

การจัดเส้นทางของโครงข่ายให้ดีที่สุดโดยวิธีฮอปฟิลด์เน็ต

ROUTING OPTIMIZATION OF TELEPHONE NETWORK
BY HOPFIELD NET

ฉัตรชัย วัฒนพันธุ์

CHATCHAI WATTHANAPHAN

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2547

ISBN 974-9703-09-1

การจัดเส้นทางของโครงข่ายให้ดีที่สุดโดยวิธีฮอปฟิลด์เน็ต

ROUTING OPTIMIZATION OF TELEPHONE NETWORK
BY HOPFIELD NET

ฉัตรชัย วัฒนพันธุ์

CHATCHAI WATTHANAPHAN

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2547

ISBN 974-9708-09-1

**ROUTING OPTIMIZATION OF TELEPHONE NETWORK
BY HOPFIELD NET**

CHATCHAI WATTHANAPHAN

**A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF ENGINEERING IN ELECTRICAL ENGINEERING
SCHOOL OF GRADUATE STUDIES
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

2004

ISBN 974-9708-09-1

COPYRIGHT 2004

SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

หัวข้อวิทยานิพนธ์	การจัดเส้นทางของโครงข่ายให้ดีที่สุดโดยวิธีฮอปฟิลด์เน็ต
นักศึกษา	นายฉัตรชัย วัฒนพันธุ์
รหัสนักศึกษา	43061128
ปริญญา	วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชา	วิศวกรรมไฟฟ้า
พ.ศ.	2547
อาจารย์ผู้ควบคุมวิทยานิพนธ์	รศ.ดร.ถวิล พึ่งมา
อาจารย์ผู้ควบคุมวิทยานิพนธ์ร่วม	ศ.มณูญ สุขเกษม

บทคัดย่อ

วิทยานิพนธ์นี้ศึกษาเกี่ยวกับการประยุกต์ใช้โครงข่ายประสาทเทียม (Neural Network) ชนิดค่าที่เหมาะสม (Optimization) ที่ชื่อฮอปฟิลด์เน็ต (Hopfield Net) ในการจัดเส้นทางของโครงข่ายโทรศัพท์ให้ดีที่สุด ตามเงื่อนไขที่ได้สร้างขึ้น โดยทำการทดลองจำลองโครงข่ายโทรศัพท์ที่ความหนาแน่นของปริมาณกราฟฟิคที่สถานะกราฟฟิคปกติ กราฟฟิคเบาบาง และกราฟฟิคหนาแน่น รวมทั้งในสถานะที่เส้นทางบางเส้นทางในโครงข่ายเกิดล้มเหลวไม่สามารถใช้งานได้ จากผลการทดลองแสดงให้เห็นว่า การเรียกตัดขาดด้วยวิธีฮอปฟิลด์เน็ตมีค่าน้อยลง 30% เมื่อเปรียบเทียบกับวิธีการแบบกราฟฟิคล้น (Overflow) และแบบแบ่งโหลด (Load sharing)

Thesis Title	Routing Optimization of Telephone Network by Hopfield Net
Student	Mr. Chatchai Watthanaphan
Student ID	43061128
Degree	Master of Engineering
Programme	Electrical Engineering
Year	2004
Thesis Advisor	Assoc. Prof. Dr.Tawil Paungma
Thesis Co-Advisor	Prof. Manoon Sukkasem

ABSTRACT

This thesis presents an application of the neural network with the optimization called “Hopfield Net” in routing of telephone network system. This is able to method optimize the routing with specific conditions. In the simulation, it consist of 3 cases from low, normal and high traffic densities, offered to the network. The results from the simulation show that the congestion in by Hopfield net method is 30% lower when compared with overflow method and load sharing method.

กิตติกรรมประกาศ

การจัดทำวิทยานิพนธ์ฉบับนี้ สามารถสำเร็จลุล่วงได้ด้วยดีด้วยความอนุเคราะห์ของ รศ. ดร. ถวิต พึ่งมา ศ. มนูญ สุขเกษม ผศ.ดร. พิเชฐ ม่วงนวล ข้าพเจ้ารู้สึกซาบซึ้งในความอนุเคราะห์และคำแนะนำอันเป็นประโยชน์ในการวิจัยนี้จากทุกๆ ท่าน และขอกราบขอบคุณเป็นอย่างสูง

ขอกราบขอบพระคุณ คุณพ่อ คุณแม่ และขอขอบคุณพี่ๆ น้องๆ ทุกคน รวมทั้งภรรยา ที่ให้โอกาสและสนับสนุนให้กำลังใจ จนข้าพเจ้าได้มีโอกาสเล่าเรียนจนถึงระดับปริญญาโทและทำวิทยานิพนธ์จนสำเร็จลุล่วงไปด้วยดี

ขอขอบคุณ นายวินัย สังฆวิจิตร และพนักงาน ส่วนปฏิบัติการและบำรุงรักษาระบบชุมสาย ฝ่ายโครงข่ายโทรคมนาคม และพนักงานโทรศัพท์จังหวัดอุทัยธานี บริษัท ทศท คอร์ปอเรชั่น จำกัด (มหาชน) ทุกๆ ท่าน ที่ให้การสนับสนุน และให้ข้อมูลในการทำวิทยานิพนธ์เป็นอย่างดีระหว่างการศึกษา

ขอขอบคุณ เพื่อนๆ พี่ๆ และน้องๆ ในห้อง T-201 ทุกคนที่ให้ความช่วยเหลือแนะนำต่างๆ พร้อมทั้งช่วยตรวจทาน และแก้ไขข้อผิดพลาดต่างๆ จนวิทยานิพนธ์ฉบับนี้สำเร็จสมบูรณ์ยิ่งขึ้น

คุณค่าและประโยชน์อันพึงมีจากวิทยานิพนธ์ฉบับนี้ ผู้วิจัยขอบแต่ คุณพ่อ คุณแม่ และผู้มีพระคุณทุกท่าน

ฉัตรชัย วัฒนพันธุ์

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VI
สารบัญรูป.....	VII
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์ของการศึกษา.....	1
1.3 เป้าหมายและขอบเขตวิทยานิพนธ์.....	1
1.4 ขั้นตอนและวิธีดำเนินการ.....	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	2
1.6 รายละเอียดของวิทยานิพนธ์.....	2
บทที่ 2 การจัดเส้นทางของโครงข่ายโทรศัพท์.....	4
2.1 ทฤษฎีกราฟฟิค.....	4
2.2 อัตราการเรียกไม่ติด.....	6
2.3 ประสิทธิภาพ.....	7
2.4 ชั้นของบริการ.....	7
2.5 การจัดเส้นทางแบบกราฟฟิคสั้น.....	8
2.6 การจัดเส้นทางแบบแบ่งโหลด.....	10
บทที่ 3 ทฤษฎีโครงข่ายประสาทเทียม.....	16
3.1 แบบจำลองของนิวรอน.....	16
3.2 ชนิดของฟังก์ชันแอคติเวชัน.....	18
3.3 กระบวนการเรียนรู้และกฎการเรียนรู้ของโครงข่ายประสาท.....	22
3.3.1 กฎการเรียนรู้.....	23
3.3.2 รูปแบบการเรียนรู้.....	30

สารบัญ (ต่อ)

	หน้า
บทที่ 4 การจัดเส้นทางของโครงข่ายโทรศัพท์ที่ใช้ฮอปฟิลด์เน็ต.....	32
4.1 ตัวอย่างการประยุกต์ใช้โครงข่ายประสาทของฮอปฟิลด์.....	32
4.2 ปัญหา Traveling Salesman Problem (TSP).....	32
4.3 การจัดเส้นทางของโครงข่ายโทรศัพท์ที่ใช้ฮอปฟิลด์เน็ต	36
4.4 การประยุกต์ใช้โครงข่ายใช้ฮอปฟิลด์เน็ต.....	37
4.5 กระบวนการเรียนรู้ของโครงข่ายประสาทฮอปฟิลด์เน็ต.....	39
4.6 การทดสอบผลการคำนวณ โดยใช้ฮอปฟิลด์เน็ต.....	41
บทที่ 5 การทดลองและผลการทดลอง.....	44
5.1 ขั้นตอนในการทดลอง.....	44
5.1.1 โมเดลของโครงข่ายที่ใช้ในการทดสอบ.....	44
5.2 การทดสอบการกำหนดเส้นทาง.....	45
5.3 การทดสอบและผลการทดสอบขณะที่สภาวะโครงข่ายปกติ.....	46
5.4 การทดสอบและผลการทดสอบขณะที่สภาวะโครงข่ายขัดข้อง.....	52
5.4.1 ผลการทดสอบขณะที่โครงข่ายขัดข้องในเส้นทาง 1-3.....	53
5.4.2 ผลการทดสอบขณะที่โครงข่ายขัดข้องในเส้นทาง 3-5.....	56
บทที่ 6 บทสรุป.....	60
6.1 สรุปผลการทดลอง.....	60
6.2 ปัญหาและข้อเสนอแนะ.....	60
เอกสารอ้างอิง.....	61
ภาคผนวก ก โปรแกรมการจัดเส้นทางให้ดีที่สุดโดยวิธีฮอปฟิลด์เน็ต.....	63
ภาคผนวก ข ผลงานวิจัยในระหว่างการศึกษาที่ได้รับการตีพิมพ์เผยแพร่.....	81
ผลงานทางวิชาการที่ได้รับการตีพิมพ์.....	89
ประวัติผู้เขียน.....	90

สารบัญตาราง

ตารางที่	หน้า
3.1 เปรียบเทียบระหว่างนิเวศธรรมชาติและแบบจำลองที่ประดิษฐ์ขึ้น.....	18
4.1 ค่าความน่าจะเป็นของการติดขัดของโครงข่าย รูปที่ 4.4.....	42
4.2 ค่าคงที่ ที่ใช้ในการทดสอบการคำนวณ.....	42
4.3 ผลการทดสอบการคำนวณ.....	43
5.1 ค่าความหนาแน่นของปริมาณกราฟฟิกของแต่ละเส้นทาง.....	45
5.2 ค่าความน่าจะเป็นของการติดขัดของแต่ละเส้นทาง.....	45
5.3 ผลการทดสอบขณะที่โครงข่ายปกติกราฟฟิกปกติและเบาบาง.....	48
5.4 ผลการทดสอบขณะที่โครงข่ายปกติกราฟฟิกหนาแน่น.....	50
5.5 ผลการทดลองขณะที่โครงข่ายปกติ (กราฟฟิกปกติและเบาบางกับกราฟฟิกหนาแน่น).....	52
5.6 ค่าความน่าจะเป็นของการติดขัดรวมทั้งเส้นทางขณะที่โครงข่ายปกติ.....	52
5.7 ผลการทดลองขณะที่โครงข่ายขัดข้องในเส้นทาง 1-3.....	54
5.8 ผลการทดลองขณะที่โครงข่ายขัดข้องในเส้นทาง 1-3.....	55
5.9 ค่าความน่าจะเป็นของการติดขัดรวมทั้งเส้นทางขณะที่โครงข่ายขัดข้องในเส้นทาง 1-3.....	55
5.10 ผลการทดลองขณะที่โครงข่ายขัดข้องในเส้นทาง 3-5.....	57
5.11 ผลการทดลองขณะที่โครงข่ายขัดข้องในเส้นทาง 3-5.....	58
5.12 ค่าความน่าจะเป็นของการติดขัดรวมทั้งเส้นทางขณะที่โครงข่ายขัดข้องในเส้นทาง 3-5.....	59

สารบัญรูป

รูปที่	หน้า
2.1 การเรียกโทรศัพท์ในท้องถิ่น.....	4
2.2 การเรียกโทรศัพท์ประจำวัน และชั่วโมงใช้ (โทรศัพท์) สูงสุด.....	5
2.3 ทราฟฟิกต่างๆ	5
2.4 Overflow system	8
2.5 แบบจำลองตัวอย่างการจัดเส้นทางแบบทราฟฟิกสั้น.....	9
2.6 ผลการจัดเส้นทางแบบทราฟฟิกสั้น.....	10
2.7 การเลือกใช้ Link Set	11
2.8 การเลือกใช้ CIC	12
2.9 การเลือกใช้ Signalling Links.....	12
2.10 เส้นทางสัญญาณผ่านปกติหรือทางตรงและทางเลือก.....	13
2.11 แบบจำลองตัวอย่างจัดเส้นทางแบบแบ่งโหลด.....	14
2.12 ผลการจัดเส้นทางแบบแบ่งโหลด.....	14
3.1 การจำลองโครงข่ายประสาทในรูปแบบคณิตศาสตร์.....	16
3.2 ผลของค่าขีดจำกัด.....	17
3.3 ฟังก์ชันเทรสโฮลด์.....	18
3.4 ฟังก์ชันเชิงเส้นอิมตัว.....	19
3.5 ฟังก์ชันลอจิสติกที่ค่าความชันต่างๆ.....	20
3.6 ความสัมพันธ์แบบต่างๆ ของโครงข่ายประสาท.....	23
3.7 รูปแบบการเรียนรู้ (ก) แบบมีการควบคุม (ข) แบบไม่มีการควบคุม.....	30
4.1 (ก) ตัวอย่างคำตอบของปัญหา TSP 5 เมือง (ข) การแทนคำตอบด้วยเมตริกซ์.....	33
4.2 ไดอะแกรมของโครงข่ายการจัดเส้นทางให้เหมาะสมโดยใช้ Hopfield Net.....	37
4.3 ไดอะแกรมของกระบวนการเรียนรู้ของโครงข่ายการจัดเส้นทางให้เหมาะสม.....	40
4.4 ตัวอย่างโครงข่ายโทรศัพท์ที่มีจำนวนโหนดในโครงข่าย 4 โหนด.....	41
4.5 เอาท์พุทของนิวโรลเน็ตเวิร์คเมื่อต้องการเรียกจากโหนดที่ 1 ไปยังโหนดที่ 2.....	42
4.6 พลังงานของนิวโรลเน็ตเวิร์ค.....	43
5.1 โมเดลของโครงข่ายที่ใช้ในการทดสอบ.....	44
5.2 รูปแบบโปรแกรมที่ใช้ในการทดสอบโครงข่ายประสาทฮอปฟิลด์เน็ต.....	46

สารบัญรูป (ต่อ)

รูปที่	หน้า
5.3 เอ้าท์พุทของนิวรอลเน็ตเวิร์กแสดงผลการจัดเส้นทางโดยวิธีฮอปไฟลด์เน็ตขณะที่โครงข่ายปกติ ทราฟฟิกปกติและเบาบาง.....	47
5.4 พลังงานของนิวรอลเน็ตเวิร์กขณะที่โครงข่ายปกติทราฟฟิกปกติและเบาบาง.....	47
5.5 ผลการจัดเส้นทางแบบ Overflow ขณะที่โครงข่ายปกติ ทราฟฟิกปกติและเบาบาง.....	48
5.6 ผลการจัดเส้นทางแบบ Load Sharing ขณะที่โครงข่ายปกติ ทราฟฟิกปกติ.....	49
5.7 เอ้าท์พุทของนิวรอลเน็ตเวิร์กแสดงผลการจัดเส้นทางโดยวิธีฮอปไฟลด์เน็ตขณะที่โครงข่ายปกติ ทราฟฟิกหนาแน่น.....	49
5.8 พลังงานของนิวรอลเน็ตเวิร์กขณะที่โครงข่ายปกติทราฟฟิกหนาแน่น.....	50
5.9 ผลการจัดเส้นทางแบบ Overflow ขณะที่โครงข่ายปกติ ทราฟฟิกหนาแน่น.....	51
5.10 ผลการจัดเส้นทางแบบ Load Sharing ขณะที่โครงข่ายปกติ ทราฟฟิกหนาแน่น.....	51
5.11 เอ้าท์พุทของนิวรอลเน็ตเวิร์กแสดงผลการจัดเส้นทางขณะที่โครงข่ายขัดข้องในเส้นทาง 1-3	53
5.12 พลังงานของนิวรอลเน็ตเวิร์กขณะที่โครงข่ายขัดข้องในเส้นทาง 1-3.....	53
5.13 ผลการจัดเส้นทางแบบ Overflow ขณะที่โครงข่ายขัดข้องในเส้นทาง 1-3.....	54
5.14 ผลการจัดเส้นทางแบบ Load Sharing ขณะที่โครงข่ายขัดข้องในเส้นทาง 1-3.....	55
5.15 เอ้าท์พุทของนิวรอลเน็ตเวิร์กแสดงผลการจัดเส้นทางขณะที่โครงข่ายขัดข้องในเส้นทาง 3-5	56
5.16 พลังงานของนิวรอลเน็ตเวิร์กขณะที่โครงข่ายขัดข้องในเส้นทาง 3-5.....	56
5.17 ผลการจัดเส้นทางแบบ Overflow ขณะที่โครงข่ายขัดข้องในเส้นทาง 3-5.....	57
5.18 ผลการจัดเส้นทางแบบ Load Sharing ขณะที่โครงข่ายขัดข้องในเส้นทาง 3-5.....	58
5.19 ผลการเปรียบเทียบค่าของการเรียกติดขัดทั้ง 3 วิธี.....	59

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันการจัดเส้นทางของโครงข่ายโทรศัพท์ ที่ใช้วิธีการจัดเส้นทาง แบบทราฟฟิคล้น (Overflow) เส้นทางของการเรียก จะไปยังเส้นทางแรกก่อน เมื่อวงจรเส้นทางแรกเต็ม ทราฟฟิคล้น จะมายังเส้นทางเพื่อเลือก หรือไม่ก็จะเป็นลักษณะของการแบ่งโหลด (Load Sharing) ดังนั้นการเลือกเส้นทางในขณะที่มีเส้นทางหลายเส้นทางต่อเชื่อมถึงกันระหว่างโหนดต้นทางไปยังโหนดปลายทางนั้น จึงไม่จำเป็นต้องกำหนดเส้นทางให้ว่าถ้าวงจรเส้นทางแรกเต็ม แล้วทราฟฟิคล้นจึงไปอีกเส้นทาง หรือจัดทราฟฟิคของแต่ละเส้นทางให้มีปริมาณทราฟฟิคเท่าๆ กัน ซึ่งเส้นทางที่เหลืออยู่ อาจจะเป็นเส้นทางที่เหมาะสมกว่า ดังนั้นจึงได้ศึกษาการจัดเส้นทางโดยใช้ฮอปฟิลด์เน็ต (Hopfield Net) ซึ่งการจัดเส้นทางแบบนี้ จะได้เส้นทางที่ดีที่สุด ซึ่งจะคำนึงถึงสถานะของปริมาณทราฟฟิคที่มีการเปลี่ยนแปลง และค่าความน่าจะเป็นของการติดขัดไม่ให้เกิน 0.01 หรือให้อยู่ในเกณฑ์มาตรฐานที่สามารถยอมรับได้ รวมทั้งในกรณีที่มีบางเส้นทางที่ใช้ในการติดต่อสื่อสารเกิดล้มเหลว ไม่สามารถใช้งานได้ โดยจะไม่คำนึงถึงวิธีการเดิม วิธีการใหม่ที่นำเสนอนี้ จะเลือกเส้นทางที่ดีที่สุดสำหรับการใช้ในการติดต่อสื่อสารตามเงื่อนไขที่ได้สร้างขึ้น

1.2 วัตถุประสงค์ของการศึกษา

เพื่อศึกษาการจัดเส้นทางของโครงข่ายโทรศัพท์ ในแบบใหม่ ที่ไม่ใช่วิธีการเดิม อีกทั้งยังได้ศึกษาค้นคว้า ออกแบบ และเขียนโปรแกรม การคำนวณในการจัดเส้นทางของโครงข่ายโทรศัพท์ ให้ดีที่สุดโดยใช้วิธีการใหม่ ซึ่งเส้นทางที่ได้ถูกเลือกนั้นจะเป็นเส้นทางเชื่อมโยงที่มีค่าความน่าจะเป็นของการติดขัดน้อยที่สุด โดยการใช้อัลกอริทึมของนิวโรลเน็ตเวิร์กในการคำนวณ โดยคาดหวังไว้ว่า จะได้เส้นทางที่ดีที่สุดตามสมการเงื่อนไขที่ได้กำหนดขึ้น

1.3 เป้าหมายและขอบเขตของวิทยานิพนธ์

1. เพื่อศึกษาวิธีการจัดเส้นทางของโครงข่ายโทรศัพท์ โดยใช้นิวโรลเน็ตเวิร์ก ชนิดฮอปฟิลด์เน็ต ซึ่งตั้งเป้าหมายไว้ว่า เส้นทางที่ได้ถูกเลือกจากวิธีการที่นำเสนอนี้ จะเป็นเส้นทางที่ดีที่สุด ที่จะนำไปใช้ในการติดต่อสื่อสาร
2. หาข้อจำกัดในการใช้งานโปรแกรมที่ได้เขียนขึ้น พร้อมเสนอแนวทาง และสาเหตุในการแก้ไขข้อจำกัดของอัลกอริทึมที่เขียนขึ้น

1.4 ขั้นตอนและวิธีดำเนินการ

- 1.ศึกษาปัญหาที่เกิดจากการจัดเส้นทางของโครงข่ายโทรศัพท์
- 2.ศึกษาทฤษฎีการจัดเส้นทางของโครงข่ายโทรศัพท์
- 3.ศึกษาการแก้ปัญหาของการจัดเส้นทางของโครงข่ายโทรศัพท์โดยวิธีฮอปฟิลด์เน็ตเวอร์ก
- 4.เขียนโปรแกรมจำลองแบบ
- 5.ทดสอบโปรแกรมและทำการทดลอง
- 6.ประเมินผลและสรุป
- 7.เขียนวิทยานิพนธ์

1.5 ประโยชน์ที่คาดว่าจะได้รับ

- 1.เรียนรู้ถึงลักษณะ และผลลัพธ์ของ การคำนวณ แบบฮอปฟิลด์เน็ต (Hopfield Net)
- 2.สามารถขยายผลที่ได้รับไปใช้งานทางด้าน วิศวกรรมระบบโทรคมนาคมได้
- 3.เป็นแนวทางในการวิจัย และพัฒนาการจัดเส้นทางเชื่อมโยงของโครงข่ายให้ดีที่สุด นอกเหนือจากวิธีการเดิมๆ

1.6 รายละเอียดของวิทยานิพนธ์

วิทยานิพนธ์ฉบับนี้ แบ่งเป็น 5 บท กล่าวคือ

บทที่ 1 บทนำ กล่าวถึงความเป็นมาและความสำคัญของปัญหา วัตถุประสงค์ของการศึกษาเป้าหมายและขอบเขตวิทยานิพนธ์ ขั้นตอนและวิธีดำเนินการ และประโยชน์ที่คาดว่าจะได้รับ

บทที่ 2 การจัดเส้นทางของโครงข่ายโทรศัพท์ กล่าวถึงทฤษฎีการจัดเส้นทางของโครงข่ายโทรศัพท์ การจัดเส้นทางแบบกราฟฟิคสั้น การจัดเส้นทางแบบแบ่งโหลด

บทที่ 3 ทฤษฎีโครงข่ายประสาทเทียม แบบจำลองนิรอรอล ชนิดของฟังก์ชัน กระบวนการเรียนรู้ และกฎการเรียนรู้ของโครงข่ายประสาท

บทที่ 4 การจัดเส้นทางของโครงข่ายโทรศัพท์โดยใช้ฮอปฟิลด์เน็ต ตัวอย่างการประยุกต์ใช้โครงข่ายประสาทของฮอปฟิลด์ ปัญหา TSP (Traveling Salesman Problem) การประยุกต์ใช้โครงข่ายฮอปฟิลด์เน็ต กระบวนการเรียนรู้ของโครงข่ายประสาทฮอปฟิลด์เน็ต และการทดสอบผลการคำนวณโดยใช้ฮอปฟิลด์เน็ต

บทที่ 5 การทดลองและผลการทดลอง การทดลองการจัดเส้นทางของโครงข่ายโทรศัพท์ การทดลองและผลการทดลองที่สถานะโครงข่ายปกติ การทดลองและผลการทดลองที่สถานะโครงข่ายขัดข้อง

บทที่ 6 บทสรุป กล่าวถึงข้อสรุปของผลการทดลองในการจัดเส้นทางของโครงข่ายโทรศัพท์ให้ดีที่สุดในวิธีฮอปไฟลด์เน็ต ตามเงื่อนไขที่ได้สร้างขึ้น นอกจากนี้ได้กล่าวถึงข้อเสนอแนะในการพัฒนาเพื่อปรับปรุงข้อจำกัดในวิธีการที่ได้เสนอขึ้น

ภาคผนวก ได้เสนอการคำนวณโดยเพิ่มจำนวนโหนดให้มากขึ้น พร้อมผลการทดสอบการคำนวณ ซึ่งยังคงแสดงให้เห็นว่าการจัดเส้นทางโดยวิธีฮอปไฟลด์เน็ต ยังคงให้เส้นทางที่ดีที่สุด เมื่อทำการเปรียบเทียบกับวิธีอื่นๆ

บทที่ 2

การจัดเส้นทางของโครงข่ายโทรศัพท์

2.1 ทฤษฎีกราฟฟิก

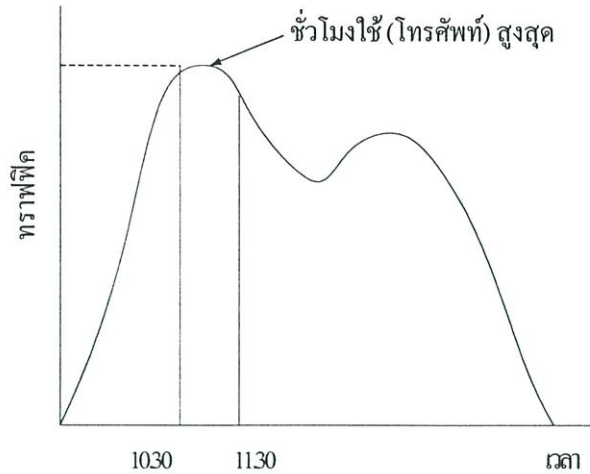
ทฤษฎีกราฟฟิกโทรศัพท์ (Telephone traffic theory) ว่าด้วยการคำนวณอุปกรณ์ที่จำเป็น เพื่อให้ชุมสายโทรศัพท์ท้องถิ่นจัดหาอุปกรณ์ดังกล่าวมาติดตั้งเพิ่มอย่างเหมาะสม ไว้ตอบสนองการเรียกหรือกราฟฟิกขอบริการ ด้วยความน่าจะเป็นของการติดขัดน้อยที่สุดในชั่วโมงใช้โทรศัพท์สูงสุด



รูปที่ 2.1 การเรียกโทรศัพท์ในท้องถิ่น

จากรูปที่ 2.1 เป็นการเรียกโทรศัพท์ในท้องถิ่น ระหว่างผู้เรียก ผ่านชุมสายท้องถิ่นอัตโนมัติ (A) ไปยังผู้รับ ซึ่งการเรียก (Call) คือการต่อสำเร็จระหว่างผู้เรียกและผู้รับ ส่วนการพยายามเรียก (Call attempt) คือการพยายามเรียกที่ถูกทำให้เกิดโดยผู้ใช้เสมอ ถ้าการพยายามเรียก ได้คำตอบจาก สถานีผู้รับ เรียกว่า การพยายามเรียกสำเร็จ (Call attempt complete) ส่วนการขอต่อซึ่งถูกปฏิเสธด้วย สัญญาณไม่ว่าง (Busy tone) เนื่องจากโครงข่ายคับคั่ง คือ การเรียกไม่ได้ (Lost call) ซึ่งการเรียกไม่ได้ จะไปกล่าวถึงในรายละเอียดความน่าจะเป็นของการติดขัด ซึ่งลงตัวกับอัตราการเรียกไม่ติด

ชั่วโมงใช้ (โทรศัพท์) สูงสุด (Busy hour) คือชั่วโมงใช้ (โทรศัพท์) สูงสุดที่อ้างถึงกราฟฟิก หรือ จำนวนของการพยายามเรียก และคือช่วงเวลาต่อเนื่อง 1 ชั่วโมง ที่มีปริมาณกราฟฟิกได้บริการ หรือการเรียกได้บริการสูงสุด

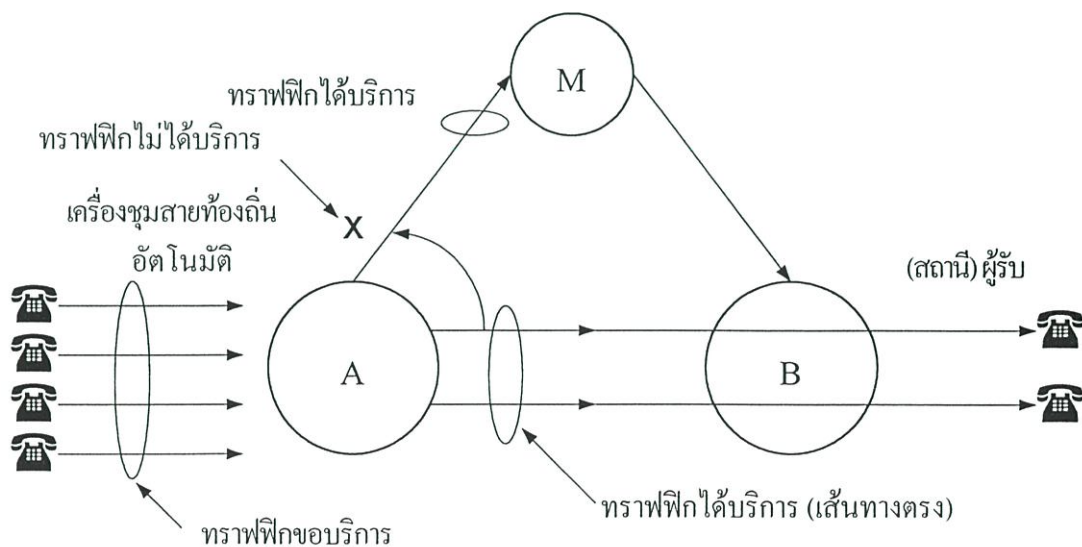


รูปที่ 2.2 การเรียกโทรศัพท์ประจำวัน และชั่วโมงใช้ (โทรศัพท์) สูงสุด

จากรูปที่ 2.2 เป็นการเรียกโทรศัพท์ประจำวัน และชั่วโมงใช้โทรศัพท์สูงสุด อยู่ที่ช่วงเวลา 10.30 น.ถึงเวลา 11.30 น.

โทรศัพท์ขอบริการ (offered traffic) คือโทรศัพท์ที่ควรจะถูกให้บริการโดยอุปกรณ์รวมต่างๆ มากเพียงพอที่จะรองรับโทรศัพท์ที่ไม่ได้จำกัดด้วยขนาดอุปกรณ์รวมจำกัด

โทรศัพท์ได้รับการ (carried traffic) คือโทรศัพท์ที่ได้รับการผ่านอุปกรณ์รวมต่างๆ ในชั่วโมงใช้ (โทรศัพท์) สูงสุด เท่ากับจำนวนเฉลี่ยอุปกรณ์ต่างๆ ทำงานพร้อมกัน โทรศัพท์ได้รับการคำนวณด้วยหน่วยเออร์แลง (Erlang)[28]



รูปที่ 2.3 โทรศัพท์ต่างๆ

จากรูปที่ 2.3 แสดงเส้นทางของทราฟฟิกต่างๆ โดยที่ A คือชุมสายท้องถิ่นอัตโนมัติ M คือชุมสายต่อผ่าน และ B คือชุมสายท้องถิ่นอัตโนมัติเหมือนกับชุมสาย A ซึ่งทราฟฟิกล้น (overflow traffic) คือส่วนของทราฟฟิกขอบริการที่อุปกรณ์รวมต่างๆ รับภาระได้ไม่หมด แต่จัดอุปกรณ์เพิ่มไว้รองรับ รับทราฟฟิกดังกล่าว เช่น รูปที่ 2.3 มีทราฟฟิกขอบริการเข้าที่เครื่องชุมสายท้องถิ่นต่อสถานี ปลายทางด้วยสายผ่านเส้นทางตรง (อุปกรณ์รวม) ได้สำเร็จ เรียกว่า ทราฟฟิกได้บริการ เมื่อเส้นทางเต็ม ทราฟฟิกขอบริการสู่ปลายทางเดียวกันมีมากกว่าจะกลายเป็น ทราฟฟิกล้นจากเส้นทางตรง เมื่อโครงข่ายจัดสายผ่านเพิ่มไว้ในเส้นทางเลือก และหรือเส้นทางสุดท้าย (final route) ทราฟฟิกล้นดังกล่าวจะกลายเป็นทราฟฟิกขอบริการเข้าเส้นทางเลือกเส้นทางสุดท้ายจะเป็นทราฟฟิกไม่ได้บริการ (lost traffic)

เออร์แลง (Erlang) คือหน่วยของทราฟฟิกได้บริการ เช่น ทราฟฟิกได้บริการ วงจรหรือสายผ่านใดต่อเนื่องเป็นเวลา 1 ชั่วโมง คือทราฟฟิก 1 เออร์แลง

$$1 \text{ Erlang} = 36 \text{ HCS (hundred call second) หรือ CCS (centum call second)}$$

2.2 อัตราการเรียกไม่ติด

อัตราการเรียกไม่ติด คือจำนวนของพารามิเตอร์ทางวิศวกรรมทราฟฟิกใช้เพื่อจัดหาให้เพียงพอขณะจัดตั้งภายใต้เงื่อนไขจำเพาะ พารามิเตอร์ชั้นของการเรียกไม่ได้

ความน่าจะเป็นของการติดขัด (Blocking probability) เนื่องจากมีการเรียกขอบริการ (Offered call) ในชั่วโมงใช้โทรศัพท์สูงสุด (Busy hour) มากเกินทรัพยากร เช่น สายผ่านในเส้นทางต่างๆ ที่จัดไว้ด้วยจำนวนจำกัด การเรียกที่เกินจำนวนดังกล่าว คือการเรียกไม่ได้ (Lost call) แสดงทางสมการคณิตศาสตร์ได้ ดังนี้

$$B = \frac{X_L}{X_O} \quad (2.1)$$

เมื่อ B = ความน่าจะเป็นของการติดขัด (blocking probability)

X_L = ทราฟฟิกไม่ได้บริการ (lost traffic) หรือ การเรียกไม่ได้ (lost calls)

X_O = ทราฟฟิกขอบริการ (offered traffic) หรือการเรียกขอบริการ (offered calls)

เนื่องจาก

$$X_O = X_C + X_L \quad (2.2)$$

X_C = ทราฟฟิกได้บริการ (carried traffic) หรือการเรียกได้บริการ (carried calls)

จะได้

$$B = 1 - \frac{X_c}{X_o} \quad (2.3)$$

หรือเมื่อต้องการหาการเรียกได้บริการ ได้ดังนี้

$$X_c = X_o(1 - B) \quad (2.4)$$

2.3 ประสิทธิภาพ (efficiency)

การหาประสิทธิภาพ เช่น ชั้นของบริการ (Grade of Service : GoS) จากความน่าจะเป็นของการติดขัด (blocking probability, B) หนึ่งในร้อย คือ 1 เปอร์เซนต์

การหาประสิทธิภาพของวงจร (η) หาได้จากอัตราส่วนของทราฟฟิกได้บริการต่อจำนวนวงจรที่ทราฟฟิกได้บริการผ่าน คูณด้วยร้อย มีหน่วยเป็นเปอร์เซนต์

$$\eta = \frac{X_c}{n} \times 100(\%) \quad (2.5)$$

เมื่อ X_c = ทราฟฟิกได้บริการ

n = จำนวนวงจรที่ทราฟฟิกได้บริการผ่าน

ภายใต้ชั้นบริการที่กำหนดให้จะพบว่า “ประสิทธิภาพวงจรของทราฟฟิกแพ็กเกตใหญ่จะดีกว่าของทราฟฟิกแพ็กเกตเล็ก”

2.4 ชั้นของบริการ (GoS : Grade of Service)

ชุมสายโทรศัพท์ท้องถิ่นต่างๆ ในพื้นที่เขตเมืองๆ หนึ่งในที่ติดตั้งสายผู้เช่าเพิ่มขึ้นอย่างต่อเนื่องระยะเวลาหนึ่งจนผู้ใช้เริ่มบ่นว่าเรียกไม่ได้มากกว่าเดิม ทำให้ผู้บริหารสั่งให้ดำเนินการตรวจสอบทุกชุมสายๆ ในชั่วโมงใช้โทรศัพท์สูงสุด (busy hour) ได้ตัวเลขความน่าจะเป็นของการติดขัด (blocking probability) ออกมาต่างๆ กัน

ความน่าจะเป็นของการติดขัด 3 ครั้ง ในการเรียก 100 ครั้ง คิดเป็น

$$\text{อัตราการเรียกไม่ได้} = \frac{\text{การเรียกไม่ได้}}{\text{การเรียกขอบริการ}} = \frac{(\text{lost call})}{(\text{offered call})} = \frac{X_L}{X_O} = \frac{3}{100}$$

เมื่อ x_L = การเรียกไม่ได้ หรือทราฟฟิกไม่ได้รับการบริการ

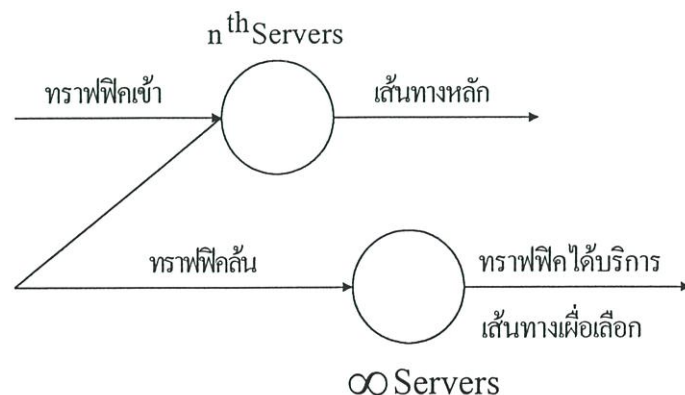
x_O = การเรียกขอบริการ หรือทราฟฟิกขอบริการ

อัตราการเรียกไม่ติดตรวจสอบจากชุมสายท้องถิ่นต่างๆ มาได้หลายค่า นำมาเทียบกัน การเรียกไม่ได้น้อยกว่าจะดีกว่า ในการเรียกขอบริการ 100 ครั้ง ด้วยกันในชั่วโมงใช้โทรศัพท์ที่สูงสุด จึงจัดเป็นชั้นของบริการ (Grade of Service) ได้เช่น 1/100, 2/100, 5/100 หรือ 0.01, 0.02, หรือ 0.05 ตามลำดับ[28]

2.5 การจัดเส้นทางแบบทราฟฟิกล้น (Overflow Traffic)

การเรียกของผู้เช่าที่อยู่ในชุมสายหนึ่งไปยังผู้เช่าในอีกชุมสายหนึ่ง การเรียกครั้งนี้อาจจะเลือกใช้ได้หลายเส้นทาง โดยที่เส้นทางแรกที่ถูกเลือกใช้เรียกว่า เส้นทางตรง (first choice route) ถ้าเส้นทางตรงไม่ว่าง การเรียกก็จะไปยังเส้นทางที่สอง เรียกว่า เส้นทางเลือกที่สอง (second choice route) ถ้าทั้งสองเส้นทางดังกล่าวไม่ว่าง การเรียกก็จะไปเลือกใช้เส้นทางที่สาม เรียกว่า เส้นทางเลือกที่สาม (third choice route) และเป็นเช่นนี้ไปเรื่อยๆ สำหรับ เส้นทางตรงนั้นจะเรียกว่า “high usage route” และเส้นทางเลือกสุดท้าย (last choice route) นั้นก็จะเรียกว่า “final route” ในกรณีของเส้นทางเลือกอื่นที่ไม่ใช่เส้นทางแรกและเส้นทางสุดท้าย ก็จะเรียกว่า “intermediate alternate route” ดังนั้น โครงข่ายที่จัดระบบการเลือกเส้นทางดังที่กล่าวมา เรียกว่า การจัดเส้นทางแบบทราฟฟิกล้น (Overflow system)[27]

การจัดเส้นทางแบบทราฟฟิกล้น ที่จะพิจารณาต่อไปนี้มีอยู่ 2 เส้นทาง ดังแสดงในรูปที่ 2.4 โดยที่ การเรียกจะใช้ เส้นทางตรง ซึ่งมี n^{th} Servers นั้นจะเป็นไปตามสมมติฐานของ Erlang B สำหรับ เส้นทางเลือกที่สอง ก็คือ การเรียกเข้ามายัง เส้นทางแรก แบบสุ่ม (random) และพบว่า n^{th} Servers ทั้งหมดใน เส้นทางแรก ไม่ว่างนั่นคือ การเรียกมายังเส้นทางเลือกที่สองก็จะเป็นทราฟฟิกล้นจากเส้นทางแรก



รูปที่ 2.4 Overflow system ซึ่งมี n^{th} servers ในเส้นทางแรก และ ∞ servers ในเส้นทางเลือกที่สอง

ตามรูปที่ 2.4 เนื่องจาก n^{th} Servers ทั้งหมดในเส้นทางตรงไม่ว่าง ค่าของทราฟฟิกขอ
บริการ ที่เรียกมาใช้ในเส้นทางเลือกที่หนึ่ง เท่ากับ A_0 ภายใต้เงื่อนไขสมดุลทางสถิติ (Statistical
equilibrium) จะสามารถหาค่าทราฟฟิกกลับ จากเส้นทางเลือกที่หนึ่ง ซึ่งมีค่า

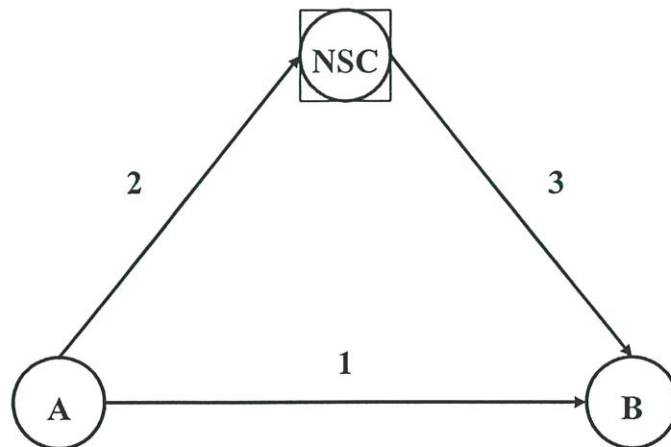
$$\text{mean } (\alpha) = A_0 B_n(A_0) \quad (2.6)$$

$$\text{Variance } (V) = \alpha \left(1 - \alpha + \frac{A_0}{n + 1 + \alpha - A_0} \right) \quad (2.7)$$

นั่นคือ ทราฟฟิกขอบริการ ของเส้นทางเลือกที่สองมีค่าเฉลี่ย (mean) และความแปรผัน (variance)
ซึ่งสามารถหาได้จากสมการที่ (2.6) และ (2.7)

$$\text{กำหนดให้ค่า Peakedness Factor (PF) = } \frac{V}{\alpha} \quad (2.8)$$

ถ้าค่าของ $PF = 1$ แล้ว ทราฟฟิกขอบริการ ของเส้นทางเลือกที่สองจะเป็นแบบสุ่ม (random) โดยทั่ว
ไปแล้วจะพบว่าค่าของ PF จะไม่เท่ากับ 1 นั่นคือ ทราฟฟิกขอบริการ ของเส้นทางเลือกที่สอง (หรือ
ทราฟฟิกกลับ จากเส้นทางเลือกที่หนึ่ง นั่นเอง) จะไม่เป็นแบบสุ่ม



รูปที่ 2.5 แบบจำลองตัวอย่างการจัดเส้นทางแบบทราฟฟิกกลับ

จากรูปที่ 2.5 A และ B คือชุมสายท้องถิ่น ส่วน NSC คือชุมสายต่อผ่าน โดยมีเส้นทางของ
ทราฟฟิกคือ 1,2 และ 3 ดังรูป

WinFIOL - [#2 D:\Thesis2003\Data Hopfield\Fig 2.6new.cmd]

Channel File Edit Block Search Run Tools Options Window Help

<ANRSP:RC=1000;

ROUTING CASE DATA

OPERATING AREA

RC	CCH	BR	PROGRAM ROUTING	SP	MISCELL
1000	YES		P01=1 R=AB10 P01=2 R=ANSC20	MM1	EST=1 EST=1

END

<STRSP:R=AB10&AB11&ANSC20&ANSC2I;

DEVICE STATE SURVEY

R	HDU	NOCC	NIDL	NBLO	RSTAT
AB10	30	30	0	0	NORES
AB11	30	30	0	0	NORES
ANSC20	30	1	29	0	NORES
ANSC2I	30	1	29	0	NORES

END

ERICSSON 26.1 Insert Port closed Log F1 - Help

Start Microsoft Word WinFIOL [#2 D:\Th... 21:46

รูปที่ 2.6 ผลการจัดเส้นทางแบบทราฟฟิกสลับ

จากรูปที่ 2.5 แสดงแบบจำลองตัวอย่างการจัดเส้นทางแบบทราฟฟิกสลับ และรูปที่ 2.6 เป็นผลที่ได้จากโปรแกรมที่ใช้ในชุมสายเพื่อจำลองการจัดเส้นทาง ของแบบจำลองดังกล่าว ซึ่งในคอตัมน์ของ Program ค่าพารามิเตอร์ P01=1 เป็นการออกแบบให้ทราฟฟิกของการเรียกเข้า ไปยังเส้นทางที่ 1 ก่อน และมีชื่อเส้นทางว่า AB10 ความหมายก็คือการเรียกจากชุมสาย A ไปยังชุมสาย B โดยใช้เส้นทางที่ 1 และ O คือ Outgoing เมื่อเส้นทางแรกเต็ม ทราฟฟิกก็จะล้น ไปยังเส้นทางเลือกที่สองซึ่งก็คือ P01=2 เป็นการออกแบบให้ทราฟฟิกของการเรียกเข้า เมื่อเส้นทางที่ 1 เต็มให้ไปใช้เส้นทางที่สอง ซึ่งมีชื่อเส้นทางว่า ANSC20 ความหมายก็คือเป็นการเรียกจากชุมสาย A ไปยังชุมสาย NSC โดยใช้เส้นทางที่ 2 และ O คือ Outgoing จากรูปที่ 2.6 จะเห็นได้ว่าเมื่อเส้นทาง AB10 เต็ม คือมีการจับใช้งานของวงจร เท่ากับ 30 วงจร ทราฟฟิกก็จะล้น ไปจับใช้งานวงจรของเส้นทาง ANSC20 ต่อไป

2.6 การจัดเส้นทางแบบแบ่งโหลด (Load sharing)

การจัดเส้นทางของโครงข่าย Signalling มีขั้นตอนต่างๆ (Routing Process) ดังนี้

(1) Link Set จะถูกเลือกตาม Priority (DPIO) โดยจะกำหนดไว้สำหรับแต่ละ DPC (Destination Point Code) ยกตัวอย่าง เช่น

Case I

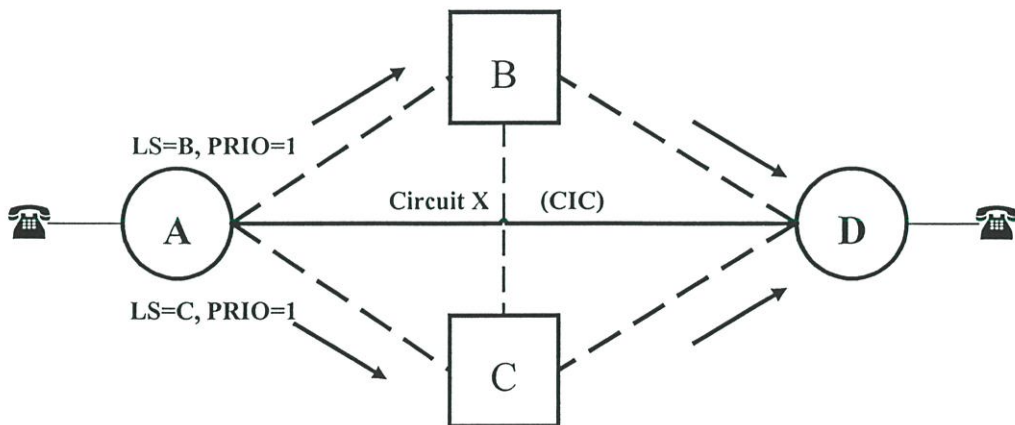
DPC=D, สามารถส่ง MSU ออกได้ทาง LS=B (PRIO=1) และ LS=C (PRIO=2) ซึ่งหมายถึงการเลือกเส้นทาง (Route) จะใช้ LS=B ตลอดเวลาที่เส้นทางนี้อยู่ในสภาพปกติ

Case II

DPC=D, สามารถส่ง MSU ออกได้ทาง LS=B (PRIO=1) และ LS=C (PRIO=1) ซึ่งหมายถึงเส้นทางทั้งสองมี Priority เท่ากัน การเลือกเส้นทางจะใช้สลับกันโดยใช้หลักการของ “การแบ่งโหลด” (Load Sharing)

การทำการแบ่งโหลด (Load Sharing) ของ SP หรือ national STP, กระทำได้โดยพิจารณาสภาพของ Bit 1 ใน CIC, ถ้า บิต = 0 ให้เลือก LS แรก และถ้า บิต = 1 ให้เลือก LS หลัง สำหรับการทำให้ Load Sharing ของ Regional STP จะพิจารณาสภาพของ บิต 2 ใน CIC

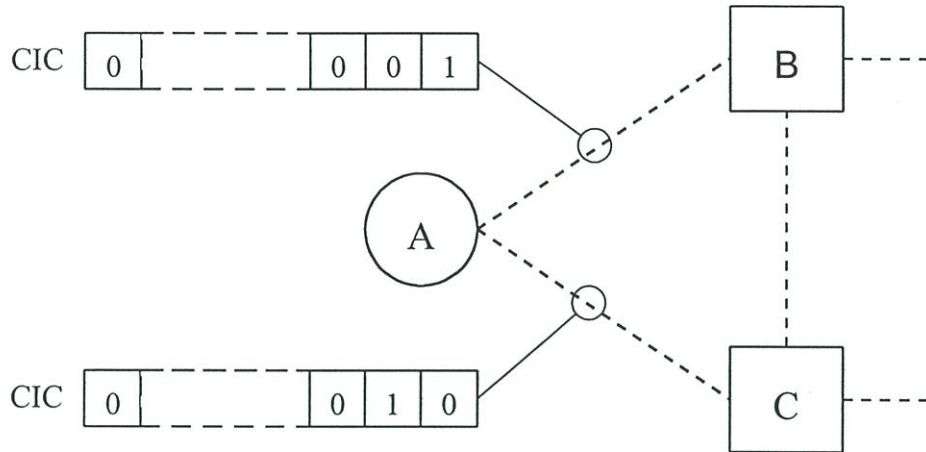
(2) หลังจากเลือก Link Set ได้แล้วก็จะทำการเลือก Signalling Link ภายใน LS ในกรณีที่ LS นั้นมี SL มากกว่าหนึ่งลิงค์ (Link) วิธีการเลือก SL จะนำเอา บิตบางส่วนของ CIC มาพิจารณาว่าจะเลือก SL ไດ



รูปที่ 2.7 การเลือกใช้ Link Set

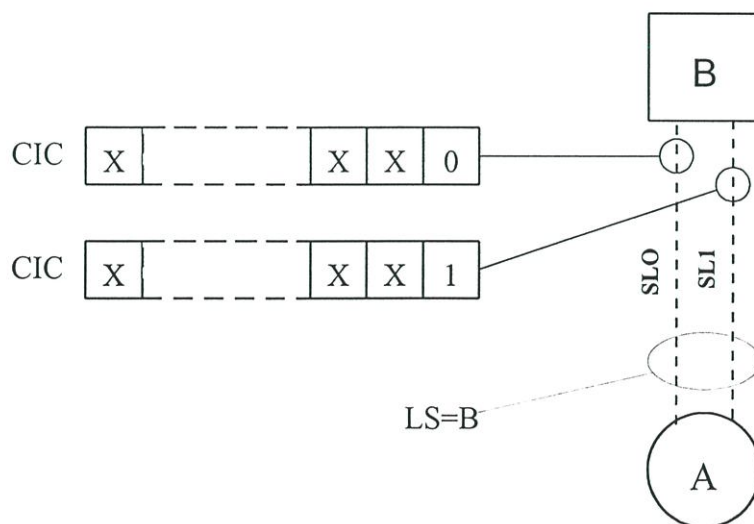
จากรูปที่ 2.7 ในกรณีที่ต้องการเรียกโทรศัพท์ จากชุมสาย A ไปยังชุมสาย D โดยผ่าน เส้นทางตรง (Direct Route) และ Signalling Point ของชุมสาย A ไม่มี Signalling Link ตรงไปยังชุมสาย

D ในเส้นทางของ Signalling ที่ไปยังชุมสาย D (DEST=D) มี 2 Link Set คือไปยัง STP B และ C ซึ่งได้กำหนด Priority ไว้เท่ากับ 1 เหมือนกันทั้งคู่



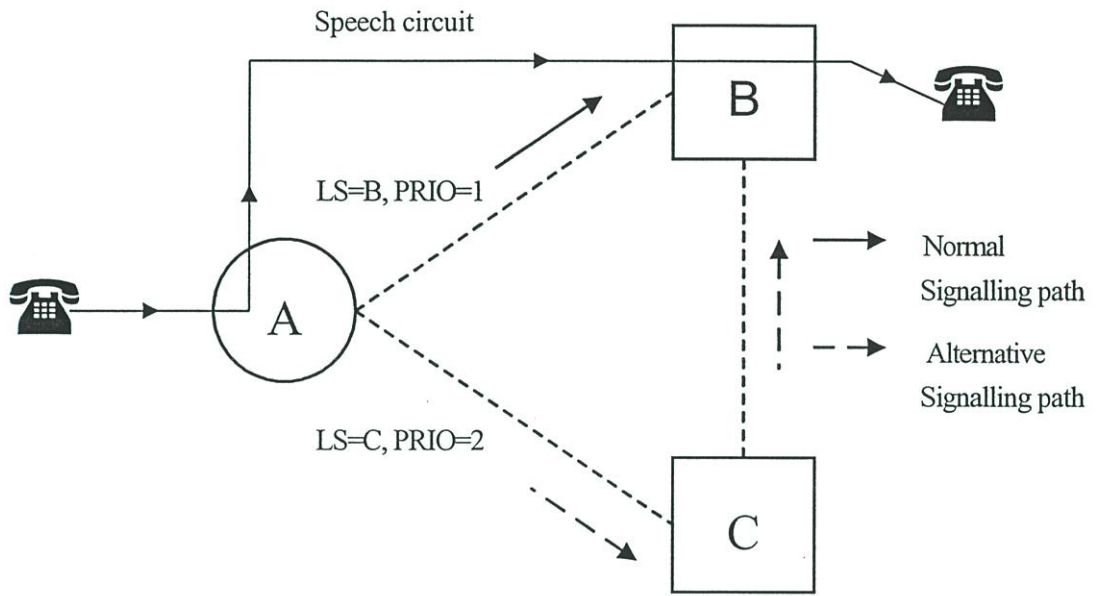
รูปที่ 2.8 การเลือกใช้ CIC

เมื่อกำหนดให้ PRIO=1 ไว้ทั้งสอง LS แสดงว่าการเลือกเส้นทางเป็นแบบการแบ่ง (Sharing) การพิจารณาเลือกเส้นทางจะดูจาก วงจร (Circuit No). ที่จับ โดยดูจากสภาพ บิตที่ 1 ว่ามีค่าเป็น 0 หรือ 1 ถ้า บิตที่ 1 เป็น 0 ให้จับ LS แรก (B) และถ้า บิตที่ 1 เป็น 1 ให้จับ LS หลัง (C) รูปต่อไปแสดงตัวอย่างการจับ วงจรหมายเลข. 1 (CIC) และ วงจรหมายเลข 2 (CIC) เมื่อชุมสาย A เลือกจับ Speed Circuit ที่ 1 Signalling Point A ก็จะส่ง MSU ไปทาง STP B และเมื่อชุมสาย A เลือกจับ Speed Circuit ที่ 2 Signalling Point A ก็จะส่ง MSU ไปทาง STP C



รูปที่ 2.9 การเลือกใช้ Signalling Links

ภายใน Link Set (LS) ยังประกอบไปด้วย Signalling Links สมมติว่า LS ระหว่างชุมสาย A และ B ประกอบด้วย SLO และ SL1 การเลือก SL ภายใน LS ก็ยังคงใช้หลักการของการแบ่งโหลด (Load Sharing) อยู่ และเป็นแบบถาวร (Permanent) โดยพิจารณาจาก บิตที่ 0 ที่ CIC ถ้ามีค่าเป็น 0 ก็เลือกใช้ SLO ถ้ามีค่าเป็น 1 ก็เลือกใช้ SL1 ดังแสดงในรูปที่ 2.9 ยกเว้นเมื่อเกิดเหตุขึ้นใน SL ใด SL หนึ่ง SL ที่เหลือก็จะต้องถูกใช้ส่ง MSU โดยไม่พิจารณา บิตที่ 0 ของ CIC



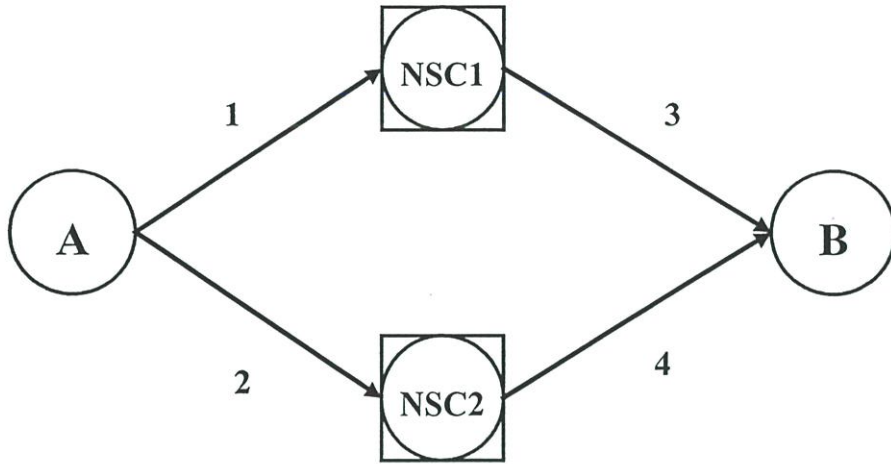
รูปที่ 2.10 เส้นทางสัญญาณผ่านปกติ หรือทางตรง และเลือก

รูปที่ 2.10 เป็นตัวอย่างของการเรียกโทรศัพท์จากชุมสาย A ไปยังชุมสาย B เส้นทางตรง (Direct Route) (ของ Speed Circuit) ซึ่งการทำงานของชุมสาย A ยกหูโทรศัพท์ และหาเลขหมาย สมมติว่าเลขหมายที่หมุนเป็นเลขหมายที่อยู่ในชุมสาย B และจากชุมสาย A มี เส้นทางตรง ไปยังชุมสาย B ชุมสาย A จะเลือกวงจรพูดภายในเส้นทาง ให้เหลือวงจรเดียว หลังจากนั้น SP A ก็จะรู้ว่าต้องส่ง MSU ไปยัง SP ปลายทางใด (Destination) ซึ่งจะได้ DEST=B และจาก DEST=B จะได้ข้อมูลดังนี้

DEST=B LS=B, PRIO=1
 LS=C, PRIO=2

SP A จะใช้เส้นทาง LS=B ซึ่งระบุไว้ว่าเป็น Normal route (PRIO=1) สำหรับใช้ในการส่ง MSU ไปยังชุมสาย B (ซึ่งเป็นทั้ง SP และ STP)

ในกรณีที่เส้นทาง LS=B เกิดสภาพของการติดขัด (Blocked) ขึ้นในขณะนั้น SP A จะใช้เส้นทาง LS=C ซึ่งระบุไว้ว่าเป็นเส้นทางเลือก (Alternative route) (PRIO=2) เป็นเส้นทางในการส่ง MSU ไปยังชุมสาย B โดยผ่านชุมสาย C (ซึ่งทำหน้าที่เป็น STP ส่งผ่านข้อมูล)



รูปที่ 2.11 แบบจำลองตัวอย่างการจัดเส้นทางแบบแบ่งโหลด

```

<ANRSP:RC=1000;

ROUTING CASE DATA
OPERATING AREA
RC  CCH BR      PROGRAM ROUTING          SP  MISCELL
1000 YES RA -50  P01=1 R=ANSC10          NM1 EST=1
      RA -50  P01=2 R=ANSC20          NM1 EST=1
      RA -50  P02=1 R=ANSC20          NM1 EST=1
      RA -50  P02=2 R=ANSC10          NM1 EST=1

END

<STRSP:R=ANSC10&ANSC1I&ANSC20&ANSC2I;

DEVICE STATE SURVEY
R      HDU  NOCC  NIDL  NBLO  RSTAT
ANSC10 30    1    29    0    NORES
ANSC1I 30    1    29    0    NORES
ANSC20 30    1    29    0    NORES
ANSC2I 30    1    29    0    NORES

END
  
```

รูปที่ 2.12 ผลการจัดเส้นทางแบบแบ่งโหลด

จากที่เคยได้อธิบายหลักการไปแล้วจากรูปที่ 2.5 และ 2.6 ซึ่งเป็นลักษณะการจัดเส้นทางแบบทราฟฟิกสั้น ส่วนรูปที่ 2.11 เป็นแบบจำลองตัวอย่างจัดเส้นทางแบบแบ่งโหลด และรูปที่ 2.12 เป็นผลการจัดเส้นทางแบบแบ่งโหลด ซึ่งได้มีการเพิ่มค่าพารามิเตอร์บางตัวเข้าไป ในคอลัมน์ของ BR (Branching type) คือ $RA=50$ ซึ่งหมายความว่าให้ แบ่งโหลดหรือทราฟฟิก ระหว่างเส้นทางที่ 1 และ 2 เส้นทางละ 50 เปอร์เซ็นต์ จากรูปที่ 2.12 มีการเรียกเข้ามาสองครั้ง แสดงให้เห็นว่ามีการจับใช้งานของวงจรเส้นทางละ 1 วงจร

บทที่ 3

ทฤษฎีโครงข่ายประสาทเทียม

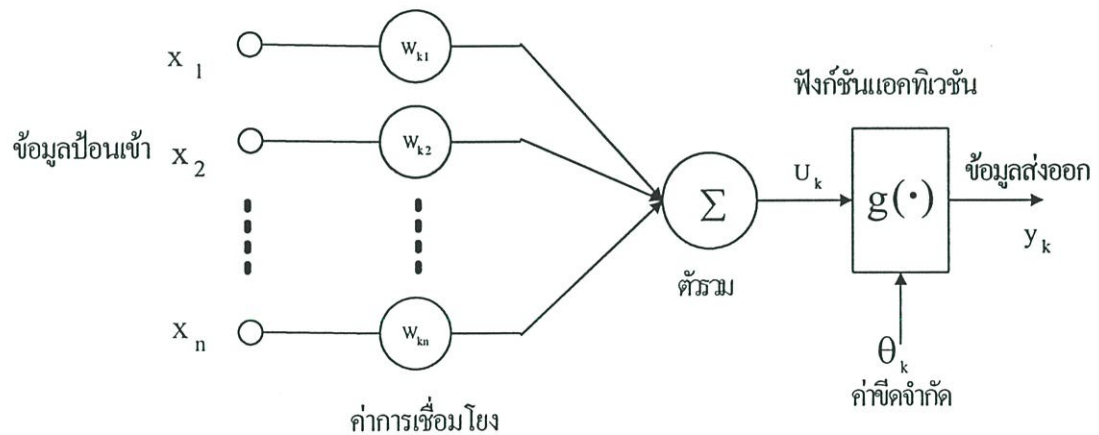
3.1 แบบจำลองของนิวรอน

นิวรอนเป็นหน่วยประมวลผลพื้นฐานที่สำคัญในการทำงานของโครงข่ายประสาท มีองค์ประกอบ 3 ส่วน ดังรูปที่ 3.1 คือ

1) กลุ่มของการเชื่อมโยงหรือไซแนปส์ การเชื่อมโยงระหว่างนิวรอนจะถูกควบคุมด้วยค่าการเชื่อมโยง กำหนดให้ x_j เป็นข้อมูลป้อนเข้าของไซแนปส์ j ที่ป้อนให้กับนิวรอน k ข้อมูลป้อนเข้าจะถูกคูณด้วยค่าการเชื่อมโยง w_{kj} โดยที่ตัวห้อยตัวแรก (k) หมายถึงนิวรอนที่ k ส่วนตัวที่สอง (j) บอกว่าเป็นข้อมูลป้อนเข้าที่ j ซึ่งอาจมาจากนิวรอนอื่นๆ ที่อยู่ข้างเคียง หาก w_{kj} มีเครื่องหมายเป็นบวกหมายถึงการกระตุ้นเสริม ถ้ามีเครื่องหมายเป็นลบหมายถึงการกระตุ้นแบบหักล้าง

2) ตัวบวก (Summation หรือ Adder) สำหรับรวมข้อมูลป้อนเข้าที่คูณด้วยค่าการเชื่อมโยงแล้ว เราเรียกว่า การรวมเชิงเส้น (Linear combiner)

3) ฟังก์ชันแอคทิเวชัน เพื่อจำกัดช่วงข้อมูลป้อนออกให้อยู่ในช่วงที่ต้องการ



รูปที่ 3.1 การจำลองโครงข่ายประสาทในรูปแบบคณิตศาสตร์

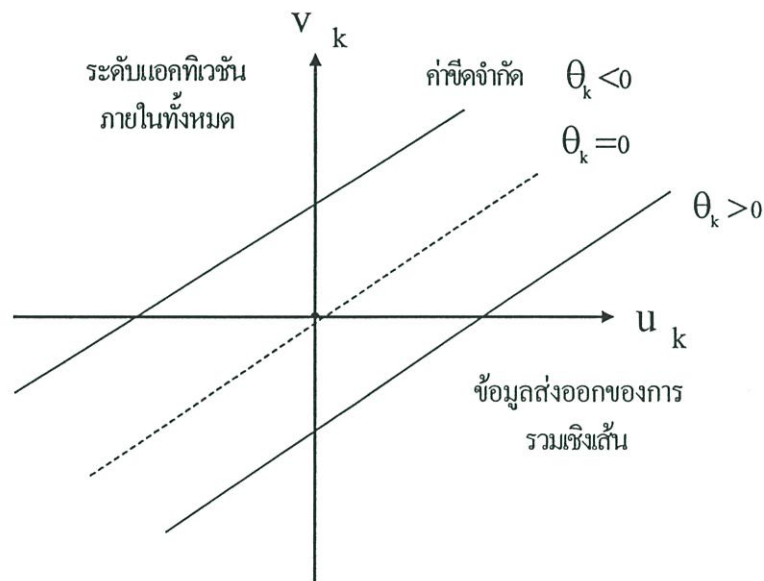
จากรูปที่ 3.1 สามารถอธิบายนิวรอน k อยู่ในรูปสมการได้ด้วยสมการ 2 สมการคือ

$$u_k = \sum_{j=1}^n w_{kj} x_j \quad (3.1)$$

$$y_k = g(u_k - \theta_k) \quad (3.2)$$

โดยที่ $x_1, x_2, x_3, \dots, x_n$ คือข้อมูลป้อนเข้า
 $w_{k1}, w_{k2}, w_{k3}, \dots, w_{kn}$ คือค่าการเชื่อมโยงของนิวรอน k
 u_k คือข้อมูลส่งออกภายในนิวรอน (Internal output) หรือข้อมูลป้อนเข้าสุทธิ (net_k)
 θ_k คือค่าขีดจำกัด (Threshold หรือ Bias หรือ Offset) ซึ่งรับมาจากภายนอก
 $g(\cdot)$ คือฟังก์ชันแอคติเวชัน
 y_k คือข้อมูลส่งออก

การป้อนค่าขีดจำกัดให้กับนิวรอน จะได้ความสัมพันธ์ระหว่างระดับแอคติเวชันภายใน (Internal activity level หรือ Internal activation potential) ค่าขีดจำกัด และข้อมูลป้อนเข้า ดังต่อไปนี้



รูปที่ 3.2 ผลของค่าขีดจำกัด

$$v_k = u_k - \theta_k \quad (3.3)$$

โดยที่ v_k คือระดับแอคติเวชันภายใน

จากนิวรอนธรรมชาติและแบบจำลองของนิวรอน ซึ่งเป็นแบบจำลองทางคณิตศาสตร์ที่ประดิษฐ์ขึ้นมา สามารถเปรียบเทียบได้ ดังตารางที่ 3.1[1-4]

ตารางที่ 3.1 เปรียบเทียบระหว่างนิเวศวิทยาและแบบจำลองที่ประดิษฐ์ขึ้น

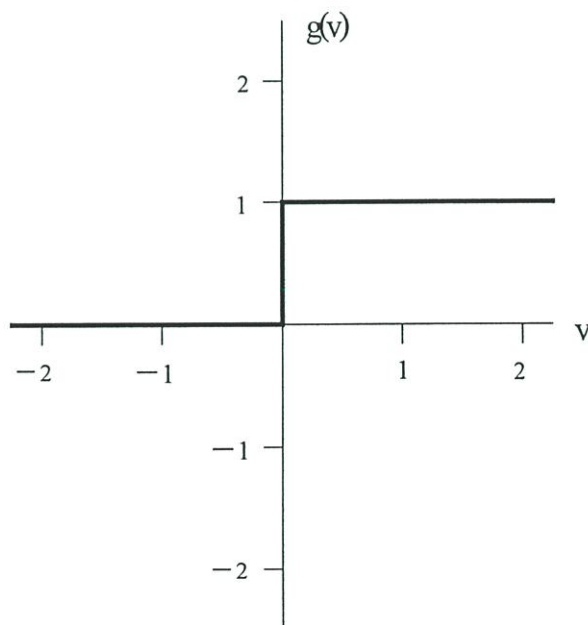
ลำดับ	นิเวศวิทยา	แบบจำลองนิเวศที่ประดิษฐ์ขึ้น
1	โซมา	ส่วนประมวลผล (ตัวรวมและฟังก์ชันแอกทิเวชัน)
2	เดนไดรต์	ข้อมูลป้อนเข้า
3	แอกซอน	ข้อมูลส่งออก
4	ไซแนป	ค่าการเชื่อมโยง
5	ความเร็วในการประมวลผลต่ำ	ความเร็วในการประมวลผลสูง
6	โครงข่ายประกอบด้วยนิเวศจำนวนมาก	โครงข่ายประกอบด้วยนิเวศจำนวนน้อย

3.2 ชนิดของฟังก์ชันแอกทิเวชัน

ฟังก์ชันแอกทิเวชันที่แทนด้วย $g(\cdot)$ เป็นตัวกำหนดค่าข้อมูลส่งออกเราอาจแบ่งฟังก์ชันแอกทิเวชันพื้นฐานออกได้เป็น 3 ชนิด คือ

- 1.) ฟังก์ชันเทรชโฮลด์ (Threshold function หรือ Hard limit function หรือ Step function)

แสดงตัวอย่างดังรูปที่ 3.3

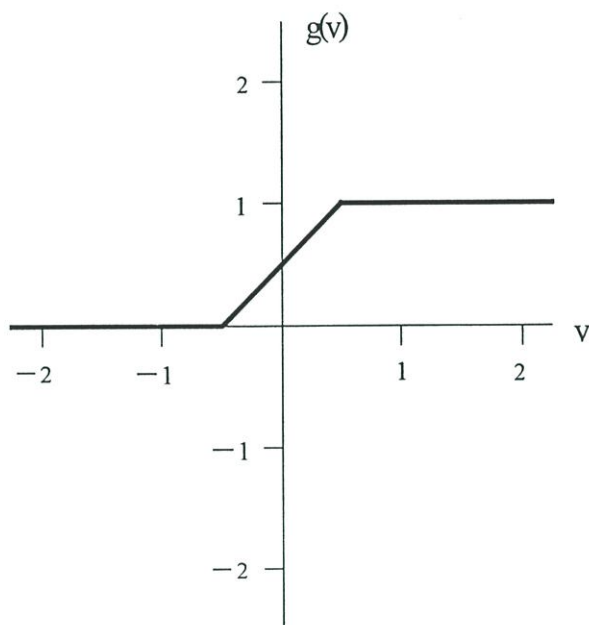


รูปที่ 3.3 ฟังก์ชันเทรชโฮลด์

โดยมีความสัมพันธ์ดังสมการที่ 3.4

$$g(v) = \begin{cases} 1; v \geq 0 \\ 0; v < 0 \end{cases} \quad (3.4)$$

2.) ฟังก์ชันเชิงเส้นอิ่มตัว (Saturating Linear Function หรือ Piecewise-Linear Function)
ตัวอย่างของฟังก์ชันแบบนี้ ดังแสดงในรูปที่ 3.4



รูปที่ 3.4 ฟังก์ชันเชิงเส้นอิ่มตัว

โดยมีความสัมพันธ์ดังสมการ

$$g(v) = \begin{cases} 1; v \geq b \\ av; -b < v < b \\ 0; v \leq -b \end{cases} \quad (3.5)$$

โดยที่ a คือค่าความชัน (Slop) ในช่วงความเป็นเชิงเส้น

b คือช่วงที่เป็นเชิงเส้นของฟังก์ชันเชิงเส้นอิ่มตัว

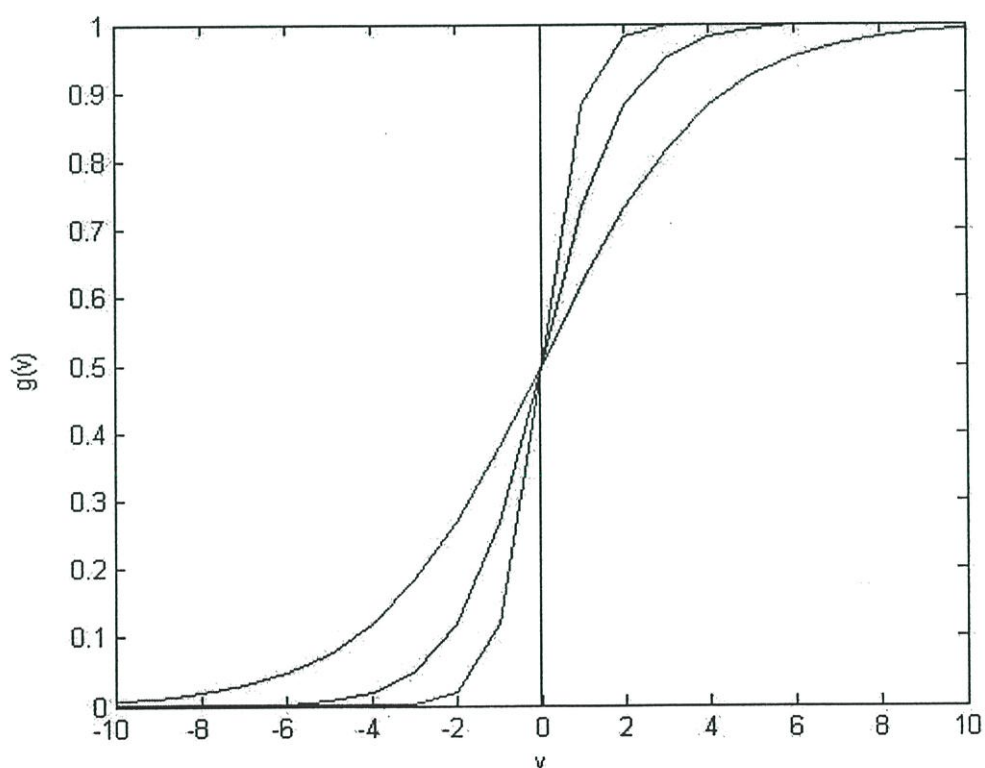
จากฟังก์ชันเชิงเส้นอิ่มตัวจะสังเกตเห็นได้ว่า เมื่อไม่มีช่วงอิ่มตัว ($b = \infty$) จะได้ฟังก์ชันเชิงเส้น นั่นคือ $g(v) = av$ และฟังก์ชันเชิงเส้นอิ่มตัวจะเปลี่ยนรูปไปเป็นฟังก์ชันเทรสโฮลด์เมื่อค่าความชันเท่ากับอนันต์ ($a = \infty$)

3.) ฟังก์ชันซิกมอยด์ (Sigmoid function) เป็นฟังก์ชันที่ได้รับความนิยมนำมาใช้เป็นฟังก์ชันแอคทิเวชันของโครงข่ายประสาทกันมาก เนื่องจากเป็นฟังก์ชันที่เพิ่มขึ้นอย่างสม่ำเสมอ (Smoothness) ฟังก์ชันซิกมอยด์ จัดเป็นฟังก์ชันแอคทิเวชันที่ไม่เป็นเชิงเส้น ตัวอย่างของฟังก์ชันซิกมอยด์คือ ฟังก์ชันลอจิสติก (Logistic function) และฟังก์ชันแทนเจนไฮเพอร์โบลิก (Hyperbolic tangent function) ซึ่งมีความสัมพันธ์ดังสมการที่ 3.6 สมการที่ 3.7

$$\text{ฟังก์ชันลอจิสติก } g(v) = \frac{1}{1 + \exp(-\lambda v)} \quad (3.6)$$

โดยที่ λ คือค่าความชัน

$$\text{ฟังก์ชันแทนเจนไฮเพอร์โบลิก } g(v) = \tanh\left(\frac{v}{2}\right) = \frac{1 - \exp(-v)}{1 + \exp(-v)} \quad (3.7)$$



รูปที่ 3.5 ฟังก์ชันลอจิสติกที่ค่าความชันต่างๆ

ฟังก์ชันแอคทิเวชันพื้นฐานทั้ง 3 ชนิด ที่กล่าวข้างต้นในสมการ (3.4),(3.5),(3.6) จะจำกัดช่วงข้อมูลส่งออกอยู่ตั้งแต่ 0 ถึง +1 ในการประยุกต์ใช้งานบางกรณีอาจมีความจำเป็นต้องใช้ฟังก์ชันที่ได้ค่า

ออกมานอกเหนือช่วงดังกล่าว เช่น ต้องการข้อมูลส่งออกอยู่ในช่วงตั้งแต่ -1 ถึง $+1$ อาจปรับเปลี่ยนฟังก์ชันพื้นฐานเหล่านี้ ตัวอย่าง เช่น ฟังก์ชันซิกนัม (Signum function) ซึ่งประยุกต์มาจากฟังก์ชันเทรสโலด์ มีความสัมพันธ์ดังสมการ

$$g(v) = \begin{cases} 1; v > 0 \\ 0; v = 0 \\ -1; v < 0 \end{cases} \quad (3.8)$$

นอกเหนือจากฟังก์ชันแอกทิเวชันพื้นฐานทั้งสาม แล้วยังสามารถนำความสัมพันธ์ระหว่างชุดข้อมูลในรูปแบบอื่นๆ มาเป็นฟังก์ชันแอกทิเวชันได้ ตัวอย่างเช่น ฟังก์ชันแอกทิเวชันแบบคอมเพิตทิฟ (Competitive)

$$g(v) = \begin{cases} 1; v_j & \text{คือค่าที่มากที่สุดในกลุ่ม } v_1, v_2, \dots, v_j, \dots, v_n \\ 0; v_j & \text{ไม่ใช่ค่าที่มากที่สุดในกลุ่ม} \end{cases} \quad (3.9)$$

ฟังก์ชันแอกทิเวชันแบบคอมเพิตทิฟนี้จะให้ค่าออกมาเท่ากับหนึ่ง เพียงค่าเดียวเท่านั้นสำหรับ v ที่มากที่สุดจากจำนวนทั้งหมดเท่ากับ n ส่วนค่าอื่นๆ จะมีค่าเป็นศูนย์

ตัวอย่างการใช้ฟังก์ชันเทรสโลด์เป็นฟังก์ชันแอกทิเวชันของนิวรอล k จะได้ข้อมูลส่งออกดังนี้

$$y_k = \begin{cases} 1; v_k \geq 0 \\ 0; v_k < 0 \end{cases}$$

เมื่อ v_k คือระดับแอกทิเวชันภายในซึ่งมีค่าดังนี้

$$v_k = \sum_{j=1}^n w_{kj} x_j - \theta_j$$

สำหรับการพิจารณาเลือกใช้ฟังก์ชันแอกทิเวชันแบบใดนั้นนับถึงปัจจุบันยังไม่มีข้อกำหนดที่แน่นอน แต่อาจจะพิจารณาจากองค์ประกอบดังต่อไปนี้

- 1.) ความเป็นเชิงเส้น หรือไม่เป็นเชิงเส้น

2.) ลักษณะของข้อมูลป้อนเข้า

3.) ขอบเขตที่ต้องการและลักษณะของข้อมูลส่งออก

4.) ปัจจัยอื่นที่มีนัยสำคัญต่อผลตอบสนองของนิเวศหรือโครงข่ายประสาทโดยรวม

ในบางครั้งอาจเลือกใช้ฟังก์ชันสโตซาสติก (Stochastic function) เป็นฟังก์ชันแบบเชิงเส้นและไม่เป็นเชิงเส้นในโครงข่ายประสาทเดียวกัน ตัวอย่างเช่น โครงข่ายประสาทแบบ RBF (Radial basis function) ซึ่งใช้ฟังก์ชันแอกทิเวชันแบบไม่เป็นเชิงเส้นในชั้นซ่อน (Hidden Layer) ของโครงข่ายประสาท ขณะที่ชั้นข้อมูลส่งออก (Output layer) ใช้แอกทิเวชันแบบเป็นเชิงเส้น

3.3 กระบวนการเรียนรู้และกฎการเรียนรู้ของโครงข่ายประสาท

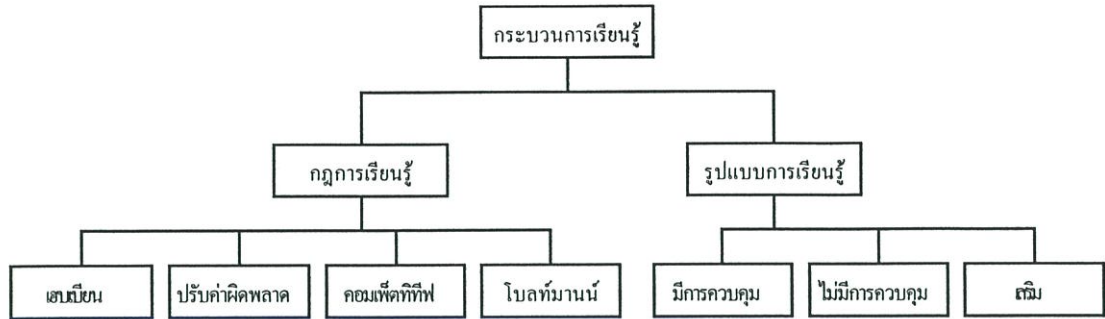
จากหัวข้อที่กล่าวมาทั้งสองส่วน ยังไม่เพียงพอที่จะใช้ประโยชน์จากโครงข่ายประสาท การที่จะทำให้โครงข่ายประสาทมีความสามารถในการแก้ปัญหาได้ ต้องปรับสอนโครงข่ายประสาท หรือกล่าวอีกอย่างก็คือ โครงข่ายประสาทต้องมีความสามารถที่จะเรียนรู้ กระบวนการปรับสอนหรือกระบวนการเรียนรู้จะเกี่ยวกับความรู้ ความรู้ที่กล่าวถึงนี้คือข้อมูลข่าวสารแบบจำลองหรือสิ่งอื่นใดที่คนหรือเครื่องจักรเก็บไว้ใช้ในกระบวนการพยากรณ์ ทำนาย ประเมินการ คาดคะเน และตอบสนองอย่างเหมาะสมต่อสถานการณ์หรือสิ่งที่กระตุ้นภายนอก นั่นคือสิ่งที่เกี่ยวกับความรู้จะมี 2 ประเด็น ประเด็นที่หนึ่ง เกี่ยวกับความรู้ว่าเป็นข่าวสารข้อมูลอะไร ประเด็นที่สองคือ วิธีการเก็บข่าวสารข้อมูลเพื่อนำมาใช้ประโยชน์นั้นทำอย่างไร เนื่องจากผลลัพธ์ที่ดีจะขึ้นอยู่กับความถูกต้องแม่นยำของตัวผลลัพธ์เองและความรวดเร็วที่จะได้ผลลัพธ์นั้นมา โครงข่ายประสาทที่ดีจึงควรเป็นแบบจำลองภาวะในการเก็บความรู้พร้อมทั้งสามารถประมวลผลได้ด้วย การเรียนรู้ของโครงข่ายประสาทเป็นกระบวนการทำให้เกิดความรู้ สามารถกล่าวได้ว่า การเรียนรู้ของโครงข่ายประสาทเป็นกระบวนการปรับค่าตัวแปรอิสระ (Free parameters) ของโครงข่ายประสาทเพื่อปรับพฤติกรรมให้สอดคล้องกับการกระตุ้นจากสิ่งแวดล้อม กระบวนการเรียนรู้ของโครงข่ายประสาทจึงเป็นกระบวนการทำซ้ำ (Iterative) เราพอจะลำดับกระบวนการเรียนรู้ได้ดังนี้

1.) โครงข่ายประสาทถูกกระตุ้นโดยสิ่งแวดล้อม

2.) โครงข่ายจะปรับตัวเพื่อตอบสนองต่อสิ่งที่มีมากระตุ้น

3.) โครงข่ายจะตอบสนองต่อสิ่งแวดล้อมด้วยภาวะใหม่หลังจากการปรับตัว

เราอาจจะกล่าวถึงกระบวนการเรียนรู้ใน 2 แนวทาง คือ กฎการเรียนรู้ (Learning rules หรือ Learning algorithm) และรูปแบบการเรียนรู้ (Learning pattern หรือ Learning paradigm)



รูปที่ 3.6 ความสัมพันธ์แบบต่างๆ ของโครงข่ายประสาท

3.3.1 กฎการเรียนรู้

กฎพื้นฐานในการเรียนรู้ของโครงข่ายประสาท [20] จะเกี่ยวกับการเปลี่ยนแปลงเวกเตอร์ค่าการเชื่อมโยง (Connection vector) $w_i = [w_{i1} w_{i2} \dots w_{in}]^T$ โดยที่ \square^T คือการทรานโพส (Transposition) การเปลี่ยนแปลงนั้นจะขึ้นอยู่กับข้อมูลป้อนเข้า x และสัญญาณการเรียนรู้ (Learning signal) l โดยปกติแล้ว l เป็นฟังก์ชันของ w_i และ c แต่ในบางครั้งอาจมีตัวแปรเพิ่มเติมคือสัญญาณควบคุมการเรียนรู้ (Teacher's signal) c_i นั่นคือ

$$\text{สัญญาณการเรียนรู้ } l = f(w_i, x, c_i) \quad (3.10)$$

จากพื้นฐานที่กล่าวมาคือ การเปลี่ยนแปลงของเวกเตอร์ของค่าการเชื่อมโยง ณ เวลา t มีความสัมพันธ์ ดังนี้

$$\Delta w_i(t) = \alpha l(t)x(t) = \alpha f[w_i(t), x(t), c_i(t)]x(t) \quad (3.11)$$

โดยที่ α คือค่าคงที่การเรียนรู้ (Learning constant)

ค่าคงที่การเรียนรู้เป็นตัวกำหนดอัตราการเรียนรู้ (Rate of learning) ให้เหมาะสมไม่เร็วหรือช้าเกินไปค่าดังกล่าวนี้จึงมีค่าเป็นบวก (Positive)

เวกเตอร์ของค่าการเชื่อมโยง ณ เวลา $t+1$ เป็นผลจากการเปลี่ยนแปลงค่าการเชื่อมโยงขั้นตอนการปรับตัวคู่ค่าการเชื่อมโยงใหม่นี้เรียกขั้นตอนนี้ว่า ขั้นตอนการเรียนรู้ (Learning step)

$$w_i(t+1) = w_i(t) + \Delta w_i(t) \quad (3.12)$$

เมื่อนำค่า $w_i(t)$ จากสมการที่ 3.11 มาแทนจะได้

$$w_i(t+1) = w_i(t) + \alpha f[w_i(t), x(t), c_i(t)]x(t) \quad (3.13)$$

เมื่อพิจารณาการปรับค่าเวกเตอร์ของค่าการเชื่อมโยงเป็นกระบวนการทำซ้ำ (Iteration) จากสมการ 3.13 สามารถเขียนได้ดังนี้

$$w_i^{k+1} = w_i^k + \alpha f[w_i^k, x^k, c_i^k]x^k \quad (3.14)$$

สำหรับตัวยก (Superscript) $k+1$ หมายถึงการคำนวณในปัจจุบัน ส่วนตัวยก k คือการคำนวณในครั้งที่ผ่านมาก่อนหน้านี้ ในการปรับค่าเวกเตอร์ของค่าการเชื่อมโยงนั้น อาจจะเริ่มต้นด้วยการสุ่มค่า (Random) ค่าการเชื่อมโยงที่เหมาะสม ตัวอย่างกฎการเรียนรู้ที่มีชื่อเสียงได้แก่

- 1.) กฎการเรียนรู้แบบเฮบเบียน (Hebbian learning rule)
- 2.) กฎการเรียนรู้แบบปรับค่าผิดพลาด (Error correction learning rule)
- 3.) กฎการเรียนรู้แบบคอมเพทิทีฟ (Competitive learning rule)
- 4.) กฎการเรียนรู้แบบโบลท์มานน์ (Boltzmann learning rule)

1.) กฎการเรียนรู้แบบเฮบเบียน

การเรียนรู้ของ Hebb นับว่าเป็นกฎการเรียนรู้ที่เก่าแก่และมีชื่อเสียง กล่าวไว้ในหนังสือ Organization of Behavior หน้า 62 ซึ่งตีพิมพ์ในปี ค.ศ.1949 [29] จะได้สัญญาณการเรียนรู้เป็นฟังก์ชันของข้อมูลส่งออกของนิวรอลดังนี้

$$l \underline{\underline{\Delta}} f(w_i^T x) \quad (3.15)$$

$$\Delta w_i = \alpha f(w_i^T x)x \quad (3.16)$$

การเปลี่ยนแปลงค่าการเชื่อมโยงแต่ละค่าหาได้จากความสัมพันธ์

$$\Delta w_{ij} = \alpha f(w_i^T x)x_j; j=1,2,\dots, n \quad (3.17)$$

จากกฎการเรียนรู้ของเฮบเบียน จะเห็นว่ารูปแบบการเรียนรู้ เป็นแบบไม่มีการควบคุม (Unsupervised) และเป็นแบบป้อนผู้ด้านหน้า เราสามารถเขียนในรูปย่อได้ดังนี้

$$\Delta w_{ij} = \alpha y_i x_j \quad (3.18)$$

ค่าดังกล่าวเป็นผลคูณระหว่างข้อมูลป้อนเข้าและข้อมูลส่งออก เราอาจจะเรียกว่า activity product rule ต่อมาในปี ค.ศ.1988 Kohonen ได้เพิ่มพจน์ที่เป็นฟังก์ชันของ w_{ij} เองเพื่อให้การปรับตัวของค่าการเชื่อมโยงให้เป็นอย่างค่อยเป็นค่อยไป สามารถเขียนสมการได้ดังนี้

$$\Delta w_{ij} = \alpha y_i x_j - \beta y_i w_{ij}; j=1,2,\dots, n \quad (3.19)$$

โดยที่ β คือค่าคงที่การเรียนรู้อีกค่าหนึ่ง

จากสมการที่ 3.19 สามารถจัดรูปแบบได้ใหม่ดังนี้

$$\Delta w_{ij} = \alpha y_i [\gamma x_j - w_{ij}]; \gamma = \frac{\beta}{\alpha} \quad (3.20)$$

จากสมการที่ 3.20 เรียกว่า กฎการเรียนรู้แบบ Generalized activity product ในกรณีที่ทราบข้อมูลส่งออกที่ต้องการ d_i คือสัญญาณควบคุมการเรียนรู้ ทำให้การเรียนรู้กลายเป็นการเรียนรู้แบบมีการควบคุม (Supervised) เรียกการเรียนรู้แบบนี้ว่า Correction learning rule ซึ่งมีความสัมพันธ์ดังนี้

$$\Delta w_{ij} = \alpha d_i x_j \quad (3.21)$$

เมื่อพิจารณาสมการที่ 3.18 และสมการที่ 3.21 จะเห็นได้ว่าแตกต่างกันที่สัญญาณการเรียนรู้ l ในสมการที่ 3.18 ใช้ข้อมูลส่งออกที่ได้จากนิเวศจริงๆ y_i ขณะที่สมการที่ 3.21 ใช้ข้อมูลส่งออกที่ต้องการ จะได้ d_i สำหรับรูปแบบในสมการที่ 3.21 เราสามารถเรียกได้ว่าเป็นการเรียนรู้แบบเฮบเบียนที่มีการควบคุม (Supervised Hebbian learning)

นอกจากการจำลองทางคณิตศาสตร์ของกฎการเรียนรู้แบบ Hebb ดังกล่าวมา ได้มีการนำเสนอหลักการเรียนรู้ของ Hebb ในแบบที่แก้ปัญหาเชิงสถิติ การเปลี่ยนแปลงของค่าการเชื่อมโยงขึ้นกับค่าโควาเรียนซ์ (Covariance) ระหว่างข้อมูลป้อนเข้าและข้อมูลส่งออก เขียนเป็นความสัมพันธ์ได้ดังนี้

$$\begin{aligned} \Delta w_{ij} &= \alpha \text{cov}[y_i, x_j]; j=1,2,\dots, n \\ &= \alpha E[(y_i - \bar{y}_i)(x_j - \bar{x}_j)] \\ &= \alpha \{E[y_i x_j] - \bar{y}_i \bar{x}_j\} \end{aligned} \quad (3.22)$$

โดยที่ E คือ Statistical expectation operator

เมื่อพิจารณาสมการที่ 3.18 และสมการที่ 3.22 จะเห็นว่าพจน์แรกต่างกันตรงที่ สมการที่ 3.22 ใช้ตัวปฏิบัติการ E กับผลคูณระหว่างข้อมูลป้อนเข้ากับข้อมูลส่งออกแทนที่จะเป็นค่าผลคูณนั้น โดยตรง สำหรับพจน์ที่ 2 ของสมการที่ 3.22 $\overline{y_i x_i}$ สามารถพิจารณาเป็นค่าขีดจำกัดที่ป้อนให้กับนิรล

2.) กฎการเรียนรู้แบบปรับค่าผิดพลาด

การเรียนรู้แบบปรับค่าผิดพลาดนับว่าเป็นการเรียนรู้ที่สำคัญวิธีหนึ่ง ค่าผลต่างระหว่างข้อมูลส่งออกที่ต้องการจะได้ d_i และข้อมูลส่งออกที่ได้รับจากโครงข่ายจริงๆ y_i เรียกว่าค่าผิดพลาด e_i นั่นคือ

$$e_i = d_i - y_i \quad (3.23)$$

เป้าหมายของการเรียนรู้แบบปรับค่าผิดพลาด คือ การหาค่าต่ำสุดของฟังก์ชันเป้าหมาย (Cost function) ซึ่งเป็นฟังก์ชันของค่าผิดพลาด เมื่อเลือกฟังก์ชันเป้าหมายที่แทนค่าผิดพลาดโดยรวมอย่างเหมาะสม ก็จะได้ผลลัพธ์ที่ดี สัญญาณการเรียนรู้ I มีค่าดังนี้

$$I \triangleq [d_i - f(w_i^T x)] f'(w_i^T x) \quad (3.24)$$

โดยที่ $f'(w_i^T x)$ คืออนุพันธ์ของฟังก์ชันแอคติเวชันเทียบกับ w_i

จะสังเกตได้ว่าการเรียนรู้แบบปรับค่าผิดพลาดเป็นการเรียนรู้แบบมีการควบคุมและฟังก์ชันแอคติเวชัน ต้องเป็นฟังก์ชันที่สามารถหาอนุพันธ์ได้เมื่อใช้ฟังก์ชันเป้าหมาย E ที่เป็นฟังก์ชันของค่าผิดพลาดกำลังสอง นั่นคือ

$$E = \frac{1}{2} [d_i - f(w_i^T x)]^2 = \frac{1}{2} [d_i - y_i]^2 \quad (3.25)$$

เมื่อหาค่าเกรเดียนต์ (Gradient) ของ E เพื่อหาค่าต่ำสุดนั้นคือ

$$\nabla E = -(d_i - y_i) f'(w_i^T x) x \quad (3.26)$$

แต่ค่าของเวกเตอร์ของเกรเดียนต์สามารถหาค่าได้จาก

$$\frac{\partial E}{\partial w_{ij}} = -(d_i - y_i) f'(w_i^T x) x_j \quad (3.27)$$

เนื่องจาก $\Delta w_{ij} = -\alpha \nabla E$ นำค่า ∇E จากสมการที่ 3.26 มาแทนจะได้

$$\Delta w_{ij} = \alpha (d_i - y_i) f'(w_i^T x) x_j \quad (3.28)$$

การเปลี่ยนแปลงของค่าการเชื่อมโยงแต่ละค่าหาได้จากความสัมพันธ์

$$\Delta w_{ij} = \alpha (d_i - y_i) f'(w_i^T x) x_j \quad (3.29)$$

การเลือกใช้ฟังก์ชันเป้าหมายดังสมการที่ 3.25 สามารถเรียกได้ว่าการเรียนรู้แบบผลต่าง (Delta learning rule) หรือการเรียนรู้แบบเพอร์เซปตรอนสำหรับฟังก์ชันแอกทิเวชันแบบต่อเนื่อง (Continuous perceptron learning rule)

ในกรณีที่ $f(w_i^T x) = w_i^T x$ สามารถเรียกได้ว่าการเรียนรู้แบบ Windrow Hoff หรือการเรียนรู้แบบ LMS (Least mean square learning rule) นั่นคือจะได้สัญญาณการเรียนรู้

$$l \triangleq d_i - w_i^T x \quad (3.30)$$

สามารถหาค่าเปลี่ยนแปลงของเวกเตอร์ของค่าการเชื่อมโยงดังนี้

$$\Delta w_{ij} = \alpha (d_i - w_i^T x) x_j \quad (3.31)$$

การเปลี่ยนแปลงของค่าการเชื่อมโยงแต่ละค่าหาได้จากความสัมพันธ์

$$\Delta w_{ij} = \alpha (d_i - w_i^T x) x_j; j=1,2,\dots,n \quad (3.32)$$

3.) การเรียนรู้แบบคอมแพ็คทิฟ

การเรียนรู้แบบคอมแพ็คทิฟนี้จะมีนิรอลที่เหมาะสมที่สุดในกลุ่มเพียงหนึ่งเดียวเท่านั้น ที่จะได้รับการกระตุ้น กล่าวคือ ณ เวลาหนึ่งจะมีเพียงนิรอลที่ได้รับการคัดเลือกเท่านั้น ที่จะมีการเรียนรู้ เราอาจจะเรียกว่าการเรียนรู้แบบผู้ชนะ (Winner take all learning) นิรอลที่ได้รับการปรับสอนคือ

นิวรอนที่มีความเหมาะสมในกลุ่ม หรืออาจเป็นนิวรอนที่เหมาะสมที่สุด นั่นคือก็อาจมีนิวรอนเพียงหนึ่งเดียว หรือหลายนิวรอนที่จะมีการเรียนรู้ในเวลาหนึ่ง เวกเตอร์ค่าการเชื่อมโยงของนิวรอนที่เหมาะสม (Winning neuron)

$$w_m = [w_{m1} w_{m2} \dots w_{mn}]^T \quad (3.33)$$

เวกเตอร์ค่าการเชื่อมโยงเปลี่ยนแปลงดังนี้

$$\Delta w_m = \alpha(x - w_m) \quad (3.34)$$

การเปลี่ยนแปลงของค่าการเชื่อมโยงแต่ละค่าหาได้จากความสัมพันธ์

$$\Delta w_{mj} = \alpha(x_j - w_{mj}); j=1,2,\dots, n \quad (3.35)$$

สำหรับเวกเตอร์ของค่าการเชื่อมโยงไปยังนิวรอนอื่นๆ จะไม่เปลี่ยนแปลงนั่นคือ

$$\Delta w_k = 0; k \neq m \quad (3.36)$$

ในการเลือกนิวรอนที่เหมาะสมนั้นจะใช้ระดับแอกทิเวชันเป็นเกณฑ์การตัดสินใจ ตัวอย่างเช่น

$$w_m^T x = \max_{i=1,2,\dots,p} (w_i^T x)$$

โดยที่ p คือจำนวนนิวรอนทั้งหมด

การเรียนรู้แบบคอมแพ็คทิฟจัดว่าเป็นการเรียนรู้ที่ไม่มีการควบคุม ตัวอย่างการประยุกต์ใช้งานได้แก่ใช้ในการจำแนกกลุ่มข้อมูลป้อนเข้าหรือใช้กับโครงข่ายแบบจัดตัวเอง (Self organizing network) เป็นต้น

4.) กฎการเรียนรู้แบบโบลท์มานน์

การเรียนรู้แบบโบลท์มานน์เป็นการเรียนรู้ที่ต่างจากที่กล่าวมา เนื่องจากเป็นขั้นตอนวิธีการเรียนรู้แบบสโตคาสติก ซึ่งประยุกต์มาจากทฤษฎีเทอร์โมไดนามิกส์ โครงข่ายแบบโบลท์มานน์ซึ่งใช้การเรียนรู้แบบโบลท์มานน์ เป็นโครงข่ายแบบย้อนกลับสถานะของนิวรอนจะเป็น เปิด/ปิด

(on/off) คุณสมบัติของโครงข่ายถูกกำหนดด้วยฟังก์ชันพลังงาน (Energy function) E ซึ่งเป็นฟังก์ชันที่บอกความสัมพันธ์ระหว่างนิวรอน ฟังก์ชันพลังงานมีรูปสมการดังนี้

$$E = -\frac{1}{2} \sum_i \sum_{\substack{j \\ j \neq i}} w_{ij} s_i s_j \quad (3.37)$$

โดยที่ s_i คือสถานะของนิวรอน i

w_{ij} คือค่าการเชื่อมโยงระหว่างนิวรอน i กับนิวรอน j

จากสมการที่ 3.37 จะไม่มีการป้อนกลับมาให้ตัวมันเอง ($i \neq j$) การทำงานเริ่มจากการสุ่มเลือกนิวรอนเพื่อทำการปรับสอน การเปลี่ยนแปลงสถานะของนิวรอน i จากปิดไปเปิด หรือจากเปิดไปเป็นปิด ที่อุณหภูมิ T มีค่าความน่าจะเป็นดังนี้

$$w(s_i \rightarrow -s_i) = \frac{1}{1 + \exp(-\Delta E_i / T)} \quad (3.38)$$

โดยที่ ΔE คือพลังงานที่เปลี่ยนแปลง

นิวรอนของโครงข่ายแบบโบลต์มานน์แบ่งออกเป็นนิวรอนที่มองเห็น (Visible neuron) และนิวรอนซ่อน นิวรอนที่มองเห็นจะเชื่อม (Interface) ระหว่างโครงข่ายประสาทภายนอก การทำงานของโครงข่ายมี 2 สถานะ คือ

- 1.) ภาวะมีเงื่อนไข (Clamped condition) ทำให้นิวรอนที่มองเห็นถูกควบคุมจากภายนอก
- 2.) ภาวะอิสระ (Free running condition) ยอมให้นิวรอนทั้งสองแบบทำงานโดยอิสระ

นิวรอนซ่อนจะทำงานที่ภาวะอิสระเสมอ

การเปลี่ยนแปลงค่าการเชื่อมโยงแต่ละค่าหาได้จากความสัมพันธ์

$$\Delta w_{ij} = \alpha(PC_{ij} - PF_{ij}) \quad (3.39)$$

โดยที่ PC_{ij} คือค่าความน่าจะเป็นที่ s_i และ s_j จะอยู่ในภาวะเปิดเมื่อนิวรอนที่มองเห็นอยู่ในภาวะมีเงื่อนไขขณะที่โครงข่ายอยู่ในสถานะสมดุลย์

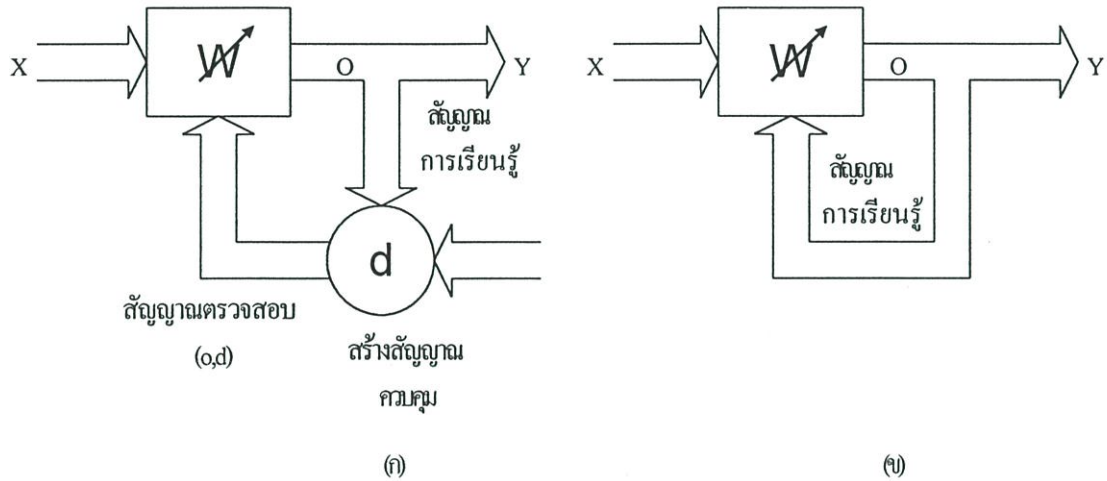
PF_{ij} คือค่าความน่าจะเป็นที่ s_i และ s_j จะอยู่ในภาวะเปิด เมื่อทุกนิวรอนอยู่ในภาวะอิสระขณะที่โครงข่ายอยู่ในสถานะสมดุลย์

การเรียนรู้แบบโบลต์มานน์ จะผ่านกระบวนการทำซ้ำจนกระทั่งจุดที่เรียกว่า สมดุลย์ทางอุณหภูมิ

3.3.2 รูปแบบการเรียนรู้

จากรูปที่ 3.6 แบ่งรูปแบบการเรียนรู้ออกเป็น 3 รูปแบบ คือ

- 1.) การเรียนรู้แบบมีการควบคุม (Supervised learning)
- 2.) การเรียนรู้แบบไม่มีการควบคุม (Unsupervised learning)
- 3.) การเรียนรู้แบบเสริม (Reinforcement learning)



รูปที่ 3.7 รูปแบบการเรียนรู้ (ก) แบบมีการควบคุม (ข) แบบไม่มีการควบคุม

จากรูปที่ 3.7 เป็นรูปแบบของการเรียนรู้ (ก) เป็นการเรียนรู้แบบมีการควบคุม และ(ข) เป็นการเรียนรู้แบบไม่มีการควบคุม การเรียนรู้ทั้งสองรูปแบบก็เพื่อปรับค่าถ่วงน้ำหนัก (W) หรือค่าการเชื่อมโยงให้อยู่ในช่วงที่เหมาะสม

1.) การเรียนรู้แบบมีการควบคุม

การเรียนรู้แบบนี้จะทราบข้อมูลส่งออกที่ต้องการ (o,d relationship generator) ทำหน้าที่เป็นครูที่คอยควบคุมการเรียนรู้ ข้อมูลส่งออกที่ต้องการคือข้อมูลที่ครูใช้เปรียบเทียบกับผลตอบสนองที่ได้รับ ผลที่ได้ ($f(o,d)$) จะใช้ในการปรับทิศทางการเรียนรู้ให้เหมาะสม จากพฤติกรรมดังกล่าวทำให้โครงข่ายประสาทที่มีการเรียนรู้แบบมีการควบคุมจึงต้องมีชุดข้อมูลป้อนเข้าและข้อมูลส่งออก เราเรียกว่า ชุดการปรับสอน (Training set หรือ Training data)

2.) การเรียนรู้แบบไม่มีการควบคุม

การเรียนรู้แบบไม่มีการควบคุมต่างจากแบบมีการควบคุม เนื่องจากเราไม่ทราบข้อมูลส่งออกที่ต้องการ การเรียนรู้จึงใช้ข้อมูลป้อนเข้า ภาวะแวดล้อม และตัวโครงข่ายเองในการปรับแก้หาคำตอบ หากจะเปรียบเทียบกับการเรียนรู้จริงก็คือการศึกษาด้วยตนเองนั่นเอง

1.) การเรียนรู้แบบเสริม

การเรียนรู้แบบเสริมเหมือนกับการเรียนรู้แบบมีการควบคุม[21] ยกเว้นข้อมูลส่งออกแทนที่จะได้ข้อมูลส่งออกที่ถูกตั้งสำหรับข้อมูลป้อนเข้าแต่ละชุด การเรียนรู้แบบเสริมนี้จะให้ค่าข้อมูลส่งออกเป็นระดับหรือคะแนนที่บอกพฤติกรรมของโครงข่ายที่มีต่อข้อมูลป้อนเข้าแต่ละชุด การเรียนรู้แบบเสริมสามารถนำไปประยุกต์ใช้กับงานระบบควบคุมได้เป็นอย่างดี

เนื้อหาในบทที่กล่าวมาเป็นสิ่งที่ควรทราบเกี่ยวกับโครงข่ายประสาท ประกอบด้วยสามส่วนด้วยกันคือ การจำลองนิเวศ ชนิดของฟังก์ชันแอคติเวชัน และกระบวนการเรียนรู้

บทที่ 4

การจัดเส้นทางของโครงข่ายโทรศัพท์โดยใช้ฮอปฟิลด์เน็ต

4.1 ตัวอย่างการประยุกต์ใช้โครงข่ายประสาทของฮอปฟิลด์ (Hopfield)

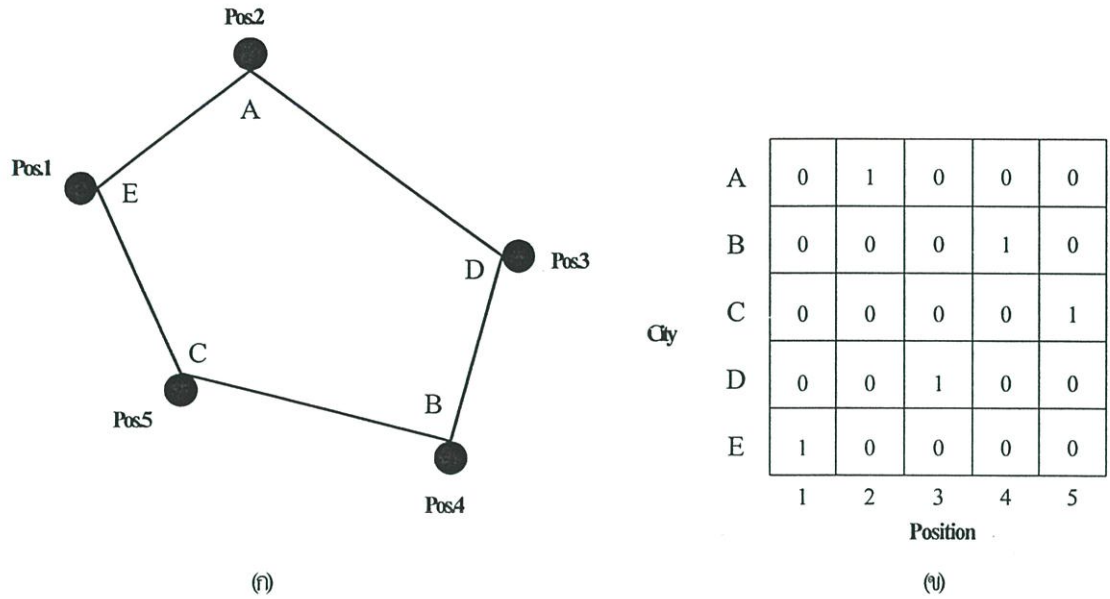
ได้มีการประยุกต์ใช้โครงข่ายประสาทฮอปฟิลด์เน็ต (Hopfield Net) ในการแก้ปัญหาต่างๆ มากมายทั้งโดย ฮอปฟิลด์และคณะ ได้แก่ ปัญหา TSP (Traveling salesman problem) ซึ่งเป็นปัญหาการหารูปแบบการเดินทางผ่านเมืองต่างๆ ให้มีระยะทางในการเดินทางครั้งนั้นสั้นที่สุด [12] ปัญหาการแปลงสัญญาณ A/D (Analog to Digital) ปัญหาการจับเก็บและเข้าถึงหน่วยความจำ (Computer memory) และนักวิจัยอื่นๆ ได้แก่ ปัญหา TSP และการจัดสรรงาน (List-matching หรือ Job assignment) การหาเส้นทางที่สั้นที่สุดสำหรับเครือข่ายการสื่อสาร (Communication network) [21,24] ในขณะนี้จะกล่าวถึงการประยุกต์ใช้โครงข่ายฮอปฟิลด์ ในการแก้ปัญหาโดยทั่วไป คือ ปัญหา TSP และจะได้นำมาประยุกต์ใช้ในการจัดเส้นทางของโครงข่ายโทรศัพท์ ที่จะกล่าวถึงต่อไป

4.2 ปัญหา TSP (Traveling Salesman Problem)

ปัญหา TSP เป็นปัญหาที่ได้รับความสนใจจากนักคณิตศาสตร์กันมานาน แต่ยังไม่มียุติแก่ปัญหาที่มีประสิทธิภาพ ปัญหา TSP เป็นปัญหาการจัดเรียงให้ได้มาซึ่งคำตอบที่เหมาะสม จึงมีคำตอบที่เป็นไปได้มากมายแต่จะมีเพียงคำตอบเดียวที่เป็นคำตอบที่ดีที่สุด และปัญหา TSP ยังจัดเป็นปัญหาแบบ NP-complete (Nondeterministic polynomial time) กล่าวคือใช้เวลาในการแก้ปัญหาเพิ่มขึ้นแบบโพลิโนเมียลตามขนาดของปัญหา เช่น ใช้เวลาในการคำนวณเท่ากับ e^n เมื่อ n คือขนาดของปัญหา

ปัญหา TSP คือปัญหาการหารูปแบบการเดินทางผ่านเมืองต่างๆ ให้มีระยะทางในการเดินทางครั้งนั้นสั้นที่สุด โดยมีข้อจำกัดคือ ต้องเดินทางผ่านเมืองทุกเมือง และแต่ละเมืองต้องเดินทางผ่านเพียงครั้งเดียว ยกเว้นเมืองที่เป็นเมืองเริ่มต้น ในรูปแบบการเดินทางแต่ละครั้งจึงเกิดเส้นทางเป็นวงรอบปิด (Closed loop) การหารูปแบบเส้นทางหรือวงรอบการเชื่อมโยงของเมืองต่างๆ ที่ให้ระยะทางที่สั้นที่สุดนั้น เป็นการพิจารณาเลือกรูปแบบเส้นทางที่เป็นไปได้ทั้งหมดนี้ใช้แก้ปัญหาได้ดี ในกรณีที่จำนวนเมืองมีไม่มากนัก ในกรณีที่จำนวนเมืองเท่ากับ n เมือง รูปแบบการเดินทางที่เป็นไปได้ทั้งหมดเท่ากับ $(n-1)!/2$ ถ้าจำนวนเมือง $n = 20$ รูปแบบการเดินทางที่เป็นไปได้ทั้งหมดประมาณ 6×10^{16} รูปแบบซึ่งนับว่าเป็นค่าที่สูงมาก พิจารณา รูปที่ 4.1 (ก) เป็นตัวอย่างคำตอบที่ดีที่สุดของปัญหา TSP ที่มีจำนวนเมือง 5 เมือง และรูปที่ 4.1 (ข) เป็นการแทนคำตอบที่ได้ใน

รูปเมตริกซ์แสดงความสัมพันธ์ระหว่างเมืองที่แทนด้วยตัวอักษร (A,B,C,D,E) และลำดับของเมืองในเส้นทางที่แทนด้วยตัวเลข (1,2,3,4,5)



รูปที่ 4.1 (ก) ตัวอย่างคำตอบของปัญหา TSP 5 เมือง (ข) การแทนคำตอบด้วยเมตริกซ์

ในปี ค.ศ.1985 Hopfield และ Tank ได้นิยามปัญหา TSP ด้วยฟังก์ชันพลังงานดังนี้

$$E = E_{con1} + E_{con2} + E_{con3} + E_{opt1} \tag{4.1}$$

โดยที่

$$E_{con1} = \frac{A}{2} \sum_x \sum_i \sum_j v_{xi} v_{xj}; i \neq j$$

$$E_{con2} = \frac{B}{2} \sum_i \sum_x \sum_y v_{xi} v_{yj}; X \neq Y$$

$$E_{con3} = \frac{C}{2} \left[\sum_x \sum_i v_{xi} - n \right]^2$$

$$E_{opt1} = \frac{D}{2} \sum_x \sum_y \sum_i d_{xy} v_{xi} (v_{y,i+1} + v_{y,i-1}); X \neq Y$$

A,B,C,D คือค่าคงที่

n คือ จำนวนเมือง

d_{XY} คือ ระยะทางระหว่างเมือง X กับเมือง Y

v_{xi} คือ ค่าของเมตริกซ์ตัวที่ $X, i, v_{xi} = 1$ หมายความว่าเมือง X อยู่ในลำดับที่ i ของการเดินทาง แต่ถ้าเมือง X ไม่ใช่เมืองที่อยู่ในลำดับที่ i ของการเดินทางค่า $v_{xi} = 0$

$i, j, X, Y = 1, 2, \dots, n$

E_{con1}, E_{con2} และ E_{con3} เป็นเงื่อนไขข้อจำกัดของปัญหานั้นคือ E_{con1} เป็นตัวบังคับให้เกิด v เท่ากับ 1 เพียงตัวเดียวในแต่ละแถว (Row), E_{con2} เป็นตัวบังคับให้เกิด v เท่ากับ 1 เพียงตัวเดียวในแต่ละหลัก (Column) ส่วน E_{con3} เป็นตัวบังคับให้เมตริกซ์ของคำตอบมีสมาชิกที่มีค่าเท่ากับ 1 เพียง n ตัวเท่านั้น เงื่อนไขทั้งสามนี้ก็คือ บังคับให้การเดินทางแต่ละครั้งต้องผ่านทุกเมืองและผ่านเพียงครั้งเดียวนั่นเอง สำหรับส่วนสุดท้าย E_{opt1} เป็นผลรวมของระยะทางระหว่างเมืองทั้งหมดในการเดินทางครั้งนั้น หากจะเทียบกับปัญหาการหาคำตอบที่เหมาะสม E_{opt1} คือฟังก์ชันเป้าหมาย และ E_{con1}, E_{con2} และ E_{con3} คือเงื่อนไข

จากฟังก์ชันพลังงานรูปการคูณแบบสเกลาร์ (Scalar product) ของเวกเตอร์ ได้ดังสมการที่

4.2

$$\frac{dE[v(t)]}{dt} = \nabla E^T(v) \dot{v} \quad (4.2)$$

ในสมการที่ 4.2 เมื่อโครงข่ายมีการขยายสูงจะพิจารณาตัดพจน์อินทิเกรตทิ้ง และใช้ตัวห้อยสองตัวเพื่อบอกความสัมพันธ์ระหว่างเมืองกับลำดับของเมือง แทนตัวห้อยตัวเดียวในสมการที่ 4.2 จะได้ฟังก์ชันพลังงานในรูปสมการกำลังสองดังนี้

$$E = \frac{1}{2} \sum_{xi} \sum_{yj} W_{xi,yj} v_{xi} v_{yj} - \sum_{xi} I_{xi} v_{xi} \quad (4.3)$$

ความสัมพันธ์ระหว่างการเปลี่ยนแปลงพลังงานกับระดับแอกทิเวชัน เป็นดังนี้

$$-\nabla E = C \frac{du}{dt} \quad (4.4)$$

จากสมการที่ 4.4 แทนที่เราจะพิจารณาการเปลี่ยนแปลงของฟังก์ชันพลังงานเราอาจจะพิจารณาการเปลี่ยนแปลงของระดับแอกทิเวชัน หรือการเปลี่ยนแปลงของข้อมูลป้อนเข้าภายในของโครงข่ายประสาท

การเปลี่ยนแปลงระดับแอกทิเวชันของโครงข่ายประสาท หรือสมการขับเคลื่อน (Equation of emotion) มีความสัมพันธ์ดังนี้

$$\Delta u_{xi} = \left[\sum_Y^{nf} \sum_j^n W_{Xj,Yj} v_{Yj} - \frac{u_{xi}}{\tau} + I_{xi} \right] \Delta t \quad (4.5)$$

โดยที่ Δu_{xi} คือ ค่าการเปลี่ยนแปลงระดับแอกทิเวชันของนิวรอน Xi

Δt คือ ช่วงเวลาที่มีการเปลี่ยนแปลงระดับแอกทิเวชันของนิวรอน Xi

จากสมการที่ 4.1 และสมการที่ 4.3 จะได้

$$\left. \begin{aligned} W_{xi,yj} = & -A\delta_{xy}(1-\delta_{ij}) - B\delta_{ij}(1-\delta_{xy}) - C - Dd_{xy}(\delta_{j,i+1} + \delta_{j,i-1}) \\ I = Cn', Cn' = & B - \frac{C}{2} \end{aligned} \right\} \quad (4.6)$$

โดยที่ δ_{xy} คือฟังก์ชันโครเนคเกอร์เดลต้า (Kronecker delta function)

$$\delta_{xy} = \begin{cases} 1; X = Y \\ 0; X \neq Y \end{cases}$$

แทนค่าทั้งสองจากสมการที่ 4.6 ลงในสมการที่ 4.5 จะได้

$$\Delta u_{xi} = \left[\begin{aligned} & -\frac{u_{xi}}{\tau} - A \sum_{j \neq i} v_{xi} - B \sum_{Y \neq X} v_{Yi} - C \sum_Y \sum_j v_{Yj} \\ & - D \sum_Y d_{xy} (v_{Y,i+1} - v_{Y,i-1}) + Cn' \end{aligned} \right] \Delta t \quad (4.7)$$

ค่าของระดับแอกทิเวชันในรอบการคำนวณครั้งต่อไปคือ

$$u_{xi}(t+1) = u_{xi}(t) + \Delta u_{xi} \quad (4.8)$$

โดยใช้ฟังก์ชันซิกมอยด์แบบแทนเจนต์ไฮเพอโบลิคเป็นฟังก์ชันแอกทิเวชัน

$$v_{xi} = g(u_{xi}) = \frac{1}{2}[1 + \tanh(\lambda u_{xi})] \quad (4.9)$$

โดยที่ λ คือค่าคงที่สำหรับปรับความชัน (Slop) ของฟังก์ชันแอคติเวชัน

จากสมการที่ 4.9 คือฟังก์ชันซิกมอยด์แบบแทนเจนต์ไฮเปอร์โบลิค เป็นฟังก์ชันแอคติเวชันในการตัดสินใจว่า เป็นการเชื่อมต่อหรือไม่ (0 หรือ 1) ซึ่งโดยปกติฟังก์ชัน $\tanh()$ จะมีค่าตั้งแต่ -1 ถึง +1 แต่เราต้องการการตัดสินใจเพียง สองสถานะเท่านั้น คือ 0 หรือ 1 จึงต้องบวก 1 (ได้ค่า 0-2) แล้วหาร 2 จะได้ค่า 0 ถึง 1 ตามต้องการ)

งานตีพิมพ์ของ Hopfield และ Tank ในปี ค.ศ.1985 ใช้ $\lambda = 50, A = B = D = 500, C = 200, \tau = 1.0$ และ $n' = 15.0$ สำหรับ $10 \leq n \leq 30$ งานของ Hopfield และ Tank นี้ได้ผลลัพธ์ที่ดีที่สุดเพียง 15% สำหรับ $n = 10$ ต่อมาในปี ค.ศ.1988 Brandt และคณะ [24] ได้นำเสนอรูปแบบฟังก์ชันพลังงานสำหรับปัญหา TSP ในรูปแบบที่แตกต่างจากรูปแบบที่นำเสนอโดย Hopfield และ Tank ผลลัพธ์ที่ได้จากการนำเสนอของ Brandt และคณะสามารถแก้ปัญหา TSP ขนาดใหญ่ขึ้นและเปอร์เซ็นต์การได้คำตอบที่ดีที่สุด มากกว่ารูปแบบที่นำเสนอโดย Hopfield และ Tank จากที่กล่าวมาเป็นผลจากการศึกษาพฤติกรรม (Performance) ของฟังก์ชันพลังงานที่ต่างกันในการแก้ปัญหา TSP โดย Proztel และคณะในปี ค.ศ.1993 [21]

4.3 การจัดเส้นทางของโครงข่ายโครงข่ายที่โดยใช้ฮอปฟิลด์เน็ต

การนำเทคโนโลยีคอมพิวเตอร์มาใช้แก้ปัญหาทางวิศวกรรมโดยทั่วไปมักอาศัยสมการทางคณิตศาสตร์เพื่อจำลองปัญหาเหล่านั้น จากนั้นจึงกำหนดขั้นตอนวิธีในการแก้ปัญหาไว้อย่างแน่นอน แต่ในบางปัญหานั้นไม่สามารถหาขั้นตอนวิธีที่มีประสิทธิภาพมาใช้ในการแก้ปัญหาได้ ได้แก่ สภาพปัญหาที่ซับซ้อนมีตัวแปรหรือองค์ประกอบที่เกี่ยวข้องหลายปัจจัย ความสัมพันธ์ระหว่างข้อมูลป้อนเข้าและข้อมูลส่งออกไม่เป็นแบบเชิงเส้น เป็นปัญหาที่มีตัวแปรหลายตัวที่เปลี่ยนแปลงอยู่ตลอดเวลาหรือบางกรณีอาจใช้เวลาในการแก้ปัญหานั้นนานมากเกินไป ทำให้ไม่สามารถประยุกต์ใช้กับงานตามสภาพความเป็นจริงได้ ปัญหาที่มีลักษณะดังกล่าวนี้อาจจะหาคำตอบได้เป็นอย่างดีด้วยเทคนิคโครงข่ายประสาท เนื่องจากคุณสมบัติเด่นของโครงข่ายประสาท คือ ความสามารถในการแก้ปัญหาที่ความสัมพันธ์ระหว่างข้อมูลป้อนเข้าและข้อมูลส่งออกมีความซับซ้อน ความแม่นยำของข้อมูลส่งออก ความรวดเร็วในการประมวลผล และความสามารถในการปรับตัวให้เหมาะสมกับสภาพปัญหาที่เปลี่ยนไป สำหรับปัญหาทางด้านวิศวกรรมโทรคมนาคม ที่มีการประยุกต์ใช้โครงข่ายประสาทเทียมในการแก้ปัญหา ได้แก่ ปัญหาในการจัดเส้นทางของโครงข่ายสื่อสาร[13,14,15]

4.4 การประยุกต์ใช้โครงข่ายฮอปฟิลด์เน็ต

ในบทความนี้ได้เสนอวิธีการจัดเส้นทางของโครงข่ายโทรศัพท์ โดยใช้นิวรอนเน็ตเวิร์กชนิดฮอปฟิลด์เน็ต ซึ่งได้คำนึงถึงค่าความน่าจะเป็นของการติดขัด ดังสมการที่ 4.10 [16-18,26-28]

$$B = \frac{X_L}{X_O} \quad (4.10)$$

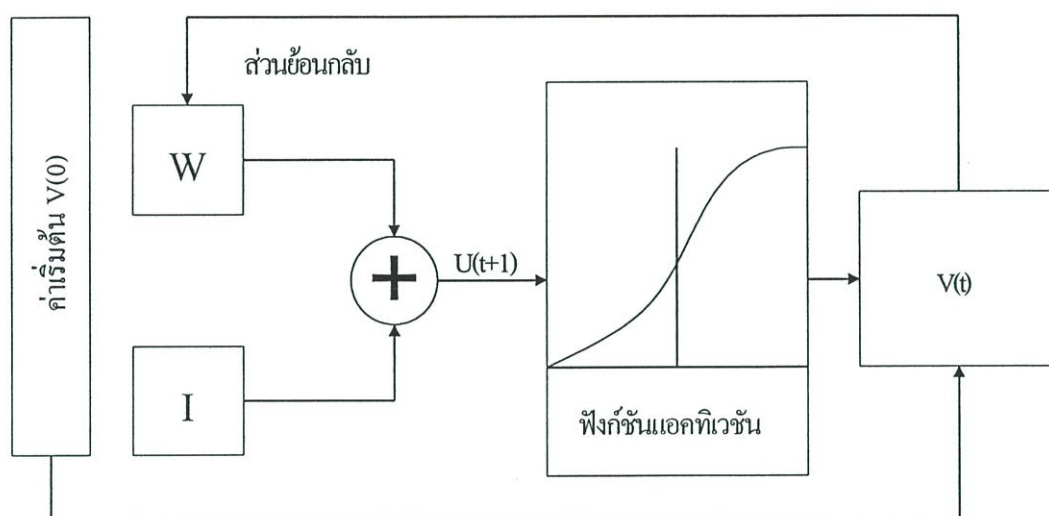
โดยที่ $X_O = X_C + X_L$

B คือค่า Blocking probability

X_O คือค่า การเรียกขอบริการ (Offered Call)

X_L คือค่า การเรียกสูญเสี (Lost Call)

X_C คือค่า การเรียกได้บริการ (Carried Call)



รูปที่ 4.2 ไดอะแกรมของโครงข่ายการจัดเส้นทางที่เหมาะสมโดยใช้ Hopfield Net

พิจารณารูปที่ 4.2 จากลักษณะดังกล่าวสามารถนำมาสร้างฟังก์ชันพลังงานได้ดังนี้

$$E = \frac{A}{2} \sum_x^{nf} [\sum v_{xi} - 1]^2 + \frac{B}{2} \sum_i^n \left[\sum_x^{nf} v_{xi} - 1 \right]^2 + \frac{C}{2} \sum_{xi}^{nf} \sum_i^n v_{xi} (1 - v_{xi}) + \frac{D}{2} \sum_x^{nf} \sum_i^n \sum_j^n L_{ij} v_{xi} v_{xj} \quad (4.11)$$

โดยที่ A, B, C, D คือค่าคงที่

n คือจำนวนโหนดทั้งหมด

nf คือจำนวนเส้นทางทั้งหมด

L_{ij} คือค่าความน่าจะเป็นของการติดขัดระหว่างโหนด i และ j

v_{xi} คือข้อมูลการส่งออกแสดงการเชื่อมโยงระหว่างโหนด i กับเส้นทาง X

ฟังก์ชันโลอะฟูนอนอฟที่มีการขยายสูงและมีสมการ การขับเคลื่อนสำหรับตัวห้อยคู่ (Double subscript) ดังสมการที่ (4.12)

$$E = -\frac{1}{2} \sum_X^{nf} \sum_Y^{nf} \sum_i^n \sum_j^n W_{xi, Yj} v_{Yj} - \sum_X^{nf} \sum_i^n v_{xi} I_{xi}$$

$$\Delta v_{xj} = \left[\sum_Y^{nf} \sum_j^n W_{xi, Yj} v_{Yj} - \frac{u_{xi}}{\tau} + I_{xj} \right] \Delta t \quad (4.12)$$

จากสมการที่ 4.11 และฟังก์ชันโลอะฟูนอนอฟในสมการที่ 4.12 จะได้

$$W_{xi, Yj} = \begin{cases} -A\delta_{XY} - B\delta_{ij} - C\delta_{XY}\delta_{ij} - DL_{ij}\delta_{XY} & \text{เมื่อ } i \in S \\ -B\delta_{ij} - C\delta_{XY}\delta_{ij} - DL_{ij}\delta_{XY} & \text{เมื่อ } i \notin S \end{cases} \quad (4.13)$$

$$I_{xi} = B - \frac{C}{2}$$

โดยที่ S คือกลุ่มโหนดที่เป็นโหนดต้นทางของแต่ละเส้นทางการเรียนรู้ของโครงข่ายที่สร้างขึ้น ได้จากการนำค่าจากสมการที่ 4.13 แทนลงในสมการ การขับเคลื่อนที่ 4.12 จะได้

$$\Delta u_{xi} = \begin{cases} \Delta u_{1xi} + \Delta u_{2xi} & \text{เมื่อ } i \in S \\ \Delta u_{2xi} & \text{เมื่อ } i \notin S \end{cases} \quad (4.14)$$

$$\text{โดยที่ } \Delta u_{1xi} = \left[-A \sum_{l \neq K}^{nf} v_{xi} \right] \Delta t$$

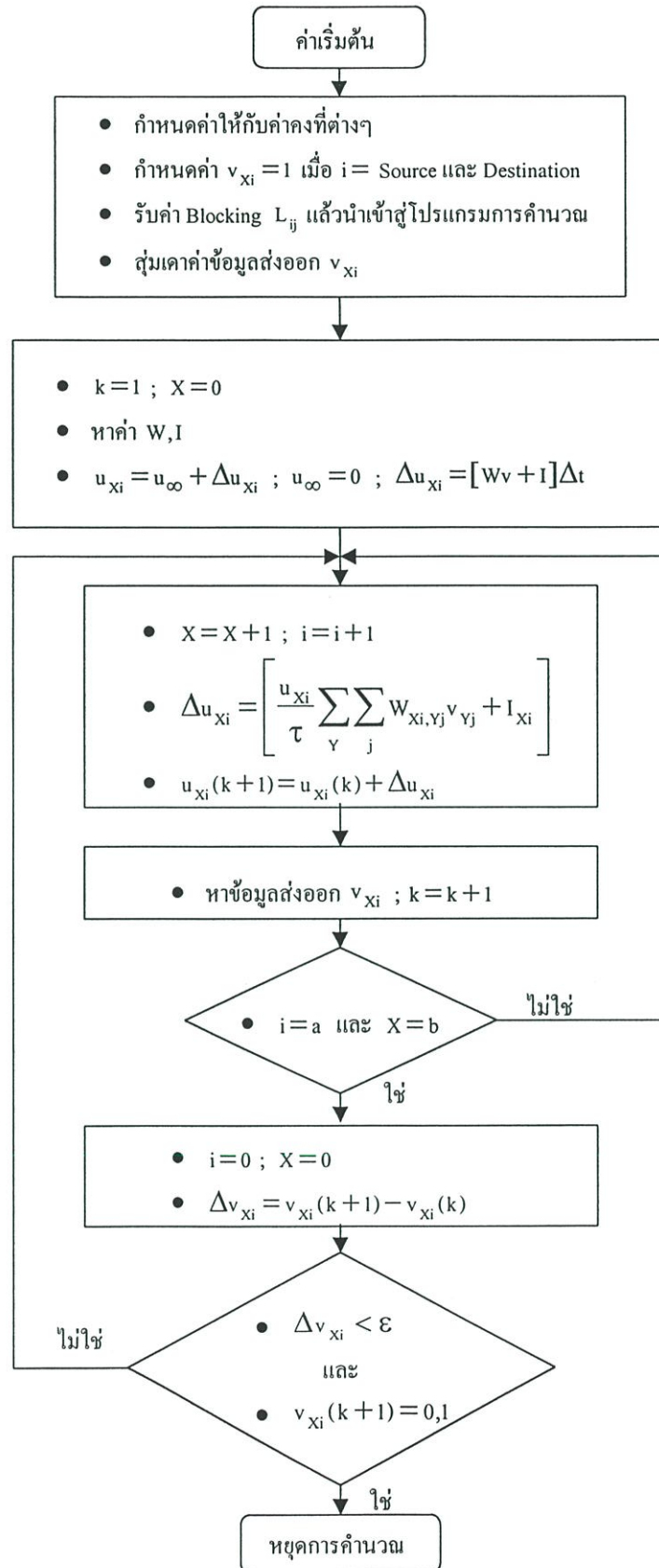
$$\Delta u_{2xi} = \left[-\frac{u_{xi}}{\tau} - B \sum_Y^{nf} v_{Yj} + C v_{xi} - D \sum_j^{nf} L_{ij} v_{xi} + B - \frac{C}{2} \right] \Delta t$$

4.5 กระบวนการเรียนรู้ของโครงข่ายประสาทฮอปฟิลด์เน็ต

กระบวนการเรียนรู้การจัดเส้นทางของโครงข่าย ที่ใช้ในการพัฒนาโปรแกรมคอมพิวเตอร์ ในงานวิจัยนี้เป็นกระบวนการทำซ้ำ โดยเริ่มจากการสุ่มเดาข้อมูลส่งออก ซึ่งจากรูปที่ 4.2 ประกอบด้วยขั้นตอนดังต่อไปนี้

- ขั้นตอนที่ 1
- กำหนดค่าให้กับค่าคงที่ต่างๆ
 - กำหนดให้ $v_{xi} = 1$ เมื่อ $i = \text{Source}$ และ Destination
 - รับค่าการเรียกติดขัดมา (L_{ij}) ที่ได้มาทำการวิเคราะห์และตรวจสอบเพื่อใช้ในการหาค่าการเชื่อมโยง
 - สุ่มเดาค่าข้อมูลการส่งออก (v_{xi})
- ขั้นตอนที่ 2
- $k = 1; i = x = 0$
 - หาค่า W, I
 - หาค่าระดับแอกทิเวชัน $u = u_{\infty} + \Delta u_{xi}; u_{\infty} = 0; \Delta u_{xi} = [Wv + I]\Delta t$
 - หาค่าข้อมูลส่งออก (v_{xi})
- ขั้นตอนที่ 3
- $X = (X + 1); i = (i + 1)$
 - หาค่าการเปลี่ยนแปลงของระดับแอกทิเวชัน
- $$\Delta u_{xi} = \left[-\frac{u_{xi}}{\tau} + \sum_Y \sum_j W_{xi,yj} v_{yj} + I_{xi} \right] \Delta t$$
- ปรับค่าระดับแอกทิเวชัน $u_{xi}(k+1) = u_{xi}(k) + \Delta u_{xi}$
- ขั้นตอนที่ 4
- หาค่าข้อมูลส่งออก $v_{xi}; k = k + 1$
- ขั้นตอนที่ 5
- ตรวจสอบว่า $i = a$ และ $X = b$ หรือไม่
 - ถ้าใช่ให้ $i = 0$ และ $X = 0$ แล้วทำขั้นตอนที่ 6
 - ถ้าไม่ใช่ ให้ทำขั้นตอนที่ 3
- ขั้นตอนที่ 6
- ตรวจสอบเงื่อนไขการหยุดคำนวณว่า $\Delta v_{xi} < \epsilon$ หรือ $v_{xi}(k+1) = 0, 1$ หรือไม่
 - โดยที่ $\Delta v_{xi} = v_{xi}(k+1) - v_{xi}(k)$
 - ถ้าใช่ ให้หยุดการคำนวณ
 - ถ้าไม่ใช่ ให้ทำขั้นตอนที่ 3

จากขั้นตอนวิธีทั้ง 6 ขั้นตอน เราสามารถเขียนไดอะแกรมได้ ดังรูปที่ 4.3



รูปที่ 4.3 ไลออะแกรมของกระบวนการเรียนรู้ของโครงข่ายการจัดเส้นทางให้เหมาะสม

เมื่อพิจารณาฟังก์ชันพลังงานที่นำเสนอในวิทยานิพนธ์จะปรากฏรายละเอียดของกระบวนการเรียนรู้ดังนี้

1.) ค่าคงที่ในขั้นตอนที่ 1 คือ $A, B, C, D, \tau, \Delta t$ และ n' และกำหนดให้ $v_{ix} = 1$ เมื่อ $i =$ Source หรือ Destination พร้อมทั้งหาค่าความน่าจะเป็นของการติดขัดเพื่อกำหนดให้กับ โปรแกรม

$$2.) \text{ ข้อมูลส่งออกที่สุ่มจากขั้นตอนที่ 1 } v_{xi} = \frac{(1 + \delta)}{(n \times nf)}$$

โดยที่ δ คือค่าที่ได้จากการสุ่มเดาเป็นค่าจำนวนน้อยๆ

n เท่ากับ จำนวน โหนดทั้งหมด

nf เท่ากับ จำนวนเส้นทางทั้งหมด

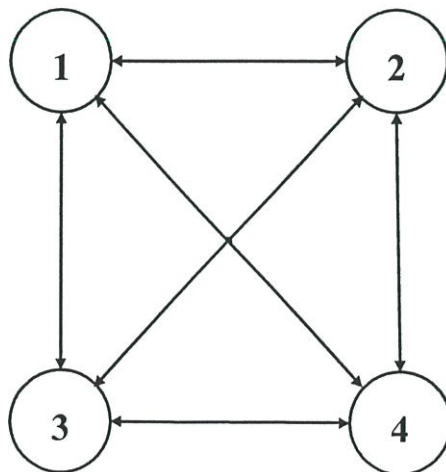
3.) ข้อมูลส่งออกในขั้นตอนที่ 2 และขั้นตอนที่ 4

$$v_{xi} = \frac{1}{2} [1 + \tan(\lambda v_{xi})]$$

4.) $a = n$ และ $b = nf$

4.6 การทดสอบผลการคำนวณโดยใช้ฮอปฟิลด์เน็ต

จากการจัดเส้นทางของโครงข่ายโทรศัพท์โดยการประยุกต์ใช้ฮอปฟิลด์เน็ต ตัวอย่างรูปที่ 4.4 เป็นตัวอย่างของ โมเดลที่ใช้ทดสอบการคำนวณ หมายเลข 1, 2, 3 และ 4 คือชุมสายท้องถิ่น ซึ่งมีเส้นทางเชื่อมต่อถึงกันดังรูป โดยมีความจุของวงจรในแต่ละเส้นทางเท่ากับ 30 วงจร และมีค่าความน่าจะเป็นของการติดขัด ดังตารางที่ 4.1 ตัวอย่างของการทดสอบการจัดเส้นทางโดยวิธีฮอปฟิลด์เน็ต ตัวอย่างนี้ ต้องการเส้นทางของการเรียก จากโหนดที่ 1 ไปยังโหนดที่ 2



รูปที่ 4.4 ตัวอย่างโครงข่ายโทรศัพท์ที่มีจำนวนโหนด ในโครงข่าย 4 โหนด

ตารางที่ 4.1 ค่าความน่าจะเป็นของการติดขัดของโครงข่าย รูปที่ 4.4

เส้นทางเชื่อมโยงระหว่างโหนด	ค่าความน่าจะเป็นของการติดขัด
1-2	0.05
1-3	0.02
1-4	0.01
2-3	0.015
2-4	0.03
3-4	0.03

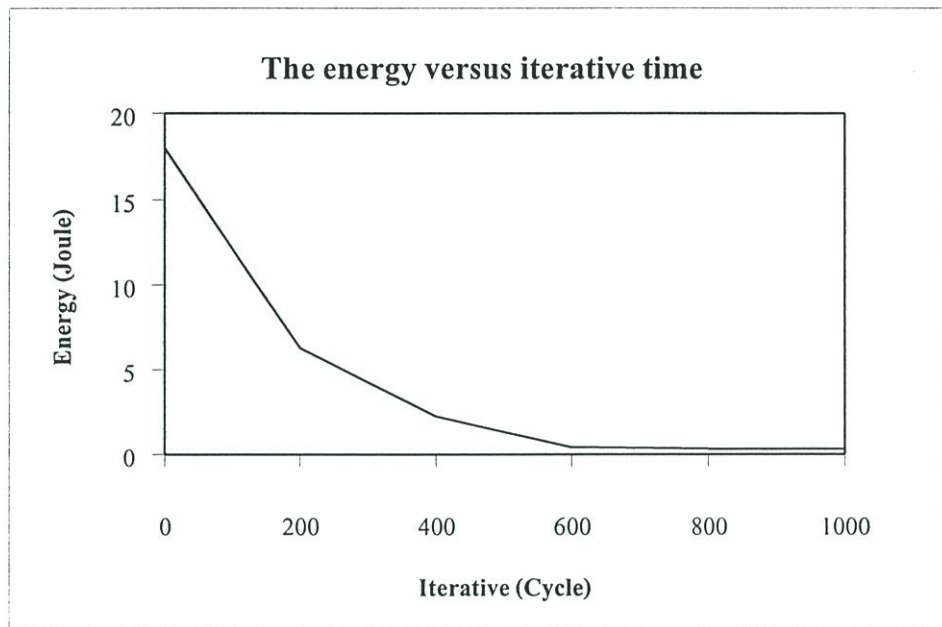
ตารางที่ 4.2 ค่าคงที่ ที่ใช้ในการทดสอบการคำนวณ

A	50
B	50
C	10
D	30
τ	1.0
Δt	0.02
จำนวนรอบ	800

	01	02	03	04
Node 1	1.0	0.0	0.0	0.0
Node 2	0.0	0.0	1.0	0.0
Node 3	0.0	1.0	0.0	0.0
Node 4	0.0	0.0	0.0	0.0

รูปที่ 4.5 เาท์พุทของนิรอล เมื่อต้องการเรียกจากโหนดที่ 1 ไปยังโหนดที่ 2

จากตัวอย่างโครงข่ายในรูปที่ 4.4 หากยังคงใช้วิธีการจัดเส้นทางในแบบเดิม (Overflow และ Load Sharing) เมื่อต้องการติดต่อสื่อสารจากโหนด 1 ไปยังโหนด 2 ซึ่งค่าตอบของเส้นทางที่ได้ก็คือ โหนดที่ 1 → โหนดที่ 2 เพราะว่ามีเส้นทางตรงที่ต่อเชื่อมถึงกันโดยตรง ซึ่งถ้าหากเรามาพิจารณาการจัดเส้นทางแบบฮอปฟิลด์เน็ตที่มีเงื่อนไขในการเลือกเส้นทาง คือเลือกเส้นทางที่มีค่าความน่าจะเป็นของการติดขัดน้อยที่สุด ดังนั้นเอาท์พุทที่ได้ ดังรูปที่ 4.5 จะแปลความหมายออกมาเป็นเส้นทางที่ใช้ในการติดต่อสื่อสารระหว่างโหนดที่ 1 กับโหนดที่ 2 เส้นทางที่ได้ถูกเลือกโดยวิธีฮอปฟิลด์เน็ตคือ โหนดที่ 1 → โหนดที่ 3 → โหนดที่ 2 ซึ่งเป็นเส้นทางที่มีผลรวมของค่าความน่าจะเป็นของการติดขัดน้อยกว่าการจัดเส้นทางในวิธีการแบบเดิม



รูปที่ 4.6 พลังงานของนิวรอลเน็ตเวอร์ค

ตารางที่ 4.3 ผลการทดสอบการคำนวณ

พลังงานเมื่อเริ่มต้นการคำนวณ (จูล)	18.02
พลังงานเมื่อสิ้นสุดการคำนวณ (จูล)	0.32
เส้นทางที่ได้จากการคำนวณ	1-3-2
จำนวนรอบ	800
ค่าความน่าจะเป็นของการติดขัด	0.025

บทที่ 5

การทดลองและผลการทดลอง

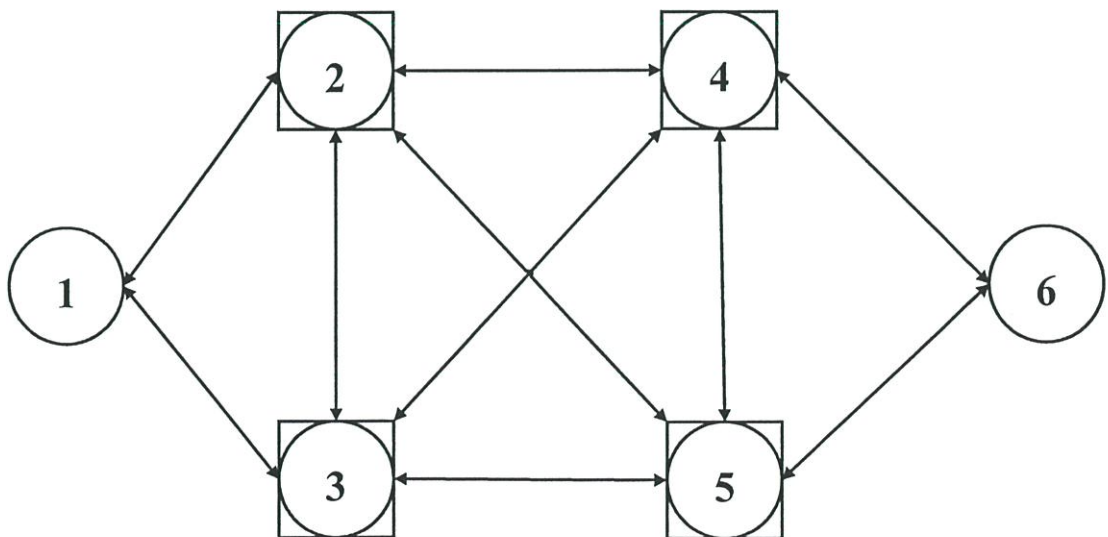
5.1 ขั้นตอนในการทดลอง

การทดลองใช้โปรแกรม Visual Basic จำลองโครงสร้างของนิเวศเน็ตเวิร์ก ตามโมเดลที่ใช้ทดสอบ ดังรูปที่ 5.1 ซึ่งมีขั้นตอนการทดลองดังนี้

- 1) ทดลองการจัดเส้นทางด้วยวิธีฮอปฟิลด์เน็ตจากการจำลองโครงสร้างตามโมเดลที่ได้สร้างขึ้น แล้วใช้คอมพิวเตอร์ในการแสดงผล
- 2) ทดลองการจัดเส้นทางด้วยวิธีแบบ Overflow ตามโมเดลที่ใช้ในการทดสอบ ด้วยการ Simulate จากชุดสาย Test Plant [26]
- 3) ทดลองการจัดเส้นทางด้วยวิธีแบบ Load Sharing ตามโมเดลที่ใช้ในการทดสอบ ด้วยการ Simulate จากชุดสาย Test Plant [26]
- 4) สรุปผลการทดลองและเปรียบเทียบผลการทดลองระหว่างวิธีการที่นำเสนอกับวิธีการเดิม

5.1.1 โมเดลของโครงข่ายที่ใช้ในการทดสอบ

โมเดลของโครงข่ายที่ใช้ในการทดสอบ ได้จำลองโครงสร้างของโครงข่ายที่มีใช้งานจริงในปัจจุบันของ บริษัท ทศท คอร์ปอเรชั่น จำกัด (มหาชน) ดังรูปที่ 5.1 ซึ่งหมายเลข 1 และ 6 คือ ชุดสายท้องถิ่น หมายเลข 2, 3, 4 และ 5 คือชุดสายต่อผ่าน NSC



รูปที่ 5.1 โมเดลของโครงข่ายที่ใช้ในการทดสอบ

จากรูปที่ 5.1 สามารถจัดกลุ่มของโหนดได้เป็น 6 โหนด และมีจำนวนเส้นทางทั้งหมด nf สามารถแทนได้ด้วยเมตริกซ์ v ซึ่งมีขนาด $nf \times n$ และ n คือจำนวนโหนด ถ้าสมาชิก $v_{xi} = 1$ หมายความว่าโหนด i ต่อกับเส้นทาง X แต่ถ้าโหนด i ไม่ต่อกับเส้นทาง X จะได้ $v_{xi} = 0$

5.2 การทดสอบการกำหนดเส้นทาง

การทดสอบการกำหนดเส้นทางให้กับโครงข่ายโทรศัพท์ โดยกำหนดการเรียกจากโหนด 1 ไปยังโหนด 6 ซึ่งแต่ละโหนดให้มีความจุของวงจรเท่ากับ 30 วงจร ในทุกๆ เส้นทาง และมีปริมาณทราฟฟิกแต่ละเส้นทางต่างๆ กัน ดังตารางที่ 5.1 ซึ่งมีค่าของการเรียกคิดขีด ดังตารางที่ 5.2 ส่วนเงื่อนไขของการทดสอบมี 2 สภาวะ คือ ขณะที่โครงข่ายปกติ ทราฟฟิกปกติกับทราฟฟิกหนาแน่น และโครงข่ายขัดข้องแต่ละเส้นทาง คือเส้นทาง 1-3 และ 3-5 ไม่สามารถใช้งานได้

ตารางที่ 5.1 ค่าความหนาแน่นของปริมาณทราฟฟิกของแต่ละเส้นทาง

เส้นทาง	ค่าของทราฟฟิก				
	เบาบาง	ปกติ	หนาแน่น	1-3 ขาด	3-5 ขาด
1-2	15.02	28.85	36.06	51.98	28.85
1-3	15.05	23.13	28.91	0	23.13
2-3	14.08	20.13	25.16	20.13	40.27
2-4	12.85	20.35	25.44	40.49	20.35
2-5	18.42	28.05	29.08	27.07	26.09
3-4	14.23	28.80	29.04	26.03	25.02
3-5	15.22	20.14	25.17	20.14	0
4-5	12.54	20.35	25.44	20.35	20.35
4-6	14.88	20.35	25.44	20.35	20.35
5-6	18.25	31.3	39.12	31.3	31.3

ตารางที่ 5.2 ค่าความน่าจะเป็นของการติดขัดของแต่ละเส้นทาง

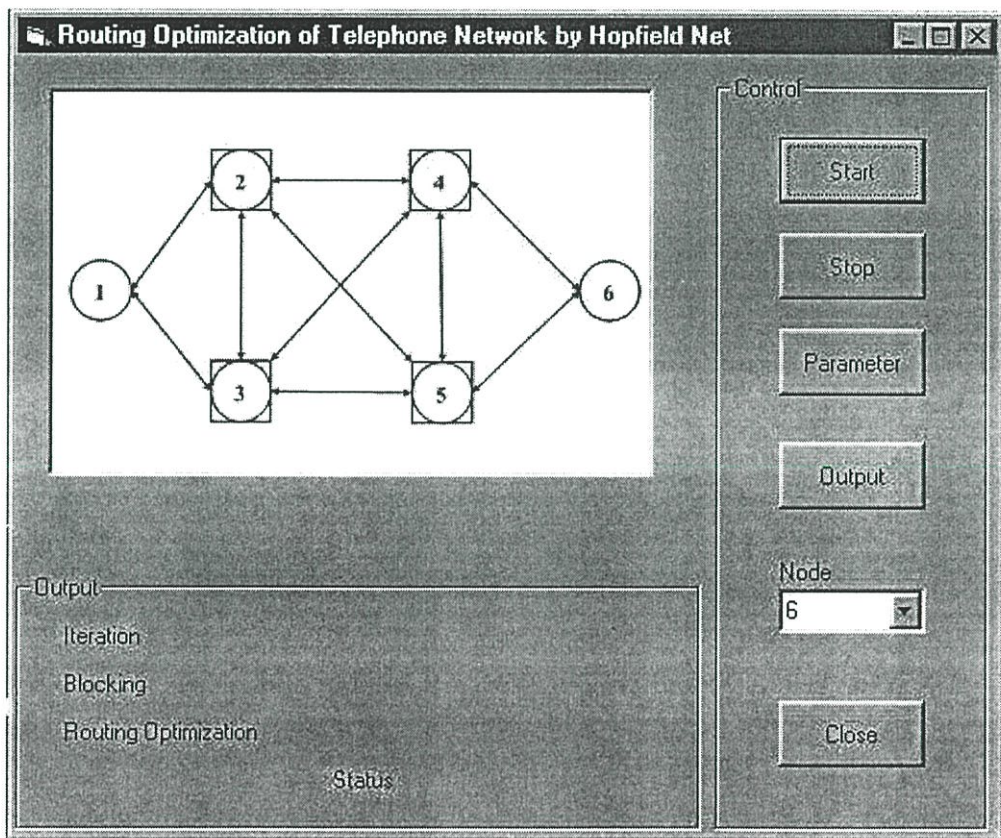
เส้นทาง	ค่าของการเรียกคิดขีด				
	เบาบาง	ปกติ	หนาแน่น	1-3 ขาด	3-5 ขาด
1-2	0.0	0.83	1.04	0.96	0.83
1-3	0.0	0.13	0.16	10	0.13
2-3	0.0	0.13	0.16	0.13	0.27

ตารางที่ 5.2 (ต่อ)

2-4	0.0	0.35	0.44	0.49	0.35
2-5	0.0	0.85	0.94	0.95	0.88
3-4	0.0	0.95	0.98	0.88	0.85
3-5	0.0	0.14	0.17	0.07	10
4-5	0.0	0.35	0.44	0.35	0.35
4-6	0.0	0.35	0.44	0.35	0.35
5-6	0.0	1.3	1.62	0.6	1.16

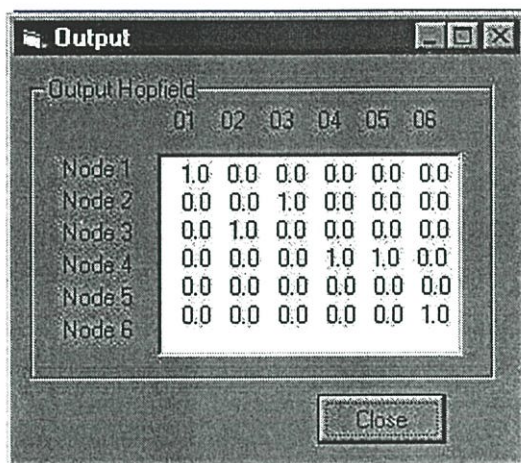
5.3 การทดสอบและผลการทดสอบขณะที่สถานะของโครงข่ายปกติ

ทดสอบขณะที่กราฟฟิคปกติและเบาบาง กับกราฟฟิคมีปริมาณความหนาแน่น (เพิ่มขึ้น 25%) ซึ่งรูปที่ 5.2 แสดงโปรแกรมที่ใช้ในการทดสอบ โครงข่ายประสาทฮอปฟิลด์เน็ต



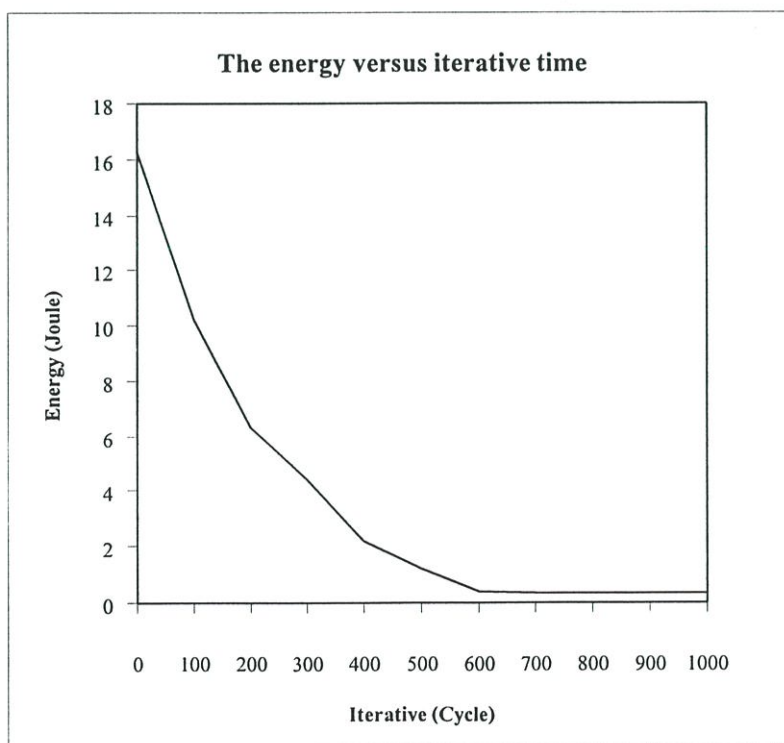
รูปที่ 5.2 รูปแบบโปรแกรมที่ใช้ในการทดสอบ โครงข่ายประสาทฮอปฟิลด์เน็ต

จากรูปโครงข่ายที่ใช้ในการทดสอบจะมีจำนวนโหนดทั้งหมด 6 โหนด ซึ่งทำการทดสอบโดยการเรียกจากโหนดที่ 1 ไปยังโหนดที่ 6 ผลการทดลองการจัดเส้นทางของโครงข่ายโทรศัพท์ให้ดีที่สุดโดยวิธีฮอปฟิลด์เน็ต ขณะที่โครงข่ายปกติ ทราฟฟิกปกติและเบาบาง ได้ผลการจัดเส้นทาง ดังรูปที่ 5.3



	01	02	03	04	05	06
Node 1	1.0	0.0	0.0	0.0	0.0	0.0
Node 2	0.0	0.0	1.0	0.0	0.0	0.0
Node 3	0.0	1.0	0.0	0.0	0.0	0.0
Node 4	0.0	0.0	0.0	1.0	1.0	0.0
Node 5	0.0	0.0	0.0	0.0	0.0	0.0
Node 6	0.0	0.0	0.0	0.0	0.0	1.0

รูปที่ 5.3 เอาท์พุทของนิวรอลเน็ตเวิร์กแสดงผลการจัดเส้นทางโดยวิธีฮอปฟิลด์เน็ต ขณะที่โครงข่ายปกติ ทราฟฟิกปกติและเบาบาง



รูปที่ 5.4 พลังงานของนิวรอลเน็ตเวิร์กขณะที่โครงข่ายปกติทราฟฟิกปกติและเบาบาง

ตารางที่ 5.3 ผลการทดสอบขณะที่โครงข่ายปกติกราฟฟิกปกติและเบาบาง

พลังงานเมื่อเริ่มต้นการคำนวณ (จูล)	16.52
พลังงานเมื่อสิ้นสุดการคำนวณ (จูล)	0.28
เส้นทางที่ได้จากการคำนวณ	1-3-2-4-6
จำนวนรอบ	700
ค่าความน่าจะเป็นของการติดขัด	0.96

จากผลการคำนวณได้ค่าของเอาต์พุตของนิเวศน์เน็ตเวิร์ค ดังรูปที่ 5.3 ซึ่งเมื่อทำการแปลความหมายของเอาต์พุตที่ได้นี้ เส้นทางที่ได้คือ โหนดที่ 1 → โหนดที่ 3 → โหนดที่ 4 → โหนดที่ 6 ซึ่งเป็นเส้นทางที่ดีที่สุด เมื่อนำมาเปรียบเทียบกับวิธีการเดิม ส่วนรูปที่ 5.4 คือค่าของพลังงานเมื่อเริ่มต้นการคำนวณ จนถึงสิ้นสุดการคำนวณ ซึ่งค่าต่างๆ เหล่านี้แสดงให้เห็น ตามตารางที่ 5.3

```

WinFIOL - [#2 D:\Thesis2003\Data Hopfield\Fig 5.5new.cmd]
Channel File Edit Block Search Run Tools Options Window Help
<ANRSP:RC=1000;

ROUTING CASE DATA
OPERATING AREA
RC  CCH BR      PROGRAM ROUTING      SP  MISCELL
1000 YES      P01=1 R=EXC120      MM1 EST=1
          P01=2 R=EXC130      MM1 EST=1

END

<STRSP:R=EXC120&EXC12I&EXC130&EXC13I;

DEVICE STATE SURVEY
R      NDV  NOCC  NIDL  NBLO  RSTAT
EXC120  30   1    29   0    NORES
EXC12I  30   1    29   0    NORES
EXC130  30   0    30   0    NORES
EXC13I  30   0    30   0    NORES

END
  
```

รูปที่ 5.5 ผลการจัดเส้นทางแบบ Overflow ขณะที่โครงข่ายปกติ กราฟฟิกปกติและเบาบาง

```

WinFIOL - [#2 D:\Thesis2003\Data Hopfield\Fig 5.6new.cmd]
Channel File Edit Block Search Run Tools Options Window Help
<ANRSP:RC=1000;

ROUTING CASE DATA
OPERATING AREA
RC   CCH BR      PROGRAM ROUTING          SP  MISCELL

1000 YES RA -50   P01-1 R=EXC120           MM1 EST=1
      RA -50     P01-2 R=EXC130           MM1 EST=1
      RA -50     P02-1 R=EXC130           MM1 EST=1
      RA -50     P02-2 R=EXC120           MM1 EST=1

END

<STRSP:R=EXC120&EXC121&EXC130&EXC131;

DEVICE STATE SURVEY
R      NDV  NOCC  NIDL  NBLO  RSTAT

EXC120 30   0   30   0  NORES
EXC121 30   0   30   0  NORES
EXC130 30   1   29   0  NORES
EXC131 30   1   29   0  NORES

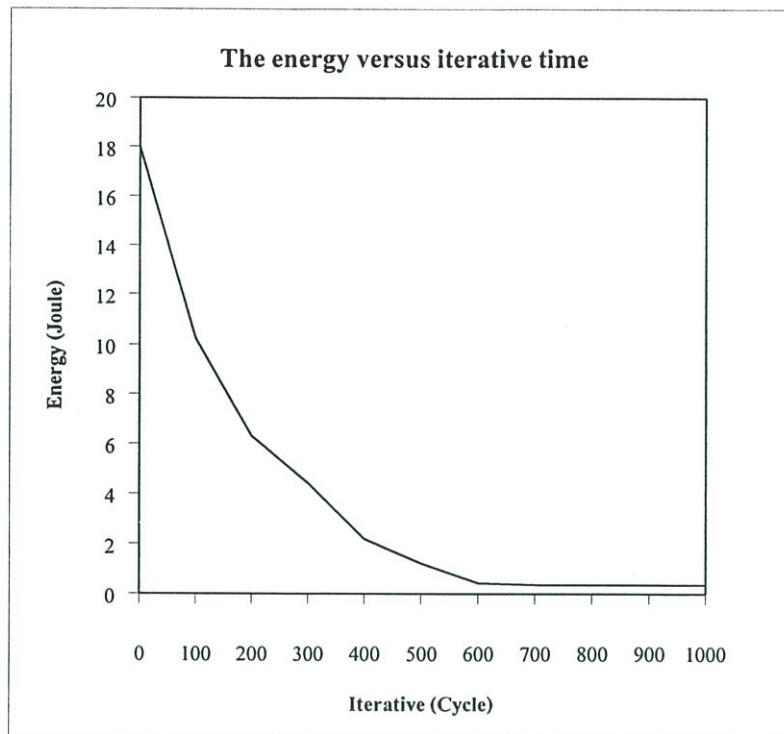
END

```

รูปที่ 5.6 ผลการจัดเส้นทางแบบ Load Sharing ขณะที่โครงข่ายปกติ ทราฟฟิกปกติและเบาบาง

	01	02	03	04	05	06
Node 1	1.0	0.0	0.0	0.0	0.0	0.0
Node 2	0.0	0.0	1.0	0.0	0.0	0.0
Node 3	0.0	1.0	0.0	0.0	0.0	0.0
Node 4	0.0	0.0	0.0	1.0	1.0	0.0
Node 5	0.0	0.0	0.0	0.0	0.0	0.0
Node 6	0.0	0.0	0.0	0.0	0.0	1.0

รูปที่ 5.7 เอาท์พุทของนิรอลเน็ตเวิร์กแสดงผลการจัดเส้นทางโดยวิธีฮอปฟิลด์เน็ต ขณะที่โครงข่ายปกติ ทราฟฟิกหนาแน่น



รูปที่ 5.8 พลังงานของนิรอลเน็ตเวิร์คขณะที่โครงข่ายปกติกราฟฟิกหนาแน่น

ตารางที่ 5.4 ผลการทดสอบขณะที่โครงข่ายปกติกราฟฟิกหนาแน่น

พลังงานเมื่อเริ่มต้นการคำนวณ (จูล)	18.00
พลังงานเมื่อสิ้นสุดการคำนวณ (จูล)	0.35
เส้นทางที่ได้จากการคำนวณ	1-3-2-4-6
จำนวนรอบ	800
ค่าความน่าจะเป็นของการติดขัด	1.2

```

WinFIOL - [#2 D:\Thesis2003\Data Hopfield\Fig 5.9new.cmd]
Channel File Edit Block Search Run Tools Options Window Help
<ANRSP:RC=1000;

ROUTING CASE DATA
OPERATING AREA
RC  CCH BR          PROGRAM ROUTING          SP  MISCELL

1000 YES           P01=1 R=EXC120          MM1 EST=1
                   P01=2 R=EXC130          MM1 EST=1

END

<STRSP:R=EXC120&EXC12I&EXC130&EXC13I;

DEVICE STATE SURVEY
R      NDU  NOCC  NIDL  NBL0  RSTAT
EXC120  30   29    0    1  NORES
EXC12I  30   29    0    1  NORES
EXC130  30    1   27    2  NORES
EXC13I  30    1   27    2  NORES

END

ERICSSON 26-1 Insert Port closed Log F1=Help

```

รูปที่ 5.9 ผลการจัดเส้นทางแบบ Overflow ขณะที่โครงข่ายปกติ ทราฟฟิกหนาแน่น

```

WinFIOL - [#2 D:\Thesis2003\Data Hopfield\Fig 5.10new.cmd]
Channel File Edit Block Search Run Tools Options Window Help
<ANRSP:RC=1000;

ROUTING CASE DATA
OPERATING AREA
RC  CCH BR          PROGRAM ROUTING          SP  MISCELL

1000 YES RA -50     P01=1 R=EXC120          MM1 EST=1
                   P01=2 R=EXC130          MM1 EST=1
                   RA -50     P02=1 R=EXC130          MM1 EST=1
                   P02=2 R=EXC120          MM1 EST=1

END

<STRSP:R=EXC120&EXC12I&EXC130&EXC13I;

DEVICE STATE SURVEY
R      NDU  NOCC  NIDL  NBL0  RSTAT
EXC120  30    1   28    1  NORES
EXC12I  30    1   28    1  NORES
EXC130  30    1   27    2  NORES
EXC13I  30    1   27    2  NORES

END

ERICSSON 27-1 Insert Port closed Log F1=Help

```

รูปที่ 5.10 ผลการจัดเส้นทางแบบ Load Sharing ขณะที่โครงข่ายปกติ ทราฟฟิกหนาแน่น

ตารางที่ 5.5 ผลการทดลองขณะที่มีโครงข่ายปกติ (ทราฟฟิกปกติและเบาบางกับทราฟฟิกหนาแน่น)

	เส้นทางที่ถูกเลือก	
	ทราฟฟิกปกติและเบาบาง	ทราฟฟิกหนาแน่น
Hopfield Net	1-3-2-4-6	1-3-2-4-6
Overflow	1-2-4-6	1-3-5-6
Load Sharing	1-3-5-6	1-3-5-6

จากรูปที่ 5.3 ถึง 5.10 คือผลของการจัดเส้นทางให้ดีที่สุดโดยวิธีฮอปฟิลด์เน็ต, Overflow และ Load Sharing ตามลำดับ ผลที่ได้ดังตารางที่ 5.3 ถึง 5.6 ซึ่งเป็นการจัดเส้นทางขณะที่มีโครงข่ายปกติ ทราฟฟิกปกติและเบาบาง กับทราฟฟิกหนาแน่น

ผลการทดลองที่ได้จากวิธีการจัดเส้นทาง แบบฮอปฟิลด์เน็ต ขณะที่มีโครงข่ายปกติ ทราฟฟิกปกติและเบาบาง กับทราฟฟิกหนาแน่น ผลที่ได้จะเหมือนกันเนื่องจากเงื่อนไขที่ได้สร้างขึ้น คือเส้นทางที่ถูกเลือกจะเป็นเส้นทางที่มีค่าความน่าจะเป็นของการติดขัดน้อยที่สุด เมื่อเทียบกับวิธีการเดิม ส่วนการจัดเส้นทางแบบทราฟฟิกดัน จะเห็นได้ว่าเมื่อทราฟฟิกเพิ่มขึ้น เส้นทางที่ถูกเลือกจะเปลี่ยนไป ดังเห็นได้จากรูปที่ 5.10 ซึ่งทราฟฟิกจะล้นไปยังอีกเส้นทาง แต่ยังคงมีค่าความน่าจะเป็นของการติดขัด มากกว่าวิธีการที่นำเสนอ ดังตารางที่ 5.6

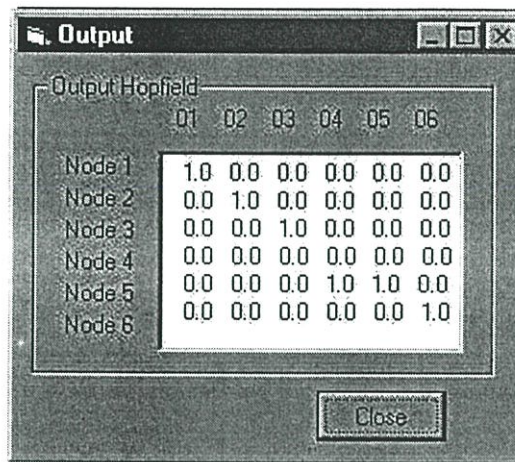
ตารางที่ 5.6 ค่าความน่าจะเป็นของการติดขัดรวมทั้งเส้นทาง ขณะที่มีโครงข่ายปกติ

	ค่าความน่าจะเป็นของการติดขัด	
	ทราฟฟิกปกติและเบาบาง	ทราฟฟิกหนาแน่น
Hopfield Net	0.96	1.2
Overflow	1.53	1.95
Load Sharing	1.57	1.95

5.4 การทดสอบและผลการทดสอบขณะที่สภาวะของโครงข่ายขัดข้อง

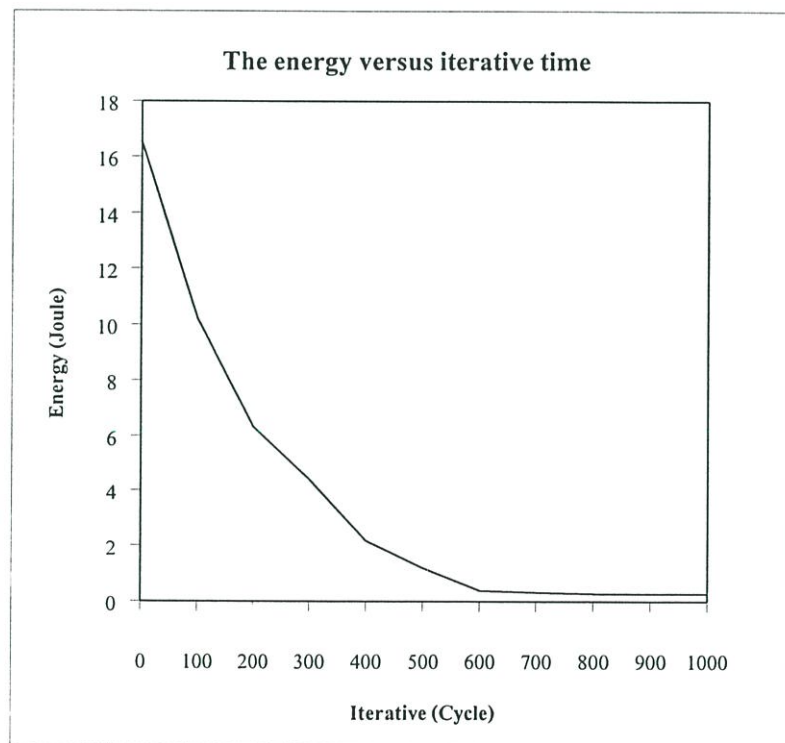
การทดสอบสภาวะของโครงข่ายขัดข้อง ทดสอบที่สภาวะทราฟฟิกปกติเพียงอย่างเดียว แต่เส้นทางของโครงข่ายที่ใช้ในการติดต่อสื่อสาร เส้นทาง 1-3 และ 3-5 นั้นเกิดการล้มเหลวไม่สามารถใช้ในการติดต่อสื่อสารได้ ผลที่ได้ รูปที่ 5.11 ถึงรูปที่ 5.18 สรุปผลการจัดเส้นทางที่ได้ทั้ง 3 วิธีดังตารางที่ 5.7 และ 5.12 ตามลำดับ

5.4.1 ผลการทดสอบขณะที่โครงข่ายจัดข้อ ในเส้นทาง 1-3



	01	02	03	04	05	06
Node 1	1.0	0.0	0.0	0.0	0.0	0.0
Node 2	0.0	1.0	0.0	0.0	0.0	0.0
Node 3	0.0	0.0	1.0	0.0	0.0	0.0
Node 4	0.0	0.0	0.0	0.0	0.0	0.0
Node 5	0.0	0.0	0.0	1.0	1.0	0.0
Node 6	0.0	0.0	0.0	0.0	0.0	1.0

รูปที่ 5.11 เอาท์พุทของนิวรอลเน็ตเวิร์กแสดงผลการจัดเส้นทางขณะที่โครงข่ายจัดข้อในเส้นทาง 1-3



รูปที่ 5.12 พลังงานของนิวรอลเน็ตเวิร์กขณะที่โครงข่ายจัดข้อในเส้นทาง 1-3

ตารางที่ 5.7 ผลการทดลองขณะที่โครงข่ายขัดข้องในเส้นทาง 1-3

พลังงานเมื่อเริ่มต้นการคำนวณ (จูล)	16.55
พลังงานเมื่อสิ้นสุดการคำนวณ (จูล)	0.28
เส้นทางที่ได้จากการคำนวณ	1-2-3-5-6
จำนวนรอบ	800
ค่าความน่าจะเป็นของการติดขัด	1.76

จากรูปที่ 5.11 ถึงรูปที่ 5.14 แสดงการจัดเส้นทางให้ดีที่สุดโดยวิธีฮอปฟิลด์เน็ต, Overflow และ Load Sharing ตามลำดับ ผลที่ได้ดังตารางที่ 5.7, 5.8 และ 5.9 ซึ่งเป็นการจัดเส้นทางขณะที่โครงข่ายขัดข้องในเส้นทาง 1-3 ซึ่งทำให้เส้นทาง 1-3 ไม่สามารถใช้ในการติดต่อสื่อสารได้

```

WinFIOL - [#2 D:\Thesis2003\Data Hopfield\Fig 5.13new.cmd]
Channel File Edit Block Search Run Tools Options Window Help
<ANRSP:RC=1000;
ROUTING CASE DATA
OPERATING AREA
RC CCH BR PROGRAM ROUTING SP MISCELL
1000 YES P01=1 R=EXC120 MM1 EST=1
P01=2 R=EXC130 MM1 EST=1
END
<STRSP:R=EXC120&EXC12I&EXC130&EXC13I;
DEVICE STATE SURVEY
R NDV NOCC NIDL NBLO RSTAT
EXC120 30 1 28 1 NORES
EXC12I 30 1 28 1 NORES
EXC130 30 0 0 30 NORES
EXC13I 30 0 0 30 NORES
END
ERICSSON 26:1 Insert Port closed Log: F1=Help
Microsoft Word - O... Microsoft PowerPoi... WinFIOL = [#2... 20:13
  
```

รูปที่ 5.13 ผลการจัดเส้นทางแบบ Overflow ขณะที่โครงข่ายขัดข้องในเส้นทาง 1-3

```

WinFIOL - [#2 D:\Thesis2003\Data Hopfield\Fig 5.14new.cmd]
Channel File Edit Block Search Run Tools Options Window Help
<AHRSP:RC=1000;

ROUTING CASE DATA
OPERATING AREA
RC   CCH BR          PROGRAM ROUTING          SP MISCELL

1000 YES RA -50      P01=1 R=EXC120          MM1 EST=1
                               P01=2 R=EXC130          MM1 EST=1
                               RA -50 P02=1 R=EXC130          MM1 EST=1
                               P02=2 R=EXC120          MM1 EST=1

END

<STRSP:R=EXC120&EXC121&EXC130&EXC131;

DEVICE STATE SURVEY
R      NDU  NOCC  NIDL  NBLO  RSTAT
EXC120 30   1    28   1    NORES
EXC121 30   1    28   1    NORES
EXC130 30   0    0    30   NORES
EXC131 30   0    0    30   NORES

END

```

รูปที่ 5.14 ผลการจัดเส้นทางแบบ Load Sharing ขณะที่โครงข่ายขัดข้องในเส้นทาง 1-3

ตารางที่ 5.8 ผลการทดลองขณะที่โครงข่ายขัดข้องในเส้นทาง 1-3

	เส้นทางที่ถูกเลือก
Hopfield Net	1-2-3-5-6
Overflow	1-2-4-6
Load Sharing	1-2-4-6

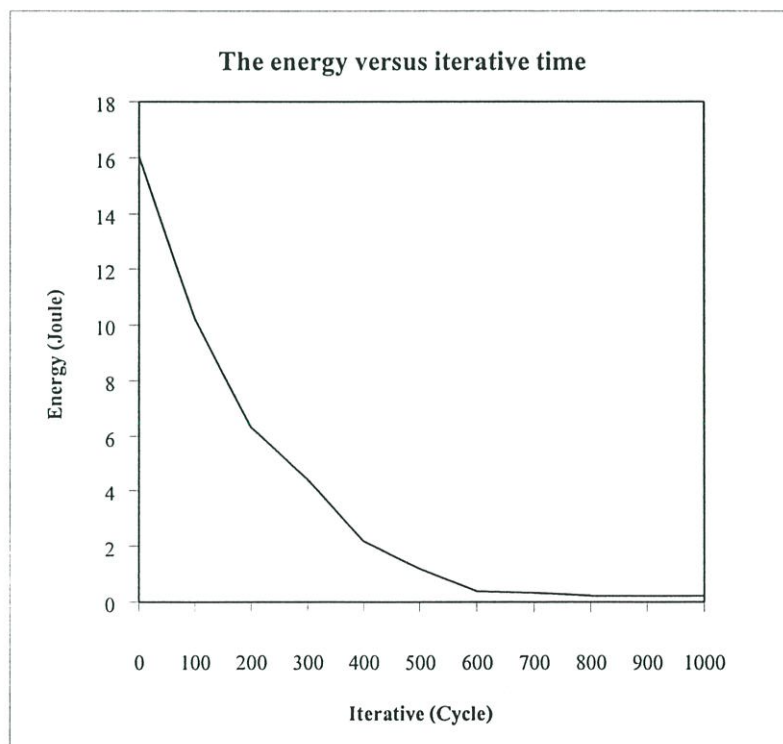
ตารางที่ 5.9 ค่าความน่าจะเป็นของการติดขัดรวมทั้งเส้นทาง ขณะที่โครงข่ายขัดข้องในเส้นทาง 1-3

	ค่าความน่าจะเป็นของการติดขัด ทราฟฟิกปกติ
Hopfield Net	1.76
Overflow	1.8
Load Sharing	1.8

5.4.2 ผลการทดสอบขณะที่โครงข่ายขัดข้อง ในเส้นทาง 3-5

	01	02	03	04	05	06
Node 1	1.0	0.0	0.0	0.0	0.0	0.0
Node 2	0.0	0.0	1.0	0.0	0.0	0.0
Node 3	0.0	1.0	0.0	0.0	0.0	0.0
Node 4	0.0	0.0	0.0	1.0	1.0	0.0
Node 5	0.0	0.0	0.0	0.0	0.0	0.0
Node 6	0.0	0.0	0.0	0.0	0.0	1.0

รูปที่ 5.15 เอาท์พุทของนิวรอลเน็ตเวิร์กแสดงผลการจัดเส้นทางโดยวิธีฮอปฟิลด์เน็ต ขณะที่โครงข่ายขัดข้องในเส้นทาง 3-5



รูปที่ 5.16 พลังงานของนิวรอลเน็ตเวิร์กขณะที่โครงข่ายขัดข้องในเส้นทาง 3-5

ตารางที่ 5.10 ผลการทดลองขณะที่ยังคงจ่ายน้ำในเส้นทาง 3-5

พลังงานเมื่อเริ่มต้นการคำนวณ (จูล)	16.00
พลังงานเมื่อสิ้นสุดการคำนวณ (จูล)	0.22
เส้นทางที่ได้จากการคำนวณ	1-3-2-4-6
จำนวนรอบ	800
ค่าความน่าจะเป็นของการติดขัด	1.1

```

WinFIOI - [#2 D:\Thesis2003\Data Hopfield\Fig 5.17new.cmd]
Channel File Edit Block Search Run Tools Options Window Help
<ANRSP:RC=1000;

ROUTING CASE DATA
OPERATING AREA
RC   CCH BR      PROGRAM ROUTING          SP MISCELL

1000 YES          P01=1 R=EXC120          MM1 EST=1
                   P01=2 R=EXC130          MM1 EST=1

END

<STRSP:R=EXC120&EXC12I&EXC130&EXC13I;

DEVICE STATE SURVEY
R      NDV  NOCC  NIDL  NBLO  RSTAT
EXC120  30   30    0    0   NORES
EXC12I  30   30    0    0   NORES
EXC130  30    1   29    0   NORES
EXC13I  30    1   29    0   NORES

END
ERICSSON 26:1 Insert Port closed Log: F1=Help
Microsoft Word - D... Microsoft PowerPot... WinFIOI - [#2 ... 20:14

```

รูปที่ 5.17 ผลการจัดเส้นทางแบบ Overflow ขณะที่ยังคงจ่ายน้ำในเส้นทาง 3-5

```

WinFIOL - [#2 D:\Thesis2003\Data Hopfield\Fig 5.18new.cmd]
Channel File Edit Block Search Run Tools Options Window Help
<ANRSP:RC=1000;

ROUTING CASE DATA
OPERATING AREA
RC  CCH BR          PROGRAM ROUTING          SP  MISCELL

1000 YES RA -50    P01=1 R=EXC120          MM1 EST=1
                P01=2 R=EXC130          MM1 EST=1
                RA -50    P02=1 R=EXC130          MM1 EST=1
                P02=2 R=EXC120          MM1 EST=1

END

<STRSP:R=EXC120&EXC12I&EXC130&EXC13I;

DEVICE STATE SURVEY
R      NDV  NOCC  NIDL  NBLO  RSTAT
EXC120 30   1   29   0   NORES
EXC12I 30   1   29   0   NORES
EXC130 30   0   30   0   NORES
EXC13I 30   0   30   0   NORES

END

```

รูปที่ 5.18 ผลการจัดเส้นทางแบบ Load Sharing ขณะที่โครงข่ายขัดข้องในเส้นทาง 3-5

จากรูปที่ 5.15 ถึงรูปที่ 5.18 ผลการจัดเส้นทางให้ดีที่สุดโดยวิธีฮอปฟิลด์เน็ต, Overflow และ Load Sharing ตามลำดับ ผลที่ได้ดังตารางที่ 5.10, 5.11 และ 5.12 ซึ่งเป็นการจัดเส้นทางขณะที่โครงข่ายขัดข้องในเส้นทาง 3-5 ซึ่งทำให้เส้นทาง 3-5 ไม่สามารถใช้ในการติดต่อสื่อสารได้

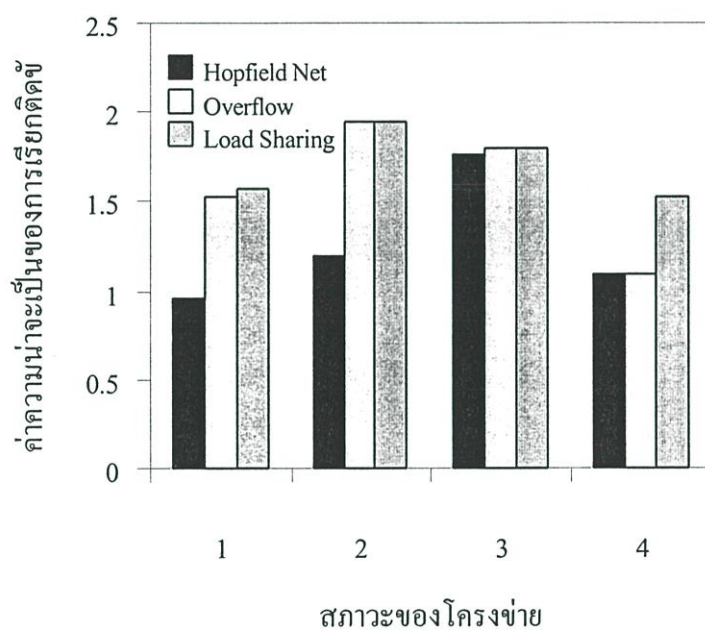
ตารางที่ 5.11 ผลการทดลองขณะที่โครงข่ายขัดข้องในเส้นทาง 3-5

	เส้นทางที่ถูกเลือก
Hopfield Net	1-3-2-4-6
Overflow	1-3-2-4-6
Load Sharing	1-2-4-6

จากการจัดเส้นทางโดยใช้ฮอปฟิลด์เน็ต เปรียบเทียบกับการจัดเส้นทางโดยวิธี Overflow และ Load Sharing จะเห็นได้ว่าเส้นทางที่ได้จากการจัดเส้นทางโดยใช้ฮอปฟิลด์เน็ต จะได้ค่าของการเรียกติดขัดโดยรวมแล้วมีค่าน้อยกว่าทั้ง 2 วิธี ดังรูปที่ 5.19

ตารางที่ 5.12 ค่าความน่าจะเป็นของการติดขัดรวมทั้งเส้นทาง ขณะที่โครงข่ายขัดข้อง
ในเส้นทาง 3-5

	ค่าความน่าจะเป็นของการติดขัด
	กราฟฟิคปกติ
Hopfield Net	1.1
Overflow	1.1
Load Sharing	1.53



รูปที่ 5.19 ผลการเปรียบเทียบค่าของการเรียกติดขัดทั้ง 3 วิธี

จากรูปที่ 5.19 รูปกราฟเป็นสถานะของโครงข่าย ซึ่งสถานะต่างๆ สามารถอธิบายได้ดังนี้

- (1.) คือสถานะโครงข่ายปกติกราฟฟิคปกติ
- (2.) คือสถานะโครงข่ายปกติกราฟฟิคหนาแน่น
- (3.) คือสถานะโครงข่ายขัดข้องในเส้นทาง 1-3
- (4.) คือสถานะโครงข่ายขัดข้องในเส้นทาง 3-5

บทที่ 6

บทสรุป

6.1 สรุปผลการทดลอง

จากผลการทดลองจะเห็นได้ว่าการจัดเส้นทางด้วยวิธีฮอปฟิลด์เน็ต จะได้เส้นทางที่ดีที่สุดตามเงื่อนไขที่ได้สร้างขึ้น ไม่ว่าปริมาณกราฟฟิกจะเปลี่ยนแปลงหรือมีบางเส้นทางเกิดล้นเหลวไม่สามารถใช้งานได้ ก็จะสามารถจัดหาเส้นทางสำรองให้ ซึ่งเส้นทางที่ได้ถูกเลือก จะมีค่าผลรวมของค่าความน่าจะเป็นของการติดขัด (Blocking probability) น้อยกว่าวิธีการแบบเดิม (Overflow และ Load Sharing) โดยเฉลี่ยแล้วน้อยกว่าวิธีการเดิม 30% ขึ้นไป ซึ่งในวิธีการแบบเดิมจะมีค่าความน่าจะเป็นของการติดขัดมากกว่าวิธีการของฮอปฟิลด์เน็ต ดังผลของการทดลองที่ได้แสดงให้เห็นในบทที่ 5 แต่วิธีการของฮอปฟิลด์เน็ต จากแบบจำลองโครงข่ายที่ใช้ในการทดลองนี้ จะใช้เส้นทางในการสื่อสารมากกว่าเนื่องจาก เงื่อนไขที่ได้สร้างขึ้น ซึ่งก็คือ เส้นทางที่ถูกเลือกให้เป็นเส้นทางที่ใช้ในการติดต่อสื่อสารระหว่างโหนดต้นทางกับโหนดปลายทางนั้น จะต้องเป็นเส้นทางที่มีค่าความน่าจะเป็นของการติดขัดรวมทั้งเส้นทางน้อยที่สุด ซึ่งเส้นทางที่ได้จากการทดลองนั้น จะใช้เป็นเส้นทางในการติดต่อสื่อสารได้อย่างมีประสิทธิภาพ

6.2 ปัญหาและข้อเสนอแนะ

ในการใช้ฮอปฟิลด์เน็ตมาประยุกต์ใช้กับปัญหาในเรื่องการหาค่าที่ดีที่สุด (Optimization) นี้ จำเป็นที่จะต้องสร้างสมการพลังงานที่ประกอบขึ้นจากสมการเงื่อนไขต่างๆ ในการคำนวณ ค่าคงที่ต่างๆ ที่กำหนดให้กับสมการเงื่อนไขเป็นเรื่องที่สำคัญมาก สาเหตุเพราะหากไม่สามารถกำหนดค่าให้เหมาะสมกับปัญหาได้แล้วพลังงานของนิวรอลเน็ตเวอร์กจะไม่เข้าสู่ค่าที่ดีที่สุด และการที่สมการพลังงานของนิวรอลเน็ตเวอร์กประกอบขึ้นจากสมการเงื่อนไขหลายๆ เทอมแล้ว จะทำให้ปัญหาในการคำนวณมีโอกาสที่จะได้รับคำตอบที่ผิดพลาดได้มาก สำหรับปัญหาในเรื่องของคำตอบของเอาท์พุทนิวรอล ที่จะต้องนำมาแปลความหมายเป็นเส้นทางที่ใช้ในการติดต่อสื่อสารนั้น จะเห็นว่าคำตอบที่ซ้ำกันในแถวเดียวกัน แต่อย่างไรก็ตาม การที่มีคำตอบที่ซ้ำกันนี้ไม่มีผลต่อ ค่าความน่าจะเป็นของการติดขัดของเส้นทางที่คำนวณได้แต่อย่างใด รวมไปถึงปัญหาของเส้นทางที่ใช้ในการติดต่อสื่อสารของแบบวิธีการที่นำเสนอนี้ จะใช้เส้นทางมากกว่าแบบวิธีการเดิม เป็นเพราะเงื่อนไขที่ได้สร้างขึ้น ซึ่งจากในวิทยานิพนธ์ฉบับนี้ได้เลือกเส้นทางที่มีค่าความน่าจะเป็นของการติดขัดในเส้นทางที่ได้ถูกเลือกให้เป็นเส้นทางที่ใช้ในการติดต่อสื่อสาร มีค่าความน่าจะเป็นของการติดขัดน้อยกว่าเส้นทางอื่นๆ

เอกสารอ้างอิง

- [1] Laurene Fausette, "Fundamentals of Neural Network Architecture Algorithm, and Applications", Prentice-Hall 1994.
- [2] Li Min Fu, "Neural Network in Computer Intelligence", McGraw-Hill 1994.
- [3] Simon Haykin, "Neural Network-A Comprehensive Foundation", Macmillan College Publishing Co., 1994
- [4] S.Y.Kung, "Digital Neural Networks", Prentice-Hall, 1994.
- [5] Shun-Ichi Amari, "Mathematical Foundations of Neurocomputing", *Proceedings of IEEE*, vol.78, No.9, September 1990.
- [6] S. Abe, "Theories of hopfield neural network", International Joint Conference on Neural Network 1, pp.557-564, 1989.
- [7] Brow, D.E, et al., "Rail network routing and scheduling using simulated annealing., International Joint Conference on Neural Network 1, pp.557-564, 1992.
- [8] K.W Cheung, and T. Lee, "Neural networks for global optimum", International Joint Conference on Neural Network 1, pp.53-63, 1992.
- [9] M. Collectt, and W. Pedrycz, "Application of neural networks for routing in telecommunication networks", Global Telecommunication Conference 2, pp.1001-1006, 1991.
- [10] T. Fritsch and W.Mandel, "Communication networks routing using neural network numerical aspect and alternative approaches ", International Joint Conference on Neural Network 1, pp.752-757, 1991.
- [11] I.S.Gerds, "Application of genetic algorithm to the problem for free routing aircraft" , Proceeding of the first IEEE Conference on Evaluationing Computation 1, pp.536-541, 1994.
- [12] J.J.Hopfield and D.W.Tank, "Neural computation of decisions in optimization problems," *Biological.Cybernetics.*, pp.141-152, 1985.
- [13] S.L.Lee, and S.Chang, "Neural network for communication routing with unreliable component", *IEEE Transaction on Neural Network* 4, pp 584-564, 1993.
- [14] H.E.Rauch and T. Winarske "Neural Networks for Routing Communication Traffic" *IEEE Contr.Syst.Mag.*, pp.26-31, Apr.1988.
- [15] J.F.Wieselthier, and Ephremindes, "A Neural networks approach to routing without

- interference in multihop radio network ”, *IEEE Transaction on Communication* 42., pp.166-177, 1994.
- [16] ITU-T E521, “Calculation of the number of circuit in a group carrying overflow traffic,” 1985.
- [17] ITU-T E525, “Designing Network to Control Grade of Service,” 1985.
- [18] NTT, Network Planning, 1990.
- [19] NEC PRACTICE, “Load sharing,” NEC ND-56427-001(E) Issue 4, June 1992.
- [20] J.M.Zurada, “Introduction to artificial neural system”, Singapore:West Publishing Company.,1992.
- [21] M.T.Hagan, H.B. Demuth, and M.Beal, “Neural network design.”, Boston:PWS Publishing Co.,1996
- [22] L.Fausett, “Fundamentals of neural network , architectures,algorithms,and applications.” New Jersey:Prentice-Hall,Inc.,1994.
- [23] S.Haykin, “Neural networks a comprehensive foundation.”, New Jersey:Macmillan Publishing Company.,1994
- [24] S. Abe, and A.H.Gee, “Global convergence of the Hopfield neural network with nonzero diagonal elements.”, *IEEE Transactions on Circuits and Systems-II* ,Analog and Digital Signal Processing 42 (January 1995):pp 39-45.
- [25] R.D.Brandt, Wang, Y.A.J. Laub, and S.K.Mitra, “Alternative networks for solving the traveling salesman problem and the list – matching problem”, *Proceedings IEEE International Conference neural network* , pp. 333-340. ,July 1988
- [26] กองระบบซอฟต์แวร์ชุมสาย ส่วนระบบชุมสาย ฝ่ายโครงข่ายโทรคมนาคม บริษัท ทศท คอร์ปอเรชั่น จำกัด (มหาชน) “ข้อมูลกราฟฟิคแบบกราฟฟิคเส้นและแบบแบ่งโหนด”, พ.ศ.2544-2547.
- [27] หน่วยวิชากราฟฟิค องค์กรการโทรศัพท์แห่งประเทศไทย, หลักสูตร “การพยากรณ์ปริมาณกราฟฟิค”, พ.ศ. 2539
- [28] มนูญ สุขเกษม “กราฟฟิคและโครงข่ายโทรศัพท์ ” ,ภาควิชาวิศวกรรมโทรคมนาคม, คณะวิศวกรรมศาสตร์, มิถุนายน 2546
- [29] D.O.Hebb, “The Organization of Behavior : A neuropsychological theory.” ,New York:Wiley, (1949).

ภาคผนวก ก

โปรแกรมการจัดเส้นทางให้ดีที่สุดโดยวิธีฮอปฟิลด์เน็ต

```
Public nnode
```

```
Public lc
```

```
Public ff
```

```
Public Function tanhyp(ByVal data As Double) As Double
```

```
    tanhyp = (Exp(data) - Exp(-data)) / (Exp(data) + Exp(-data))
```

```
End Function
```

```
Public Function gfunc(ByVal data As Double)
```

```
    gfunc = 0.5 * (1 + tanhyp(data))
```

```
End Function
```

```
Private Sub CloseMainCmd_Click()
```

```
End
```

```
End Sub
```

```
Private Sub Combo1_Click()
```

```
    n = Combo1.ListIndex
```

```
    nnode = Combo1.List(n)
```

```
For i = 1 To nnode
```

```
    For j = 1 To nnode
```

```
        v(i, j) = "0.0"
```

```
        v4(i, j) = "0.0"
```

```
    Next j
```

```
Next i
```

```
Select Case nnode
```

```
Case 4
```

```
    Outpfrm2.Text1.Height = 1115
```

Outpfrm2.Text1.Width = 1695
Outpfrm2.Frame1.Height = 2055
Outpfrm2.Frame1.Width = 3015
Outpfrm2.Width = 3555
Outpfrm2.Height = 3390
Outpfrm2.Command1.Left = 1800
Outpfrm2.Command1.Top = 2400

Outpfrm2.Label5.Visible = False
Outpfrm2.Label6.Visible = False
Outpfrm2.Label7.Visible = False
Outpfrm2.Label8.Visible = False
Outpfrm2.Label9.Visible = False
Outpfrm2.Label10.Visible = False
Outpfrm2.Label11.Visible = False
Outpfrm2.Label12.Visible = False
Outpfrm2.Label13.Visible = False
Outpfrm2.Label14.Visible = False
Outpfrm2.Label15.Visible = False
Outpfrm2.Label16.Visible = False
Outpfrm2.Label17.Visible = False

Outpfrm2.Label25.Visible = False
Outpfrm2.Label26.Visible = False
Outpfrm2.Label27.Visible = False
Outpfrm2.Label28.Visible = False
Outpfrm2.Label29.Visible = False
Outpfrm2.Label30.Visible = False
Outpfrm2.Label31.Visible = False
Outpfrm2.Label32.Visible = False
Outpfrm2.Label33.Visible = False
Outpfrm2.Label34.Visible = False

Outpfrm2.Label35.Visible = False
Outpfrm2.Label36.Visible = False
Outpfrm2.Label37.Visible = False

Case 6

Outpfrm2.Text1.Height = 1515
Outpfrm2.Text1.Width = 2295
Outpfrm2.Frame1.Height = 2295
Outpfrm2.Frame1.Width = 3495
Outpfrm2.Width = 3930
Outpfrm2.Height = 3420
Outpfrm2.Command1.Left = 2280
Outpfrm2.Command1.Top = 2520

Outpfrm2.Label7.Visible = False
Outpfrm2.Label8.Visible = False
Outpfrm2.Label9.Visible = False
Outpfrm2.Label10.Visible = False
Outpfrm2.Label11.Visible = False
Outpfrm2.Label12.Visible = False
Outpfrm2.Label13.Visible = False
Outpfrm2.Label14.Visible = False
Outpfrm2.Label15.Visible = False
Outpfrm2.Label16.Visible = False
Outpfrm2.Label17.Visible = False

Outpfrm2.Label27.Visible = False
Outpfrm2.Label28.Visible = False
Outpfrm2.Label29.Visible = False
Outpfrm2.Label30.Visible = False
Outpfrm2.Label31.Visible = False
Outpfrm2.Label32.Visible = False

Outpfrm2.Label33.Visible = False
Outpfrm2.Label34.Visible = False
Outpfrm2.Label35.Visible = False
Outpfrm2.Label36.Visible = False
Outpfrm2.Label37.Visible = False

Case 17

Outpfrm2.Text1.Height = 4155
Outpfrm2.Text1.Width = 6135
Outpfrm2.Frame1.Height = 5055
Outpfrm2.Frame1.Width = 7255
Outpfrm2.Width = 7575
Outpfrm2.Height = 6240
Outpfrm2.Command1.Left = 5760
Outpfrm2.Command1.Top = 5280

Outpfrm2.Label7.Visible = True
Outpfrm2.Label8.Visible = True
Outpfrm2.Label9.Visible = True
Outpfrm2.Label10.Visible = True
Outpfrm2.Label11.Visible = True
Outpfrm2.Label12.Visible = True
Outpfrm2.Label13.Visible = True
Outpfrm2.Label14.Visible = True
Outpfrm2.Label15.Visible = True
Outpfrm2.Label16.Visible = True
Outpfrm2.Label17.Visible = True

Outpfrm2.Label27.Visible = True
Outpfrm2.Label28.Visible = True
Outpfrm2.Label29.Visible = True
Outpfrm2.Label30.Visible = True

```

Outpfrm2.Label31.Visible = True
Outpfrm2.Label32.Visible = True
Outpfrm2.Label33.Visible = True
Outpfrm2.Label34.Visible = True
Outpfrm2.Label35.Visible = True
Outpfrm2.Label36.Visible = True
Outpfrm2.Label37.Visible = True

```

```
End Select
```

```
'MsgBox "node" + nnode
```

```
End Sub
```

```
Private Sub Command1_Click()
```

```
v4(1, 1) = "1.0"
```

```
v4(2, 3) = "1.0"
```

```
v4(3, 2) = "1.0"
```

```
Outpfrm2.Visible = True
```

```
'Outpfrm2.Text1.Text = " " & v4(1, 1) & " " & v4(1, 2)
```

```
Outpfrm2.Text1.Text = " " & v4(1, 1) & " " & v4(1, 2) & " " & v4(1, 3) & " " & v4(1, 4) +
vbCrLf & " " & v4(2, 1) & " " & v4(2, 2) & " " & v4(2, 3) & " " & v4(2, 4) + vbCrLf & " "
& v4(3, 1) & " " & v4(3, 2) & " " & v4(3, 3) & " " & v4(3, 4) + vbCrLf & " " & v4(4, 1) & "
" & v4(4, 2) & " " & v4(4, 3) & " " & v4(4, 4) + vbCrLf
```

```
End Sub
```

```
Private Sub Form_Load()
```

```
Print
```

```
lc = 0
```

```
ff = 0
```

```
nnode = 6
```

```
For i = 1 To nnode
```

```
    For j = 1 To nnode
```

```
        v(i, j) = "0.0"
```

```
    Next j
```

```
Next i
```

```
Outpfrm2.Text1.Height = 1515
```

```
Outpfrm2.Text1.Width = 2295
```

```
Outpfrm2.Frame1.Height = 2295
```

```
Outpfrm2.Frame1.Width = 3495
```

```
Outpfrm2.Width = 3930
```

```
Outpfrm2.Height = 3420
```

```
Outpfrm2.Command1.Left = 2280
```

```
Outpfrm2.Command1.Top = 2520
```

```
Outpfrm2.Label7.Visible = False
```

```
Outpfrm2.Label8.Visible = False
```

```
Outpfrm2.Label9.Visible = False
```

```
Outpfrm2.Label10.Visible = False
```

```
Outpfrm2.Label11.Visible = False
```

```
Outpfrm2.Label12.Visible = False
```

```
Outpfrm2.Label13.Visible = False
```

```
Outpfrm2.Label14.Visible = False
```

```
Outpfrm2.Label15.Visible = False
```

```
Outpfrm2.Label16.Visible = False
```

```
Outpfrm2.Label17.Visible = False
```

```
Outpfrm2.Label27.Visible = False
```

```
Outpfrm2.Label28.Visible = False
```

```
Outpfrm2.Label29.Visible = False
Outpfrm2.Label30.Visible = False
Outpfrm2.Label31.Visible = False
Outpfrm2.Label32.Visible = False
Outpfrm2.Label33.Visible = False
Outpfrm2.Label34.Visible = False
Outpfrm2.Label35.Visible = False
Outpfrm2.Label36.Visible = False
Outpfrm2.Label37.Visible = False
```

```
End Sub
```

```
Private Sub Label6_Click()
```

```
    If ff = 0 Then
```

```
        routefail(1, 3) = 10
```

```
        Label6.Caption = Fail
```

```
        Label6.BackColor = &HFF&
```

```
        Label6.ForeColor = &H80000012
```

```
        ff = 1
```

```
    Else
```

```
        routefail(1, 3) = 0
```

```
        Label6.Caption = Normal
```

```
        Label6.BackColor = &H80000004
```

```
        ff = 0
```

```
    End If
```

```
    '-----
```

```
    '-----
```

```
    Select Case routefail(1, 3)
```

Case 10

```

For i = 1 To nnode
  For j = 1 To nnode
    v(i, j) = "0.0"
  Next j
Next i

```

Case Else

```

For i = 1 To nnode
  For j = 1 To nnode
    v(i, j) = "0.0"
  Next j
Next i

```

End Select

End Sub

Private Sub OutpCmd_Click()

```
'v(1, 1) = Str(v(1, 1))
```

```
Outpfrm2.Visible = True
```

```

Outpfrm2.Text1.Text = " " & v4(1, 1) & " " & v4(1, 2) & " " & v4(1, 3) & " " & v4(1, 4) +
vbCrLf & " " & v4(2, 1) & " " & v4(2, 2) & " " & v4(2, 3) & " " & v4(2, 4) + vbCrLf & " "
& v4(3, 1) & " " & v4(3, 2) & " " & v4(3, 3) & " " & v4(3, 4) + vbCrLf & " " & v4(4, 1) & "
" & v4(4, 2) & " " & v4(4, 3) & " " & v4(4, 4) + vbCrLf

```

```
'Outpfrm2.Visible = True
```

```

'Outpfrm2.Text1.Text = " " & v(1, 1) & " " & v(1, 2) & " " & v(1, 3) & " " & v(1, 4) & " "
& v(1, 5) & " " & v(1, 6) + vbCrLf & " " & v(2, 1) & " " & v(2, 2) & " " & v(2, 3) & " " &
v(2, 4) & " " & v(2, 5) & " " & v(2, 6) + vbCrLf & " " & v(3, 1) & " " & v(3, 2) & " " & v
(3, 3) & " " & v(3, 4) & " " & v(3, 5) & " " & v(3, 6) + vbCrLf & " " & v(4, 1) & " " & v(4,

```

```
2) & " " & v(4, 3) & " " & v(4, 4) & " " & v(4, 5) & " " & v(4, 6) + vbCrLf & " " & v(5, 1)
& " " & v(5, 2) & " " & v(5, 3) & " " & v(5, 4) & " " & v(5, 5) & " " & v(5, 6) + vbCrLf &
" " & v(6, 1) & " " & v(6, 2) & " " & v(6, 3) & " " & v(6, 4) & " " & v(6, 5) & " " & v(6,
```

```
6) + vbCrLf
```

```
End Sub
```

```
Private Sub ParamCmd_Click()
```

```
Paramfrm.Visible = True
```

```
End Sub
```

```
Private Sub StartCmd_Click()
```

```
'Hopfield Net
```

```
Dim node As Integer
```

```
Dim nfeed As Integer
```

```
'Neural network parameter
```

```
'Step1: Define constant variable
```

```
A = Paramfrm.Text1.Text
```

```
B = Paramfrm.Text2.Text
```

```
C = Paramfrm.Text3.Text
```

```
D = Paramfrm.Text4.Text
```

```
lamda = 0.05
```

```
dt = 0.02
```

```
tau = 1
```

```
erro = 0.0001
```

```
node = Combo1.Text
```

```
u(0, 0) = -0.014
```

```
u0 = 0.02
```

```
ndash = node + 2
```

```
'-----
```

```
'Initiation
```

```
'-----
```

```
'Cities distance (dxy)
```

```
For i = 1 To node
```

```
    dxy(i, i) = 0
```

```
Next i
```

```
For X = 1 To node - 1
```

```
    For Y = X + 1 To node
```

```
        dxy(X, Y) = Int((Rnd + 1) * 10)
```

```
        dxy(Y, X) = dxy(X, Y)
```

```
    Next Y
```

```
Next X
```

```
Outpfrm2.Visible = True
```

```
Outpfrm2.Text1.Text = " " & dxy(1, 1) & " " & dxy(1, 2) & " " & dxy(1, 3) & " " & dxy(1, 4) & " " & dxy(1, 5) & " " & dxy(1, 6) & vbCrLf & " " & dxy(2, 1) & " " & dxy(2, 2) & " " & dxy(2, 3) & " " & dxy(2, 4) & " " & dxy(2, 5) & " " & dxy(2, 6) & vbCrLf & " " & dxy(3, 1) & " " & dxy(3, 2) & " " & dxy(3, 3) & " " & dxy(3, 4) & " " & dxy(3, 5) & " " & dxy(3, 6) & vbCrLf & " " & dxy(4, 1) & " " & dxy(4, 2) & " " & dxy(4, 3) & " " & dxy(4, 4) & " " & dxy(4, 5) & " " & dxy(4, 6) & vbCrLf & " " & dxy(5, 1) & " " & dxy(5, 2) & " " & dxy(5, 3) & " " & dxy(5, 4) & " " & dxy(5, 5) & " " & dxy(5, 6) & vbCrLf & " " & dxy(6, 1) & " " & dxy(6, 2) & " " & dxy(6, 3) & " " & dxy(6, 4) & " " & dxy(6, 5) & " " & dxy(6, 6) & vbCrLf
```

```
'find u(x,i)
```

```
For X = 1 To node
```

```
    For i = 1 To node
```

```
        u(X, i) = u(0, 0) + ((Rnd - 1) / 10 * u0)
```

```
    Next i
```

```
Next X
```

```

'-----
'Iteration
'-----

'u(x,i)=u(x,i)+du(x,i)
'du(x,i)=-u(x,i)-termA-termB-termC-termC

sumvx = 0
sumvy = 0
sumvyj = 0
sumdv = 0

For X = 1 To node
  For i = 1 To node

    'find termA
    For j = 1 To node
      If j = i Then
        j = j + 1
      Else
        End If
      sumvx = sumvx + v(X, j)
    Next j
    termA = A * sumvx

    'find termB
    For Y = 1 To node
      If Y = X Then
        Y = Y + 1
      Else
        End If
      sumvy = sumvy + v(Y, i)

```

```

Next Y
termB = B * sumvy

'find termC
For Y = 1 To node
  For j = 1 To node
    sumvyj = sumvyj + v(Y, j)
  Next j
Next Y
termC = c * sumvyj

'find termD
For Y = 1 To node
  dv = dxy(X, Y) * (v(Y, i + 1) + v(Y, i - 1))
  sumdv = sumdv + dv
Next Y
termD = D * sumdv
'Debug.Print termD

W = -termA - termB - termC - termD
Ix = c * ndash

'find activation level u(x,i)
'u(0,0)=0
du(X, i) = (-u(X, i) + W + Ix) * dt

u(X, i) = u(X, i) + du(X, i)

'find data out v(x,i)

v(X, i) = gfunc(u(X, i) * lamda)
'Debug.Print v(x, i, 1)

```

```
Next i
Next X
DoEvents

'-----

sumWV = 0

step3:

For k = 1 To 20 'round for adept error
For X = 1 To nfeed
  For i = 1 To node
    s = 0
    sumvy = 0
    sumvyj = 0
    sumlv = 0

    'node i is an original node
    Select Case X
      Case 1
        Select Case i
          Case 1
            s = 1

          Case Else
            s = 0

        End Select
      Case 2
        Select Case i
          Case 3
```

s = 1

Case Else

s = 0

End Select

Case 3

Select Case i

Case 4

s = 1

Case Else

s = 0

End Select

End Select

'find termA

'original node

'ss(1) = 1

'ss(2) = 3

' ss(3) = 4

For l = 1 To nfeed

If l = X Then

l = l + 1

Else

End If

' ll = ss(l)

'termA = A * v(X, ll, k)

Next l

'find termB

For Y = 1 To nfeed

If Y = X Then

Y = Y + 1

Else

End If

' sumvy = sumvy + v(Y, i, k)

Next Y

termB = B * sumvy

'find termC

For Y = 1 To nfeed

For j = 1 To node

' sumvyj = sumvyj + v(Y, j, k)

Next j

Next Y

termC = c * sumvyj

'find termD

For j = 1 To node

' lv = ploss(i, j) * v(X, j, k)

sumlv = sumlv + lv

Next j

termD = D * sumlv

If s = 1 Then

W = -termA - termB - termC - termD

ElseIf s <> 1 Then

W = -termB - termC - termD

End If

Ix = c * ndash

```

'find activation level u(x,i)

    ' du(X, i) = (-u(X, i) / tau + W + Ix) * dt
    'Debug.Print x, i
    'Debug.Print du(x, i)
    ' u(X, i) = u(X, i) + du(X, i)

'find data out v(x,i)
    ' v(X, i, k + 1) = gfunc(u(X, i) * lamda)
    ' dv(X, i) = v(X, i, k + 1) - v(X, i, k)
    'Debug.Print v(x, i, 1)

Next i
Next X

'check error
'if dv < erro then
'break
'Else

Next k

k = k - 1
For X = 1 To nfeed
For i = 1 To node
    'Text1.Text = Text1.Text & " " & "v(" & X & "," & i & "," & k & ") = " & v(X, i, k) &
vbCrLf

    'Text1.Text = Text1.Text & " " & "deltav(k) = " & dv(x, i) & vbCrLf
Next i
Next X

```

```
lc = 0
nnact = 1
For i = 1 To nmode * 10
lc = lc + 1
Next i
v(1, 1) = 11
End Sub

Private Sub StopCmd_Click()
c = 0
End Sub
```

ภาคผนวก ข

ผลงานวิจัยในระหว่างการศึกษาที่ได้รับการตีพิมพ์เผยแพร่



ISSN 0125-1724

วิศวกรรม

ลาดกระบัง

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

LADKRABANG ENGINEERING JOURNAL

ปีที่ 20 ฉบับที่ 1

มีนาคม 2546

1.	วงจรรวมต้านทานแบบลอยตัวที่ควบคุมด้วยกระแสใช้มอสทรานซิสเตอร์ สมนึก หัวหาญ กอบชัย เดชหาญ ไชโย ธรรมรัตน์	1
2.	วงจรรยกกำลังที่ใช้วงจรรายพาดกระแสควบคุมด้วยกระแส วิจิตรา เพ็ชรกิจ สิทธิพร เพ็ชรกิจ วิวัฒน์ กิรานนท์	7
3.	วงจรมัลติเพล็กซ์แบบลอยตัวโดยใช้วงจรรวมสลีเยอร์แบบมอสเฟตและการประยุกต์ ภัทรา เพียรชอบ อมร จิรเสรีอมรกุล วัลลภ สุระกำพลธร	13
4.	การศึกษาเปรียบเทียบวงจรฟลายแบคคอนเวอร์เตอร์ในโหมดการนำกระแสแบบต่อเนื่อง (CCM) และแบบไม่ต่อเนื่อง (DCM) ศุภชัย ทอวิมานพร ชนินทร์ บุญลักษณ์นุสรณ์	19
5.	เอทีเอ็มสวิตช์ที่มีบัพเฟอร์ทางด้านขาเข้าซึ่งมีนโยบายการจัดลำดับความสำคัญของข้อมูลภายใต้กราฟฟิกแบบคล้ายคลึงตัวเอง ปิณฑพงษ์ ประภามณฑล สุวิพล สิทธิชีวภาค เกียรติกร วงศ์โรจนภรณ์	25
6.	การจัดเส้นทางของโครงข่ายโทรศัพทให้ดีที่สุดโดยวิธีฮอปฟิลด์เน็ต ฉัตรชัย วัฒนพันธ์ มนูญ สุขเกษม ถวิล พึ่งมา	31
7.	การปรับปรุงคุณภาพของแต่ละคลาสที่ได้จากตัวจำแนกความเหมือนสูงสุด สมบัติ อีสระวานิชย์ พุศศักดิ์ ชิวสุวิทย์	37
8.	การแปลงภาพระดับสีเทาเป็นภาพสองระดับโดยใช้วิธีของเอนโทรปีด้วยบอร์ดสตาร์ทเตอร์คิท TMS 320C31 เพชร ใจหาญ พุศศักดิ์ ชิวสุวิทย์	43
9.	การแปลงเวฟเล็ดและโครงข่ายประสาทเทียมสำหรับแบ่งประเภทฟอลต์ในสายส่ง อรุณพล เก่าพิทักษ์กุล ศุภี บรรจงจิตร สมชาติ จิรวิภากร สุธีวัฒน์ นงนุช	49
10.	การระบุตำแหน่งฟอลต์ในสายส่งโดยใช้การแปลงเวฟเล็ดและโครงข่ายประสาทเทียม อรุณพล เก่าพิทักษ์กุล ศุภี บรรจงจิตร สมชาติ จิรวิภากร อัครชัย บรรจงศิลป์	55
11.	การหาฟอลต์ในสถานีไฟฟ้าแรงสูงหนองจอกโดยใช้โครงข่ายประสาทเทียม พูนิชจร บุญมี มารุต ตันติเดชมงคล สมชาติ จิรวิภากร นิทัศน์ กฤษณจินดา	61
12.	การวิเคราะห์ฟอลต์ที่ตลาดหม้อแปลงไฟฟ้า 3 เฟสโดยใช้โปรแกรม ATP/EMTP จิตร์ อินทร์พยุ่ง มณฑล ลีลาจินดาไกรฤกษ์ บุญเลิศ สือเจย ปัญญา มาลีวัตร สิริวิช ทัดสวน	67
13.	การวิเคราะห์ผลการออกแบบหม้อแปลงจำหน่ายเชิงเศรษฐศาสตร์ บุญเลิศ สือเจย ปัญญา มาลีวัตร สมศักดิ์ สิริโพรานานนท์ ศุภี บรรจงจิตร	73
14.	การออกแบบราคาที่เหมาะสมของหม้อแปลงจำหน่ายโดยใช้เงินดิกอัลกอริธึม ชวรัตน์ เก่งธรรมกิจ ต่อพงศ์ ภูมิภาค ชัยวุฒิ ฉัตรอุทัย	79
15.	การออกแบบโปรแกรมวิเคราะห์การใช้ไฟฟ้าของโรงงานอุตสาหกรรม มารุต ตันติเดชมงคล สมชาติ จิรวิภากร	85
16.	การป้องกันโหลดเกินในระบบไฟฟ้ากำลังโดยวิธีการปรับตารางการป้อนแหล่งจ่ายและไลน์สวิตซ์ซิง สิทธิเดช วชิราศรีศิริกุล มณฑล ลีลาจินดาไกรฤกษ์	91
17.	การวิเคราะห์และจำลองพฤติกรรมของมอเตอร์ไฟฟ้ากระแสตรงแบบไม่มีแปรงถ่าน นิติพงษ์ สมไชยวงศ์ ประภาส อุดคภูมิพันธ์	97
18.	Semantic Query for Content-Based Image Retrieval Using Genetic Algorithm Cutleya Duangate Orachat Chitsobhuk Suphamit Chittayasothorn	103
19.	Robust Inverse Response Compensator Pratch Tontirittiphol Mongkol Janchookiat Kiattisak Kumwachara	109
20.	อิทธิพลของตำแหน่งการฉีดอากาศทุติยภูมิต่ออุณหภูมิการเผาไหม้ในเตาเผาฟลูอิดไดซ์เบดแบบอากาศหมุนวน นิยม ศรีศิริสิทธิ์กุล พงษ์เจต พรหมวงศ์ ภูรินทร์ อัครกุลธร	115

การจัดเส้นทางของโครงข่ายโทรศัพท์ที่ดีที่สุด โดยวิธีฮอปฟิลด์เน็ต

Routing Optimization of Telephone Network by Hopfield Net

ฉัตรชัย วัฒนพันธุ์ มนูญ สุขเกษม ถวิล พึ่งมา

ภาควิชาวิศวกรรมโทรคมนาคม คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง กรุงเทพมหานคร 10520

บทคัดย่อ

บทความนี้เสนอผลการศึกษาเกี่ยวกับการประยุกต์ใช้โครงข่ายประสาทเทียม (Neural Network) ชนิดค่าที่เหมาะสม (Optimization) ที่ชื่อฮอปฟิลด์เน็ต (Hopfield Net) ในการจัดเส้นทางของโครงข่ายโทรศัพท์ที่ดีที่สุด ตามเงื่อนไขที่ได้สร้างขึ้น โดยทำการทดลองจำลองโครงข่ายโทรศัพท์ที่ความหนาแน่นของปริมาณกราฟฟิกที่สภาวะกราฟฟิกปกติ และกราฟฟิกมีการเปลี่ยนแปลง ในเส้นทางที่ใช้ติดต่อสื่อสาร รวมทั้งในสภาวะที่เส้นทางบางเส้นทางในโครงข่ายเกิดล้มเหลวไม่สามารถใช้งานได้ จากผลการทดลองแสดงให้เห็นว่า ความน่าจะเป็นการเรียกติดขัดมีค่าน้อยลง 10% เมื่อเปรียบเทียบกับวิธีการแบบกราฟฟิกล้น (Overflow) และแบบแบ่งโหลด (Load sharing)

Abstract

This paper presents an application of the neural network with the optimization called "Hopfield Net" in routing of telephone network system. This method optimizes the routing with specific conditions. In the simulation, consist of 3 cases from low, normal and high traffic densities, offered to the network. The results from the simulation showed that the congestion in the network is 10% lower compared with Overflow method and Load sharing method.

1. บทนำ

ในปัจจุบันการจัดเส้นทางของโครงข่ายโทรศัพท์ ที่ใช้วิธีการจัดเส้นทาง แบบกราฟฟิกล้น (Overflow) เส้นทางของการเรียก จะไปยังเส้นทางแรกก่อน เมื่อวงจรเส้นทางแรกเต็ม กราฟฟิกล้นจะมายังเส้นทางเพื่อเลือก หรือไม่ก็จะเป็นลักษณะของการแบ่งโหลด (Load sharing) ดังนั้นการเลือกเส้นทางในขณะที่มีเส้นทางหลายเส้นทางต่อถึงกันระหว่างโหนดต้นทางไปยังโหนดปลายทางนั้น จึงไม่จำเป็นต้องกำหนดเส้นทางให้ว่าถ้าวงจรเส้นทางแรกเต็มแล้ว กราฟฟิกล้นจะไปอีกเส้นทาง หรือจัดกราฟ

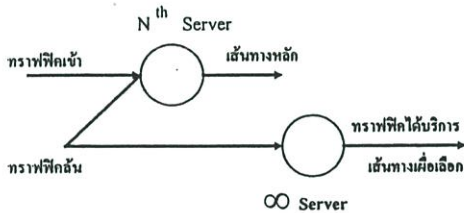
ฟิกของแต่ละเส้นทางให้มีปริมาณกราฟฟิกเท่าๆ กัน ซึ่งเส้นทางที่เหลืออยู่หรือเส้นทางบางเส้นทาง อาจจะเป็นเส้นทางที่เหมาะสมกว่า ดังนั้นจึงได้ศึกษาการจัดเส้นทางโดยใช้ฮอปฟิลด์เน็ต (HN) ซึ่งการจัดเส้นทางแบบนี้ จะได้เส้นทางที่ดีที่สุด ซึ่งจะคำนึงถึงสภาวะของปริมาณกราฟฟิกที่มีการเปลี่ยนแปลง และค่าความน่าจะเป็นของการเรียกติดขัด ให้อยู่ในเกณฑ์มาตรฐานที่ยอมรับได้ รวมทั้งในกรณีที่อาจมีบางเส้นทางที่ใช้ในการติดต่อสื่อสารเกิดล้มเหลว ไม่สามารถใช้งานได้ โดยจะไม่คำนึงถึงวิธีการเดิม วิธีการใหม่ที่

นำเสนอแนะ จะเลือกเส้นทางที่ดีที่สุดสำหรับการติดต่อสื่อสาร ตามเงื่อนไขที่ได้สร้างขึ้น

2.การจัดเส้นทางในแบบเดิม

2.1 การจัดเส้นทางแบบ ทราฟฟิคคั่น

เมื่อผู้เรียก (SUB A) ทำการเรียกไปยังชุมสายท้องถิ่นอื่นๆ หรือ ทราฟฟิคขาออก จะสามารถกระทำได้หลายเส้นทาง ดังแสดงในรูปที่ 1 โดยเส้นทางแรกเรียกว่า เส้นทางตรง (Direct Route) ทราฟฟิคคั่นจะไปใช้เส้นทางเลือกหรืออาจจะเรียกว่าเส้นทางที่ 2 และหรือเส้นทางเลือกสุดท้าย ตามลำดับ



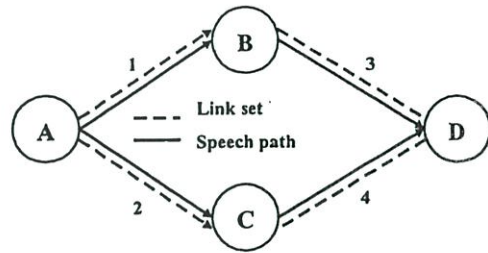
รูปที่ 1 แผนผังแสดงทราฟฟิคคั่น (Overflow Traffic)

การจัดเส้นทางแบบนี้เรียกว่าการจัดเส้นทางที่พิจารณาจากปริมาณทราฟฟิคคั่น ปริมาณทราฟฟิคจะพิจารณาจากทราฟฟิคใน 2 เส้นทาง แสดงดังรูปที่ 1 โดยทราฟฟิคขอบริการที่เส้นทางตรง Nth Servers (ตามสมมติฐานของเออแลง B) จะเป็นทราฟฟิคแบบสุ่ม และถ้า Nth Server ทั้งหมดในเส้นทางตรงเต็ม ทราฟฟิคก็จะล้นมายังเส้นทางที่สอง ซึ่งจะมี ∞ Servers เป็น ทราฟฟิคแบบสุ่ม [1-2,6]

2.2 การจัดเส้นทางแบบ แบ่งโหลด

จากรูปที่ 2 แสดงการจัดเส้นทางแบบแบ่งโหลด โดยการจัดเส้นทางแบบนี้จะเป็นการแชร์การใช้งานของจำนวนวงจรระหว่างสองเส้นทาง ให้มีปริมาณการใช้เท่าๆ กัน หรืออาจแบ่งเป็นเปอร์เซ็นต์แต่ละเส้นทาง ส่วนมากจะแชร์ที่ 50% ในกรณีที่ว่าวงจรของเส้นทางที่ 1 และ 2 มีจำนวนเท่ากัน ดังเช่น ถ้ามีการเรียก ออกจากชุมสาย A ไปยังชุมสาย D ทั้งหมด

จำนวน 4 Call การเรียกครั้งที่ 1 จะถูกส่งไปยังเส้นทางที่ 1 ก่อน ครั้งที่ 2 จะไปยังเส้นทางที่ 2 ลักษณะนี้สลับกันไป โดยครั้งที่ 3 ก็จะไปยังเส้นทางที่ 1 อีก และครั้งที่ 4 ก็จะกลับไปยังเส้นทางที่ 2 คือจัดให้แต่ละเส้นทางมีปริมาณทราฟฟิคเท่าๆ กัน[3-4]



รูปที่ 2 การจัดเส้นทางของโครงข่ายโดยวิธี Load Sharing

3. นิวรอลเน็ตเวิร์คแบบ Hopfield Net

ในปี ค.ศ.1985 Hopfield และ Tank [5] ได้นิยามปัญหา Traveling Salesman Problem (TSP) ด้วยฟังก์ชันพลังงานดังนี้

$$E = \frac{A}{2} \sum_x \sum_i \sum_j v_{xi} v_{xj} + \frac{B}{2} \sum_i \sum_x \sum_y v_{xi} v_{yi} + \frac{C}{2} \left(\sum_x \sum_i v_{xi} - n \right)^2 + \frac{D}{2} \sum_x \sum_y \sum_i d_{xy} v_{xi} (v_{y,i+1} + v_{y,i-1}) \quad (1)$$

โดยที่ A,B,C,D คือค่าคงที่
 n คือจำนวนเมือง
 d_{xy} คือระยะทางระหว่างเมือง X กับเมือง Y
 v_{xi} คือเมตริกซ์ตัวที่ x,i
 v_{xi} เท่ากับ 1 หมายความว่าเมือง X อยู่ลำดับที่ i ของการเดินทาง แต่ถ้าเมือง X ไม่ใช่เมืองในลำดับที่ i ของการเดินทางจะได้ค่า v_{xi} = 0

พิจารณาฟังก์ชันพลังงานสมการที่ 1 เทียบกับสมการไลอะปูนอฟ (Lyapunov) ที่มีการขยายสูง (High gain) ดังสมการที่ 2 [5]

$$E = \frac{1}{2} \sum_{xi} \sum_{yj} W_{xi,yj} V_{xi} V_{yj} - \sum_{xi} I_{xi} V_{xi} \quad (2)$$

จากสมการที่ 1 และ 2 จะได้

$$\left. \begin{aligned} W_{xi,yj} &= -A\delta_{XY}(1-\delta_{ij}) - B\delta_{ij}(1-\delta_{XY}) \\ &\quad - C - Dd_{XY}(\delta_{j,i+1} + \delta_{j,i-1}) \\ I &= Cn', Cn' = B - \frac{C}{2} \end{aligned} \right\}$$

(3)

โดยที่ δ_{xy} คือฟังก์ชันโครเนคเกอร์เดลต้า (Kronecker

$$\text{delta function) } \delta_{xy} = \begin{cases} 1; X = Y \\ 0; X \neq Y \end{cases}$$

การเปลี่ยนแปลงระดับแอกทิวิตี (Activity level) ของโครงข่ายประสาท มีค่าตามสมการการขับเคลื่อน (Equation of motion) ดังสมการที่ 4

$$\Delta U_{xi} = \left[\sum_y \sum_i W_{xi,yi} - \frac{U_{xi}}{\tau} + I_{xi} \right] \Delta t \quad (4)$$

ΔU_{xi} คือ ค่าการเปลี่ยนแปลงระดับแอกทิวิตีของนิวรอน xi

Δt คือ ช่วงเวลาที่มีค่าการเปลี่ยนแปลงระดับแอกทิวิตีของนิวรอน xi

แทนค่าจากสมการที่ 3 ลงในสมการที่ 4 จะได้

$$\Delta U_{xi} = \left[\begin{aligned} &-\frac{U_{xi}}{\tau} - A \sum_{j \neq i} V_{xi} - B \sum_{y \neq x} V_{yj} - \\ &C \sum_y \sum_j V_{yj} - D \sum_y d_{xy} (V_{y,i+1} - V_{y,i-1}) \\ &+ Cn' \end{aligned} \right] \Delta t \quad (5)$$

ระดับแอกทิวิตีในรอบการคำนวณครั้งต่อไป คือ

$$U_{xi}(t+1) = U_{xi}(t) + \Delta U_{xi} \quad (6)$$

โดยฟังก์ชันซิกมอยด์ (Sigmoid) แบบแทนเงินที่ไฮเปอโบลิกเป็นฟังก์ชันแอกทิเวชัน (Activation)

$$V_{xi} = g(U_{xi}) = \frac{1}{2} (-[1 + \tanh(\lambda V_{xi})]) \quad (7)$$

โดยที่ λ คือค่าคงที่สำหรับปรับความชัน (Slop) ของฟังก์ชันแอกทิเวชัน จะใช้ค่าตั้งแต่ 0 จนถึง 50 ตามความเหมาะสม

3.1 การประยุกต์ใช้โครงข่าย Hopfield Net

ในบทความนี้ได้เสนอวิธีการจัดเส้นทางของโครงข่ายโทรศัพท์ โดยใช้นิวรอนแอกทิเวชันฮอปฟิลด์เน็ต ซึ่งได้คำนึงถึงค่าความน่าจะเป็นของการเรียกติดขัด (Blocking probability) ดังสมการที่ 8

$$B = \frac{X_L}{X_0} \quad (8)$$

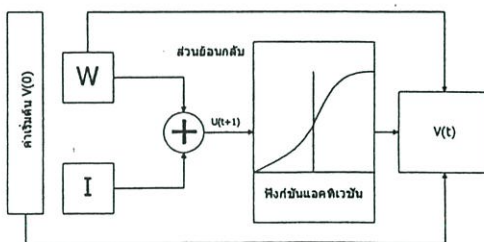
โดยที่ $X_0 = X_c + X_L$

B คือค่าค่าความน่าจะเป็นของการเรียกติดขัด

X_0 คือค่าทราฟฟิกขอบริการ

X_L คือค่าทราฟฟิกสูญเสีย

X_c คือค่าทราฟฟิกได้รับบริการ



รูปที่ 3 โค้ดแกรมของโครงข่ายการจัดเส้นทางให้เหมาะสมโดยใช้ Hopfield Net

พิจารณารูปที่ 3 จากลักษณะดังกล่าวสามารถนำมาสร้างฟังก์ชันพลังงานได้ดังนี้

$$E = \frac{A}{2} \sum_x \left[\sum V_{xi} - 1 \right]^2 + \frac{B}{2} \sum_i \left[\sum_x V_{xi} - 1 \right]^2 + \frac{C}{2} \sum_{xi} \sum_i V_{xi} (1 - V_{xi}) + \frac{D}{2} \sum_x \sum_i \sum_j L_{ij} V_{xi} V_{xj} \quad (9)$$

โดยที่ A,B,C,D คือค่าคงที่

n คือจำนวน โหนดทั้งหมด

nf คือจำนวนเส้นทางทั้งหมด

L_{ij} คือค่าความน่าจะเป็นของการเรียกติดขัดระหว่าง โหนด i และ j

V_{xi} คือข้อมูลการส่งออกแสดงการเชื่อมโยงระหว่าง โหนด i กับเส้นทาง X

ฟังก์ชันโลอะฟูนอฟที่มีการขยายสูง และมีสมการ การขับเคลื่อนสำหรับตัวห้อยคู่ (Double subscript)

$$E = -\frac{1}{2} \sum_x \sum_y \sum_i \sum_j W_{xi,yj} V_{yj} - \sum_x \sum_i V_{xi} I_{xi}$$

$$\Delta V_{xi} = \left[\sum_y \sum_j W_{xi,yj} V_{yj} - \frac{U_{xi}}{\tau} + I_{xi} \right] \Delta t \quad (10)$$

จากสมการที่ 9 และฟังก์ชันโลอะฟูนอฟในสมการที่ 10 จะได้

$$W_{xi,yj} = \begin{cases} -A\delta_{xy} - B\delta_{ij} - C\delta_{xy}\delta_{ij} - DL_{ij}\delta_{xy} \\ -B\delta_{ij} - C\delta_{xy}\delta_{ij} - DL_{ij}\delta_{xy} \end{cases}$$

$$I_{xi} = B - \frac{C}{2} \quad \begin{matrix} \text{เมื่อ } i \in S \\ \text{เมื่อ } i \notin S \end{matrix} \quad (11)$$

โดยที่ S คือกลุ่มโหนดที่เป็นโหนดต้นทางของแต่ละเส้นทาง การเรียนรู้ของโครงข่ายที่สร้างขึ้น ได้จากการนำค่าจากสมการที่ 11 แทนลงในสมการ การขับเคลื่อนในสมการที่ 10 จะได้

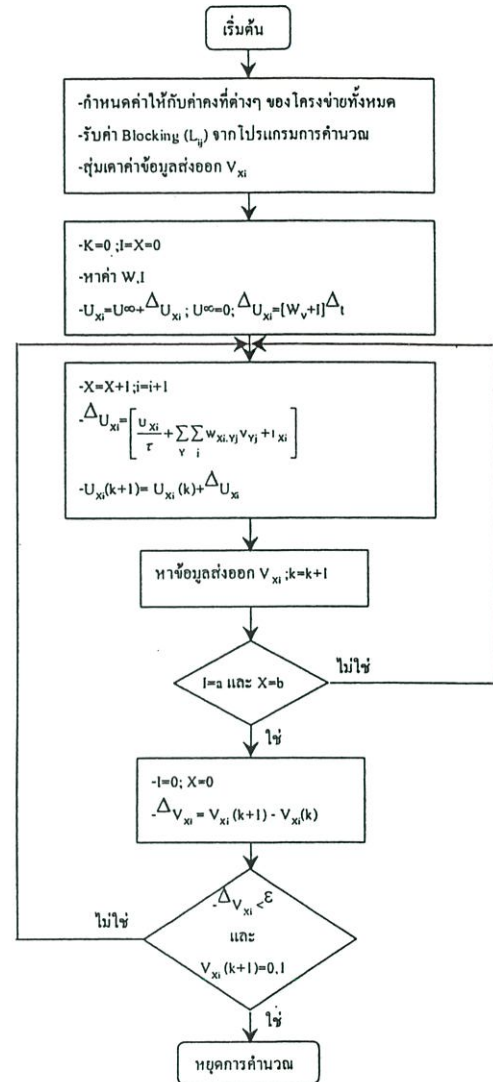
$$\Delta u_{xi} = \begin{cases} \Delta u1_{xi} + \Delta u2_{xi} & \text{เมื่อ } i \in S \\ \Delta u2_{xi} & \text{เมื่อ } i \notin S \end{cases} \quad (12)$$

โดยที่

$$\Delta u1_{xi} = \left[-A \sum_{i \neq k} V_{xi} \right] \Delta t$$

$$\Delta u2_{xi} = \left[-\frac{U_{xi}}{\tau} - B \sum_y V_{yj} + CV_{xi} - D \sum_j L_{ij} V_{xi} + B - \frac{C}{2} \right] \Delta t$$

จากรูปที่ 3 มีขั้นตอนของกระบวนการเรียนรู้ของการจัดเส้นทางคง โคอะแกรมรูปที่ 4



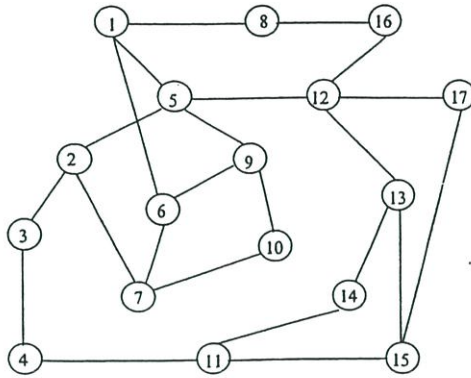
รูปที่ 4 โคอะแกรมของกระบวนการเรียนรู้ของโครงข่ายการจัดเส้นทางให้เหมาะสม

4.การทดลองและผลการทดลอง

การทดลองใช้โปรแกรม Visual Basic จำลองโครงสร้างของนิรวลเน็ตเวิร์ก ตามโมเดลที่ใช้ทดสอบ ดังรูปที่ 5 และมีกระบวนการเรียนรู้ของโครงข่ายการจัดเส้นทางให้เหมาะสม ตามโคอะแกรมรูปที่ 4 ส่วนการจัดเส้นทางแบบ Overflow และ Load

Sharing ให้เลือกเส้นทางที่หมายเลขของโหนดที่น้อย เป็นเส้นทางแรกก่อน แล้วจึงไปเลือกเส้นทางลำดับต่อไป

4.1 โมเดลของโครงข่ายที่ใช้ในการทดสอบ



รูปที่ 5 โมเดลของโครงข่ายที่ใช้ในการทดสอบ

จากรูปที่ 5 สามารถจัดกลุ่มของโหนดได้ เป็น 17 โหนด และมีจำนวนเส้นทางทั้งหมด nf สามารถแทนได้ด้วยเมตริกซ์ V ซึ่งมีขนาด $nf \times n$ และ n คือจำนวนโหนด ถ้าสมาชิก $V_{xi} = 1$ หมายความว่าโหนด i ต่อกับเส้นทาง X แต่ถ้าโหนด i ไม่ต่อกับเส้นทาง X จะได้ $V_{xi} = 0$

ตารางที่ 1 ค่าความน่าจะเป็นของการเรียกติดขัดของแต่ละเส้นทาง

เส้นทาง	ความน่าจะเป็นของการเรียกติดขัด			จำนวน วงจร
	ปกติ	เปลี่ยน แปลง	เส้นทาง 2-3 ขาด	
1-5	0.189	0.189	0.189	30
1-6	0.053	0.053	0.053	60
2-3	0.053	0.053	10	60
2-7	0.028	0.028	0.081	120
3-4	0.189	0.189	0.189	30
4-11	0.053	0.053	0.053	60
5-2	0.189	0.189	0.189	30
6-2	0.189	0.189	0.189	30
6-7	0.053	0.053	0.053	60
7-3	0.189	0.189	0.189	30

7-15	0.189	0.417	0.189	30
11-14	0.189	0.189	0.189	30
15-11	0.053	0.053	0.053	60

4.2 การทดสอบการกำหนดเส้นทาง

การทดสอบการกำหนดเส้นทางให้กับโครงข่ายโทรศัพท์ โดยกำหนดให้โหนดที่มีวงจรเชื่อมโยงเท่ากับ 30 วงจร 60 วงจร และ 120 วงจร ให้มีทราฟฟิกเข้าเท่ากับ 65% ,40% และ 45% ตามลำดับ ซึ่งมีความน่าจะเป็นของการเรียกติดขัด ดังตารางที่ 1 ส่วนเงื่อนไขของการทดสอบมี 3 สถานะ คือ ขณะที่โครงข่ายปกติทราฟฟิกปกติ ขณะที่โครงข่ายปกติทราฟฟิกมีการเปลี่ยนแปลง และโครงข่ายขัดข้องในเส้นทางที่ใช้ในการติดต่อสื่อสาร ซึ่งเส้นทางนั้นไม่สามารถใช้งานได้ และจะกำหนดให้เส้นทางนั้นมีความน่าจะเป็นของการเรียกติดขัดเท่ากับ 10

4.3 การทดสอบและผลการทดสอบขณะที่สถานะของโครงข่ายปกติ

ทดสอบขณะที่ทราฟฟิกปกติกับทราฟฟิกมีการเปลี่ยนแปลงในเส้นทาง 7-15 ผลที่ได้ดังตารางที่ 2

ตารางที่ 2 ขณะที่โครงข่ายปกติ

	เส้นทางที่ถูกเลือกขณะที่ทราฟฟิก	
	ปกติ	เปลี่ยนแปลง
Hopfield Net	2-7-15-11-14	2-3-4-11-14
Overflow	2-3-4-11-14	2-3-4-11-14
Load Sharing	2-3-4-11-14	2-3-4-11-14

4.4 การทดสอบและผลการทดสอบขณะที่สถานะของโครงข่ายปกติ ทดลองเปลี่ยนโหนดต้นทางและโหนดปลายทาง

กำหนดให้เรียกจากโหนด 1 ไปยังโหนด 4 ผลที่ได้ดังตารางที่ 3

4.5 การทดสอบและผลการทดสอบขณะที่สถานะของโครงข่ายปกติ และเส้นทางเชื่อมโยงระหว่างโหนดที่ 2 และ 3 เกิดล้มเหลวไม่สามารถใช้งานได้

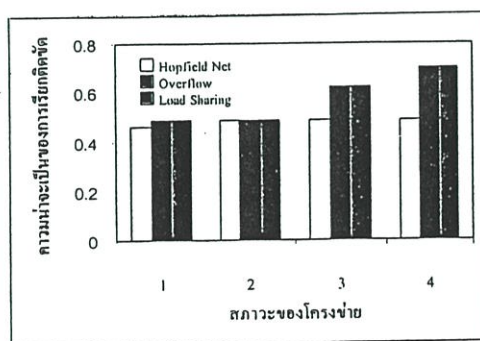
จากการกำหนดเส้นทางในหัวข้อที่ 4.4 เมื่อเส้นทางเชื่อมโยงระหว่างโหนดที่ 2 และ 3 เกิดล้มเหลวไม่สามารถใช้งานได้ ผลที่ได้เป็นดังตารางที่ 4 ตารางที่ 3 ขณะที่โครงข่ายปกติทดลองเปลี่ยนโหนดต้นทางและโหนดปลายทาง

	เส้นทางที่ถูกเลือก ขณะกราฟฟิคปกติ
Hopfield Net	1-6-2-3-4
Overflow	1-5-2-3-4
Load Sharing	1-5-2-3-4

ตารางที่ 4 ขณะที่โครงข่ายขัดข้องในเส้นทาง 2-3

	เส้นทางที่ถูกเลือก ขณะโครงข่ายขัดข้อง
Hopfield Net	1-6-7-3-4
Overflow	1-6-2-7-3-4
Load Sharing	1-6-2-7-3-4

จากการจัดเส้นทางโดยใช้ ฮอปฟิลด์เน็ต เปรียบเทียบกับการจัดเส้นทางโดยวิธี Overflow และ Load Sharing จะเห็นได้ว่าเส้นทางที่ได้จากการจัดเส้นทางแบบใช้ ฮอปฟิลด์เน็ต จะได้ค่าความน่าจะเป็นของการเรียกติดขัดโดยรวมแล้วมีค่าน้อยกว่าทั้ง 2 วิธี ดังรูปที่ 6



รูปที่ 6 ผลการเปรียบเทียบค่าความน่าจะเป็นของการเรียกติดขัดทั้ง 3 วิธี

จากรูปที่ 6 สถานะของโครงข่ายจากกราฟเป็นดังนี้ (1) คือสถานะโครงข่ายปกติกราฟฟิคปกติ,

(2) คือสถานะโครงข่ายปกติกราฟฟิคมีการเปลี่ยนแปลงในเส้นทางที่ใช้ในการติดต่อสื่อสาร, (3) คือสถานะโครงข่ายปกติกราฟฟิคปกติ เปลี่ยนโหนดต้นทางและโหนดปลายทาง และ (4) คือสถานะโครงข่ายขัดข้องในเส้นทางที่ใช้ติดต่อสื่อสาร ในที่นี้คือเส้นทาง 2-3 ขัดข้องไม่สามารถใช้งานได้

5.สรุป

จากผลการทดลองจะเห็นได้ว่าการจัดเส้นทางด้วยวิธีฮอปฟิลด์เน็ต จะได้เส้นทางที่ดีที่สุดตามเงื่อนไขที่ได้สร้างขึ้น ไม่ว่าจะปริมาณกราฟฟิคจะเปลี่ยนแปลงหรือมีบางเส้นทางที่ใช้ในการติดต่อสื่อสารเกิดล้มเหลวไม่สามารถใช้งานได้ ก็จะสามารถจัดหาเส้นทางสำรองให้ ซึ่งเส้นทางที่ถูกเลือก จะมีค่าผลรวมของความน่าจะเป็นการเรียกติดขัด (Blocking probability) น้อยกว่าวิธีการแบบ Overflow และ Load Sharing โดยเฉลี่ย 10% ขึ้นไป และยังใช้เส้นทางในการติดต่อสื่อสารระหว่างโหนดต้นทางและโหนดปลายทาง น้อยกว่าทั้งสองวิธี ซึ่งเส้นทางที่ถูกเลือกโดยวิธีฮอปฟิลด์เน็ต จะเป็นเส้นทางที่ใช้ในการติดต่อสื่อสารได้อย่างมีประสิทธิภาพ

6. เอกสารอ้างอิง

- [1] ITU-T E521, "Calculation of the number of circuit in a group carrying overflow traffic," 1985.
- [2] ITU-T E525, "Designing Network to Control Grade of Service," 1985.
- [3] NTT, Network Planning, 1990.
- [4] NEC Practice, "Load sharing," NEC ND-56427-001(E) Issue 4, June 1992.
- [5] J.J. Hopfield and D.W. Tank, "Neural computation of decisions in optimization problems," *Biolog. Cybern.*, vol.52, pp.141-152, 1985.

ผลงานทางวิชาการที่ได้รับการตีพิมพ์

- 1.ฉัตรชัย วัฒนพันธุ์ พิเชฐ ม่วงนวล มนูญ สุขเกษม บทความวิชาการเรื่อง “การวิเคราะห์ข้อมูลใน ISUP ของโครงข่ายบริการร่วมระบบดิจิทัล” วิศวกรรมลาดกระบัง ปีที่ 17 ฉบับที่ 2 มิถุนายน 2543.
- 2.ฉัตรชัย วัฒนพันธุ์ มนูญ สุขเกษม ถวิล พึ่งมา บทความวิชาการเรื่อง “การจัดเส้นทางของโครงข่ายโทรศัพท์ให้ดีที่สุดโดยวิธีฮอปฟิลด์เน็ต” วิศวกรรมลาดกระบัง ปีที่ 20 ฉบับที่ 1 มีนาคม 2546.

ประวัติผู้เขียน

ชื่อ-นามสกุล	นายฉัตรชัย วัฒนพันธุ์
วัน เดือน ปี เกิด	5 พฤษภาคม พ.ศ.2513 ที่จังหวัดอุทัยธานี
ที่อยู่	152 หมู่ 4 ตำบลเมืองกาฐัง อำเภอบ้านไร่ จังหวัดอุทัยธานี 61180
ประวัติการศึกษา	2534 อดสาหกรรมศาสตรบัณฑิต สาขาวิศวกรรมไฟฟ้า มหาวิทยาลัยสยาม
ความชำนาญเฉพาะด้าน	1.) Telephone System 2.) Network Planning 3.) ISDN Technology
ประสบการณ์การทำงานและผลงานวิจัย	
พ.ศ.2534-2540	ตำแหน่ง วิศวกร องค์กรโทรศัพท์แห่งประเทศไทย โดยประจำอยู่ที่แผนกอุปกรณ์ชุมสาย โทรศัพท์จังหวัดอุทัยธานี จังหวัดอุทัยธานี
พ.ศ.2541-2547	ตำแหน่ง วิศวกร องค์กรโทรศัพท์แห่งประเทศไทย โดยประจำอยู่ที่กองระบบซอฟต์แวร์ชุมสาย ส่วนระบบชุมสาย ฝ่ายโครงข่ายโทรคมนาคม ซึ่งในปัจจุบันได้เปลี่ยนมาเป็น บริษัท ทศท คอร์ปอเรชั่น จำกัด (มหาชน) กรุงเทพฯ
ปัจจุบัน	ตำแหน่ง วิศวกร บริษัท ทศท คอร์ปอเรชั่น จำกัด (มหาชน) โดยประจำอยู่ที่ ส่วนปฏิบัติการและบำรุงรักษาระบบชุมสาย ฝ่ายโครงข่ายโทรคมนาคม กรุงเทพฯ