i}_{pri_cl_in} + \sum_{j=1}^{N_{far}} T^{j}_{pri_far_bus} \quad (1)$$

เมื่อ N_{cl} คือ จำนวนของรีเลย์ที่อยู่ใกล้จุดลัดวงจร

N_{far} คือ จำนวนของรีเลย์ที่อยู่ไกลจุดลัดวงจร

$T^{i}_{pri_cl_in}$ คือ เวลาเริ่มทำงานของรีเลย์ที่อยู่ใกล้จุดลัดวงจร

$T^{i}_{pri_far_bus}$ คือ เวลาเริ่มทำงานของรีเลย์ที่อยู่ไกลจุดลัดวงจร

$$T^{i}_{pri_cl_in} = T^{i}_{pri_far_bus} = \frac{\alpha * TDS}{\left(\frac{I}{PS * CT_{pri_rating}}\right)^{\beta} - \gamma} \quad (2)$$

$\alpha = 0.14,\ \beta = 0.02$ และ $\gamma = 1$

CT_{pri_rating} คือ กระแสที่ไหลผ่าน หม้อแปลงกระแสที่ต่อกับรีเลย์

I คือ กระแสลัดวงจรสูงสุด

เงื่อนไขของตัวแปร

1. $TDS^{i}_{min} \le TDS^{i} \le TDS^{i}_{max}$
   เมื่อ $TDS^{i}_{min} = 0.05\ TDS^{i}_{max} = 1.1$
2. $PS^{i}_{min} \le PS^{i} \le PS^{i}_{max}$
   เมื่อ $PS^{i}_{min} = 1.25\ PS^{i}_{max} = 1.50$
3. $T_{primary}$ มีค่าระหว่าง 0.05 ถึง 1.00
4. $T_{backup} - T_{primary} - CTI \ge 0$

เมื่อ T_{backup} คือ เวลาการทำงานของรีเลย์ที่ทำหน้าที่เป็นรีเลย์สำรอง (backup) $T_{primary}$ คือ เวลาการทำงานของรีเลย์ที่ทำหน้าที่เป็นรีเลย์หลัก (primary) และ CTI คือ เวลาการจัดความสัมพันธ์ของรีเลย์ (CTI = 0.3)

3. วิวัฒนาการผลต่าง (Differential evolution. DE)

ขั้นตอนการหาค่าเหมาะสมด้วย DE [2] มีดังนี้

3.1. Initialization

สร้างชุดคำตอบของตัวแปรตัดสินใจ (decision variables. X_j) จำนวน NP ชุด หนึ่งชุดประกอบด้วยตัวแปรตัดสินใจ D ตัว แล้วคำนวณหาค่าฟังก์ชันวัตถุประสงค์ของแต่ละชุดคำตอบ

3.2. Mutation

สร้างเวกเตอร์สัมผัส (target vector. $X_{i,G}$) จำนวน NP ชุด (ชุดละ D ตัว) สุ่มเลือก 3 เวกเตอร์ ($X_{r1,G}$, $X_{r2,G}$, $X_{r3,G}$) ที่ไม่ซ้ำกันกับเวกเตอร์สัมผัสจากชุดคำตอบที่สร้างไว้ เพื่อทำการกลายพันธุ์ (Mutation. $V_{i,G+1}$) โดยใช้ความสัมพันธ์ในสมการที่ 3

$$V_{i,G+1} = X_{r1,G} + F(X_{r2,G} - X_{r3,G}) \quad (3)$$

เมื่อ F คือ weighing factor มีค่าระหว่าง 0 ถึง 2

3.3. Crossover

เป็นกระบวนการข้ามสายพันธุ์(crossover) เพื่อให้คำตอบมีความหลากหลายขึ้น สร้างได้จากความสัมพันธ์ตามสมการที่ 4

$U_{ji,G+1} = (U_{1i,G+1}, U_{2i,G+1}, U_{3i,G+1}, \ldots, U_{Di,G+1})$

$$U_{ji,G+1} = \begin{cases} V_{ji,G+1} & if\ (randb(j) \le CR\ or\ j = rnbr(i)) \\ X_{ji,G} & if\ (randb(j) > CR\ or\ j \ne rnbr(i)) \end{cases} \quad (4)$$

เมื่อ $U_{ji,G+1}$ = เวกเตอร์ทดสอบ(Trial vector)

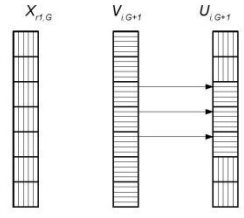
$X_{ji,G+1}$ = Mutant vector

$V_{ji,G}$ = Target vector

$randb(j)$ = การสุ่มตัวจำนวนจริงที่มีค่าระหว่าง 0 ถึง 1 ครั้งที่ j

CR = Crossover Constant มีค่าเป็นเลขจำนวนจริง ระหว่าง 0 ถึง 1

$rnbr(i)$ = ค่า Index จากการสุ่มเลือก มีค่าเป็นเลขจำนวนเต็ม ระหว่าง 0,1,…,D-1j = 0,1,…,D-1



รูปที่ 1 กระบวนการ crossover ระหว่าง target vector และ mutant vector ที่มีตัวแปรการตัดสินใจ (D) เท่ากับ 7

3.4. Selection

การคัดสรรคำตอบซึ่งเวคเตอร์ที่ให้คำตอบที่ดีกว่าจะถูกเก็บไว้ต่อไป วิธีการคือเปรียบเทียบค่าฟังก์ชันของ trial vector($U_{i,G+1}$) กับ target vector($V_{i,G+1}$) ในกรณีที่ trial vector ให้ค่า function value ที่ดีกว่า มันก็จะแทนที่ target vector ใน generation ต่อไป จากนั้น ทำซ้ำขั้นตอนที่ 2 ถึง 4 จนครบทุกชุดคำตอบ แล้วนำคำตอบชุดใหม่แทนที่คำตอบชุดเก่า ทำซ้ำกระบวนการจนพบเงื่อนไขการหยุด

4. การจำลองผลและวิเคราะห์ผลการทดลอง

ในการทดสอบอัลกอริทึมบทความนี้ใช้ระบบทดสอบขนาด 6 บัส [5] ซึ่งเป็นการต่อเชื่อมระบบแสดงดังรูปที่ 5 ใช้รีเลย์ป้องกันกระแสเกินแล้วทำการหาค่ากระแสในสายส่งที่ไหลไปยังบัสต่าง ๆ ระบบทดสอบ 6 บัส มีสายส่งทั้งหมด 7 เส้น ติดตั้งรีเลย์ป้องกันทั้งหมด 14 ตัว และจากสมการ 2 จะพบว่ามีการติดตั้งรีเลย์ทั้งหมด 14 ตัว N_{ci} = 14 และ N_{bc} = 14 ทำให้เกิดตัวแปรในสมการวัตถุประสงค์ทั้งหมด 28 ตัว เมื่อใช้วิธีวิวัฒนาการผลต่าง (Differential Evolution Algorithm) แก้ปัญหาฟังก์ชันวัตถุประสงค์เพื่อหาค่าต่ำสุดในการทำงานของรีเลย์ทั้งระบบแล้ว จะได้ผลเฉลยที่เป็นค่าเหมาะสมในการจัดลำดับการงานของรีเลย์ คือ ค่าตัวคูณเวลา และค่ากระแสทำงานแสดงในตารางที่ 1

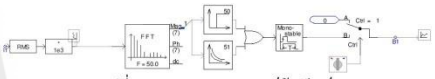
ตารางที่ 1 จำนวนเท่าเวลาปรับตั้ง และ กระแสเริ่มต้นทำงาน

หมายเลขรีเลย์	ค่าตัวคูณเวลา	ค่าปรับตั้งกระแส
1	0.2143	1.4993
2	0.3423	1.5000
3	0.4141	1.4998
4	0.3100	1.5000
5	0.1489	1.4998
6	0.5801	1.4999
7	0.3077	1.4911
8	0.3761	1.4975
9	0.0500	1.3384
10	0.2874	1.4998
11	0.4096	1.5000
12	0.4371	1.4998
13	0.4695	1.4998
14	0.3072	1.4963
ค่าฟังก์ชันวัตถุประสงค์ =		22.48

จากตารางที่ 1 จะทำให้เราทราบค่าที่เหมาะสมที่สุดในการปรับตั้งของรีเลย์ในแต่ละตัว เมื่อได้ค่าดังกล่าวแล้วจึงนำมาสร้างระบบทดสอบด้วยโปรแกรม PSCAD/EMTDC โดยแยกออกเป็น 2 ส่วน ประกอบด้วย

1. ระบบไฟฟ้ากำลังเป็นระบบทดสอบขนาด 6 บัส ดังรูปที่ 5
2. ระบบควบคุมเบรกเกอร์ด้วยรีเลย์กระแสเกิน ดังรูปที่ 2

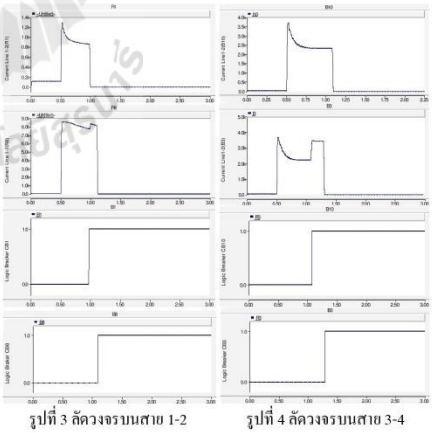
ในทดสอบการจำลองผลเวลาในการทำงานของรีเลย์เมื่อเกิดสภาวะลัดวงจรทำการจำลองการลัดวงจรบนสายส่ง1-2ใกล้บัส 1 และสายส่ง 3-4 ใกล้บัส 4 ณ เกิดการลัดวงจรที่เวลา 0.5 วินาที ดังนั้นจากข้อมูลการปรับตั้งรีเลย์ที่ได้จากการหาค่าเหมาะที่สุดในตารางที่ 1 นำมาแทนค่าปรับตั้งรีเลย์ในสมการที่ (2) จะทำให้ทราบเวลาการทำงานของรีเลย์ในแต่ละตัว จากนั้นทำการจำลองผลขณะเกิดการลัดวงจรเปรียบเทียบเวลาในการทำงานของรีเลย์จากการคำนวณกับการจำลองผลด้วยโปรแกรม PSCAD/EMTDC แสดงดังตารางที่ 2 โดยแบ่งเป็นกรณีศึกษาดังนี้



รูปที่ 2 วงจรควบคุมเบรกเกอร์ด้วยรีเลย์

กรณีที่ 1 เกิดลัดวงจรบนสายส่ง1-2 ใกล้บัส 1

รีเลย์ที่ทำหน้าที่เป็นป้องกัน คือ R8 และ R1 ทำการจำลองการลัดวงจรบนสาย1-2ใกล้บัส 1 ที่เวลา 0.5 วินาที รีเลย์ R1 ตรวจพบกระแสลัดวงจร จึงสั่งให้ CB1 ให้ตัดวงจรที่เวลา 0.97 วินาที และรีเลย์ R8 สั่งให้ CB8 ให้ตัดวงจรที่เวลา 1.08 วินาที ดังรูปที่ 3



รูปที่ 3 ลัดวงจรบนสาย 1-2 ใกล้บัส1 (CB8, CB1)

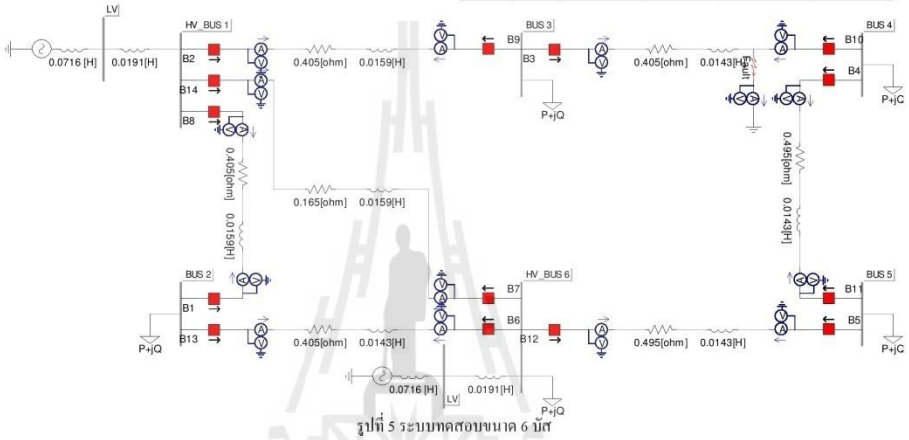
รูปที่ 4 ลัดวงจรบนสาย 3-4 ใกล้บัส4 (CB3,CB10)

กรณีที่ 2 เกิดลัดวงจรบนสายส่ง3-4 ใกล้บัส 4

รีเลย์ที่ทำหน้าที่เป็นรีเลย์ป้องกัน คือ R3 และ R10 ทำการจำลองการลัดวงจรบนสาย 3-4 ใกล้บัส 4 ที่เวลา 0.5 วินาที รีเลย์ R3 ตรวจพบกระแสลัดวงจร จึงสั่งให้ CB3 ให้ตัดวงจรที่เวลา 1.28 วินาที และรีเลย์ R10 สั่งให้ CB10 ให้ตัดวงจรที่เวลา 1.07 วินาที ดังรูปที่ 4

ตารางที่ 2 การเปรียบเทียบค่าเวลาการทำงานของรีเลย์

กรณี	รีเลย์	เวลาในการทำงานของรีเลย์		เวลาตัดวงจร	
		คำนวณ (วินาที)	จำลอง (วินาที)	คำนวณ (วินาที)	จำลอง (วินาที)
1	R1	0.41	0.47	0.91	0.97
	R8	0.63	0.58	1.13	1.08
2	R3	0.82	0.78	1.32	1.28
	R10	0.51	0.57	1.01	1.07



รูปที่ 5 ระบบทดสอบขนาด 6 บัส

7. สรุป

จากการหาค่าเหมาะสมที่สุด (TDS, PS) ด้วยวิธีการวิวัฒนาการผลต่าง (DE) ได้ผลเฉลยที่อยู่ในขอบเขตที่ตั้งไว้ เมื่อนำผลเฉลยที่ได้นั้นมาปรับตั้งค่ารีเลย์กระแสเกินแบบเวลาผกผันกับกระแส ในระบบทดสอบ 6 บัส ได้ว่า รีเลย์ R8, R1 ,R10 และ R3 สามารถทำงานตามเวลาที่ได้คำนวณไว้ล่วงหน้าโดยมีค่าเวลาแตกต่างกันเล็กน้อย ทำให้สรุปได้ว่าผลเฉลยที่ได้จากการหาค่าเหมาะที่สุดด้วยวิธีการวิวัฒนาการผลต่างนั้นมีค่าถูกต้อง

เอกสารอ้างอิง

[1] R. Thangaraj, M. Pant, K. Deep "Optimal coordination of over-current relays using modified differential evolution algorithms" *journal*, Jan 16,2010

[2] S. Chittaladakorn, A. Prempree "Optimal Design of water Distribution Piping system using different evolution algorithm" *การประชุมวิชาการวิศวกรรมโยธาแห่งชาติครั้งที่ 10* , พฤษภาคม 2548

[3] F. Razavi, H. A. Abyaneh, M. Al-Dabbagh, R. Mohammadi, H. Torkaman "A new comprehensive genetic algorithm method for optimal overcurrent relays coorfination" *ScienceDirect Electrical Power Systems* 78(2008) 713-720

[4] A.Y. Abdelaziz, H.E.A. Talaat, A.I. Nosseir, Ammar A. Hajjar, " An adaptive protection scheme for optimal coordination of overcurrent relays", Electric *Power Systems Research 61 (2002)* , page1–9

[5] S. Rodporn, T. Kulworawanichpong, A. Oonsivilai and R. Oonsivilai, "Optimal Coordination of Over-Current Relays Using Differential Evolution", *Conference on Energy Network of Thailand 7th*, 2011

Optimal Coordination of Over-Current Relays Using Differential Evolution

Somboonsup Rodporn
Power System Research Unit, School of Electrical Engineering Institute of Engineering
Suranaree University of Technology, SUT
Nakhon Ratchasima, Thailand 30000
E-mail: somboon_tun@hotmail.com

Dusit Uthitsunthorn
Power System Research Unit, School of Electrical Engineering Institute of Engineering
Suranaree University of Technology, SUT
Nakhon Ratchasima, Thailand 30000
E-mail: dusit.sut@gtmail.com

Thanatchai Kulworawanichpong
Power System Research Unit, School of Electrical Engineering Institute of Engineering
Suranaree University of Technology, SUT
Nakhon Ratchasima, Thailand 30000
E-mail: Thanatchai@gmail.com

Ratchadaporn Oonsivilai
School of Food Technology Institute of Agricultural of Technology
Suranaree University of Technology, SUT
Nakhon Ratchasima, Thailand 30000
E-mail: roonsivi@sut.ac.th

Anant Oonsivilai
Power System and Energy Research Unit, School of Electrical Engineering Institute of Engineering
Suranaree University of Technology, SUT
Nakhon Ratchasima, Thailand 30000
E-mail: anant@sut.ac.th

Abstract—In this paper, the study of optimal coordination of directional over-current relays is proposed. The relay coordination problem is non-linear. It typically consists of two groups of control variables (Time Dial Setting: TDS and Plug Setting: PS). The objective function used is the operation time of the main relay in all considered fault cases. The differential evolution (DE) is employed to solve for solutions of optimal relay coordination. It is a stochastic model based on finding the answer from some random method. This algorithm was originally developed from Genetic Algorithm (GA), but the structure is simpler. A 9-bus test system was employed as the test system. The results showed that the DE can find the best solution for optimal relay coordination problems when comparing with those of GA and BFGS.

Keywords-Differential Evolution Algorithm; Coordination over-current relay; Genetic Algorithm; Optimization

I. INTRODUCTION

Over-current relay is an important protective device in power system. It is used to protect electric power apparatus in power system when a fault or an overload occurs. Power system protection divided protected zone to protect, which can be divided into zone to protect the generator, sub-station, transmission line etc. Each zone is responsible for prevention and protections operate in separate zone of responsibility as quickly as possible from the system when fault occur in the system. This system is called primary protection system and if primary protection system fails or does not operate must have a backup protection system, which is responsible for the operations of the primary protection system to prevent damage to occur. Therefore should note that the relay position has served as primary relay or backup relay in case of a fault occurred. And to provide more effective protection, Relays must be coordinated in power systems. Determine the most appropriate setting for each relay. The problem is variables of Time dial setting (TDS) and Plug setting (PS). This paper, a mathematical method to find the optimal setting. The objective function of the problem is time operate every the relays. [1]

This paper organizes a total of six sections. Next section, Section II discusses objective function and constraints of the problem. Section III, describes differential evolution, de . Section IV is optimization of problem. Section V described solution. Section VI is conclusion.

II. OBJECTIVE FUNCTION AND CONSTRAINTS OF THE PROBLEM

A. Objective function

The optimal coordination problem of directional over current relay (DOCR) using optimization technique consists of minimizing an objective function (performance function (1)) subject to certain coordination criteria and limits on problem variables. The relay, which is supposed to operate first to clear the fault, is called the primary relay. A fault close to relay is known as the close-in fault for the relay and a fault at the other end of the line is known as a far-bus fault for the relay. Conventionally, objective function in coordination studies is constituted as the summation of operating times of all primary relays, responding to clear all close-in and far-bus faults. The objective function is defined as follows:

Minimize $$O_{bj} = \sum_{i=1}^{N_{cl}} T^{i}_{pri_cl_in} + \sum_{j=1}^{N_{far}} T^{j}_{pri_far_bus} \quad (1)$$

Where

N_{cl} is number of relays responding for close-in fault.

N_{far} is number of relays responding for far-bus fault.

$T^{i}_{pr_cl_in}$ is an operating time of primary relay for close-in fault.

$T^{j}_{pri_far_bus}$ is an operating time of primary relay for far-bus fault.

$$T^{i}_{pri_cl_in} = T^{i}_{pri_far_bus} = \frac{\alpha * TDS}{(\frac{I}{PS * CT_{pri_rating}})^{\beta} - \gamma} \quad (2)$$

$$\alpha = 0.14,\ \beta = 0.02 \text{ and } \gamma = 1$$

CT_{pri_rating} is the current flowing through each current transformer of the relay.

I is max fault of each relay.

B. Constraints

Define constraints [3]-[4] are as follows:

a) $0.05 \le TDS^{i} \le 1.1$

b) $1.25 \le PS^{i} \le 1.50$

c) $T_{primary}$ is between 0.05 to 1.00

d) $T_{backup} - T_{primary} - CTI \ge 0$

where

T_{backup} is operating time of backup relay, $T_{primary}$ is operating time of primary relay. CTI is coordination time interval, is equal to 0.3 seconds and i is the number of relay.

III. DIFFERENTIAL EVOLUTION, DE

According to the description by Storn and Price [2], the classical differential evolution algorithm can be outlined in the following.

A. Initialization

Create answer to the decision variables.(X_i) There are NP sets. The decision variables, D. The calculation of the objective function of each answer.

B. Mutation

Create tangent vectors of NP sets. (target vector, $X_{i,G}$ variables, D) 3 random vector that is unique to the tangent vectors. ($X_{r1,G}$, $X_{r2,G}$, $X_{r3,G}$) Mutation (Mutation. $V_{i,G+1}$) by using the equation.(3)

$$V_{i,G+1} = X_{r1,G} + F(X_{r2,G} - X_{r3,G}) \quad (3)$$

F is weighing factor between 0 to 2

C. Crossover

The crossover Answer varied according to Equation (4).

$$U_{ji,G+1} = (U_{1i,G+1}, U_{2i,G+1}, U_{3i,G+1}, \dots, U_{Di,G+1})$$

$$U_{ji,G+1} = \begin{cases} V_{ji,G+1} & \text{if (randb(j)} \le \text{CR or j = rnbr(i)} \\ X_{ji,G} & \text{if (randb(j)} > \text{CR or j} \ne \text{rnbr(i)} \end{cases} \quad (4)$$

where

$U_{ji,G+1}$ = Trial vector

$X_{ji,G+1}$ = Mutant vector

$V_{ji,G}$ = Target vector

randb(j) = is a randomly chosen index to ensure that at least one of the variables should be changed between 0-1.

CR = Crossover Constant There is a real number between 0-1.

rnbr(i) = is the index of the random integer value between 0 – D-1.

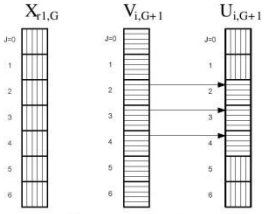


Figure 1. crossover between target vector, mutant vector decision variables (D=7)

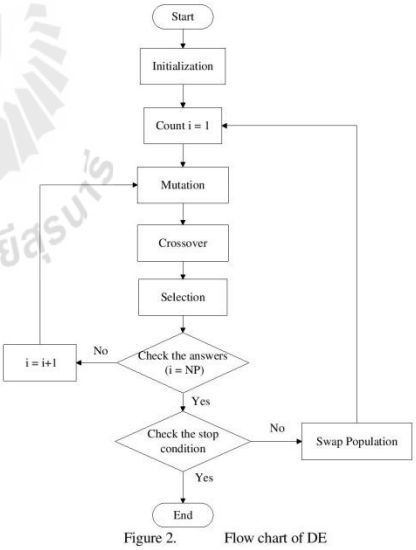


Figure 2. Flow chart of DE

D. Selection

The objective function values obtained from trial vector ($U_{i,G+1}$) and target vector ($V_{i,G+1}$) . Vector that gives a better answer than to be stored. Repeat steps 2 through 4. The conditions that must be stopped.

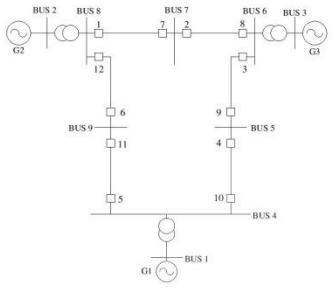


Figure 3. 9-bus test system

IV. DATAIL OF THE 9-BUS TEST SYSTEM

In Fig.3, twelve over-current relays are equipped in a 9-bus test system. In this paper, the objective function as described by (5) is selected together with five constraints given in (6)-(10). Design parameters, a-h, are pre-defined in Table I and II.

$$O_{bj} = \sum_{i=1}^{12} T^{i}_{pri_cl_in} + \sum_{j=1}^{12} T^{j}_{pri_far_bus} \quad (5)$$

$$T_{pri_cl_in} = \frac{0.14 * TDS}{(\frac{a}{PS * b})^{0.02} - 1} \quad (6)$$

$$T_{pri_far_bus} = \frac{0.14 * TDS}{(\frac{c}{PS * d})^{0.02} - 1} \quad (7)$$

The values of constants, a and c, are the maximum faults while b and d are the load currents. These parameters are of primary relay conditions. They are summarized in Table I.

$$T_{backup} - T_{primary} - CTI \geq 0 \quad (8)$$

$$T_{backup} = \frac{0.14 * TDS}{(\frac{e}{PS * f})^{0.02} - 1} \quad (9)$$

$$T_{primary} = \frac{0.14 * TDS}{(\frac{g}{PS * h})^{0.02} - 1} \quad (10)$$

The values of constants, e and g, are the maximum faults while f and h are the load currents. These parameters are of backup relay conditions. They are summarized in Table II.

TABLE I. SOME PARAMETERS OF PRIMARY RELAY

Close bus			Far bus		
Relay	a_i	b_i	Relay	c_i	d_i
R1	67.86	2.62	R7	25.93	2.62
R2	62.22	2.97	R8	54.15	2.97
R3	112.1	3.62	R9	28.7	3.62
R4	20.77	2.89	R10	28.9	2.89
R5	51.97	2.91	R11	15.45	2.91
R6	39.13	2.60	R12	28.16	2.60
R7	35.80	2.62	R1	41.63	2.62
R8	102.65	2.97	R2	38.55	2.97
R9	57.05	3.62	R3	40.75	3.62
R10	55.27	2.89	R4	14.65	2.89
R11	21.42	2.91	R5	29.72	2.91
R12	73.06	2.60	R6	20.10	2.60

TABLE II. SOME PARAMETERS OF BACKUP RELAY

Backup relay			Primary relay		
Relay	e_i	f_i	Relay	g_i	h_i
R3	40.30	3.62	R4	20.77	2.89
R10	28.50	2.89	R9	57.05	3.62
R1	41.23	2.62	R2	62.2	2.97
R8	53.70	2.97	R7	35.80	2.62
R12	27.86	2.60	R11	21.42	2.91
R5	29.35	2.91	R6	39.13	2.60
R3	27.45	3.62	R4	14.65	2.89
R12	19.56	2.60	R11	15.45	2.91
R10	19.92	2.89	R9	28.7	3.62
R1	25.16	2.62	R2	38.55	2.97
R8	38.70	2.97	R7	25.93	2.62
R5	14.72	2.91	R6	20.10	2.60

V. SIMULATION RESULTS

In this paper, the 9-bus test system was employed. The results are collected and evaluated by calculating the minimum, maximum, average and standard deviation of those results obtained from 30 trials. To assess the effectiveness of the proposed method, genetic algorithms (GAs) and Newton's search method using BFGS updating formula are used. The obtained results of optimal coordination relays from each method can be illustrated in Table III and also in Figs.4 – 6, as the convergence curves. As a result, the average objective function found by the DE is 6.8493, while they are 6.8582 and 7.9283 for GA and BFGS, respectively. When comparing the execution time, The DE spent only 358.25 s, while the GA and the BFGS used 631.21 s and 12.58 s respectively. The Table IV show operating time of relay. Which was calculated using TDS and PS of DE. The solution shows that the primary relay and backup relay to the constraints (8).

VI. CONCLUSIONS

In this paper, some efficient search method named differential evolution (DE) is proposed for solving the problem of optimal coordination of over-current relays. The main outstanding feature of DE different from GA is a simple structure. This can reduce considerable execution time to find the solutions. In this paper, with the 9-bus test system the DE can find the best average solution of 30 trials when compares to those

obtained by using GA and BFGS. This confirmed the effectiveness of the DE algorithm to solve the problem of optimal coordination of over-current relays.

TABLE III. OPTIMAL RESULTS

Relay	DE		GA		BFGS	
	TDS	PS	TDS	PS	TDS	PS
R1	0.1466	1.4567	0.1524	1.2520	0.1544	1.2684
R2	0.0500	1.2507	0.0470	1.2495	0.0492	1.2680
R3	0.1569	1.2826	0.1450	1.2527	0.1793	1.2525
R4	0.0500	1.2514	0.0398	1.2499	0.0481	1.3145
R5	0.1324	1.2661	0.1309	1.2495	0.1725	1.2905
R6	0.0500	1.2541	0.0472	1.2500	0.0682	1.2961
R7	0.0500	1.2542	0.0434	1.2498	0.0509	1.2782
R8	0.1728	1.2706	0.1659	1.2511	0.2340	1.2574
R9	0.0500	1.2521	0.0460	1.2507	0.0547	1.2910
R10	0.1196	1.4931	0.1276	1.2495	0.1633	1.2905
R11	0.0500	1.2514	0.0445	1.2499	0.0484	1.3222
R12	0.1424	1.4851	0.1450	1.3106	0.1546	1.2556
Min	6.8375		6.6417		7.9283	
Max	6.9757		6.9232		7.9283	
$\overline{X}$	6.8493		6.8582		7.9283	
S.D.	0.0241		0.0484		0	
Time	358.25		631.21		12.58	

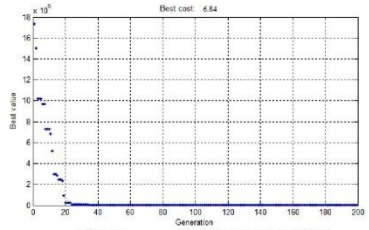


Figure 4. Objective function of DE

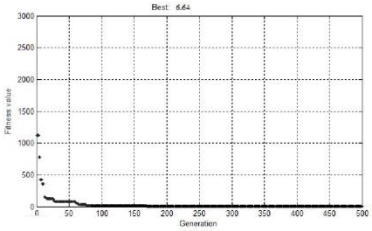


Figure 5. Objective function of GA

TABLE IV. TIME OPERATING OF RELAYS

Line		Relay		Fault Point(% length Line)									
				0%		25%		50%		75%		100%	
		Primary	Backup	Primary	Backup	Primary	Backup	Primary	Backup	Primary	Backup	Primary	Backup
4-5	close	R10	-	0.3202	-	0.3527	-	0.3799	-	0.4073	-	0.4323	-
	far	R4	R3	0.2470	0.6073	0.2309	0.5577	0.2187	0.5330	0.2069	0.5104	0.1969	0.4975
4-9	close	R5	-	0.3410	-	0.3644	-	0.3883	-	0.4119	-	0.4347	-
	far	R11	R12	0.2387	0.6044	0.2257	0.5723	0.2139	0.5434	0.2034	0.5175	0.1940	0.4945
5-6	close	R9	R10	0.1348	0.4355	0.1461	0.4902	0.1586	0.5521	0.1725	0.6283	0.1862	0.7073
	far	R3	-	0.4949	-	0.4507	-	0.4126	-	0.3738	-	0.3341	-
6-7	close	R8	-	0.3540	-	0.3773	-	0.4002	-	0.4225	-	0.4422	-
	far	R2	R1	0.1462	0.5339	0.1389	0.4982	0.1326	0.4703	0.1265	0.4443	0.1208	0.4210
7-8	close	R7	R8	0.1431	0.4436	0.1484	0.4610	0.1541	0.4779	0.1602	0.4954	0.1659	0.5077
	far	R1	-	0.4193	-	0.4010	-	0.3828	-	0.3645	-	0.3464	-
8-9	close	R12	-	0.3292	-	0.3694	-	0.4099	-	0.4509	-	0.4918	-
	far	R6	R5	0.1890	0.6597	0.1735	-	0.1604	0.5315	0.1484	0.4816	0.1374	0.4374

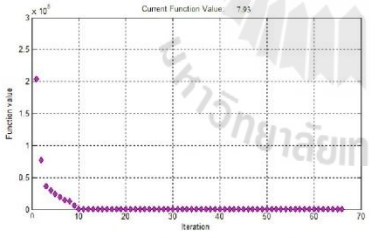


Figure 6. Objective function of BFGS

REFERENCES

[1] R. Thangaraj, M. Pant, K. Deep "Optimal coordination of over-current relays using modified differential evolution algorithms" journal, Jan 16,2010

[2] R. Storn and K. Price, "Differential evolution – A simple and efficient heruistic for global optimization over continuous spaces", Journal of Global Optimization, vol.11, pp.341-359, 1997.

[3] F. Razavi, H. A. Abyaneh, M. Al-Dabbagh, R. Mohammadi, H. Torkaman "A new comprehensive genetic algorithm method for optimal overcurrent relays coorfination" ScienceDirect Electrical Power Systems 78(2008) 713-720

[4] A.Y. Abdelaziz, H.E.A. Talaat, A.I. Nosseir, Ammar A. Hajjar, " An adaptive protection scheme for optimal coordination of overcurrent relays", Electric Power Systems Research 61 (2002) , page1–9

[5] R. M. Chabanloo, H. A. Abyaneh, S. S. H.Kamanger "A New Genetic Algorithm Method for Optimal Coordination of Overcurrent and Distance Relays Overcurrent Relays", 2nd IEEE International Conference on Power and Energy, December 1-3, 2008

ประวัติผู้เขียน

นายสมบูรณ์ทรัพย์ รอดพร เกิดเมื่อวันที่ 27 มกราคม พ.ศ. 2530 ที่จังหวัดปราจีนบุรี สำเร็จการศึกษาระดับมัธยมศึกษาตอนปลาย จากโรงเรียนเบญจมราชรังสฤษฎิ์จังหวัดฉะเชิงเทรา และสำเร็จการศึกษาระดับปริญญาวิศวกรรมศาสตรบัณฑิต (วิศวกรรมไฟฟ้า) จากมหาวิทยาลัยเทคโนโลยีสุรนารี จังหวัดนครราชสีมา เมื่อ พ.ศ. 2552 และเข้าศึกษาต่อในระดับปริญญาโท สาขาวิชาวิศวกรรมไฟฟ้า ณ สถาบันเดิม ขณะศึกษาได้เป็นผู้สอนปฏิบัติการของสาขาวิชาวิศวกรรมไฟฟ้า สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี จำนวน 3 รายวิชา ได้แก่ 1) ปฏิบัติการระบบไฟฟ้ากำลัง 1 2) ปฏิบัติการระบบไฟฟ้ากำลัง 2 3) ปฏิบัติการ Micro controller Lab โดยมีความสนใจในด้านการไหลของกำลังไฟฟ้าเหมาะที่สุด และมีผลงานทางวิชาการที่ได้รับการตีพิมพ์เผยแพร่ในขณะศึกษาดัง ภาคผนวก ง

มหาวิทยาลัยเทคโนโลยีสุรนารี